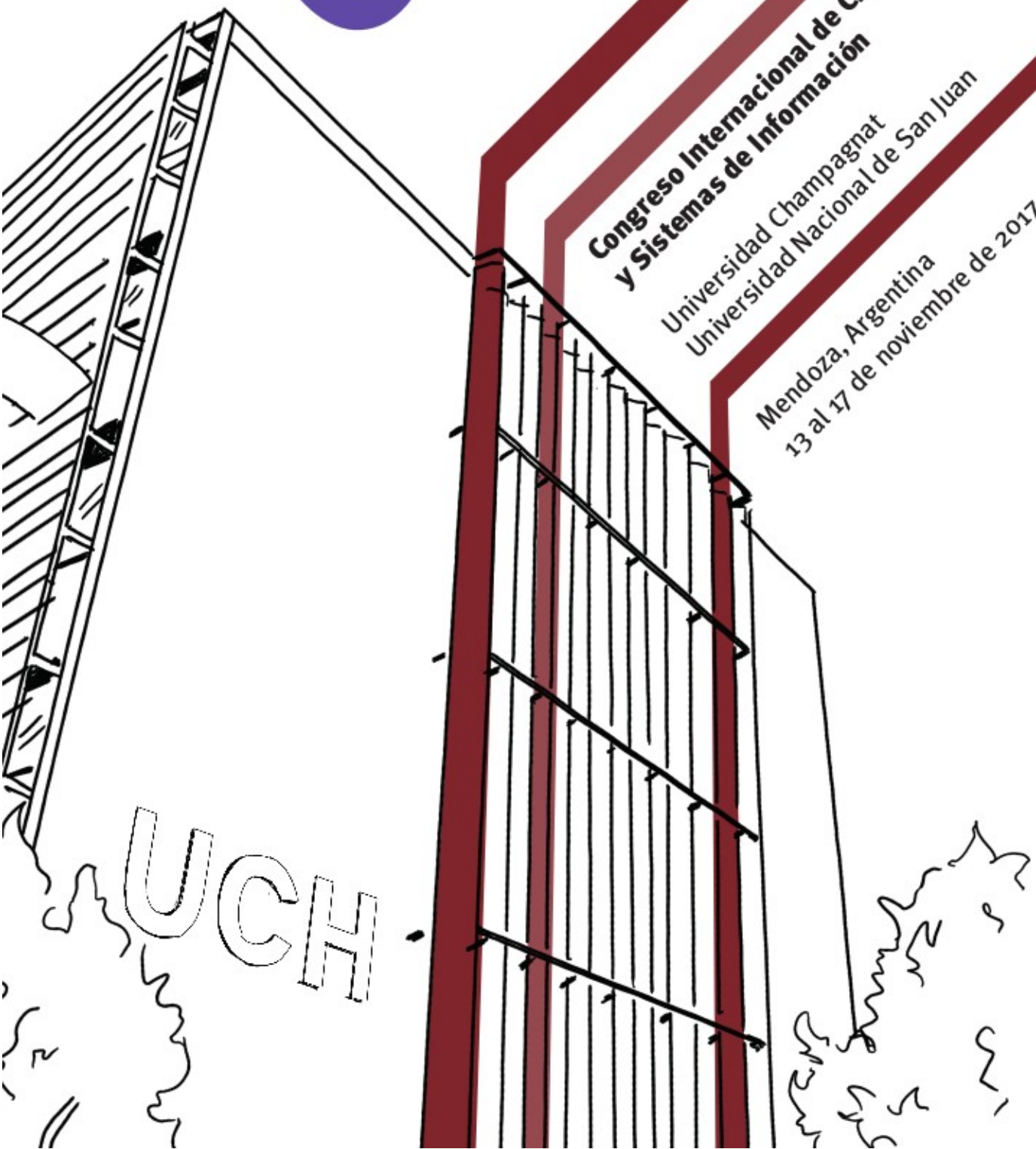
CICCSI 2017

Congreso Internacional de Ciencias de la Computación y Sistemas de Información

Universidad Champagnat
Universidad Nacional de San Juan

Mendoza, Argentina
13 al 17 de noviembre de 2017



Congreso Internacional de Ciencias de la Computación y Sistemas de Información CICCI 2017

Mendoza, Argentina, del 13 al 17 de noviembre de 2017

Organizadores:

Universidad Champagnat, Mendoza, Argentina.
Universidad Nacional de San Juan, San Juan Argentina.

Fecha de Catalogación: 12/06/2018

Anales del Congreso Internacional de Ciencias de la Computación y Sistemas de Información - CICCSI 2017 / Fernando Pinciroli ... [et al.]; compilado por Fernando Pinciroli ... [et al.]. - 1a edición para el alumno - Godoy Cruz: FUSMA Ediciones, 2017.

Libro digital, PDF

Archivo Digital: descarga
ISBN 978-987-45683-5-9

1. *Computación. 2. Sistemas de Información. I. Pinciroli, Fernando II. Pinciroli, Fernando, comp.*
CDD 005.1

(página dejada intencionalmente en blanco)

Comité científico

José Luis Barros Justo, Universidade de Vigo, España
Fernando Pinciroli, Universidad Champagnat, Argentina
Laura Zeligueta, Universidad Champagnat, Argentina
Javier Rosenstein, Universidad Champagnat, Argentina
Miguel Méndez Garabetti, Universidad Champagnat, Argentina
Mario Mirallas, Universidad Champagnat, Argentina
Carolina Canessa, Universidad Champagnat, Argentina
Marcelo Palma, Universidad Champagnat, Argentina
Elisa Galdame, Universidad Champagnat, Argentina
María Inés Lund, Universidad Nacional de San Juan, Argentina
Laura Nidia Aballay, Universidad Nacional de San Juan, Argentina
Nelson Rodríguez, Universidad Nacional de San Juan, Argentina
Manuel Ortega, Universidad Nacional de San Juan, Argentina
Alejandra Malberti, Universidad Nacional de San Juan, Argentina
Raúl Klenzi, Universidad Nacional de San Juan, Argentina
Alejandra Otazú, Universidad Nacional de San Juan, Argentina
Silvana Aciar, Universidad Nacional de San Juan, Argentina
Juan Aranda, Universidad Nacional de San Juan, Argentina
María Murazzo, Universidad Nacional de San Juan, Argentina
Daniel Díaz, Universidad Nacional de San Juan, Argentina
Roberto Guerrero, Universidad Nacional de San Luis, Argentina
Fabiana Píccoli, Universidad Nacional de San Luis, Argentina
Sandra Rodríguez, Universidad Nacional de La Rioja, Argentina
Rosana Giménez, Universidad del Aconcagua, Argentina

Patricia Ontiveros, Universidad Tecnológica Nacional-FRM, Argentina
Osvaldo Marianetti, Universidad de Mendoza, Argentina
Fabiane Benitti, Universidade Federal de Santa Catarina, Brasil
Ania Cravero, Universidad de La Frontera, Chile
Samuel Sepúlveda, Universidad de La Frontera, Chile
Alonso Toro Lazo, Universidad Católica de Pereira, Colombia
Luis Eduardo Peláez Valencia, Universidad Católica de Pereira, Colombia
Juan Carlos Blandón Andrade, Universidad Católica de Pereira, Colombia
Álex A. Torres Bermúdez, Corporación Universitaria Comfacauca, Colombia
José Luis Arciniegas Herrera, Universidad del Cauca, Colombia
Martín Monroy Ríos, Universidad de Cartagena, Colombia
Mayela Coto, Universidad Nacional de Costa Rica, Costa Rica
Sonia Mora, Universidad Nacional de Costa Rica, Costa Rica
Santiago Matalonga, University of the West of Scotland, Escocia
Gilberto Borrego Soto, Universidad Autónoma de Baja California, México
Gisela Torres, Universidad Tecnológica de Panamá, Panamá
Gerardo Maturro, Universidad ORT, Uruguay

Comité académico

Coordinador general:
Raymundo Forradellas, Universidad Nacional de San Juan, Argentina

Coordinadores de programa:
José Luis Barros Justo, Universidade de Vigo, España
Laura Zeligueta, Universidad Champagnat, Argentina
María Inés Lund, Universidad Nacional de San Juan, Argentina

Coordinadores de talleres:
Javier Rosenstein, Universidad Champagnat, Argentina
Nelson Rodríguez, Universidad Nacional de San Juan, Argentina

Coordinadores del encuentro científico:
Fernando Pinciroli, Universidad Champagnat, Argentina
Laura Nidia Aballay, Universidad Nacional de San Juan, Argentina

Coordinadores del encuentro de profesores:
Manuel Ortega, Universidad Nacional de San Juan, Argentina
Elisa Galdame, Universidad Champagnat, Argentina

Coordinador del encuentro de graduados:
Juan Berdugo, Universidad Champagnat, Argentina

Comité organizador

Fernando Pinciroli, Universidad Champagnat, Argentina

Susana Dueñas, Universidad Champagnat, Argentina

Laura Zeligueta, Universidad Champagnat, Argentina

María Inés Lund, Universidad Nacional de San Juan, Argentina

Verónica Miguez, Universidad Champagnat, Argentina

Miguel Méndez Garabetti, Universidad Champagnat, Argentina

Javier Rosenstein, Universidad Champagnat, Argentina

Elisa Galdame, Universidad Champagnat, Argentina

Prólogo

El Congreso Internacional de Ciencias de la Computación y Sistemas de Información, CICCSI, organizado por la Universidad Champagnat, de la provincia de Mendoza y por la Universidad Nacional de San Juan, pretende ser la actividad académica de referencia de la región y ubicarse entre las más importantes de la Argentina, cuyo objetivo es reunir en un evento de periodicidad anual a investigadores, profesionales y alumnos de diferentes países, al promover un espacio para la difusión de los conocimientos generados en los ámbitos académico y de la industria.

En esta edición de CICCSI, llevada a cabo en las instalaciones de la Universidad Champagnat, en Mendoza, del 13 al 17 de noviembre de 2017, se buscó potenciar el intercambio entre investigadores, docentes y estudiantes de informática con la industria, representada por empresas, organizaciones y profesionales del sector.

Así fue que las actividades realizadas en el evento se organizaron de acuerdo con esa idea rectora, motivo por lo que se incluyeron dieciocho conferencias internacionales dictadas por referentes de la academia y de la industria, tres cursos con certificaciones profesionales y dos conferencias magistrales a cargo de Arie van Bennekum, de Holanda, coautor del Manifiesto

Ágil para el desarrollo de software, y de Luis Canessa, ingeniero proveniente de la empresa Airbus, de los Estados Unidos.

Por parte de la industria, en las conferencias se destacó la presencia del Centro de Operaciones Aeroespacial Alemán, de las fábricas de software Everis, Aconcagua Software Factory, Belatrix Software y GlobalLogic, y de las empresas Mercado Libre, Red Link y Edeste. En cuanto a las conferencias de parte de la academia, los expositores provinieron la Universidad Champagnat, la Universidad Nacional de San Juan y la Universidad Nacional de San Luis, por Argentina, y de la Universidad del Cauca, del Politécnico Grancolombiano, de la Universidad Católica de Pereira y de la Corporación Universitaria Comfacauca, por Colombia.

Además, se desarrollaron un encuentro de investigadores en Ciencias de la Computación y otro de investigadores en Sistemas de Información, un encuentro de docentes universitarios, un encuentro de graduados y otro para colegios secundarios, donde el Polo TIC de Mendoza les hizo una presentación de modo de acercarlos a nuestra disciplina.

En cuanto a los artículos científicos que recibimos y que fueron evaluados por un comité académico de primera línea, presentamos en esta obra los trabajos recibidos de Panamá, Brasil, Argentina, Colombia y Paraguay, cubriendo una amplia variedad de temas de las dos ramas que abarca CICCSI y que destaca en su nombre.

Finalmente, además de felicitar a los autores de los artículos que presentamos en esta obra, queremos hacerle llegar nuestra mayor gratitud para todos los organizadores, miembros del comité y colaboradores para que CICCSI pudiera llevarse a cabo, agradecemos enormemente a los sponsors del congreso, a las organizaciones que lo auspiciaron, a las honorables cámaras de Diputados y de Senadores de la provincia de Mendoza y de Diputados de la provincia de San Juan, que declararon de interés al evento, y también queremos expresarles nuestro mayor agradecimiento a la Red UNCI, al Colegio de Informáticos de San Juan y a la IEEE, quienes también nos acompañaron con sus respectivos auspicios.

Noviembre de 2017

Fernando Pinciroli
Laura Aballay
Laura Zeligueta
María Inés Lund
Miguel Méndez-Garabetti

(página dejada intencionalmente en blanco)

Índice

(página dejada intencionalmente en blanco)

Diseño de una Ontología para Validar la Calidad de Datos en Sistemas Web

Marta E. Cabrera Villafañe[1], Alejandra G. Espinosa[2], Sandra I. Rodríguez[2].

Centro de Investigación y Desarrollo Informático (CIDI)
UNIVERSIDAD NACIONAL DE LA RIOJA
marta.cabreravilla@gmail.com; alegabrielaespinosa@gmail.com

Resumen: Uno de los factores esenciales, para que una empresa se posicione en altos niveles de competitividad en el mercado, es el acceso rápido, eficiente y fácil a la información valiosa y útil. Para alcanzar este objetivo, es necesario diseñar una ontología para medir la calidad de los datos en sistemas Web para lograrlo se utilizará la metodología Methontology. Es necesario destacar que la calidad es una característica fundamental que debe poseer los datos que son utilizados como materia prima por los procesos de producción de información, esta calidad de la información es una particularidad fundamental que debe tener un producto de información para que su utilización cumpla con los requisitos de los usuarios.

Palabras Claves: Diseño de Ontología; Calidad de Datos; Sistemas Web.

Introducción

En la actualidad es posible encontrar gran variedad de software de E- Commerce con aplicaciones similares que pueden ser utilizadas en, las diferentes áreas de la empresa, como por ejemplo: ventas, marketing finanzas etc. Son numerosas las empresas que se han beneficiado por la implementación de un sistema de E-Commerce, Pronosticándose que con el tiempo se convertirá en una necesidad para toda empresa.
La Internet y todos los servicios que ofrece proporcionan un sin número de datos que si son bien utilizados forman un patrimonio importante para el progreso económico de la empresa. Según los gerentes y personal encargado de la toma de decisiones, una empresa mantendrá su nivel de competitividad si los clientes tienen un acceso rápido y fácil a información útil y valiosa brindada por la empresa. Una forma de alcanzar este objetivo consiste en proporcionar datos, obtenidos con el uso de E- Commerce, con alta calidad. Según Miguel Pérez Subías [1], el e-commerce, no se centra en la Web sino en el cliente. Por esta razón, las empresas implementan técnicas de E-Commerce que permiten a la organización: Registrar, Medir, Actualizar y Analizar, gran cantidad de información relacionada con el cliente [2]. Además las empresas pueden añadir a sus bases de datos de cliente datos externos, recuperados. Se puede hacer esta afirmación porque una ontología permite: i) Describir sin ambigüedades un dominio de discurso y

ii) Establecer un vocabulario compartido y consensuado. Una vez realizada esta tarea se procederá a la definición de métricas de calidad [3].
Calidad es una característica fundamental que deben poseer los datos que son utilizados como materia prima por los procesos de producción de información. La calidad de la información es una particularidad fundamental que debe tener un producto de Información para que su utilización cumpla con los requisitos de los Usuarios.
El artículo está organizado como sigue. La sección 2 presenta una descripción de la metodología propuesta, que incluye la metodología de trabajo y los objetivos planteados. La sección 3 describe la implementación y los resultados obtenidos. Por último la sección 4 expone las conclusiones por los servicios de información, que permitan comprender mejor las necesidades de los mismos.
La línea de investigación presentada en este artículo tiene como objetivo principal brindar una solución al siguiente problema: "Calidad de Datos como valor estratégico de la Información en E-Commerce". Para alcanzar este objetivo se piensa que la definición de una Ontología de Medición de la Calidad de Datos es una tarea fundamental [4].

Metodologías Empleadas

La metodología utilizada para la construcción de la ontología que permitirá la obtención de datos de calidad utilizados en sitios web, es la metodología Methontology [5], se la eligió por ser una de las metodologías más difundida y madura que cuenta con actividades y técnicas para el proceso de desarrollo de ontologías, validada en diversos proyectos de investigación como por ejemplo: Metodología y Métodos para la construcción de Ontologías [6]; Un Modelo Ontológico para el Aprendizaje Colaborativo en la Educación Interactiva en la Distancia [7], etc. El principal objetivo de la ontología propuesta es proveer las bases conceptuales para desarrollar métricas válidas para esta dimensión. Esta metodología incorpora los aspectos que se detallan a continuación:

1. Identificar el propósito de la ontología: El objetivo de esta etapa es dejar en claro por qué se construye la ontología y que uso se le pretende dar. También es útil en este momento identificar y caracterizar el rango de posibles usuarios y aplicaciones de la ontología.
2. Construir la Ontología: En esta etapa se distinguen tres sub-tareas, ellas son: Capturar, Evaluar y Documentar la Ontología. La primera está relacionada con: a) Identificar los conceptos claves y relaciones en el dominio de interés, b) La producción de definiciones textuales precisas y carentes de ambigüedad que describan tales conceptos y relaciones y c) Identificar los términos para referirse a tales conceptos y relaciones.
3. Evaluar los ambientes de software asociados y la documentación con respecto al marco de referencia relacionado con la ontología.
4. Estudiar el establecimiento de pautas para documentar la ontología construida teniendo presente el propósito de la misma.

Para llevar a cabo las tareas mencionadas previamente se realizaron dos actividades muy importantes. La primera de ellas fue la selección de la metodología para la construcción de la ontología, La segunda, consistió en la aplicación de dicha metodología para desarrollar la ontología en cuestión.

Especificación y Creación de la Ontología

Para desarrollar la ontología se empleó la metodología methontology, descripta en la sección anterior, qué proporciona guías para especificar ontologías en un nivel de conocimiento, especialmente provee técnicas para especificar la conceptualización.
Los principales pasos propuestos por esta metodología son: Identificar el propósito de la ontología, Construcción de la ontología, Implementación y Evaluación. Como se muestra a continuación:

1. Se define los principales requerimientos de la ontología elaborados en la etapa de Especificación.
2. Se detalla la etapa de Conceptualización y se define los términos de la ontología y sus relaciones mostrándose las representaciones semiformales y los axiomas obtenidos en esta etapa.
3. Se procede a analizar aspectos técnicos de la etapa de implementación de la ontología y finalmente se expone algunas conclusiones sobre el desarrollo de la ontología obtenida.

Especificación

En esta etapa se debe tener en cuenta los principales requerimientos de la ontología, como son: el domino modelado, propósito, alcance y las fuentes de conocimiento utilizadas. A continuación se detalla cada uno de estos puntos para la ontología.
Dominio modelado: son elementos para la medición y/o estimación de artefactos de software (procesos, productos y recursos) que permiten satisfacer distintas necesidades de información.
Propósito de la ontología: esta ontology persigue distintos propósitos, siendo los más importantes: a)Suministrar el intercambio de información relacionadas a métricas e indicadores de software; b)Satisfacer la base de conocimiento para el diseño y construcción de ontología de medición de calidad del dato, que facilite la documentación, recuperación y búsqueda semántica de la información relacionada, sirviendo de apoyo a los procesos de aseguramiento de la calidad definiendo y especificando requerimientos no funcionales, implementando test de calidad entre otros y c)Está ontología puede ser útil como sub-ontología de una ontología de proceso de evaluación y medición más amplio.
Alcance: comprende todo los conceptos, atributos y relaciones necesarios para la catalogación y explotación de la información relacionadas a métricas en el ámbito de la Ingeniería de Software e Ingeniería Web.

Conceptualización

A continuación se muestra la conceptualización del dominio de métricas, presentando primero estrategias y técnicas utilizadas para este fin, y una descripción detallada del dominio para facilitar su comprensión.
Técnicas empleadas en el dominio de Conceptualización: En esta etapa se utilizaron técnicas y estrategias como las tablas sugeridas por la metodología methontology, y diagramas UML. La utilización de UML, como leguaje de representación de ontologías ha sido propuesta por el autor Cranelfield [8].
Se utilizó UML para modelar artefactos de Ingeniería de Software más declarativos como son las ontologías propuesta por Kogut P. [9]. Existen otras razones para la utilización del lenguaje UML, como notación para la conceptualización de una ontología que es necesario conocer: I)UML es una notación gráfica que surge con el aporte de la práctica de varios años en análisis y diseño de software de diversas compañías en un amplio campo de industria y dominio; II)Es un estándar abierto, administrado por el Object Management Group (OMG); III)Provee mecanismos para definir extensiones de aplicaciones específicas como es el modelo de ontología; IV) Las ontologías incluyen taxonomías de conceptos (comprendiendo jerarquías, clases y subclases), relaciones entre conceptos (asociaciones) y definiciones de atributos de conceptos; pudiéndose modelar con el lenguaje UML esta información de la ontología, más precisamente con diagramas de clase; V) Está soportado por la utilización de herramientas CASE, ampliamente difundidas y accesibles para los profesionales del software, a comparación con herramientas más específicas para ontología como Ontolingua y Protege, requiriendo experiencia en la representación del conocimiento.

La elaboración de esta ontología ha sido realizada teniendo en cuenta los siguientes pasos: 1°) Definición del glosario de conceptos partiendo de la fuente de conocimiento citadas, teniendo en cuenta los objetivos y el alcance de la ontología; 2°) Definir las interrelaciones semánticas entre dichos conceptos representándolas mediante diagrama de clases del lenguaje UML, elaborando una tabla de clases y relaciones con las diferentes clases que se van identificando, por ejemplo Clase: Atributo de calidad, Unidad de medida, Seguridad, etc.; 3°) Examinar los conceptos relacionados para identificar las partes comunes entre dos o más conceptos. Decidir si estas partes son conceptos, si así lo fuere, incluirlos en el glosario de conceptos; 4°) Identificar los atributos de cada concepto para construir la tabla de atributos, incluyéndolos en el diagrama de clase de UML.; 5°) Identificar los axiomas estructurales que aporten conocimiento adicional a la ontología y por último demostrar la completitud y corrección de toda la tabla, analizando el modelo conceptual UML, los objetivos y alcances de la ontología.

Descripción conceptual de la Medición y Evaluación

En esta sección, se presenta una descripción conceptual mostrando que se entiende por métrica, indicador, medición y otros conceptos de la ontología, para una mejor

interpretación de los términos. Para llevar a cabo esta tarea se muestran las estructuras y relaciones existentes entre las necesidades de información de un proyecto de evaluación y los elementos relevantes: entidades, atributos, métricas, métodos de cálculo y de medición, etc., que permiten obtener los informes que satisfagan dichas necesidades de información.
Un atributo es una propiedad medible de una entidad. Donde se sabe que una entidad puede tener muchos atributos, pero sólo algunos de ellos podrían ser de interés para evaluar un concepto calculable. Un atributo de una entidad puede ser cuantificado por una o varias métricas. Donde cada observación o medición consiste en asignar (obtener) el valor de un atributo en un determinado instante del tiempo, usando la definición de una de sus métricas asociadas. El valor del atributo obtenido es llamado medida.
Se podría decir que una métrica **m** representa una correspondencia m: A →X, siendo la flecha, la función de correspondencia y que conlleva homomorfismo. **A** corresponde al atributo del mundo empírico; **X** es la variable (representa el mundo formal), a la cual se le asigna el valor categórico o numérico que cuantifica al atributo. Para poder llevar a cabo esta correspondencia, se necesita una definición precisa de una actividad de medición específica, que establezca explícitamente un método y una escala (se necesita la definición de métricas).
Las métricas están asociadas a una determinada escala; las escalas se clasifican en cinco tipos: Absoluta, Nominal, Ordinal, Intervalo y de Proporción.
Para poder evaluar o estimar un concepto calculable se utiliza un indicador. Siendo necesario realizar la distinción semántica entre métricas e indicador, una métrica representa una correspondencia (mapeo), entre un atributo del mundo empírico y una variable en el mundo formal. El indicador representa un nuevo mapeo que comprende desde el valor de una métrica (mundo formal), a otra variable de un nuevo mundo formal, a la cual se le asigna valores numéricos que representan la interpretación de dicho valor de métrica y que soporta criterios de decisión para una necesidad o requerimiento especifico de usuario; por lo tanto, la correspondencia para indicadores elementales va de **X →Ie** (donde X es el valor de la métrica; **Ie** es la variable que contendrá el valor del Indicador Elemental).

Definición de Términos y sus Relaciones

En esta sección se definen a modo de ejemplo algunos de los términos de la ontología de métricas e indicadores ilustrando cada concepto con un ejemplo y mostrando su relación con otros conceptos. *Atributo:* es una propiedad mensurable, físico o abstracta, de una entidad.
Relaciones: Uno o más atributos se asocian a una o más entidades que caracterizan; Un atributo es cuantificado por una o más métricas y Uno o más atributos se combinan en un concepto calculable.
Ejemplos de atributos de la entidad: "Página Web", pueden ser: Tamaño de Imágenes, Tamaño del código HTML.
Concepto Calculable: es una relación abstracta entre atributos de entidades y necesidades de información.

Relaciones: Un concepto calculable combina uno o más atributos de entidades; Uno o más conceptos calculables son definidos con el objetivo de satisfacer una necesidad de información concreta; Un concepto calculable es especificado por uno o más modelos de conceptos; Un concepto calculable es evaluado o estimado por un indicador. *Ejemplos de concepto calculabl*e**:** Usabilidad (de un producto de software), productividad (por ejemplo de un grupo de desarrollo), Calidad, Costo.
Cálculo: es la actividad que utiliza una definición de indicador para producir un valor de indicador.
Relaciones: Cada cálculo produce un valor de indicador específico; Cada cálculo está relacionado a la descripción de un indicador que emplea para llevar a cabo la actividad.
Ejemplos: Actividad que consiste en usar la especificación del indicador "Preferencia en actividad de desarrollo" para obtener un valor en el campo numérico (formal), que estime o evalúe el atributo Calidad del dato en una página web. Esta acción dará como resultado un valor de indicador.
Métricas Directas: es una métrica de un atributo que no depende de una métrica de otro atributo.
Relaciones: Una métrica directa incluye un método de medición específico.
Ejemplos: Cantidad de línea de código fuente y Cantidad de enlaces rotos internos.
Comentario: En el estándar [10], se encuentra definido "medidas directas" y no "métricas directas"
*Indicador Elemental***:** es un indicador que no depende de otros indicadores para evaluar o estimar un concepto calculable.
Relaciones: Un indicador elemental puede interpretar una métrica específica. Y Un indicador elemental es modelado por un modelo elemental.
Ejemplos: Nivel de Productividad y Nivel de Completitud de desarrollo.
Cometario: Este concepto no se encuentra especificado hasta el momento en ninguno de los estándares consultados.
*Indicador Global***:** es un indicador que deriva de otros indicadores para evaluar o estimar un concepto calculable. *Relaciones:* Un indicador global se puede componer en base a otros indicadores relacionados y Un indicador global es modelado por un modelo global.
Ejemplos: Calidad y Costo.
Cometario: Este concepto no se encuentra especificado hasta el momento en ninguno de los estándares consultados, si está introducido en tesis doctoral de Olsina L. [11].

A continuación se muestran las siguientes tablas con los ejemplos correspondientes:

Tabla 1. Diccionario de Términos de la Ontología de la Medición de la Calidad del Dato.

Concepto	Descripción
Atributos	**Unapropiedad mensurable, física o abstract, de una entidad.**
Concepto Calculable	**Relación abstracta entre atributos de entidades y necesidades de información**

Cálculo	**Actividad que usa una definición de indicador para producir un valor de indicador**
Métricas Directas	**Es una métrica de un atributo que no depende de una métrica de otro atributo**
Indicador Elemental	**No depende de otros indicadores para evaluar o estimar un concepto calculable.**
Entidad	**Un objeto que va a ser categorizado mediante la medición de sus atributos.**
Función	**Es un algoritmo o formula que especifica cómo combinar dos o más métricas.**
Modelo Global	**Es un algoritmo o función con criterio de decisión asociados que modela un indicador global.**
Medición	**Actividad que utiliza la definición de una métrica para producir un valor medido.**
Método de Medición	**Una secuencia de operaciones lógicas y posibles heurísticas, especificadas para permitir la realización de una descripción de métrica por parte de una actividad de medición.**

Tabla 2. Tabla de Relaciones de la Ontología de la Medición de la Calidad del Dato.

Nombre	Descripción
Relacionada con	**Uno o más atributos medibles se asocian con uno o más entidades.**
Automatizada por	**Métodos que utiliza varias o ninguna herramienta de software.**
Asociaciones	**Un concepto calculable asocia (combina) uno o más atributos medibles.**
Contenido	**Una métrica o un indicador contienen una escala específica.**
Describe	**Conceptos calculables se definen para satisfacer una necesidad de información correcta. De esta manera, un concepto calculable describe una necesidad de información correcta.**
Cuantifica	**Una o más métricas cuantifican un atributo.**

Tabla 3. Tabla de Atributos de la Ontología de la Medición de la Calidad del Dato.

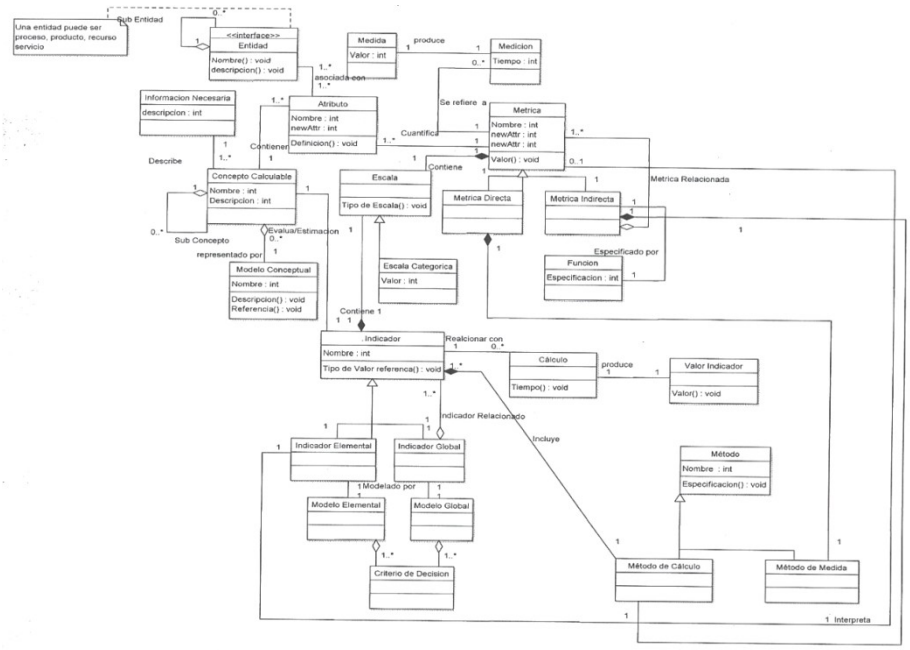
Concepto	Atributos	Descripción
Cálculo	**Intervalo**	**Instante del tiempo en que se realiza el calculo**

Escala Categórica	Valores Permitidos	Lista de Métricas que indican los valores válidos de una escala categórica
Modelo de Concepto	**Nombre**	**Nombre de un modelo de concepto para ser especificado**
	Especificación	**Una representación formal o semiformal de un modelo de concepto**
Criterios de Decisión	**Descripción**	**Una descripción no ambigua del significado del criterio de decisión**
	Rango	**Valores numéricos especificando los límites para un criterio dado.**
Entidad	**Nombre**	**Nombre de una entidad para ser identificada**
	Descripción	**Una descripción no ambigua del significado de la entidad**
Función	**Especificación**	**Una representación formal o semiformal de una función, sinónimo o fórmula.**
Medida	**Valor**	**Resultado numérico o categórico asignado a una métrica, denominado, dato**
Medición	**Precisión**	**Instante del tiempo en ser realizada la medición**
Método	**Nombre**	**Nombre de un método para ser identificado**
	Especificación	**Una descripción formal o semiformal de un método**
	Nombre	**Nombre de una métrica para ser interpretado**
	Valor de Interpretación	**Una declaración textual no ambigua para ayudar a entender el significado del valor obtenido**

Representación de la Ontología

Luego de haber definido los conceptos y relaciones de la ontología, se indicaron los atributos de cada concepto y se construyó una tabla de atributos, incluyéndolos a la vez al diagrama de clase UML.
En este diagrama se denota la caracterización y el objetivo de la ontología; las medidas del dato; la forma de medir y el o la acción de medir.
Como se puede observar la ontología queda representada como resultado de la etapa de conceptualización, obteniéndose una primera versión de la ontología de la medición de la calidad del dato, que fue semiformalizada utilizando algunas representaciones intermedias, independiente del dominio y de la implementación, como se muestran en las Tablas creadas:
Tabla1: Diccionario de Términos, Tabla 2: Relaciones de medición de la Ontología, Tabla 3: Atributos de la Ontología y por último se muestra el diagrama de UML, completo de la ontología de la medición de la calidad del dato, mostrando todos sus términos, atributos e indicadores.
En este diagrama se denota la caracterización y el objetivo de la ontología; las medidas del dato; la forma de medir y la acción de medir.

Tabla 4. Diagrama de la ontología de la Medición de la Calidad del Dato



Conclusiones

Cabe destacar la importancia que tiene, contar con una ontología para el ámbito de la calidad del Dato en sitios Web ya que provee una base conceptual para compartir, reusar y organizar toda la información relacionada a la calidad del dato. Se piensa especificar la ontología, por medio de la herramienta Protégé, por ser un editor de ontología libre y de código abierto además es un sistema de adquisición de conocimientos.
Esta ontología pretende servir como base conceptual para compartir y organizar información obtenida como resultado de diversos trabajos de investigación, en el área de medición y evaluación de software, de una forma extensible, interoperable y automáticamente procesable.

Agradecimiento

El presente trabajo de investigación fue aprobado y financiado por el Consejo de Investigación Científica y Tecnológica (CICyT) perteneciente a la Universidad

Nacional de La Rioja, según Resolución 052/017, por lo que se agradece el apoyo brindado para poder incursionar en la temática presentada y poder ampliar el campo de la investigación con los alumnos. Está integrado por docentes investigadores (3) y 3 alumnos que cursan el último año de la carrera de Licenciatura en Sistemas de Información.

Referencias

1. Pérez Subía Miguel; Revista de Novas Tecnoloxicas de Galicia, N°7, Febrero 2003, .S.S.N. (edición impresa):1579-7546
2. ISO07. El estándar ISO/IEC 15939
3. Zu98, Zuse, Horst Walter. A Framework of Software Measurement. Gruyte*;* ISBN Web. http://www.w3.org/2001/sw/. 2001.
4. Jaime Alberto Guzmán Luna. M.S. Mauricio López Bonilla, Ing. Ingrid Durley Torres. Scientia et Technica Año XVII, No 50, abril de 2012. Universidad Tecnológica de Pereira. ISSN 0122-1701, Metodología y Métodos para la Construcción de Ontología.
5. Fernandez-Lopez M., Gomez-Perez A., Pazos-Sierra J. (1999), Building a chemical ontology using METHONTOLOGY and the ontology design environment, IEEE Intelligent Systems & their applications 4 (1) 37–46.
6. Muñoz A. Sandia B y Paez G. www.Saber.ula.ve/bistram/123456789/33754/1/5.
7. Género, M; García, F; Olsina, L.; Calero, C; Martın, M. An Ontology for Software Measurement. 15th International Conference on Software Engineering and Knowledge Engineering. ISBN: 1-891706-12-8, pp.307-317. 2003.
8. Cranel Field, S and Purvis M, UML as an Ontology Modelling Language, In Proceedings of the Workshop on Intelligent Information, 16th International Joint Conference of Artificial Intelligence (IJKI-99). 1999.
9. Kogut, P.; Granefield, S.; Hart, L; Dutra, M; Blaclawski, K; Kokar, M. and Smith, UML for Ontology Development. http:/www.sandsoft.com/doc/Ker4. 2003.
10. ISO (International Organization for Standardization)/ IEC Comisión electrotécnica Internacional Comisión. Suiza. 1999.
11. Olsina, Luis. Quantitative Methodology for Evaluation and comparison of Web Site Quality. Doctoral Thesis. Facultad de Ciencias Exactas, Universidad Nacional de La Plata. Argentina. 2000.

Realidad Aumentada orientada al reconocimiento de objetos como herramienta de apoyo para los museos

José R. Peralta[1], Diego A. Brandan[1], Analia N. Herrera Cognetta[2], Fabiola P. Paz[2]

[1] Alumnos de la carrera Licenciatura en Sistemas
[2] Cátedra de Trabajo Final de Sistemas
Facultad de Ingeniería – Universidad Nacional de Jujuy (UNJu)
{jose.rodi12, aymeku, aniherrera012, fabyppaz}@gmail.com

Abstract. La Realidad Aumentada mejora la visión que el usuario tiene del mundo real con información adicional o complementaria superponiendo al entorno real la información que le interesa visualizar [1]. En este trabajo se presenta una aplicación móvil que implementa realidad aumentada orientada al reconocimiento de objetos, sin el uso marcadores, en un ambiente museístico. Esta aplicación permite reconocer los objetos de un museo que están presentes en el recorrido de la exposición. Para esto, realiza un proceso de escaneo de los objetos expuestos, buscando una serie de patrones característicos almacenados previamente en una base de datos, luego, al reconocer este patrón de puntos, muestra un cuadro de información multimedia que flota en la pantalla del dispositivo, al lado del objeto escaneado. Además la aplicación permite al usuario, contar con información complementaria en distintos formatos (audio, video y texto), logrando así una experiencia más enriquecedora en la visita en un museo.

Keywords: Realidad aumentada, reconocimiento de objetos, dispositivos móviles para RA, aplicación RA para museos, información complementaria en formato multimedia.

1. Introducción

La realidad aumentada (RA) es la integración de la información digital con el entorno del usuario en tiempo real. A diferencia de la realidad virtual, que crea un ambiente totalmente artificial, la realidad aumentada utiliza el entorno existente y superpone nueva información sobre el mismo [2].

Según Ronald Azuma [3] para lograrlo se necesita:

- Combinar el mundo real y el mundo virtual
- Ser interactivo en tiempo real
- Registro en el espacio 3 dimensiones

Desde una perspectiva más amplia la RA es una aplicación interactiva que combina la realidad con información sintética tal como imágenes 3D, audios, videos, textos, sensaciones táctiles, en tiempo real y de acuerdo al punto de vista del usuario [4].

Un sistema de realidad aumentada se compone de una cámara, una unidad de cálculo y una pantalla. La cámara captura una imagen, el sistema aumenta el objeto virtual en la parte superior de la imagen y muestra el resultado [5].

Por tal motivo, un sistema de RA necesita un medio para capturar la imagen del mundo real (una cámara de video o una webcam), una máquina capaz de crear imágenes sintéticas y procesar la imagen real añadiendo esta información (un procesador y software específico para esto) y un medio para proyectar la imagen final (una pantalla) [6].

La información virtual, tiene que estar enlazada al mundo real, es decir, un objeto virtual siempre debe aparecer en cierta ubicación relativa al objeto real. La visualización de la escena aumentada (mundo real + sintético) debe hacerse de manera coherente [3].

Hoy en día, la mayoría de las investigaciones que trabajan con RA utilizan imágenes de vídeo en directo, que el sistema procesa digitalmente para añadir gráficos generados por ordenador una vez que el sistema reconoce los marcadores.

Este trabajo utiliza al objeto real como el marcador a reconocer, es decir, se utilizan los puntos característicos de cada objeto para generar RA, los cuales son obtenidos y guardados previamente en una base de datos con ayuda de la librería Vuforia. En otras palabras, la aplicación es capaz de reconocer un objeto real perteneciente a un museo, sin la utilización de marcadores, y en su reemplazo utilizar bases de datos, que contienen puntos y que, en su conjunto, definen al objeto, permitiendo su identificación dentro de un ambiente real.

1.1 Realidad Aumentada sin uso de marcadores

La realidad aumentada sin marcadores es un sistema que compara un grupo de puntos representativos de cada objeto del ambiente con los ya almacenados previamente en una base de datos. Esta comparación se realiza en forma continua, hasta hallar una coincidencia, se calcula la ubicación y orientación de la cámara respecto del objeto, y luego superpone información digitalizada en la parte superior de la imagen y la muestra en la pantalla del dispositivo móvil. A continuación en las Fig.1.1 y Fig.1.2 se muestra el escaneo de un objeto para determinar sus puntos característicos.

Fig. 1.1.Escaneo de un objeto real

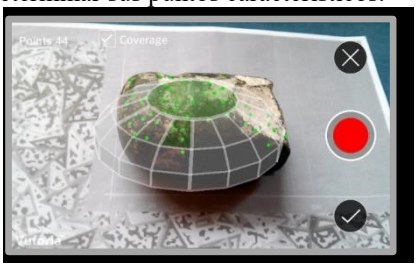


Fig. 1.2. Determinación puntos característicos del objeto

El truco en realidad aumentada es el uso de una cámara virtual idéntica a la cámara real del sistema. De esta manera los objetos virtuales de la escena se proyectan en la misma forma que los objetos reales y el resultado es convincente. Para ser capaz de imitar la cámara real, el sistema necesita conocer las características ópticas de la cámara.

De acuerdo a lo descripto anteriormente y aprovechando las posibilidades que nos ofrece RA, este trabajo se basa en el desarrollo de una aplicación de realidad aumentada de reconocimiento de objetos reales, sin el uso de marcadores, dentro de un museo, para incrementar la información sobre los objetos expuestos, a través de la superposición de información adicional, ya sea en forma de imágenes, videos y/o audios. Con esta propuesta el usuario podrá interactuar y lograr una mejor experiencia durante su recorrido dentro de un museo.

2. Realidad Aumentada en Museos

En los años recientes se pudo observar que la realidad aumentada obtuvo de las otras disciplinas un interés cada vez mayor. La capacidad de insertar información complementaria superpuesta en el espacio real, ha convertido a esta nueva tecnología en uno de los recursos museográficos más vanguardistas, gracias a que favorece tanto la interacción entre los visitantes y el objeto cultural de una forma atractiva y, a la vez, didáctica, cumpliendo con la principal función de estos espacios, la difusión de contenidos culturales. Esta nueva tecnología comenzó a ser utilizada por los museos como una herramienta para ofrecer al visitante un ambiente más amigable que le permita una mayor inmersión, generando la posibilidad de que interactúen directamente con los objetos expuestos de una forma atractiva y a la vez didáctica [67].

Aprovechando las capacidades que proporciona la RA este trabajo pretende brindar a los museos de la región una herramienta poderosa que les permita captar la atención del público, más aún de la población joven, a la que generalmente solo le atrae lo relacionado a nuevas tecnologías (Smartphones) y redes sociales. Permitiendo a los visitantes la posibilidad de obtener mayor información sobre los objetos en exposición de forma interactiva, formativa y novedosa.

3. Herramientas seleccionas para la aplicación ObjetAR

El reconocimiento de objetos complejos requiere de dispositivos con un alto nivel de procesamiento y una licencia adecuada para realizarlo. Actualmente existen pocas librerías que lo soportan y a un costo elevado, a excepción de algunas.

En base a los estudios realizados, se determinó que la librería que mejor se adapta a este trabajo es Vuforia SDK debido a que posee una licencia libre, además de ser un sistema flexible, diseñado para funcionar bien en los dispositivos móviles, compatible con IOS y Android [7]. Cuenta con una gran comunidad de desarrolladores y además proporciona una extensión para trabajar con Unity 3D que posee un poderoso motor de renderizado

Se utilizaron las herramientas de Vuforia y Unity3D. A continuación en la Fig.1.3 se visualiza el funcionamiento interno de lo realizado en este trabajo.

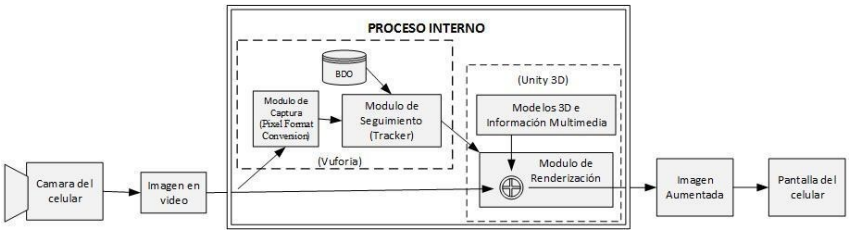


Fig.1.3- Funcionamiento interno de la aplicación ObjetAR

A través de la cámara del dispositivo se capta una escena (un video en vivo) que es procesada internamente por el móvil. En el caso presentado, es la imagen captada durante el recorrido de una exposición, cuyo proceso se detalla a continuación.

1. El SDK de Vuforia a través de su módulo de captura (Pixel Format Conversion) convierte el formato de la cámara a un formato apto para ser renderizado y rastreado internamente. Junto con la conversión se produce un incremento de la resolución con el fin de disponer de la imagen capturada en diferentes resoluciones, para ser correctamente tratada por el Módulo de Seguimiento o Tracker.
2. Vuforia SDK analiza la imagen a través del Tracker y busca coincidencias en la base de datos, la cual está compuesta por Targets (objetos a ser reconocidos).
3. Esta base de datos (BDO) es obtenida a partir de un proceso de escaneo de los objetos que se van a detectar, utilizando un aplicativo provisto por la librería vuforia llamado Vuforio Object Scanner, el cual capta los puntos característicos (vértices) representativos de los mismos.
4. Esta comparación se realiza constantemente y en tiempo real a través del método (OnTrackableStateChanged) y una vez detectado un objeto se utiliza el método OnTrackingFound para indicarle al Módulo de Renderizado que debe “aumentar”, según la posición y orientación de la cámara. Cuando la cámara pierde de vista al objeto, se llama al método OnTrackingLost, para señalar que se desactivan los elementos aumentados. Así mismo estos dos métodos son los que Vuforia permite modificar, a fin de realizar la programación de cómo se realizará el aumento, en este caso para poder mostrar información multimedia, cada uno con su respectiva lógica.

5. El módulo de renderización o interpretación combina la imagen original y los componentes virtuales y luego muestra en pantalla la imagen aumentada sobre la realidad captada por video.
6. Este proceso es llevado a cabo gracias a la integración de Vuforia con Unity 3D, el cual contiene un poderoso motor de renderizado. Esto permite personalizar la forma en que se muestra la información multimedia.
7. Luego del proceso de renderizado, se obtiene la imagen aumentada que es exhibida por la pantalla del dispositivo.

3.1 Generación de bases de datos para el reconocimiento de objetos reales

Al realizar el escaneo de los objetos para determinar sus puntos representativos, se tuvieron en cuenta las texturas presentes en los mismos, los colores monocromáticos, opacos y con pocos rasgos en la superficie de los objetos exhibidos en el museo. Debido a la complejidad de las características propias del objeto, se hacía difícil obtener una única base de datos para un determinado objeto. Es decir, uno de los principales problemas fue el tipo de objeto con el que se trabajaba, ya que se generaban distintas bases de datos, dependiendo de la iluminación que tuviera durante el escaneo, principalmente debido a las sombras proyectadas sobre la superficie del objeto, que se interpretaban como puntos. En la práctica, esto ocasionaba un reconocimiento erróneo, sobre todo si la iluminación durante el reconocimiento era diferente a la presente durante el proceso de escaneo.

Además, en este trabajo, se usaron dispositivos con características inferiores a la capacidad aconsejada por la librería Vuforia. Esto es, procesador de 2.5 GHz 32bits Quad-Core, memoria de 2Gb LPDDR3 y un cámara de 16 Megapixeles. Estas características responden a dispositivos de alta gama, inaccesibles para el equipo de desarrollo. Por lo que el reconocimiento erróneo se vio potenciado por el dispositivo utilizado en el desarrollo y pruebas del trabajo, ya que este reunía sólo la mitad de las características aconsejadas.

Con el fin de superar los inconvenientes citados, durante el proceso de escaneo se decidió obtener una iluminación total de objeto como se muestra en la Fig.1.4 Agregando puntos de luz alrededor del objeto para obtener una mejor iluminación del mismo y captar así todos sus puntos representativos. Se obtuvo de esta manera una base de datos de objeto más completa, que asegura un reconocimiento más eficiente del objeto durante el tracking. En el caso de los museos la posición e iluminación de los objetos se mantiene estática, por lo que resulta más práctico realizar el escaneo en el mismo lugar para favorecer el tracking.

Fig.1.4- Iluminación de un objeto durante el escaneo

Otro inconveniente fue el volumen de las bases de datos, en la práctica, resulto desfavorable para el rendimiento de la aplicación debido a la capacidad de computo con la que se contaba. Al ser de gran tamaño, eran demasiados los puntos a comparar, produciendo un retardo de al menos 5 segundos en su reconocimiento, lo que resultaba inaceptable.

Para salvar este inconveniente y teniendo en cuenta que los objetos a reconocer forman parte de una exposición, es decir, que su posición es fija, no era necesario tener una captación completa del objeto. Por lo que se decidió restringir el área de reconocimiento de cada objeto, considerando solo la superficie que es visible durante una exposición. Con esto se logró reducir considerablemente el tiempo de reconocimiento de los objetos.

4 Aplicación ObjetAR para el museo

Para hacer uso de la aplicación ObjetAR el usuario visitante deberá contar con un dispositivo móvil que contenga android 4.2 o superior, un procesador de 1.2 GHz 32bits Dual-Core, memoria de 1 Gb LPDDR2 y una cámara de 8 megapixeles en adelante y tener descargada e instalada previamente la aplicación antes de su visita al museo. Cuando el usuario abre la aplicación, deberá tomar el dispositivo en dirección al objeto de exposición para su escaneo y en milésimas de segundos reconocerá el patrón de acuerdo a la forma del objeto y mostrará la información multimedia asociada a dicho objeto.

Uno de los objetivos principales de la aplicación ObjetAR es que el usuario tenga información por medio del dispositivo de manera más agradable y divertida. Para ello se trabajó con la *reconstrucción objetos* sobre pequeños fragmentos, estos fragmentos son los que se hallaron y lograron conservar y que se exponen en el museo. También *Información multimedia en diversos formatos,* con acceso a ella mediante la *reproducción manual y automática* dependiendo del objeto en exposición. Por último se incorporaron *botones virtuales,* para manejar la información multimedia asociada a los objetos.

Cuando el usuario abre la aplicación, se le presenta un menú como se observa en Fig.1.5, el que le permite seleccionar la forma en que desea ver información del objeto. Estas opciones son:

- Reproducción automática de acceso a información multimedia.
- Reconstrucción de objetos históricos.
- Reproducción manual de acceso a información multimedia.
- Botones virtuales de acceso a información multimedia.

Fig.1.5- Menú principal de la aplicación ObjetAR

Información multimedia:

Las Fig.1.6 y Fig.1.7 muestran la información complementaria que se le asigna a cada objeto en exhibición dentro de un museo. Tal información puede estar en formato de audio, vídeo, textos y/o imágenes (fotos, 3D). Esta forma de acceso a la información, aporta entretenimiento y atracción a la hora de realizar un recorrido por el museo.

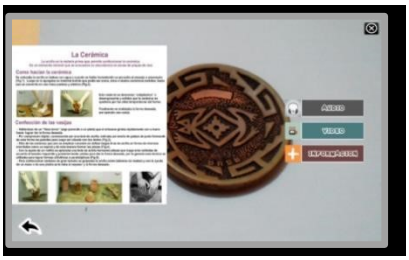


Fig. 1.6- Información multimedia en formato de texto de un objeto

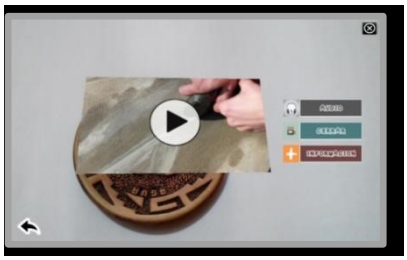


Fig. 1.7- Información multimedia en formato de video de un objeto

4.1 Reproducción manual de acceso a información multimedia

Cada objeto expuesto en el museo tiene asociada información en diversos formatos, lo que provee al usuario la posibilidad de elegir el modo de acceder a la misma, por ejemplo, reproduciendo, pausando un audio, ampliando un video o imagen, entre otras opciones.

En la Fig.1.8 se observa un menú que se activa al reconocer el objeto, en él existe una gama de opciones que posibilitan la consulta de información referente al objeto en distintos formatos.

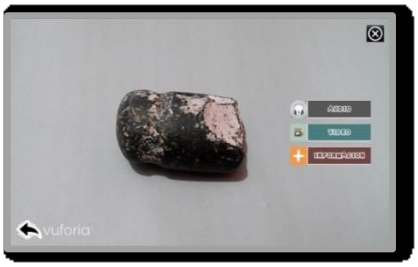


Fig.1.8- Menú de opciones de un objeto reconocido

A continuación en las Fig.1.9, Fig.1.10 y Fig.1.11, se muestra lo descripto anteriormente, para uno de los objetos reales con cada uno de los botones de acceso a información multimedia (audio y video) y de información (textos y fotos) respectivamente (Aplicación ObjetAR).

Fig.1.9- Botón manual a "Audio" de un objeto

Fig.1.10- Botón manual a "Video" de un objeto

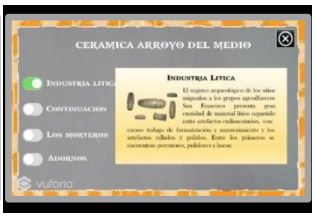


Fig.1.11- Botón manual a "Información" de un objeto

4.2 Reproducción automática de acceso a información multimedia

Como se trata de objetos en exposición, para mostrar información se jugó con diferentes perfiles y/o vistas del objeto, que se generan de acuerdo a su posición y lugar dentro del museo. En cada una de estas vistas se insertó información en un determinado formato, que se reproducen automáticamente cuando el dispositivo enfoca dicha vista del objeto.

Como puede verse en las Fig.1.12, Fig.1.13 y Fig.1.14 al reconocer el objeto y dependiendo del perfil del objeto, se ejecuta un medio para mostrar información. Cuando el usuario mueve el dispositivo, perdiendo de vista la cara del objeto, automáticamente se termina la ejecución del aumento, desapareciendo la información de la pantalla.

Fig.1.12- Reproducción automática "video" del objeto

Fig.1.13- Reproducción automática de "Información".

Fig.1.14- Reproducción de automática de "audio"

4.3 Reconstrucción de objetos históricos

Esta opción fue pensada para trabajar con objetos incompletos o fragmentos de ellos. La idea es que al reconocer este fragmento se muestre un modelo 3D representativo de cómo era la pieza completa y así mismo poder interactuar con ella rotando y escalándola con los dedos de la mano. A continuación en las Fig.1.15, Fig.1.16 y Fig.1.17, siguientes se muestran algunos ejemplos de objetos desarrollado para la aplicación ObjetAR.

Fig.1.15- Reconstrucción de la cerámica de la cultura San Francisco

Fig.1.16- Reconstrucción de la cerámica del complejo alfarero, Arroyo del Medio

Fig.1.17- Vista de la reconstrucción de objetos en modelo 3D de varias cerámicas.

4.4 Botones Virtuales

La idea de esta opción fue la de mostrar información en forma novedosa e interactiva, con el uso de los botones virtuales como se puede observar en la Fig. 1.18 y 1.19. Esto consiste en definir un sector del objeto a reconocer y usarlo como botón,

de manera que cuando la mano se interponga entre el objeto y la visión de la cámara, se ejecute cierta acción. Para este caso, una animación que desplaza una serie de pantallas informativas del objeto reconocido.

Fig. 1.18- Objeto en exposición a ser escaneado

Fig. 1.19- Información multimedia haciendo uso de los botones virtuales

Las pruebas fueron realizadas para el museo Pablo Balduin de la localidad de San Pedro, provincia de Jujuy. Los responsables del museo participaron con mucho interés permitiendo trabajar con los objetos reales allí expuestos, con resultados muy satisfactorios.

5 Conclusiones

Se tuvieron en cuenta requerimientos funcionales de la librería y las capacidades de cómputo del dispositivo móvil utilizado para la realización de este trabajo (menor al aconsejado), se realizaron, en consecuencia, ciertas restricciones en el desarrollo, para lograr un correcto funcionamiento de la aplicación. Por una parte, con el fin de obtener una base de datos de objeto más liviana, se restringió el área de reconocimiento de cada objeto, escaneando solo la superficie que es visible durante una exposición. Por otra parte, para eliminar los retardos existentes en el pasaje de una escena a otra, se decidió estructurar la aplicación sobre una misma escena. Si bien la aplicación fue desarrollada bajo un conjunto de restricciones, se deduce que, se potenciaría aún más su buen funcionamiento si se montara sobre un dispositivo que cuente con el hardware recomendado. Ya que, a modo de prueba, se pudo instalar y ejecutar el prototipo en dispositivos de media y alta gama como son el Samsung J7 y el Motorola Moto Z Play, obteniendo un funcionamiento óptimo en cada caso.

El hecho más relevante, si se quiere, fue trabajar **sin marcadores**. Esto requiere de un esfuerzo extra para generar un patrón de reconocimiento, ya que cuando se trabaja con marcadores, existen conjuntos de patrones preestablecidos. Para reconocer un objeto se generó entonces una base de datos, con puntos característicos obtenidos a

través de un proceso de escaneo. Además para realizarlo, fue necesario tener en cuenta la iluminación existente en el ambiente, al momento del escaneo, del reconocimiento y también la textura del objeto. Todo este trabajo trajo consigo la ventaja de lograr una mayor inmersión en el uso de la realidad aumentada, al no alterar el ambiente donde se la utiliza. Esto lo hace ideal para su aplicación en múltiples ambientes, donde el uso de marcadores no sea aconsejable como es el caso de los museos. Para finalizar, en este trabajo se demuestra la posibilidad de desarrollar una aplicación capaz de enriquecer la visión que se tiene de un objeto, a través de información complementaria, disponible en diversos formatos multimedia para ambientes museísticos.

References

1. Williams Parsons, Siân Jones, Xiaosong Zhang, Jimmy Cheng-Ho Lin, Rebecca J. Leary, et al. (2008). An Integrated Genomic Analysis of Human Glioblastoma Multiforme. the American Association for the Advancement of Science, Washington, DC.
2. Whatistechtarget. (2016). AugmentedReality (AR). Recuperado el 20 de agosto de 2016, de http://whatis.techtarget.com/definition/augmented-reality-AR
3. Azuma Ronald. (2001). Augmented Reality: Approaches and Technical Challenges. Book chapter in Fundamentals of Wearable Computers and Augmented Reality, Woodrow Barfield and Thomas Caudell. Editors: Lawrence Erlbaum Associates, ISBN 0-8058-2901-6. Chapter 2, pp. 27-63.
4. María José Abásolo, Alejandro Mitaritonna, Javie Giacomantone, Armando De Giusti, Marcelo Naiouf et al. (2013). Realidad Virtual, Realidad Aumentada y TVDi. Facultad de informática UNLP.La Plata, Buenos Aires, Argentina.
5. SanniSiltanen. (2012). Theory and applications of marker-based augmented reality. Sanni Siltanen. Espoo 2012. VTT Science 3. 198 p. + app. 43 p.
6. Andrea Gallego & Leandro Aguilar. (2013). Realidad Aumentada en el Museo. Tesis de Licenciatura en Informática. Universidad Nacional de la Plata. Buenos Aires, Argentina.
7. Vuforia Developer Library. (2016). Library. Recuperado el 3 de septiembre de 2016, dehttps://developer.vuforia.com/

Plataforma como servicio – PaaS para la gestión de tecnologías en el desarrollo y despliegue de aplicaciones web

Fredy H. Vera-Rivera[1,2], Carlos Mauricio Gaona Cuevas [1]

[1] Universidad del Valle, Escuela de Ingeniería de Sistemas y Computación,
Ciudad Universitaria Meléndez, Calle 13 # 100-00, Cali - Colombia{fredy.vera, mauricio.gaona}@correounivalle.edu.co

[2] Universidad Francisco de Paula Santander, Departamento de Sistemas e Informática,
Avenida Gran Colombia No. 12E-96 Barrio Colsag, Cúcuta Colombia fredyhumbertovera@ufps.edu.co

Abstract. Las Plataformas como Servicio (PaaS) permiten el uso de herramientas de desarrollo a través de internet, accediendo en cualquier momento, desde cualquier lugar y sin necesidad de configurarlas localmente, permitiendo reducir la complejidad de desplegar y desarrollar aplicaciones web. Se construyó Sandbox – UFPS, una PaaS para el desarrollo y despliegue de aplicaciones web. Se realizó un estudio de las características de las PaaS y se identificaron los proveedores más importantes del mercado, luego se describió la plataforma Sandbox – UFPS y se resaltaron sus beneficios, entre los cuales se encontraron que reduce la complejidad de las tareas de despliegue y configuración y las automatiza para los desarrolladores. Como parte de la PaaS se implementa una biblioteca de componentes reutilizables que permiten agilizar la construcción de las aplicaciones web y promover la reutilización. Se propone un modelo de nube híbrida para mejorar la disponibilidad, el rendimiento, el respaldo y la tolerancia a fallos.

Keywords: Plataforma como servicio (PaaS), Sandbox-UFPS, desarrollo web, biblioteca basada en componentes, nube híbrida.

1 Introducción

El desarrollo de aplicaciones web ha tenido un crecimiento importante en los últimos años siendo estas usadas en muchas áreas del conocimiento al igual que se ha incrementado la presencia en la web de las empresas para ofrecer sus productos y servicios a través de Internet. Por lo tanto, es imprescindible que los desarrolladores de software cuenten con metodologías, herramientas, frameworks y componentes reutilizables que faciliten esa construcción de aplicaciones de calidad que entreguen al cliente valor y beneficios en el menor tiempo posible.

Las herramientas y frameworks que requieren los desarrolladores de software pueden ser provistas por una plataforma como servicio - PaaS - la cual ofrece sistemas operativos para su uso, ya configurados e instalados, frameworks de aplicaciones, APIs (ApplicationProgramming Interface) y herramientas de desarrollo, a menudo adaptados a las necesidades de los desarrolladores.

En una primera versión se diseñó e implementó la plataforma Sandbox-UFPS para apoyar a los estudiantes de Ingeniería de Sistemas, de una universidad pública colombiana, que no cuentan con los recursos suficientes para adquirir una cuenta de hosting o un servidor en la nube para realizar el desarrollo, despliegue y pruebas de sus proyectos de software. Desde sus inicios el número de aplicaciones alojadas en Sandbox-UFPS se ha incrementado en un 58%, pasando de 129 a 203 aplicaciones, con una tendencia de uso que aumenta año a año.

Dada la acogida que ha tenido la plataforma, los cambios frecuentes que se dan en las tecnologías de la información y la comunicación y los problemas que han manifestado los usuarios con relación a la disponibilidad y operatividad de la plataforma se hace necesario realizar una mejora y actualización en los servicios y herramientas que se encuentran en la plataforma.

Para mejorar el funcionamiento y disponibilidad de la plataforma SandBox-UFPS se realizó, como primera medida, un análisis de las características de las plataformas como servicios - PaaS - identificando a los proveedores más importantes del mercado, a continuación se introduce la implementación de la plataforma Sandbox - UFPS para agilizar la construcción de aplicaciones web por medio de la implementación de una biblioteca de componentes reutilizables y la mejora en la disponibilidad, el rendimiento, el respaldo y la tolerancia a fallos con la propuesta de un modelo de nube híbrida, culminando con la presentación de resultados y conclusiones.

2 Metodología del trabajo de investigación

El presente trabajo de investigación se basa principalmente en los conceptos de la investigación tecnológica aplicada, en la cual se parte de un problema que se requiere mejorar e intervenir, en este caso es mejorar el funcionamiento y la disponibilidad de la plataforma Sandbox-UFPS, se selecciona una teoría bajo la cual se puede resolver ese problema (cloud computing, metodologías ágiles, desarrollo basado en componentes); se deriva de esta teoría un sistema de acciones y de previsiones (prototipo). Por último se ensaya y se prueba el prototipo con el fin de probar que se resuelve el problema planteado. En la figura 1 se presenta el modelo de la investigación realizada.

Para el desarrollo de esta investigación se partió de la fundamentación teórica para establecer los conceptos, definiciones y bases del trabajo de investigación, se realizó un diagnóstico del problema planteado identificando sus causas y consecuencias; posteriormente, se realizó un estudio de las plataformas de desarrollo que se encuentran en el mercado, se identificaron las características importantes de ellas y se plantearon las mejoras sobre la plataforma Sandbox-UFPS. Por último se implementaron las funcionalidades, herramientas y servicios identificados en la fase anterior.

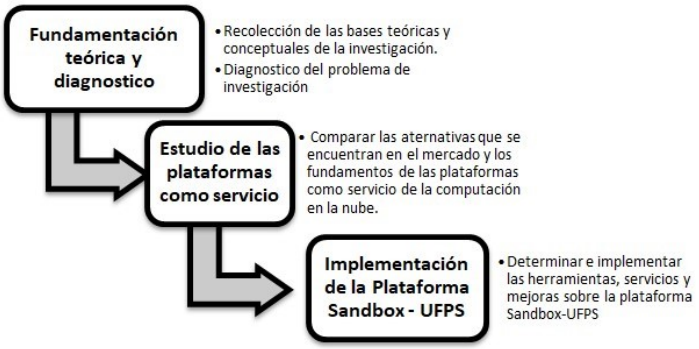


Fig. 1. Metodología del trabajo de investigación.

3 Fundamentación teórica y diagnóstico

Los sistemas web son cada vez más complejos y sofisticados, por tanto su desarrollo conlleva asociada una complejidad en la cual se deben tener en cuenta atributos de calidad como son: la disponibilidad, el respaldo, la eficiencia, el nivel de servicio y la tolerancia a fallos. Para administrar y mejorar estos atributos la computación en la nube ofrece un conjunto de servicios que ayudan al desarrollador a minimizar esta complejidad y mejorar la calidad del servicio.

Para que un proyecto de desarrollo de aplicaciones web sea exitoso se deben tener en cuenta, entre otros, los siguientes aspectos importantes:

1. Uso de una metodología de desarrollo de software adecuada. La tendencia es usar metodologías ágiles de desarrollo y prácticas ágiles que ayuden a la planificación, gestión y control del proyecto de desarrollo.
2. Gestión del trabajo colaborativo. Es importante la gestión del código fuente, de las versiones de desarrollo, del trabajo que adelanta cada miembro del equipo.
3. Manejo de los lenguajes de programación web. Los lenguajes base son HTML5, CSS3, JavaScript, junto con un lenguaje del lado del servidor que permita la construcción del “backend” o lógica del negocio de la aplicación.

4. Manejo de bases de datos relacionales y no relacionales. Depende del proyecto usar una u otra o en ocasiones la combinación de ambas. Al igual que la gestión, configuración y administración del servidor de base de datos.
5. Gestión y automatización de pruebas. Existen varias herramientas y aplicaciones que permiten facilitar y automatizar este proceso.
6. Gestión del despliegue y calidad del servicio. Una aplicación web de calidad debe garantizar una alta disponibilidad, ser tolerante a fallos y tener políticas de almacenamiento y respaldo de la información, de tal forma que su funcionamiento se garantice siempre.
7. Reutilización de código, de componentes y/o de funcionalidades.

Una vez planteado el punto de partida de este trabajo, el cual se centra en el desarrollo web y en la gestión de herramientas, lenguajes y servidores para el despliegue de una forma rápida y transparente de esas aplicaciones web. Ahora se plantea el problema del presente trabajo de investigación.

El ambiente donde se ubica este trabajo de investigación es el académico, en una Universidad, en un programa de Ingeniería de Sistemas, en el cual los estudiantes y docentes de ese programa tienen la necesidad de realizar proyectos de desarrollo de aplicaciones web ya sea como proyectos de clase para diferentes materias, prácticas profesionales y empresariales, proyectos de grado o de finalización de semestre, proyectos de investigación y de extensión, que pretenden resolver problemas de la región. Con la desventaja que la mayoría de estudiantes al ser de estratos socioeconómicos bajos no tienen la capacidad de adquirir una cuenta de hosting o un servidor en la nube para realizar el desarrollo, el despliegue y las pruebas de sus proyectos. Por este motivo surge el proyecto para facilitar a los estudiantes y docentes una plataforma donde puedan construir y desplegar sus proyectos.

Por lo enunciado surge la pregunta ¿Dónde se distribuye la aplicación? Se planteó una aplicación Web que permitiera automatizar las labores de despliegue y configuración remotamente, este fue el primer paso para la construcción de la plataforma SandboxUFPS. Sandbox-UFPS es una plataforma de desarrollo en la nube, cumple con las características para ser considerada una Plataforma como Servicio, en la cual los usuarios no deben instalar nada en sus computadores, las herramientas las pueden usar desde la plataforma, pueden tener acceso a través de internet desde cualquier lugar, a cualquier momento y desde cualquier dispositivo [1]. Aunque está lejos de los líderes en el mercado internacional de plataformas como servicio, tales como force.com, Google ApplicationEngine, Microsoft Azure y Amazon Web Services, sin embargo, se convierte en una alternativa viable para nuestro medio.

4 Plataformas como Servicios - PaaS

Las plataformas como servicio hacen parte de los niveles de servicio que ofrece la computación en la nube, en primer lugar se repasa el concepto de computación en la nube, George Reese la define como: "La Nube no es simplemente una manera más elegante de describir a la Internet. Si bien la Internet es un fundamento clave para la Nube, la Nube es algo más que la Internet. La Nube es donde vas para usar la tecnología cuando la necesitas, por el tiempo que lo necesites, y ni un minuto más. No instalas nada localmente, y no pagas por la tecnología cuando no la estás utilizando" [2].

Las plataformas como servicio (PaaS), corresponden a todos aquellos componentes orientados al desarrollo y despliegue de aplicaciones sobre la Nube, en la cual los usuarios no instalan ninguna herramienta localmente para realizar sus desarrollos. Está enfocada a los desarrolladores de software y de aplicaciones web o móvil en la nube. PaaS está construida sobre la capa de infraestructura como servicio (IaaS), ofrece sistemas operativos para su uso, ya configurados e instalados, frameworks de aplicaciones y API (ApplicationProgramming Interface) para el desarrollo de software. Con PaaS se pretende minimizar la complejidad del despliegue de aplicaciones sobre máquinas virtuales [3]. A continuación se detallan las características fundamentales de las plataformas como servicios resumidas de [4], [5], [6], [7], [8]:

- Suministran los medios necesarios para soportar el ciclo de vida de construcción y entrega de servicios, aplicaciones Web y móviles sobre la nube.
- Comprende por lo menos tres elementos distintos: herramientas de desarrollo, una plataforma virtual de ejecución de aplicaciones y un componente de almacenamiento tipo FaaS – File as a Service.
- PaaS proporciona un conjunto de herramientas y recursos, a menudo adaptados a las necesidades del usuario, un conjunto de aplicaciones. Este modelo no requiere la compra de hardware o instalación de software - todos los programas necesarios se encuentran en el proveedor. El cliente de su lado tiene acceso a la interfaz a través de Internet y un cliente ligero.
- Se basa en una arquitectura Multitenant, en la cual se ofrecen recursos técnicos, instancias de código y herramientas de desarrollo comunes para múltiples clientes.
- Permite Interfaces de usuario configurables, soporta la creación de interfaces de usuario flexibles sin la necesidad de escribir código complejo.
- Los proveedores de PaaS deben suministrar lenguajes de programación para el desarrollo de aplicaciones, almacenamiento y recuperación de datos y sistemas operativos donde la aplicación pueda funcionar.
- Las tareas de administración, configuración y actualización de los servidores de aplicación y de la infraestructura que los soporta son transparentes para el usuario, es decir el desarrollador no se debe preocupar por estas tareas. Se preocupa por su creación y por el despliegue de lo demás se encarga el proveedor del servicio.

- Los proveedores de PaaS deben suministrar opciones flexibles de precios y pago por uso.
- Permite la configuración y mantenimiento ilimitado de bases de datos, provee la facilidad de crear, modificar o extender el modelo de datos a través de unos cuantos clics.
- Tiene capacidades para definir fácilmente procesos de flujo de trabajo y especificar reglas empresariales. Estas capacidades están automatizadas y son fáciles de usar.
- Control multinivel de seguridad para compartir funcionalidades y recursos dentro de las aplicaciones y componentes de la plataforma.
- Contiene una librería integrada de contenidos. Elementos comunes que amplían el conjunto de funcionalidades principales de la aplicación, mejoran el intercambio de información y aceleran el tiempo de desarrollo.
- Tiene un modelo flexible de integración de servicios. Permite una integración perfecta de la aplicación y las funcionalidades de la nube a través de un modelo flexible de integración de servicios web.
- Permite analíticas. Capacidad para aprovechar los datos agregados entre empresas y aplicaciones para análisis.
- Permite editar las características del sistema operativo y actualizar fácilmente.
- Sirve como un entorno de alojamiento, pruebas, implementación y desarrollo de aplicaciones.

Las PaaS se basan en plataformas de desarrollo de aplicaciones que permiten el uso de recursos externos para crear y alojar aplicaciones. Algunos ejemplos de PaaS son tomados de [3], [9], [10], [11].

- **OpenShift:** desarrollada por Red Hat, soporta Ruby, Java, PHP, Node.js, Python, Perl, entre otros.
- **CloudBees:** plataforma para crear, implementar y administrar aplicaciones Java. Permite el manejo de DevOps a nivel empresarial.
- **Engine Yard:** plataforma para construir e implementar aplicaciones Ruby y PHP que se pueden ampliar con complementos.
- **Google App Engine:** plataforma para desarrollar y ejecutar aplicaciones Java, Python y Go en la infraestructura de Google. Permite el uso de las Api de los servicios de google como son Gmail, Google Drive, Google Maps, Calendar, entre otros. [12], [13].
- **Heroku:** plataforma para implementar aplicaciones Java, Ruby, Python, Clojure, node.js y Scala que se pueden ampliar con recursos complementarios.
- **Microsoft Windows Azure:** servicios de computación y almacenamiento bajo demanda, así como una plataforma de desarrollo y despliegue para aplicaciones web, móviles y diferentes servicios. [14]

- **Salesforce.com - Force.com:** plataforma para construir y ejecutar aplicaciones y componentes comprados de AppExchange o aplicaciones personalizadas. [12], [15], [16].
- **Amazon Web Services:** permite implementar y escalar servicios y aplicaciones web desarrollados con Java, .Net, PHP, Node.js, Python, Ruby, Go y Docker en servidores como Apache, Nginx, Passenger e Internet information Server (IIS).
  El servicio de Amazon que ofrece PaaS es AWS ElasticBeanstal, entre otros. [17], [8].

Implementación de la Plataforma Sandbox - UFPS

Sandbox - UFPS, es una plataforma de uso académico que permite la administración de servidores de aplicación y de red, donde se integran herramientas necesarias para realizar las prácticas en las diferentes materias del programa de Ingeniería de Sistemas, las cuales involucran el desarrollo de aplicaciones que requieran persistencia en diferentes motores de base datos, así como servidores de aplicaciones y servidores web en tecnologías Java, Python y PHP. Además, cuenta con herramientas de control de versiones y de trabajo en equipo.

Como parte del proyecto, se desarrolló una herramienta software que permite gestionar, administrar y controlar los recursos que ofrece Sandbox - UFPS a los estudiantes [1]. Sandbox – UFPS es una plataforma de desarrollo en la nube que permite la integración y uso por parte de los estudiantes y docentes, de las herramientas necesarias para el desarrollo y despliegue de aplicaciones web. Adicionalmente, los estudiantes continuamente desarrollan algoritmos y productos software que pueden ser reutilizados por otros estudiantes o usuarios de la plataforma; estos componentes software ya desarrollados son documentados en los diferentes proyectos que se realizan en las materias y en el programa de Ingeniería de Sistemas, no son conocidos ni utilizados por los demás estudiantes, por este motivo se creó una biblioteca de componentes software reutilizables que permite catalogar y mostrar la forma de uso de estos componentes. Con el desarrollo de este repositorio es posible tener un conjunto de unidades software o aplicaciones modulares que se puedan ensamblar para formar otros sistemas nuevos más grandes y complejos, en vez de tener que desarrollar el sistema nuevo desde cero. Al poder utilizar estos componentes software, que ya han sido probados y verificados se puede disminuir el tiempo de desarrollo y hacer sistemas informáticos más confiables y seguros. En la tabla 1 se describen las características principales de la plataforma Sandbox-UFPS.

Para mejorar el rendimiento y la disponibilidad se propone un modelo de nube híbrida, el cual combina la infraestructura local con una infraestructura en la nube, para mejorar la disponibilidad, el respaldo y la tolerancia a fallos [18].

Tabla 1. Características de la plataforma Sandbox - UFPS

Característica	Descripción
Lanzamiento inicial:	2013
Lanzamiento última versión:	2017
Lenguajes soportados:	Java, JSP, JEE, PHP, Python.
Bases de datos soportadas:	Mysql, Postgres, MongoDB
Sistemas operativos soportados:	Linux
Control de versiones:	Git, SVN, Gitlab.
Reutilización de componentes software:	Colossal: Biblioteca de componentes software.
Gestor de contenidos:	Joomla
Costo:	Gratuito para estudiantes y docentes

En la figura 2 se puede apreciar la arquitectura de la plataforma Sandbox-UFPS. Esta arquitectura define la forma como están organizados todos los componentes y elementos que hacen parte de Sandbox-UFPS.

Los usuarios principales de la aplicación son los estudiantes y docentes del programa de Ingeniería de Sistemas junto con el administrador de la plataforma quien es el encargado de la gestión y configuración de Sandbox-UFPS.

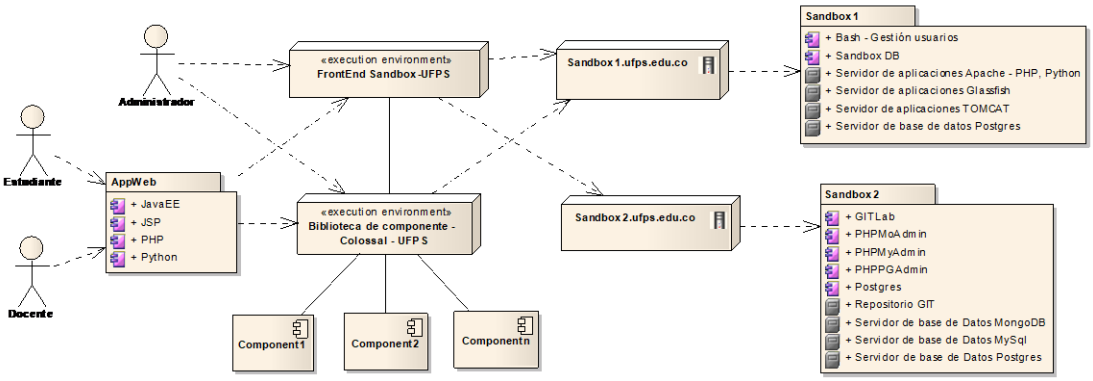


Fig. 1. Arquitectura de la plataforma Sandbox - UFPS

Sandbox-UFPS contiene dos aplicaciones principales: (1) El FrontEnd en el cual los estudiantes y docentes pueden crear sus proyectos de desarrollo, seleccionar las instancias de las tecnologías que van usar en ese proyecto y acceden a las herramientas de desarrollo respectivas de cada tecnología para la construcción de su aplicación. (2) La biblioteca o librería de componentes software reutilizables, la cual es un catálogo de unidades de software disponibles para ser utilizadas en el desarrollo de las aplicaciones web, se puede encontrar diferentes clases de componentes como son: servicios web,

controles personalizados para la interfaz de usuario, paquetes, librerías y Apis. Como trabajo futuro se piensa incluir una herramienta para la gestión del proyecto de desarrollo, que incluya prácticas ágiles.

Estas dos aplicaciones junto con las aplicaciones creadas por los usuarios son desplegadas en dos servidores: sandbox1 y sandbox2. Sandbox1 contiene la instancia de la base de datos de la plataforma junto con los servidores de despliegue de la plataforma, en sandbox2 están las herramientas disponibles para los usuarios como son: PhpMyAdmin, PhpPGAdmin, PhpMOAdmin, los servidores de bases de datos Postgres, Mysql y MongoDb, junto con el repositorio GIT para la gestión de versiones y del código fuente.

Esta infraestructura es replicada en un proveedor de computación en la nube con el fin de mejorar la disponibilidad, el respaldo y la tolerancia a fallos. La nube hibrida es una composición de dos o más clouds (privado, comunitario o público) que permanecen como entidades únicas, pero son enlazadas a la vez por tecnologías estandarizadas que permiten la portabilidad de datos y aplicación, su principal ventaja es la de permitir aumentar la capacidad del cloud privado con los recursos del cloud público para poder mantener niveles de servicio adecuados, frente a rápidas fluctuaciones de carga de trabajo [19]. A continuación se detallan las características de Sandbox – UFPS por medio de las cuales se puede considerar como una PaaS:

- Sirve como un entorno de alojamiento, pruebas, implementación y desarrollo de aplicaciones. Estas tareas se realizan de forma automática y transparente para el usuario.
- Soportar el ciclo de vida de construcción y entrega de servicios para el desarrollo de aplicaciones Web.
- Comprende un conjunto de herramientas de desarrollo, servidores de aplicaciones y recursos, en el cual no se requiere la compra de hardware o instalación de software por parte de estudiantes y docentes. El acceso es a través de internet por medio del FrontEnd de Sandbox-UFPS.
- Suministra un conjunto de lenguajes de programación para el desarrollo de aplicaciones web, almacenamiento y recuperación de datos
- Permite la configuración y mantenimiento de bases de datos, provee la facilidad de crear, modificar o extender el modelo de datos a través de unos cuantos clics.
- Contiene una librería integrada de contenidos llamada Colossal – UFPS, biblioteca o repositorio de componentes software reutilizables.

5 Resultados

Para validar el impacto de la plataforma Sandbox – UFPS se realizó una encuesta a estudiantes y docentes del programa de Ingeniería de Sistemas, la población corresponde a 500 posibles usuarios. Con un nivel de confianza del 95% y un margen de error del 15%, el tamaño de la muestra fueron 42 usuarios elegidos al azar.

Al indagar ¿Para qué cursos usan Sandbox-UFPS? Se pudo observar que los cursos o asignaturas en las cuales más se usa son aquellas a partir de 5to semestre, siendo Análisis y Diseño de Sistemas, Desarrollo Web, Base de Datos, Ingeniería de Software y Seminarios Integradores las que muestran una mayor demanda (83,3%).

Al preguntar ¿Encuentra en Sandbox-UFPS las herramientas de trabajo necesarias para la aplicación desarrollada? Los usuarios manifiestan encontrar siempre o casi siempre con un 70% el conjunto de herramientas necesarias para la aplicación que necesitan desarrollar. En cuanto al número de aplicaciones desarrolladas por los usuarios, el (90,48%) de los encuestados han usado Sandbox-UFPS, un (19,05%) al menos una aplicación, y la mayoría de usuarios ha alojado entre 2 y 4 aplicaciones web (50%) mostrando un uso interesante de la plataforma. En la tabla 2 se puede apreciar el número de aplicaciones alojadas desde el año de su lanzamiento.

Tabla 2. Número de aplicaciones alojadas por año en Sandbox - UFPS

Año	**2013**	**2014**	**2015**	**2016**
Número de aplicaciones	129	156	186	203

Se puede apreciar que la tendencia de uso aumenta con cada año, lo que implica mantener un proceso de mejora continua en los servicios y herramientas que los usuarios encuentran en la plataforma. Se puede apreciar que el 81% usuarios usan conexión a base de datos, siendo este el servicio más usado, el 76% usa el servicio de base de datos MySQL, el 69% conexión al servidor, el 64% de los usuarios usan alojamiento de sus aplicaciones, estos servicios son los que requieren mayor atención por parte de los administradores de la plataforma. Se pudo apreciar que la disponibilidad es el servicio que presenta más problemas (69%) de los usuarios manifestaron este problema, junto con el registro y acceso a Sandbox-UFPS (38%). Por este motivo si hizo la propuesta de un modelo de nube híbrida que mejora la disponibilidad y tolerancia a fallos. Se pudo evidenciar que los estudiantes y docentes tienen un lugar donde desplegar sus aplicaciones junto con un conjunto de herramientas y servicios para sus desarrollos.

6 Conclusiones

- Las plataformas como servicio facilitan el trabajo de los desarrolladores de aplicaciones web porque automatizan muchas tareas complejas a la hora de realizar el despliegue de la aplicación. Estas tareas las lleva a cabo el proveedor del servicio permitiendo al desarrollador concentrarse en la escritura del código y en la lógica del negocio propia de su aplicación.
- Sandbox-UFPS se puede considerar una PaaS enfocada para los estudiantes y docentes del programa de Ingeniería de Sistemas, pero está lejos de ofrecer las características que se encuentran en las PaaS del mercado. Sandbox-UFPS facilita el desarrollo, despliegue y ejecución de las aplicaciones web, reduciendo la complejidad de estas tareas y automatizándolas para el usuario. En la plataforma los docentes pueden de forma fácil y automatizada revisar los desarrollos que hacen sus estudiantes, teniendo control de su aprendizaje.
- El uso de nubes híbridas para mejorar la disponibilidad, el respaldo y la tolerancia a fallos de sistemas informáticos es una alternativa muy interesante ya que permite tener recursos de la computación en la nube combinados con recursos tradicionales, los costos se pueden reducir considerablemente ya que los servicios de respaldo y réplica de la nube sólo se utilizan o se pagan cuando falle la instancia local o también cuando la capacidad de la instancia local se vea rebozada. Esto reduce considerablemente los pagos del servicio en la nube.
- El uso de componentes software reutilizables catalogados en una librería o biblioteca permite agilizar el desarrollo de aplicaciones web y facilitar el aprendizaje de los estudiantes. La biblioteca de componentes juega un papel importante en el uso y en la reutilización de código, es el corazón del desarrollo basado en componentes.

Referencias

1. F. H. Vera-Rivera, B. R. Perez-Gutierrez, and F. J. Torres-Bermudez, “Sandbox UFPS - cloud development platform for server management, creation and deployment of web applications of academic use,” *Res. Comput. Sci.*, vol. 101, pp. 65–75, 2015.
2. G. Reese, *Cloud application architectures*. O’Reilly Media, 2009.
3. Q. Zhang, L. Cheng, and R. Boutaba, “Cloud computing: State-of-the-art and research challenges,” *J. Internet Serv. Appl.*, vol. 1, no. 1, pp. 7–18, 2010.
4. J. Mestas J, “La tendencia XaaS: Todo como Servicio,” 2010. [Online]. Available: http://geekswithblogs.net/gotchas/archive/2010/08/09/la-tendencia-xaas-todo-comoservicio.aspx. [Accessed: 29-Mar-2017].
5. L. Youseff, M. Butrico, and D. Da Silva, “Toward a Unified Ontology of cloud computing held in conjunction with SC08,” in *Grid Computing Environments Workshop (GCE08)*, 2008.

6. Krasis, “Los retos de la escalabilidad y disponibilidad en las aplicaciones,” *MSDN - Microsoft*, 2014. [Online]. Available: https://msdn.microsoft.com/eses/windowsazure/gg318629.aspx. [Accessed: 29-Mar-2017].
7. Microsoft, “Microsoft Azure: Cloud Computing Platform & Services,” 2016. [Online]. Available: https://azure.microsoft.com/en-us/?b=16.24. [Accessed: 05-Aug2016].
8. Amazon Web Services, “Productos y servicios de la nube: Amazon Web Services (AWS),” 2016. [Online]. Available: https://aws.amazon.com/es/products/. [Accessed: 14-Aug-2016].
9. D. T and G. R, “Platform-as-a-Service (PaaS): Model and Security Issues,” *Int. J. Adv. Appl. Sci.*, vol. 4, no. 1, pp. 13–23, 2015.
10. Springer Gabler Verlag, “Cloud Computing: STATE OF THE ART AND SECURITY ISSUES,” *Gabler Wirtschaftslexikon*, vol. 40, no. 2, 2015.
11. D. Grzonka, “The Analysis of OpenStack Cloud Computing Platform : Features and Performance,” *J. telecomunications Inf. Technol.*, vol. 3, pp. 52–57, 2015.
12. Gartner, “Magic Quadrant for Enterprise Application Platform as a Service, Worldwide,” *Gartner*, 2016. [Online]. Available: https://www.gartner.com/doc/reprints?id=1-2C8JHBP&ct=150325&st=sb. [Accessed: 30-Mar-2017].
13. Google, “Google App Engine Documentation | App Engine | Google Cloud Platform,” 2016. [Online]. Available: https://cloud.google.com/appengine/docs. [Accessed: 05Aug-2016].
14. Microsoft - MSDN, “¿Qué es Azure?,” 2015. [Online]. Available: http://msdn.microsoft.com/es-es/library/azure/dd163896.aspx. [Accessed: 05-Aug2016].
15. salesforce.com INC., “Force.com - Create Mobile Apps for Your Business - Salesforce.com,” 2016. [Online]. Available: http://www.salesforce.com/platform/products/force/. [Accessed: 05-Aug-2016].
16. Salesforce.com, “App Cloud: preguntas más frecuentes - Salesforce.com LatinAmerica,” 2017. [Online]. Available: https://www.salesforce.com/mx/platform/faq/. [Accessed: 04-Apr-2017].
17. Amazon Web Services, “Introducción a AWS - Introducción a AWS,” *Amazon web services*, 2016. [Online]. Available: http://docs.aws.amazon.com/es_es/gettingstarted/latest/awsgsg-intro/gsg-awsintro.html. [Accessed: 05-Aug-2016].
18. F. H. Vera-Rivera, B. Perez-Gutierrez, and V. Urbina, “Modelo De Nube Híbrida (Hybrid Cloud) De Infraestructura Como Servicio Para Mejorar El Rendimiento De La Plataforma Sandbox - Ufps,” In *Conferências Iadis Ibero-Americanas Computação Aplicada 2016*, 2016, no. 2, pp. 237–244.
19. S. A. Cabrera, E. M. Abad, H. Danilo Jaramillo, G. A. Poma, and J. C. Verdum, “Incidencia de atributos de calidad de software en el diseño, construcción y despliegue de ambientes arquitectónicos Cloud,” *2015 10th Iber. Conf. Inf. Syst. Technol. Cist. 2015*, no. August 2016, 2015.

Evidencias de la disciplina Informática en las producciones finales de carrera del año 2015

Sonia I. Mariño[1], Pedro L. Alfonzo[1]

[1] Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura
Universidad Nacional del Nordeste, Corrientes, Argentina
simarinio@yahoo.com, plalfonzo@hotmail.com

Resumen. En Informática los estudios empíricos producen información representativa de la realidad y orientada a la toma de decisiones y al mejoramiento de las prácticas que de ellos emergen. La Ingeniería del Software Basada en la Evidencia se basa en leyes naturales, resultados experimentales y fórmulas empíricas para proponer y respaldar soluciones a los problemas presentados. En este trabajo se propone adaptar la Ingeniería del Software Basada en la Evidencia a las producciones de graduación de los estudiantes, aun cuando estas traten otras áreas de conocimiento diferentes de la Informática a fin de resolver problemáticas del entorno demandadas por el gobierno, la industria y la academia. En el documento se contextualiza el dominio de conocimiento, se explica el desarrollo de la metodología propuesta en el caso de estudio, se muestran los resultados obtenidos al analizar las tesinas defendidas por los estudiantes en el año 2015, finalmente se presentan las conclusiones derivadas del presente estudio.

Keywords: Educación Superior, Informática, evidencias en Informática, trabajos de graduación, gestión académica.

1 Introducción

Las universidades interaccionan con el gobierno y la sociedad en el desarrollo de sus regiones, diseñando soluciones que respondan a necesidades sociales con creatividad e innovación. En [1] exponen que "para estudiar la vinculación de la universidad con el entorno en términos interactivos y sin visiones normativistas o prescriptivas, resulta útil entender que el proceso de identificación de demandas forma parte del proceso de construcción de problemas sociales".

Así, en el ámbito universitario es notorio como estas organizaciones a través de sus equipos de docentes, científicos y tecnólogos indagan en torno a los requerimientos del mercado, estableciendo líneas de I+D+i. Definen aquellas prioritarias según distintos indicadores, y así se formulan proyectos pertinentes orientados a innovar en conocimientos, procesos, productos y servicios.

El modelo de la Triple Hélice [2] plantea una evolución en la relación universidad-sociedad, focalizada en procesos de conocimientos y económicos, siendo fundamental

el protagonismo de los sectores universitarios. Este modelo se aplicó ampliamente en dominios donde la Academia interviene en distintos modos.

En [3] se expresa que "la innovación es entendida a partir de la complementación de saberes internos y externos a las firmas". Así, aquellas que tienden sus redes hacia sus contextos presentan mayores posibilidades de transformar sus procesos, productos y resultados. Este concepto es aplicable en los contextos universitarios, entendiendo que, se innova en los espacios de Educación Superior al detectar e incorporar nuevas necesidades del medio, como problemáticas susceptibles de resolver sustentadas en los conocimientos científicos-tecnológicos. Además, aquellas organizaciones comprendidas por los gobiernos y empresas innovan mejorando sus productos y servicios en pos de lograr una sociedad más adaptada y equitativa.

Las universidades como creadoras de conocimiento, incluso aportan a la generación de tecnología a través de la transferencia explicita que realizan o mediante el capital intelectual que de ellas emerge y se incorpora en los gobiernos, las empresas y la sociedad.

En la propuesta que se expone, estos conocimientos se materializan en las soluciones TIC ideadas para cumplimentar un requerimiento académico y lograr la graduación. Se enfatiza que estos productos del intelecto deben resolver problemáticas detectadas por: los estudiantes -potenciales profesionales del Sector de Servicios y Sistemas Informáticos (SSI)-, sus profesores orientadores, los docentes de la asignatura Proyecto Final de Carrera, o desde requerimientos iniciados por el gobierno, la empresa o la sociedad con quienes los actores mencionados interactúan en la búsqueda de un problema objetivo.

Así, en [4] se exponen siete ejes de políticas públicas para entender la relación Estado-Subsector Software en la Argentina. La Ley de Promoción de Software es uno de los factores claves para lograr mejoras en las empresas relacionadas con las TIC ([5], [6]).

Se coincide con [7] que las soluciones crean conocimiento para la innovación. En el contexto de las producciones de graduación, éstas se diseñan para resolver una problemática particular y con ello modifican el proceso que se traduce en mejores prácticas y desarrollos.

Particularmente, en esta presentación se propone adecuar la Ingeniería del Software Basada en la Evidencia (ISBE) [8] para crear conocimiento en torno a las producciones de fin de carrera, y que pueda mejorar la práctica de la disciplina Informática orientada a la solución de problemáticas del contexto.

1.1 Descripción del contexto académico

Trabajo Final de Aplicación (TFA) y Proyecto Final de Carrera (PFC) son sendas asignaturas de los planes de estudios de la carrera Licenciatura en Sistemas de Información Plan 1999 y Plan 2009 ([9], [10]) desde las cuales se generan los proyectos de fin de carrera o tesinas. Una tesina o disertación de grado, siguiendo al Tesauro de la UNESCO consistiría en un diploma universitario de primer nivel.

Su objetivo general es completar la formación académica y profesional de los alumnos, posibilitando la integración y utilización de los conocimientos adquiridos durante sus años de estudio para la resolución de problemas de índole profesional, académico y científico.

Favorece la formación de los futuros graduados de acuerdo a los requerimientos del mundo del trabajo, y constituye el nicho para el diseño y la elaboración de productos tecnológicos en el marco de actividades en que la Universidad es el principal generador.

Una característica diferenciadora y superadora de otras propuestas curriculares vinculadas a la generación de producciones finales de carrera, es que las asignaturas TFA y PFC abordan el diseño del proyecto tecnológico y su producción. Además, durante el cursado se aplica un estricto monitoreo que continúa hasta la defensa del trabajo para lograr un sustancial avance, aun cuando los estudiantes hayan finalizado el cursado anual de la asignatura.

Durante el cursado, el plantel docente transmite conceptos de metodología de la investigación, enfatizando la orientación de los mismos hacia proyectos de I+D en Informática. Es decir, en la asignatura se abordan conceptos básicos para la formulación del mencionado proyecto, enfatizando etapas involucradas en la metodología de la investigación tendiendo hacia un matriz de índole profesional, debido a la carrera en la cual se inserta. Parafraseando a [11] en la formación de los estudiantes se "tiende a potenciar el desarrollo profesional, lo cual lo encamina a su perfeccionamiento laboral y ciudadano", en un sentido del compromiso social desde la Universidad al medio a la cual se debe.

El plantel docente de la asignatura orienta y realiza un acompañamiento a los alumnos, desde la elaboración y formulación del proyecto, transitando por la producción, hasta la finalización del mismo. Es así, como este proceso permite a los alumnos construir conocimiento en colaboración con otros actores como los orientadores y los docentes de las mencionadas asignaturas, proponiendo aquellas respuestas más adecuadas ante las situaciones que se presentan.

1.2 Ingeniería de Software Basada en Evidencia

La Ingeniería del Software (IS) como una de las disciplinas de la Ciencia Informática comprende los aspectos de la producción de software desde las etapas iníciales de la especificación de un sistema, hasta el mantenimiento de éste después de su implementación [12].

En [8] y [13] se define el objetivo de la Ingeniería de Software Basada en Evidencia como:

"proporcionar los medios por los que la mejor evidencia actual de la investigación se pueda integrar con la experiencia práctica y los valores humanos en el proceso de la toma de decisiones sobre el desarrollo y mantenimiento de software"

En [14] se presentó una aproximación al análisis de las producciones en el marco de las tesinas defendidas en el año 2014. Por lo expuesto en párrafos precedentes, en este documento se presenta una adecuación de la ISBE para determinar evidencias en las producciones defendidas por los estudiantes en el año 2015 para graduarse. Es decir,

en este contexto de I+D+i, los conceptos, los métodos y las técnicas tratados en asignaturas de la carrera, se aplican en el desarrollo de artefactos software que resuelven problemáticas del entorno socio-económico-cultural en ámbitos que atañen a las empresas, los gobiernos y la sociedad.

2 Metodología

En esta sección se describe el método, los procesos realizados y los hallazgos al realizar un análisis exploratorio en torno a la producción de tesinas finalizadas en el año 2015 de la Licenciatura en Sistemas de Información.

Se optó por métodos primarios, es decir, se realizaron estudios sobre las producciones de los graduados en el año 2015; con el objetivo de obtener evidencia respecto a los artefactos elaborados para resolver problemas del mundo real, y plasmados en las tesinas defendidas.

El experimento que guía el trabajo, consiste en indagar en las elecciones de proyectos de los graduados, quienes a través de sus artefactos aportan a la Industria del Software. Por ello se establece como pregunta que guía la indagación:

- ¿cuáles son las áreas de conocimiento elegidas por los estudiantes en el abordaje del trabajo final de graduación y los desarrollos derivados de esta producción?

Para lograr los resultados se adecuó el proceso definido en [8], que consta de las fases expresadas en la Fig. 1.

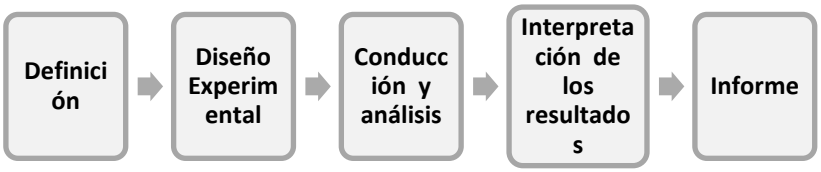


Fig. 1. Proceso de la investigación empírica (Fuente: Elaboración propia basada en [8]).

2.1 Proceso de la investigación empírica en torno a los proyectos de graduación

A continuación se expone el proceso de la investigación empírica en torno a los proyectos de graduación desarrollado para la obtención y exposición de los resultados. Éstos se obtuvieron de la ejecución de las etapas 3 y 4 de la metodología y se explicitan en la sección 3.

Etapa 1. Definición

Se define como objetivo de I+D+I: establecerlas áreas y los temas de interés de los estudiantes.

Siguiendo a [8] "trabajar con evidencias científicas en lugar de con suposiciones se permitirá que el desarrollo de software se convierta en una verdadera disciplina de ingeniería". Lo expuesto contextualizado al dominio del estudio implica que esta técnica guía la investigación para determinar el alcance de la transferencia de conocimiento logrado desde el ámbito universitario al ámbito de influencia.

Por ello, el propósito es lograr información respecto a la aplicación del conocimiento en la resolución de problemas demandados y plasmados en los artefactos producidos en el año 2015 en el marco de las asignaturas TFA y PFC.

En referencia a los objetos:

- Selección de objetos. Los objetos experimentales son los productos construidos por los estudiantes y explicitados en el informe de sus tesinas de grado.
- Selección de sujetos, consiste en 29 tesinas del periodo 2015.

Etapa 2. Diseño experimental

Se estableció como pregunta de investigación que guía el trabajo ¿cuáles son las áreas de conocimiento elegidas por los estudiantes en el abordaje del trabajo final de graduación y los desarrollos derivados de esta producción?

Además, esta pregunta se complementa desde la Ingeniería de Software Basada en Evidencia indicando: "Qué es lo que funciona, para quién, dónde, cuándo y por qué". A continuación se detallan algunas cuestiones que orientan la definición de variables a relevar en los trabajos seleccionados:

- qué es lo que funciona: el producto,
- para quién: el destinatario, representado por las empresas, organizaciones del gobierno, organizaciones del medio, sujetos demandantes de tecnologías de la información y sus productos.
- cuándo: el periodo de indagación, definido por el año 2015
- por qué: Se trata de trabajos finales de graduación producidos por los estudiantes bajo la supervisión de un profesor orientador y de los docentes de la asignatura. Cabe aclarar que los alumnos pueden seleccionar el área y el tema en el cual desarrollar el trabajo, lo que implícitamente podría considerarse como área de preferencia para el futuro desarrollo profesional.

Las que se complementan, considerando

- técnico: tecnologías, complejidad o sistemas definidos.
- social: habilidad individual (por ejemplo, la facilidad con la que los estudiantes identifican un problema o a partir de un planteamiento lo transforman en un proyecto de graduación), autonomía de los estudiantes al seleccionar el área y tema de desarrollo del proyecto de tesina.
- ambiental: posibilidad de transferir el producto a las organizaciones demandantes, o de insertarse en el mercado con este artefacto de las TI.

Por lo expuesto, se definen como variables de interés: el área de conocimiento, la sub-área de la disciplina, el tema de aplicación de los conceptos teóricos, el dominio de

posible transferencia del conocimiento profundizado o adquirido y el problema a resolver con el artefacto tecnológico.

Etapa 3. Conducción y análisis
En la recolección de datos se analizaron los informes de graduación –fuentes primarias- que describen el producto tecnológico diseñado y desarrollado. Estos datos se complementaron con los obtenidos desde una base de datos administrada por las asignaturas. Su posterior análisis produjo información que aporta a la toma de decisiones.
Realizado el estudio, se procedió al procesamiento, la reducción de los datos y la generación de estadísticos descriptivos concernientes a las evidencias estudiadas; lo que derivó en el análisis de los resultados como actividad previa a la interpretación y al reporte.

Etapa 4. Interpretación de los resultados
El análisis de los proyectos modelizados a partir de situaciones reales y defendidos en el periodo señalado, permite identificar:

- Cómo los desarrollos de I+D generados en el marco de trabajos de graduación abordan problemas de la industria, del gobierno o del contexto en el que se insertan los graduados.
- Cómo y a través de qué productos de las TI se logra la transferencia de conocimiento.
- Cómo los estudiantes – principales desarrolladores de software - utilizan e integran los estudios obtenidos en la formación de grado y los materializan en estos artefactos.
- Cuáles son los problemas de la industria o del contexto, que demandan solución, y son identificados y tratados por los estudiantes.

En referencia a las limitaciones se menciona el periodo de estudio focalizado en las producciones analizadas en el trabajo. Además, cabe aclarar que los temas abordados por los estudiantes en la generación del proyecto de graduación dependen de sus intereses, de los interlocutores que pudieron brindarles problemáticas a tratar, y de un relevamiento de posibles temas que las asignaturas producen y difunden entre los interesados en cada ciclo lectivo.
En referencia a las amenazas a las conclusiones, se tratan de restricciones metodológicas vinculadas a cómo se construyó la muestra, es decir, el tipo o el número de tesinas evaluadas que afectan la validez externa.
Una posible amenaza de validez interna podría suponerse en función a algunos elementos desconocidos en el estudio. Particularmente precisar para cada tesina cómo se identificó el problema y quien lo identificó. En futuros trabajos se propone solucionar esta amenaza preguntando al autor el origen de la problemática planteada.

Etapa 5. Reporte de los resultados

Los hallazgos del estudio se informan considerando la importancia de compartir el conocimiento construido y en próximos estudios replicarlo.

3 Resultados

La Informática como disciplina científica y tecnológica se presenta transversal hacia distintos dominios y es amplia su utilización para la resolución de problemas que impactan significativamente en los gobiernos, las empresas y la sociedad. El análisis de los datos de las tesinas defendidas en el año 2015 refleja como principales consumidores del conocimiento producido y plasmado en los trabajos finales de graduación a la academia, los gobiernos, las empresas y la sociedad. A continuación se expresan los datos generados.

La categorización de las tesinas defendidas siguiendo este criterio permitió determinar que estas producciones se concentran principalmente en el área de Ingeniería del Software (26), mientras que en el área Computación se enfocaron solo 2 trabajos y el restante trató la Seguridad. Lo mencionado se refleja en la Fig. 2.

En la Fig. 3 se aprecia que un total de 18 trabajos del Área de la Ingeniería del Software surgieron como soluciones propuestas para su implementación en organizaciones del gobierno y empresas, impactando indirectamente en la sociedad. Cabe aclarar que los autores de estos trabajos se desempeñan laboralmente en estos espacios de actuación, lográndose otro objetivo como es la transferencia de las producciones intelectuales desde la Academia. Los restantes 8 trabajos relacionados a la Ingeniería del Software fueron aplicados en el ámbito de la Universidad.

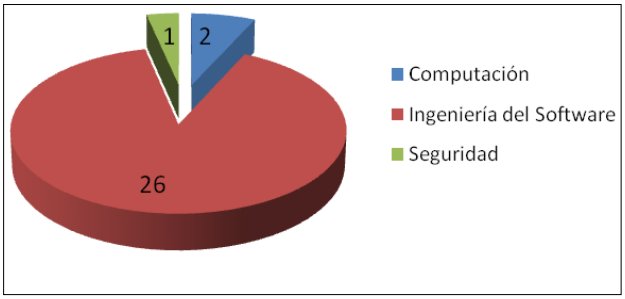


Fig. 2. Porcentajes de temas según disciplina de la Informática

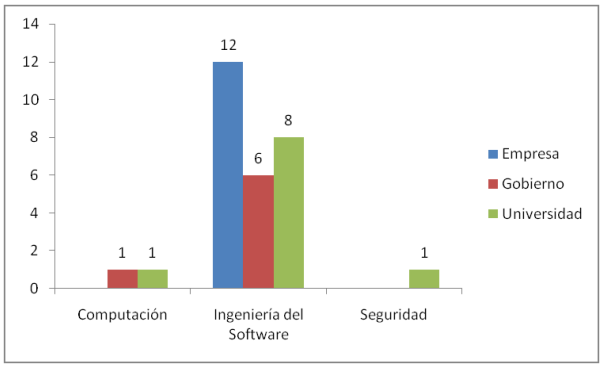


Fig. 3. Ámbito de aplicación de los trabajos defendidos.

Actualmente, el avance de la Informática se refleja en que la mayoría de las producciones intelectuales tratan algunos aspectos de la Ingeniería de Software.

Por otra parte las producciones comprendidas en el campo de la IS son mayoritariamente soluciones en torno a proyectos de gestión (15). Asimismo, otros 7 se enfocaron en la gestión de la UNNE en particular, y otros 2 en la gestión de proyectos (ver Fig. 4). Este enfoque podría explicarse considerando que estas temáticas son ampliamente estudiadas y desarrolladas en asignaturas previas de la carrera, siendo el objetivo de la tesina la integración, profundización y/o abordaje de saberes disciplinares sumando el aporte a la sociedad.

En referencia a los trabajos del área Computación, estos constituyeron soluciones definidas para atender una problemática del área de Gobierno y una aplicación en el ámbito de la Universidad. Cabe aclarar que, los autores de los trabajos se desempeñan en los mencionados ámbitos. En referencia a la tesina vinculada a Seguridad se trató de un desarrollo definido para el ámbito académico y validado en una asignatura de la carrera.

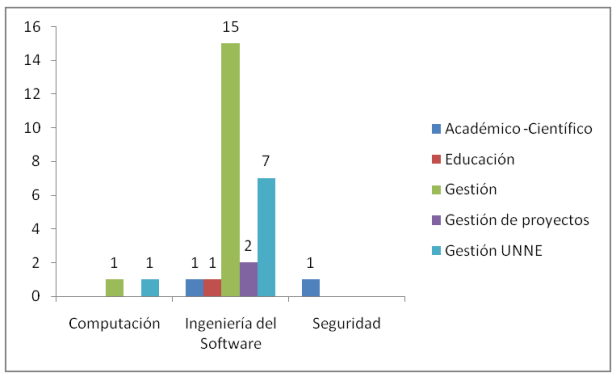


Fig. 4. Campo de aplicación de las producciones

5 Conclusiones

La propuesta metodológica expuesta y sustentada en una adecuación de la Ingeniería del Software basada en la Evidencia y los resultados derivados del estudio podrán utilizarse en la gestión curricular de la asignatura y de la carrera para aportar conocimiento en torno a los saberes y las producciones de los graduados del año 2015.

La replicabilidad del estudio en otros años generará conocimiento disciplinar vinculado al contexto en que se inscribe la carrera; y así se sustentaría la toma de decisiones que repercutan positivamente en la solución de problemáticas identificadas en el ámbito de actuación de estudiantes y graduados. Éstas se constituyen en aportes al Sector de Servicios y Sistemas Informáticos de la región dado que los artefactos de las TIC diseñados y desarrollados en el contexto descripto son objeto de innovación y transferencia hacia aquellos ámbitos que los requieren, y de esta manera se contribuye en procesos de captura, almacenamiento, procesamiento y difusión de la información.

Por otra parte, los espacios de Educación Superior se enriquecen, dado que la búsqueda de problemáticas abordables desde las tesinas de finalización de carrera indica las necesidades de soluciones tecnológicas requeridas por el contexto, y los nuevos métodos y herramientas que deben tratarse desde ámbitos de la educación universitaria. Es así como se refleja una sinergia desde y hacia las Universidades reflejada en los trabajos de finalización del grado.

Referencias

1. Romero, L., Buschini, J., Vaccarezza, L., Zabala, J. P.: La universidad como agente político en su relación con el entorno municipal. Ciencia, Docencia y Tecnología, 26 (51), ISSN: 1851-1716. (2015).
2. Etzkowitz, H., Leydesdorff, H.: A Triple Helix of University-Industry- Government relations. The future location of Research. Book of Abstracts, Science Policy Institute, State University of New York. (1997).
3. Barletta, F., Pereira, M., Robert, V., Yoguel, G.: Argentina: Dinámica reciente del sector de software y servicios informáticos, Revista CEPAL, no. 110, pp. 137-155. (2013).
4. Dughera, L., Ferpozzi, H., Gajst, N., Mura, N., Yannoulas, M., Yansen, G., Zukerfeld, M.: Una aproximación al subsector del Software y Servicios Informáticos (SSI) y las políticas públicas en la Argentina. 10° Simposio sobre la Sociedad de la Información. 41 Jornadas Argentinas de Informática e Investigación Operativa (JAIIO). La Plata, Argentina. (2012).
5. Baruffaldi, J., Iwakawa, S., Iwakawa, N., Ramírez, A.: Una experiencia de vinculación universidad-industria: desafiando la educación, IX Jornadas de Vinculación Universidad-Industria (JUI 2015) - JAIIO 44. (2015)
6. Iglesias, N., Coronel, J., Ezpeleta, J., Angelone, L., Bulacio, P., Tapia, E.: Experiencia vinculación universidad – industria: Desarrollo de tecnología ISOBUS para la industria nacional de maquinarias agrícolas", 9º Jornadas de Vinculación Universidad. 44 Jornadas Argentinas de Informática e Investigación Operativa (JAIIO). Rosario, Argentina. (2015).
7. Gubiani, J., Morales, A., Selig, P.: A pesquisa universitaria e aplicaçãon a inovação. 11vo Simposio Sociedad de la Información 2013, 42 Jornadas Argentinas de Informática e Investigación Operativa (JAIIO). (2013).

8. Genero, M.: Ingeniería del Software Basada en la Evidencia, Escuela Superior de Informática, Universidad de Castilla-La Mancha. Material en diapositiva. (2016).
9. Programa de la asignatura Trabajo Final de Aplicación, FACENA, UNNE
10. Programa de la asignatura Proyecto Final de Carrera, FACENA, UNNE
11. Valdés Rodríguez, M. C., Darin, S.: Una herramienta estratégica para el crecimiento profesional en la sociedad del conocimiento: la formación transversal curricular de competencias comunicativas. Simposio Sociedad de la Información. 37 JAIIO. Jornadas Argentinas de Informática. (2008).
12. Sommerville, I.: Ingeniería del Software. Ed. Prentice Hall. (2011).
13. Kitchenham, B., Budgen, A., Brereton, D.: Evidence-Based Software Engineering and Systematic Reviews, CRC Press. (2016).
14. Mariño, S. I., Alfonzo, P. L.: Ingeniería de software basado en evidencia: soportes como producto académico, Enl@ce: Revista Venezolana de Información, Tecnología y Conocimiento, 14(1):50- 68, (2017)

A Novel Multi-Objective Two-Echelon Green Location-Routing Problem from a City Government Perspective

Rafael Aquino[1], Benjamín Barán[1], Fabio López-Pires[1,2], Fernando Sandoya[3], and Jorge Luis Chicaiza[4]

[1]Polytechnic School, National University of Asunción, San Lorenzo, Paraguay
[2]Itaipu Technological Park, Hernandarias, Paraguay
[3]Department of Mathematics, ESPOL Polytechnic University, Guayaquil, Ecuador
[4]Department of IT in Production and Logistics, Tech. Univ. Dortmund, Germany
raquinos@outlook.com, bbaran@pol.una.py, fabio.lopez@pti.org.py,
fsandoya@espol.edu.ec, jorge.chicaiza@epn.edu.ec

Abstract. This work presents a novel two-echelon, multi-product, Green Location-Routing Problem formulation from a city government perspective, for the optimization of five objective functions, two of them related to pollutant emissions minimization. Additionally, it is demonstrated that the use of city distribution centers (CDCs), compared to direct shipping, is a better strategy for a congested city as Asunción in Paraguay. Initial experimental results using an exhaustive search alternative prove an 8-23% reduction of carbon monoxide (CO) emissions, a 6-22% reduction of carbon dioxide (CO2) emissions and an 8-17% reduction in shipping costs, given an initial investment in CDCs.

1. Introduction

The Location-Routing Problem (LRP) is an NP-Hard combinatorial optimization problem [15] considered as an extension of the well-known Vehicle Routing Problem (VRP) [10]. The main difference between them is that the LRP optimizes not only routing but also depot placement [15].

The *single-echelon* LRP considers a set of potential depots and a set of customers. In a *two-echelon* (2E) LRP, the *first echelon* is composed of manufacturers and depots (also called *satellites* or *city distribution centers*, *CDCs* [5])[1], while the *second echelon* is composed of depots and customers. At first, goods are transported from manufacturers to depots, and distributed from there to the final customers. Figure 1 shows an example of a *two-echelon* LRP where α and β, located at the top, represent manufacturers from which goods are shipped to depots A and B. Customers 1 to 5 are served from depots A and B. In particular, Figure 1 shows a possible solution where two vehicles provide all needed goods to the 5 customers, using only depot A, avoiding the cost of establishing depot B.

[1] The term *CDC* will be preferred through this document.

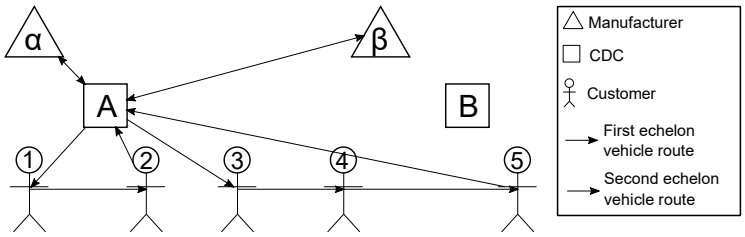


Fig. 1. An example of a two-echelon LRP. Note that the *CDC* Bis not used.

Shipping goods directly from factories to final customers using a fleet of heavy vehicles might not be a suitable strategy for cities with traffic congestion issues, such as Asunción in Paraguay. Trucks may not be able to move through certain zones of the city because of safety or legal reasons, space restrictions or environmental concerns. Furthermore, customers generally need small quantities of products, therefore, using large vehicles may not be a cost-effective solution.

A working vehicle emits carbon monoxide (CO) and carbon dioxide (CO2), among many other pollutants [12]; therefore, road traffic is one of the major sources of *greenhouse gas* [8]. A growing research line, known as "Green Logistics", aims to minimize harmful environmental effects of transportation [7].

Most real-life decisions cannot be accurately modeled with a single objective function [13]. A large vehicle fleet can reduce delivery time, but would increase emission of pollutants. Smaller vehicles fleets usually mean less investments and environmental impact, but these objectives are measured in different units. In this context, a Multi-Objective formulation seems the most adequate approach when modeling objectives which are contradictory or of disjoint nature.

Given that many objective functions have been studied in the literature for problems with similar characteristics, a reasonable approach for solving a TwoEchelon *Green* LRP would be Multi-Objective Optimization. Thus, the main contributions of this work are:

1. A novel *Green* formulation of the 2E Multi-Product LRP, from a city government perspective, considering the minimization of five objective functions, formally presented in Table 3.
2. Initial experimental results showing the benefits of using CDCs, helping citygovernment to improve quality of life.

2. Related Works and Motivation

Most papers deal with single objective models, but there is a trend of increasing attention towards multi-objective LRPs [6]. In [13], Lopes et al. (2013) recognize that most real-life decisions cannot be accurately modeled with a single objective function. Inspired in [13], Table 1 provides a summary of different main objectives considered so far in multi-objective LRPs, grouped by identified categories.

In [4] Caballero et al. (2007) consider a five-objective LRP with uncapacitated depots and describe an application concerning the installation of waste incineration

facilities in southern Spain. The objectives include social rejection to vehicle routes, rejection to facility installation and cost minimization functions.

Table 1. Summary of the main objective functions addressed in multi-objective LRP models, inspired in [13].

Category	Objective
Cost minimization	Number of depots Facility installation cost Transportation cost Travel distance Travel time Transportation load (weight per distance) Distance traveled by customers accessing depots Total monetary costs
Environmental aspects	Transportation risk/nuisance minimization Minimization of risk caused by proximity to facilities Minimization of risk derived from transportation and location Maximization of population satisfaction level /demand served
Equity distribution (minimization objectives)	Maximum transportation risk Maximum proximity to obnoxious facilities Maximum total risk Work time imbalance Load imbalance Unmet customer demand
Others	Other objectives regarding a specific model

In [16], Tavakkoli-Moghadam et al. (2010) present a bi-objective LRP with optional customers. The first objective aims to minimize the total system costs. Thesecond objective maximizes the total served customer demand.

A bi-objective two-echelon, multi-period LRP with time windows for optimizing economic and environmental objectives in a perishable food supply chain network is introduced in [9]. The first objective function minimizes the total cost of a solution while the second objective minimizes the total environmental impact. To the best of authors' knowledge, this is the only Two-Echelon LRP model that can be considered as *Green*.

Govindan et al. [9] (2014) consider environmental impact of facilities and manufacturers as well as transportation: every node and every arc have an environmental impact value. The arc value is not related to the vehicle fleet, so it does not properly model the effects of vehicle fleet on the environment as vehicle emissions vary significantly depending on vehicle type, as described in [12]. As a novelty, the model proposed in the present work explicitly takes this variability into account by considering different emission levels depending on vehicle type.

None of the models reviewed in the present section ([9], [16] and [4]) consider multiple products, so they are better suited to the needs of a particular business, rather

than a city government. On the contrary, the model proposed in this work considers that customers have multiple-product demands, and aims to be especially useful for city-governments when planning and running a city.

3. Proposed Multi-Objective Multi-Product Green 2E-LRP

Table 2. Parameters and decision variables used in the proposed formulation.

(a) Model parameters

Parameter		Description
First Echelon (α is used as superscript)	$t_{l,i}^{\alpha}$	Travel time between nodes $l,i \in (L \cup I)$.
	$c_{l,i}^{\alpha}$	Distance between nodes $l,i \in (L \cup I)$.
	$F_{l,i}^{\alpha}$	Fixed delivery cost from manufacturer $l \in L$to CDC $i \in I$.
	V_{l}^{α}	Cost per delivered unit (e.g., per kg) from manufacturer $l \in L$.
	C^{α}	Vehicle operating cost by distance unit (e.g., per km).
	Q^{α}	Maximum capacity of the vehicles.
	T^{α}	Maximum travel time of the vehicles.
	D^{α}	Maximum travel distance of the vehicles.
	E_{p}^{α}	Vehicles' emission factor of pollutant $p \in P$.
	U_{i}^{α}	Unload time required by the vehicles at CDC $i \in I$.
Second Echelon (β is used as superscript)	$t_{i,j,k}^{\beta}$	Travel time between nodes $i,j \in (I \cup J)$, of vehicle $k \in K$.
	$c_{i,j}^{\beta}$	Distance between nodes $i,j \in (I \cup J)$.
	$F_{i,j}^{\beta}$	Fixed delivery cost from CDC $i \in I$to customer $j \in J$.
	V_{i}^{β}	Cost per delivered unit (e.g., per kg) from CDC $i \in I$.
	C_{k}^{β}	Operating cost by distance unit (e.g., per km), of vehicle $k \in K$.
	Q_{k}^{β}	Maximum capacity of vehicle $k \in K$.
	T_{k}^{β}	Maximum travel time of vehicle $k \in K$.
	D_{k}^{β}	Maximum travel distance of vehicle $k \in K$.
	$E_{k,p}^{\beta}$	Emission factor of vehicle $k \in K$, of pollutant $p \in P$.
	$U_{k,j}^{\beta}$	Unload time at customer $j \in J$, required by vehicle $k \in K$.
Other variables	O_i	Establishing cost of CDC $i \in I$.
	S_i	Maximum storage capacity of CDC $i \in I$.
	$d_{j,l}$	Demand of customer $j \in J$, of product of manufacturer $l \in L$.

(b) Decision variables that define a specific solution *X*

Variable		Description
First Echelon	$x_{l,i,r}^{\alpha}$	Equals 1 if node $l \in (L \cup I)$ immediately precedes node $i \in (L \cup I)$ on the route of vehicle $r \in R$, and 0 otherwise.
	$z_{l,i}^{\alpha}$	Equals 1 if CDC $i \in I$is served from manufacturer $l \in L$, and 0 otherwise.
	$w_{l,i,r}^{\alpha}$	Fraction of the demand of CDC $i \in I$, of products from manufacturer $l \in L$, transported by vehicle $r \in R$. $\left(w_{l,i,r}^{\alpha} \in [0,1]\right)$.

Second Echelon	$x_{i,j,k}^{\beta}$	Equals 1 if node $i \in (I \cup J)$ immediately precedes node $j \in (I \cup J)$ on the route of vehicle $k \in K$, and 0 otherwise.
	y_i	Equals 1 if a CDC is established at a CDC site $i \in I$, otherwise 0.
	$z_{l,i}^{\beta}$	Equals 1 if customer $j \in J$is served from CDC $i \in I$, otherwise 0.

Given a set of manufacturers, a set of potential CDC sites and a set of customers with known demand of several products, it must be determined: which CDCs to establish, assignment of customers to CDCs, vehicle tours for serving the customers' demand (second echelon) and vehicle tours for supplying CDCs with products from manufacturers (first echelon). The solution to the problem must be such that: demand is satisfied without exceeding vehicle capacities, maximum route length and duration, while CDC capacities are not exceeded.

With respect to manufacturers, the proposed model assumes that: *a*) there is no limit on manufacturer production capacity and *b*) every manufacturer produces a specific product, and every product is unique to its manufacturer.

Assumptions about vehicles are: *a*) an homogeneous vehicle fleet is considered at the first echelon, *b*) an heterogeneous vehicle fleet is considered at the second echelon, *c*) only closed tours are considered; they must begin and end at the same facility, *d*) two pollutants are considered for this work, carbon monoxide (CO) and carbon dioxide (CO_2), although the model provides flexibility to consider more pollutants if necessary, *e*) split delivery[2] is allowed only at the first echelon and *f*) no time window are considered at any level.

The sets used in the proposed model are: 1) *L*, the set of manufacturers, 2) *I*, the set of CDCs, 3) *J*, the set of customers, 4) *R*, the set of first-echelon vehicles, 5) *K*, the set of second-echelon vehicles and 6) *P*, the set of pollutants considered. The parameters used in the proposed model and decision variables that define a specific solution *X* are presented in Table 2. Due to space limitations, objective functions and problem constraints are presented without a detailed explanation in Tables 3 and 4 respectively.

4. Benefits on using 2E over single-echelon distribution

The main stakeholders of a logistic system, are: a) the manufacturers, b) the city government, c) the vehicle fleet operator and d) the customers.

On a *Two-echelon distribution scheme*, manufacturers are mostly interested in the cost of shipping goods from their facilities to the CDCs. The city government is responsible for establishing and running CDCs, therefore it is interested in the CDC establishing cost. City governments are also interested in the environmental aspects of a logistics system, namely air pollution caused by delivery vehicles. Vehicle operators are responsible for transportation. They are concerned with the usage and wear of their fleet, and they are interested in being profitable; they are looking at total delivery value minus total vehicle operational costs. Finally, the customers are focused on the delivery costthat vehicle operators charge.

[2] Split delivery: the demand of one customer may be satisfied by more than one vehicle.

On a *single-echelon distribution scheme*, given that no CDCs are in use, large trucks will also be used to distribute customer demand. A fleet of this characteristics is generally more expensive in terms of monetary and environmental costs, therefore: *a*) the vehicle fleet would suffer greater usage wear because of longer trips from manufacturers to final customers, *b*) a heavy vehicle fleet has a greater operational cost, so customers may pay a higher price for delivery and *c*) the city government will not invest in establishing CDCs thus saving money. Moreover, the environmental impact caused by the vehicle fleet emissions will increase and the city traffic may suffer from heavy vehicles entering and parking in highly congested areas as downtown.

Table 3. Objective functions used in the proposed formulation.

CDC establishing cost
This cost is calculated as the sum of the establishing cost of every opened CDC. $$F_1(X) = \sum_{i \in I} y_i \cdot O_i.$$ (1)
Vehicle operating costs
A vehicle's operating cost is calculated as the product between the total traveled distance and its cost factor (C^α at the first echelon, C_k^β at the second). The total operating cost is given as the sum of all vehicles' individual operational costs. $$F_2(X) = \sum_{r \in R} \sum_{l \in (L \cup I)} \sum_{i \in (L \cup I)} c_{l,i}^\alpha \cdot x_{l,i,r}^\alpha \cdot C^\alpha + \sum_{k \in K} \sum_{i \in (I \cup J)} \sum_{j \in (I \cup J)} c_{i,i}^\beta \cdot x_{i,i,k}^\beta \cdot C_k^\beta.$$ (2) The first term corresponds to the first-echelon operating cost, while the second corresponds to the second-echelon operating costs.
Carbon monoxide (CO) and carbon dioxide (CO_2) emissions
A vehicle's emissions of a given pollutant $p \in P$ are calculated as the product between the total traveled distance and an emission factor (which is usually given in grams per kilometer) E_p^α at the first echelon and $E_{p,k}^\beta$ at the second echelon. The total emissions of a given pollutant are given as the sum of all individual vehicles' emissions. $$F_3(X) = \sum_{r \in R} \sum_{l \in (L \cup I)} \sum_{i \in (L \cup I)} c_{l,i}^\alpha \cdot x_{l,i,r}^\alpha \cdot E_1^\alpha + \sum_{k \in K} F_1(X) = \sum_{i \in I} y_i \cdot O_i. \sum_{i \in (I \cup J)} \sum_{j \in (I \cup J)} c_{i,i}^\beta \cdot x_{i,i,k}^\beta \cdot E_{k,1}^\beta.$$ (3) $$F_4(X) = \sum_{r \in R} \sum_{l \in (L \cup I)} \sum_{i \in (L \cup I)} c_{l,i}^\alpha \cdot x_{l,i,r}^\alpha \cdot E_2^\alpha + \sum_{k \in K} \sum_{i \in (I \cup J)} \sum_{j \in (I \cup J)} c_{i,i}^\beta \cdot x_{i,i,k}^\beta \cdot E_{k,2}^\beta.$$ (4) E_1^α and $E_{k,1}^\beta$ corresponds to CO emission factors for the first and second echelon, respectively, while E_2^α and $E_{k,2}^\beta$ to CO_2 emission factors for the first and second echelon, respectively. **Observation:** Other pollutants can be added to the set *P* as necessary, and then the corresponding objective functions will follow the form of equations (3) and (4) to consider those new pollutants. The model provides enough flexibility to suit the needs of different users.
Shipping costs
The shipping costs are composed of: *a*) variable costs, which are calculated as the product between total shipped quantity of goods and a unitary delivery cost and *b*) fixed costs, which are given as fixed for each manufacturer-CDC and CDC-customer pair. $$F_5(X) = \sum_{l \in L} \sum_{i \in I} \left(\sum_{j \in J} d_{j,l} \cdot z_{i,j}^\beta \right) \cdot z_{l,i}^\alpha \cdot V_l^\alpha + \sum_{i \in I} \sum_{j \in J} \left(\sum_{l \in L} d_{j,l} \right) \cdot z_{i,i}^\beta \cdot V_i^\beta + \sum_{l \in L} \sum_{i \in I} z_{l,i}^\alpha \cdot F_{l,i}^\alpha + \sum_{i \in I} \sum_{j \in J} z_{i,i}^\beta \cdot F_{i,i}^\beta.$$ (5) The first two terms of Equation 5 correspond to the first and second-echelon total variable costs, respectively, while the last two corresponds to the total fixed cost of the first and second echelons, respectively.

Table 4. Constraints used in the proposed formulation

Two problem instances will be used for comparison. Instance 1 will consider the following: *a*) a single-echelon distribution scheme, with no CDCs and *b*) only heavy vehicles. On the other hand, Instance 2 will consider the two-echelon model above presented.

Vehicle capacity

At the first echelon, the sum of individual demands of the served customers of each CDC visited in a route must not exceed the vehicle's maximum capacity Q^{α},

$$\sum_{i\in I}\sum_{l\in L} w_{l,i,r}\cdot\left(\sum_{j\in I} d_{j,l}\cdot z^{\beta}_{i,j}\right)\leq Q^{\alpha},\quad \forall r\in R. \tag{6}$$

At the second echelon, every vehicle must be able to carry the total demand, of all requested products, of every visited customer,

$$\sum_{j\in I}\sum_{i\in(I\cup J)}\sum_{l\in L} d_{j,l}\cdot x^{\beta}_{i,j,k}\leq Q^{\beta}_{k},\quad \forall k\in K. \tag{7}$$

Route length

At the first echelon, the length of every vehicle route $r \in R$ must not exceed the vehicle's limit D^{α},

$$\sum_{l\in(L\cup I)}\sum_{i\in(L\cup I)} c^{\alpha}_{l,i}\cdot x^{\alpha}_{l,i,r}\leq D^{\alpha},\quad \forall r\in R. \tag{8}$$

At the second echelon, the length of every vehicle route $k \in K$ must not exceed the vehicle's limit D^{α}_{k},

$$\sum_{i\in(I\cup J)}\sum_{j\in(I\cup J)} c^{\beta}_{i,j}\cdot x^{\beta}_{i,j,k}\leq D^{\beta}_{k},\quad \forall k\in K. \tag{9}$$

Route Duration

A route's total duration, at each echelon, is given as the sum of the time spent unloading goods at every node, and the time spent traveling node to node. The route's total duration must not exceed the vehicle's limit T^{α} at the first echelon,

$$\sum_{i\in I}\sum_{l\in(L\cup I)} U^{\alpha}_{i}\cdot x^{\alpha}_{l,i,r}+\sum_{l\in(L\cup I)}\sum_{i\in(L\cup I)} t^{\alpha}_{l,i}\cdot x^{\alpha}_{l,i,r}\leq T^{\alpha},\quad \forall r\in R, \tag{10}$$

and it must not exceed the vehicle's limit T^{α}_{k}, for every vehicle $k \in K$, at the second echelon,

$$\sum_{j\in I}\sum_{i\in(I\cup J)} U^{\beta}_{k,j}\cdot x^{\beta}_{i,j,k}+\sum_{i\in(I\cup J)}\sum_{j\in(I\cup J)} t^{\beta}_{i,j,k}\cdot x^{\beta}_{i,j,k}\leq T^{\beta}_{k},\quad \forall k\in K. \tag{11}$$

CDC Capacity

The total demand of all customers served from a CDC $i \in I$, must not exceed the CDC's maximum storage capacity S_i,

$$\sum_{j\in I}\sum_{l\in L} z^{\beta}_{i,j}\cdot d_{j,l}\leq S_{i},\quad \forall i\in I. \tag{12}$$

The two considered instances are built around the Gran Asunción area in Paraguay. Two big farms are chosen as manufacturers, two existing city supply centers as CDCs and five supermarkets as customers. Figure 1, presented in Section 1, shows an example using the given entities. $L = \{\alpha, \beta\}$ is the set of manufacturers, $I = \{A, B\}$ the set of CDCs and $J = \{1,2,3,4,5\}$ the set of customers. As CO and CO2 emissions are optimized, $P = \{CO, CO_2\}$ is the set of pollutants.

Due to space restrictions, data for the considered instances may be found online at http://www.ug.edu.ec/mdp/fss/.

Remarks on the test dataset are: *a*) maximum allowed travel time per vehicle is set to 12 hours, *b*) emission factor for the vehicles are taken from [11] and operational costs are based on fuel consumption values taken from [1], [2] and [3], *c*) travel times are

calculated using a reference speed of 25 km/h (value within normal urban speed range [11]), *d*) it is assumed that all vehicles travel at the same speed and *e*) shipping costs and unload times are directly proportional to the vehicle fleet operating costs and type (light or heavy).

An exhaustive search, that tests every possible solution to the problem, is performed for each considered instance to find an optimal, non-dominated solution set. All tests were implemented on Microsoft Visual C++ and run on an Intel Core i7 4790K @ 4.4GHz, 16 GB RAM, Microsoft Windows 10 Home 64 Bit computer.

4.1. Experimental Results and Analysis

Table 5. Sets of optimal non-dominated solutions for the two instances. *Sol.* stands for *Solution*.

(a) Single-echelon instance solutions set SS.

Sol. $s \in S$	**Objective functions**			
	$F_2(s)$	$F_3(s)$	$F_4(s)$	$F_5(s)$
s_1	225.70	171926.10	31.73	163.50
s_2	269.70	206016.54	38.02	148.20

(b) Two-echelon instance solutions set S*S*.

Sol. $s^* \in S^*$	**Objective functions**				**Dominance over the set** S.
	$F_2(s)$	$F_3(s)$	$F_4(s)$	$F_5(s)$	
s_1^*	167.78	123213.54	23.44	151.80	Dominates s_1. Not comparableto s_2.
s_2^*	212.95	157763.66	29.81	135.90	All solutions of S.
s_3^*	214.34	157187.36	29.86	135.90	All solutions of S.
s_4^*	172.72	121778.36	23.67	151.80	Dominatess_1.Not comparable to s_2.
s_5^*	220.57	155902.08	30.19	135.90	All solutions of S.
s_6^*	221.85	155268.48	30.23	135.90	All solutions of S.
s_7^*	177.08	119628.64	23.78	151.80	Dominates s_1. Not comparable to s_2.
s_8^*	222.65	154359.84	30.19	135.90	All solutions of S.

s_9^*	221.91	155223.12	30.23	135.90	All solutions of S.

The optimal, non-dominated solution sets for single and two-echelon instances are presented in Table 5. Given that a single-echelon solution will always have zero CDC establishing cost, the corresponding objective function ($F_1(X)$) is not included in the comparison. The set $S = \{s_1, s_2\}$ is formed from the singleechelon instance solutions, and the set$S^* = \{s_1^*, \cdots, s_9^*\}$ from the two-echelon instance solutions. Considering the Pareto dominance criteria [14], 6 of 9 solutions of S^* (roughly 67%) dominate the entire set of single-echelon solutions S. The rest of S^* are non-comparable to the solutions of S. It is worth emphasizing that no solution of S is optimal (non-dominated) in the Pareto sense, when considering objective functions $F_2(X)$ to $F_5(X)$.

The decision maker can choose one of the dominant solutions from the set S^*, such as s_2^*; in that case, a comparison of objective function values of solutions from set $S$with respect to s_2^* is presented in Table 6. According to the presented results, it can be seen that:

a) a *two-echelon* distribution scheme combined with a light-vehicle fleet for final-customer distribution produces less pollutant emissions, from 8 to 23% less carbon monoxide (CO) and from 6 to 22% less carbon dioxide (CO_2),
b) by using CDCs (*two-echelon distribution*), shipping costs are reduced by 8 to 17% and vehicle operating costs by 5 to 21%, according to data presented in Table 6 and
c) the CDCs establishing costs are a city government investment that will be amortized over time.

Finally, it is worth mentioning that in addition to the advantages of pollution reduction there exists possibilities of centralized traffic improvements, as well as an 8-17% saving in shipping costs that can be used to afford CDC costs while the other part is distributed among the stakeholders.

Table 6. An example of an objective function comparison of single-echelon solutions S with respectto a two-echelon solution $s_2^* \in S^*$. *Sol*. stands for *Solution*, *Op*. for *Operating* and *Shp*. for *Shipping*.

Sol.	**Objective functions**				**Op. cost saving (%)**	**Shp. cost saving (%)**	**Comment**
	$F_2(s)$	$F_3(s)$	$F_4(s)$	$F_5(s)$			
s_1	225.07	171926.10	31.73	163.50	5	17	Clearly, $s_2^*$is better in every considered objective.
s_2	269.70	206016.54	38.02	148.20	21	8	
s_2^*	212.95	157763.66	29.81	135.90	-	-	

5. Conclusions and Future Work

This work presented a novel Green two-echelon, multi-product LRP model formulation, from the city government perspective, with five objective functions, two of them related to the minimization of pollutant emissions. This model provides enough flexibility to include additional objective functions that minimizes several other pollutant emissions. After comparing the single-echelon and two-echelon's best solutions, it was demonstrated that the use of CDCs is a better strategy than direct-shipping, considering economic and environmental aspects, with the potential of improving traffic (not quantitized in this work).

Once the CDCs are established in a city (the problem studied in the present article), new studies will be needed, based on the results of their operation, to establish new ones.

Emission factor is a simple yet valuable approach for modeling vehicle emissions. Vehicle emissions variations depend on many different factors such as speed, driving style, engine temperature and others. A model capable of taking these factors into account will be more accurate but more complex, and therefore, left for future work.

Another echelon can be added between CDCs and customers: loading/unloading bays, which are small street areas specially reserved for cargo vehicle parking.

Finally, given that an exhaustive search is not scalable for larger, real-world problems, it is worth mentioning that a Multi-Objective Evolutionary Algorithm (MOEA) is under research to solve the proposed 2E-LRP from a city government perspective.

References

1. Emissions from Volvo's trucks. http://www.volvotrucks.com/content/dam/volvo/volvo-trucks/markets/global/pdf/our-trucks/Emis_eng_10110_14001.pdf, accessed: 2017-05-03
2. Fiat Fiorino e-Brochure. http://www.fiatprofessional.com/com/CMSEN/Pdf/Fiorino_40p_INGint_04_3_2854_08.pdf, accessed: 2017-05-02
3. Fuso Canter e-Brochure. http://canter.co.uk/Projects/c2c/channel/files/372913_Fuso_Canter_3C13_Single_Manual.pdf, accessed: 2017-05-03
4. Caballero, R., Gonzalez, M., Guerrero, F.M., Molina, J., Paralera, C.: Solving amulti-objective location routing problem with a metaheuristic based on tabu search. Application to a real case in Andalusia. European Jrnl. of Op. Res. 177(3) (2007)
5. Cattaruzza, D., Absi, N., Feillet, D., González-Feliu, J.: Vehicle routing problemsfor city logistics. EURO Journal on Transportation and Logistics 6(1) (2017)
6. Drexl, M., Schneider, M.: A survey of variants and extensions of the location-routing problem. European Jrnl. of Op. Res. 241(2) (2015)
7. Eglese, R., Bektas, T., et al.: Green vehicle routing. Vehicle Routing: Problems,Methods, and Applications 18 (2014)

8. Esteves-Booth, A., Muneer, T., Kubie, J., Kirby, H.: A review of vehicular emissionmodels and driving cycles. Jrnl. of Mech. Engineering Sci. 216(8) (2002)
9. Govindan, K., Jafarian, A., Khodaverdi, R., Devika, K.: Two-echelon multiplevehicle location-routing problem with time windows for optimization of sustainable supply chain network of perishable food. Internat. Jrnl. of Prod. Econ. 152 (2014).
10. Hassanzadeh, A., Mohseninezhad, L., Tirdad, A., Dadgostari, F., Zolfagharinia, H.: Location-routing problem. In: Facility Location: Concepts, Models, Algorithms andCase Studies, chap. 17, pp. 359–417. Physica-Verlag Heidelberg, first edn. (2009)
11. Heijne, V., Ligterink, N., Stelwagen, U.: 2016 emission factors for diesel euro-6 passenger cars, light comm. vehicles and euro-vi trucks. Tech. rep., TNO (2016)
12. Hickman, J., Hassel, D., Joumard, R., Samaras, Z., Sorenson, S.: Meet methodologyfor calculating transport emissions and energy consumption. Tech. rep., EC (1999)
13. Lopes, R., Ferreira, C., Santos, B., Barreto, S.: A taxonomical analysis, currentmethods and objectives on location-routing problems. Int. Trn. Op. Res. 20 (2013)
14. von Lücken, C., Barán, B., Brizuela, C.: A survey on multi-objective evolutionaryalgorithms for many-objective problems. Comput. Opt. and Apps. 58(3) (2014)
15. Marinakis, Y.: Location-routing problem. In: Encyclopedia of Optimization, pp.1919–1925. Springer, second edn. (2009)
16. Tavakkoli-Moghaddam, R., Makui, A., Mazloomi, Z.: A new integrated mathematical model for a bi-objective multi-depot location-routing problem solved by a multi-objective scatter search algorithm. Jrnl. of Manuf. Systs. 29 (2010)

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Tipología influyente en el rendimiento académico de alumnos universitarios

Susana Ruiz[1], Myriam Herrera[2], María Romagnano[3], Lilian Mallea[4], María Inés Lund[3]

[1]Departamento de Geofísica y Astronomía de la FCEFN. - UNSJ
[2]Departamento de Informática de la FCEFN - UNSJ
[3]Instituto de Informática de la FCEFN -UNSJ
[4]Departamento de Matemática de FFHA - UNSJ

sbruizr@yahoo.com.ar; lamallea@gmail.com; {mherrera, maritaroma, mlund}@iinfo.unsj.edu.ar

Resumen. El presente trabajo aborda un informe estadístico centrado en caracterizar el rendimiento académico de alumnos universitarios, a partir de la determinación de variables asociadas, aplicando técnicas estadísticas del Análisis Multivariado. Los análisis realizados se basan en datos provenientes de una encuesta, realizada en el año 2015, a los alumnos de la Facultad de Ciencias Exactas Físicas y Naturales (FCEFN) y de la Facultad de Filosofía Humanidades y Artes (FFHyA) de la Universidad Nacional de San Juan
(UNSJ). Mediante un Análisis Factorial de Correspondencias Múltiples, Análisis de Conglomerados y Análisis de Discriminación Logística se puedo identificar tipologías de alumnos y variables influyentes que diferencian a los alumnos según su rendimiento. Los resultados contribuyen al aporte de herramientas que permitan realizar un diagnóstico válido para orientar de manera efectiva las intervenciones que realice la institución educativa.

Keywords: rendimiento, alumnos, clasificación, discriminación, factores, clúster.

1 Introducción

En muchas investigaciones, independientemente del área de conocimiento, es habitual contar con la necesidad de identificar cuáles son las características que diferencian ciertos grupos de sujetos u objetos respecto de otros, y así contar con predicciones futuras [1].

Este trabajo tiene como propósito esencial mostrar los resultados obtenidos en la determinación de variables que mejor explican la atribución de la diferencia de los grupos de alumnos universitarios de la FCEFN y FFHA de la UNSJ, según su buen o mal rendimiento. Para ello se aplican técnicas del Análisis Multivariado (AM) a datos provenientes de una encuesta titulada “Encuesta de Factores de Riesgo y Calidad de Vida de Estudiantes Universitarios” realizada a alumnos de dichas facultades.

El Análisis Factorial de Correspondencias Múltiples (AFCM) es una herramienta del AM de gran utilidad en la investigación por encuestas, tanto por su potencial en términos exploratorios como por su adecuación para el tratamiento de variables categóricas. Permite establecer relaciones (correspondencias) que existen entre las variables (y entre sus modalidades). Puede interpretarse como una manera de representar las variables en un espacio de dimensión menor, mediante la definición de ejes factoriales y utilizando la métrica Chi-cuadrado. También como un procedimiento objetivo de asignar valores numéricos a variables cualitativas [2].

Por otro lado, tanto el Análisis de Conglomerados como el Análisis Discriminante, a lo que algunos autores ubican entre las técnicas estadísticas del AM más potentes para aplicar en investigaciones sociales, son técnicas que nos permiten clasificar sujetos u objetos a partir de características similares. Las técnicas mencionadas se pueden diferenciar en la manera para extraer conocimiento útil, escondido en esos datos. El Análisis Discriminante cuenta con grupos de datos conocidos, así como observaciones de unidades cuya pertenencia a los grupos, en términos de los grupos conocidos, es desconocida inicialmente y tiene que ser determinada a través del análisis de los datos. Este tipo de problemas de clasificación es comúnmente conocido como reconocimiento de patrones asistido o aprendizaje con una guía. En terminología estadística se conoce con el título de "Análisis Discriminante" (AD).

Hay problemas de clasificación donde los grupos son ellos mismos desconocidos a priori y el principal propósito del análisis es determinar los grupos a partir de los propios datos, de modo que las unidades dentro del mismo grupo sean en algún sentido más similares u homogéneas que aquellas que pertenecen a grupos diferentes. Este tipo de problema de clasificación es referido como reconocimiento de patrón no supervisado o conocimiento sin guía, y, en terminología estadística cae bajo el título de "Análisis de Conglomerados" (AC).

Se puede afirmar que en general un indicador directo de la calidad de la enseñanza es el rendimiento académico medido a través del nivel alcanzado por los estudiantes [3]. El rendimiento académico del estudiantado universitario constituye un factor imprescindible en el abordaje del tema de la calidad de la educación superior. Vista la importancia del tema, en este trabajo se aplican y complementan las distintas técnicas mencionadas para analizar y caracterizar lo que llamamos rendimiento académico universitario desde la perspectiva del alumno.

2. Metodología

Teniendo en cuenta que se quiere caracterizar el rendimiento académico hallando tipologías del alumnado, como también determinar variables influyentes y definir una función discriminante que explique el rendimiento de los alumnos universitarios a partir de dichas variables, se organiza el presente trabajo en dos etapas:

1) Caracterización del alumnado según su perfil de rendimiento.

2) Definición y evaluación de modelos que permitan identificar variables influyentes y discriminar a alumnos según su rendimiento.

Se realiza un estudio exploratorio uni y bidimensional de los datos de la encuesta. Como existe una evidente asociación de las variables tratadas, se hace necesaria la aplicación de técnicas del Análisis Multivariado.

La información proveniente de la encuesta, incluye preguntas relacionadas al rendimiento académico del estudiante. La encuesta fue elaborada con la herramienta web de encuestas online EncuestaFácil.com, y cuenta con varias secciones. Las variables consideradas se pueden agrupar en: variables que caracterizan la Facultad, Universidad y la carrera/s que cursa el estudiante; variables que representan características personales del estudiante y de su familia (edad, sexo, su relación con pares, etc.). Variables asociadas al rendimiento (rendimiento, promedio, etc.); y variables que representan el esfuerzo y motivación del estudiante (Ej. hs. de estudio, asistencia a la universidad, etc.). Se puede consultar la encuesta en: https://www.encuestafacil.com/RespWeb/Qn.aspx?EID=2197195. Las preguntas consideradas no son mutuamente excluyentes entre sí. Por lo que cada pregunta es una variable en sí misma; las respuestas alternativas a las preguntas de cada sección sí son mutuamente excluyentes y cada una de estas respuestas es una modalidad de las variables cualitativas a la que pertenece. En este trabajo cada encuestado no interesa en sí mismo sino como representante de cierta categoría o grupo de población.

En la primera etapa, mediante el Análisis Factorial de Correspondencias Múltiples y un AC se busca dar la tipología de los alumnos según su perfil de rendimiento y su caracterización. Dos alumnos son próximos si poseen buen rendimiento o no, y si tienen similares características según las variables consideradas. Con el AFCM se pretende: hallar la semejanza de los alumnos; estudiar la relación entre las variables; resumir las características observadas en un pequeño número de variables; y comparar modalidades de diferentes variables. Se distinguen dos ámbitos: el que concierne al estudio de cómo se agrupan los alumnos teniendo en cuenta el rendimiento académico del alumno y las variables más asociadas. A estas variables le llamaremos "variables activas" en la conformación de los clusters en el AC. El segundo ámbito es relativo a otras variables, distintas de las activas, que llamaremos "variables suplementarias" que pueden contribuir en la caracterización de los grupos de alumnos predefinidos, en otros aspectos que pueden ser relevantes.

Una vez que se obtienen los factores que concentran la mayor proporción de inercia, mediante el AFCM, se puede aplicar un AC. Dicho análisis trata, a partir de una tabla de datos (individuos-variables), situar a los individuos en grupos homogéneos o conglomerados, de manera que los que se puedan considerar similares, sean asignados a un mismo clúster o grupo. Tanto el AFCM como AC se realizan con la ayuda del software estadístico SPAD-N, siguiendo los pasos indicados en la Figura 1. SPAD (Système Portable pour l'Analyse de Données), permite implementar una estrategia de análisis adecuada al tratamiento exploratorio multivariante de grandes tablas de datos. Su concepción es original y adaptada para un proceso natural de aprendizaje a partir de los datos (data learning) [4].

La existencia de asociación entre variables se puede pensar como una fuente de oportunidades para plantear Modelos de Discriminación. El objetivo es definir un modelo para clasificar mediante la construcción de un Modelo de Regresión Lineal Generalizado (MLG), tal que a partir de una observación "x" de n-variables asociadas al rendimiento académico del alumno, proporcione la probabilidad de que el rendimiento académico sea, por ejemplo, malo (éxito). Dicho modelo se construye maximizando la probabilidad de la muestra. A través de un Modelo de Regresión Logística (MRL) se puede estimar la probabilidad de un suceso que depende de los valores de ciertas covariables o variables asociadas.

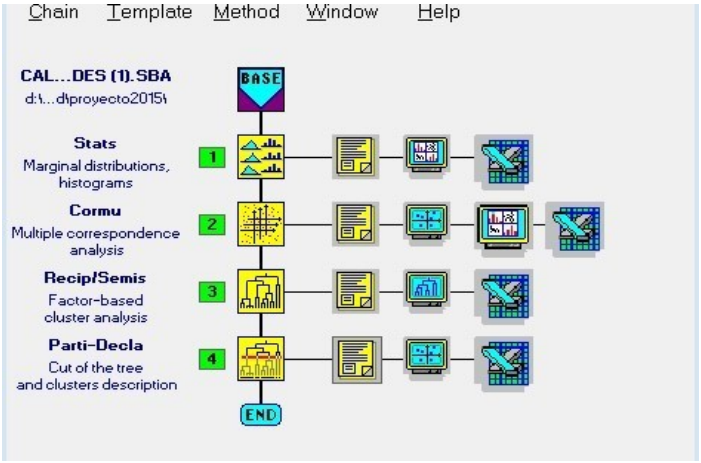


Fig. 1. Pasos en la clasificación jerárquica con SPAD-N

En la segunda etapa de este estudio, el suceso (o evento) de interés es A: "El Rendimiento Académico del alumno es malo" que puede presentarse o no en cada uno de los alumnos de la población donde se realizó la encuesta. Se considera la variable binaria "y" (de tipo Bernoulli) que toma los valores: y = 1, si el suceso A se presenta; y = 0, si A no se presenta. Sea p la probabilidad de que y=1, en un ensayo. Suponiendo que la probabilidad p depende de los valores de ciertas variables, $X_1,\ldots,X_k$; si $\mathbf{x} = (\mathbf{x_1}; \ldots; \mathbf{x_k})$ son las observaciones correspondientes a un alumno sobre las variables, entonces la probabilidad de acontecer A dado x es p(y = 1/x) que simbolizamos con p(x). La probabilidad que no suceda A dado x será p(y = 0/x) = 1p(x).

Teniendo en cuenta los valores en que varía p(x) y el tipo de variables explicativas en este estudio, resulta conveniente suponer un modelo lineal con la llamada transformación logística de la probabilidad, $\ln\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + x_1\beta_1 + \cdots + x_k\beta_k$, siendo los β_i , con$1 \leq i \leq k$ los parámetros del MRL.

Según Díaz & Demétrio [5], el objetivo en el proceso de ajuste de un MLG determinado debe ser obtener el mejor trade-off entre el número de variables y sus parámetros que deben incluirse en la estructura lineal, manteniendo el menor número

posible de ellos y la habilidad del modelo para representar a los datos, manteniendo el ajuste lo más adecuado posible [5]. Como test de ajuste de un modelo MLG, primeramente se observa el estadístico deviance [6], diferencia entre los máximos de los log-verosimilitud para el modelo saturado y en investigación (modelo con k parámetros), esto es $D = 2(\hat{L}(n) - \hat{L}(k))$. Otra expresión para este estadístico es $S_k = \frac{D(y,\hat{\mu})}{\Phi}$, conocido como la deviance para el modelo de investigación. Para ver si un modelo ajusta bien, se compara el valor S_k con el punto crítico de la $\chi^2_{(n-k)gl}$, para un nivel de significación α. Luego si $S_k > \chi^2_{(n-k)gl\,\alpha}$ el modelo de investigación es rechazado.

Para analizar la contribución de un término más en el modelo, se emplea la diferencia de deviances, $S_k - S_q$, la cual debe ser comparada con el punto crítico de la para ese nivel α. También se utiliza la cantidad $\chi^2_{(q-k)gl}$ denotada con AIC [7], de la presentada por Chambers & Hastie [7], forma $D + 2k\hat{\phi}$ con $\hat{\phi}$parámetro de escala.

Para determinar un MRL se aplica la función genérica "glm" del programa R.

Como el modelo de discriminación no es único, es conveniente evaluar y comparar resultados. Para ello se aplica el método Hold-out que tiene en cuenta las tasas de errores en la clasificación (APER), tanto para la muestra para definir el modelo, como en la muestra test. La tasa de error aparente se define como la fracción de observaciones en las muestras de entrenamiento clasificadas erróneamente por la función de clasificación muestral determinada por $APER = \frac{n_{1M}+n_{2M}}{n_1+n_2}$, donde n_{1M} es el número de observaciones de la población 1 clasificados erróneamente como observaciones de la población 2 y n_{2M} el número de observaciones de la población 2 clasificados erróneamente como observaciones de la población 1 [8].

En este estudio, la selección de la muestra para el entrenamiento se determina en forma aleatoria a partir de una rutina "simple" del programa R, y representando aproximadamente el 70% del total de datos de la encuesta.

3 Desarrollo y Resultados

A partir de la puesta en práctica de una encuesta sobre factores de riesgo y calidad de vida a 74 alumnos universitarios pertenecientes a la FFHA y FCEFN de la UNSJ, donde aproximadamente el 88 % afirma tener "buen rendimiento académico" y el 12% "mal rendimiento", se quiere caracterizar a los alumnos en cuanto a su rendimiento, con el objetivo de identificar grupos y diferenciarlos, en base al análisis de variables asociadas, utilizando las técnicas estadísticas del AC y el Análisis de Discriminación Logística.

En la base de datos se pueden distinguir 95 variables, de las cuales se trabaja con 30 de ellas que se consideran más adecuadas a la caracterización del rendimiento académico del estudiante de ambas facultades, 7 de las cuales se seleccionan como "variables activas" para la construcción de los clúster o grupos. Las 24 variables

restantes se consideran “variables suplementarias” para caracterizar a los grupos. Las variables seleccionadas son todas categóricas y el número de modalidades o categorías quedan detalladas a través de la Figura 2.

El criterio de selección de cada variable categórica como activa, fue a través de observar evidencias, bajo un nivel de significación del 5%, en la asociación de la variable seleccionada y la variable rendimiento académico. Esto requirió previamente del análisis y la transformación de la base de datos en forma conveniente. Se combinaron categorías de una misma variable cuando las frecuencias observadas resultaron menores a 5, y se aplicaron pruebas Chi-cuadrado de independencia para tablas de contingencias dos por dos. En total se realizaron 29 pruebas utilizando el software SPSS. Las variables así seleccionadas van a caracterizar los ejes factoriales en este estudio.

```
SELECTION OF CASES AND VARIABLES
ACTIVE CATEGORICAL VARIABLES
     6 VARIABLES      14 ASSOCIATED CATEGORIES
--------------------------------------------------------------------------------
  14 . Acceso a PC en vivienda                                (   2 CATEGORIES )
  15 . Tienes Internet en Vivienda                            (   2 CATEGORIES )
  25 . Valor Alimento                                         (   2 CATEGORIES )
  38 . Cuantas hs a estudiar                                  (   2 CATEGORIES )
  47 . Relación c/compañeros                                  (   2 CATEGORIES )
  57 . Tu Rendimiento académico es                            (   2 CATEGORIES )
--------------------------------------------------------------------------------
SUPPLEMENTARY CATEGORICAL VARIABLES
    24 VARIABLES      91 ASSOCIATED CATEGORIES
--------------------------------------------------------------------------------
   1 . Año que cursas                                         (   6 CATEGORIES )
   3 . Género                                                 (   3 CATEGORIES )
   4 . Edad                                                   (   3 CATEGORIES )
   8 . Tienes Hijos                                           (   2 CATEGORIES )
  19 . Cómo costeaste estudios                                (   4 CATEGORIES )
  26 . Valor Estudio                                          (   2 CATEGORIES )
  31 . Con que frecuencias vas a Univ                         (   4 CATEGORIES )
  32 . Prom con Aplazos                                       (   5 CATEGORIES )
  33 . Importancia de Aplazos                                 (   6 CATEGORIES )
  34 . Nivel exigencia carrera                                (   3 CATEGORIES )
  35 . Cuántas hs dedicas a clase teo                         (   4 CATEGORIES )
  36 . Cuántas hs dedicas a la clase pract                    (   4 CATEGORIES )
  37 . Cuántas hs dedicas a la consulta                       (   4 CATEGORIES )
  40 . Material Estudio Económicamente accesible              (   3 CATEGORIES )
  41 . Material Estudio Actualizado                           (   3 CATEGORIES )
  42 . Material Estudio de Calidad                            (   3 CATEGORIES )
  43 . Material de fácil comprensión                          (   3 CATEGORIES )
  44 . Material de Biblioteca                                 (   5 CATEGORIES )
  46 . Elección de la carrera                                 (   2 CATEGORIES )
  48 . Relación c/docente                                     (   4 CATEGORIES )
  49 . Estres por estudiar                                    (   4 CATEGORIES )
  54 . La carrera mejora tu futuro                            (   3 CATEGORIES )
  76 . Hs/dia Sentado                                         (   4 CATEGORIES )
  81 . Percepción de Salud                                    (   7 CATEGORIES )
```

Fig. 2. Caracterización de variables activas y suplementarias

En las Figuras 3 y 4 se muestran algunos resultados del AFCM correspondiente a la primera etapa de trabajo.

Del análisis se observa:

- Para el primer eje factorial, las variables activas que más peso ejercen sobre el eje, y concentran mayor cantidad de inercia son: “Tu Rendimiento Académico es” (contribución 23.6), “Tienes Internet en vivienda” (contribución 21.2), siguiendo “Acceso a PC en vivienda” y “Relación con compañeros” (ambas con contribución 19). Los valores de los cosenos al cuadrado de cada modalidad, en la medida que se aproximan al valor 1, indican buena calidad en la representación de la modalidad en el eje. Por lo que todas las modalidades quedan bien representadas en el eje. Las modalidades de mayor peso ubicadas en el semieje positivo son: “Bueno” (respecto a la variable “Rendimiento Académico”), “Si” (tiene Internet en su vivienda), “Sí” (tiene

acceso a PC en vivienda) y "Buena" (relación con compañeros). En el primer eje factorial, que reúne el 39.70% de la inercia total de la nube de puntos (ver Figura 3), se oponen los alumnos que: tienen buen rendimiento, PC e Internet en vivienda y consideran tener buena relación con sus compañeros, con aquellos que poseen mal rendimiento, no tienen acceso a PC ni a Internet en su vivienda y tiene mala, regular o ninguna relación con sus compañeros.

- En el factor 2 la variable relevante es "Cuántas horas le dedica a estudiar" (contribución 63.4). Teniendo en cuenta los valores de los cosenos al cuadrado, las dos modalidades de dicha variable quedan bien representadas. El segundo eje, que reúne el 18.88% de la inercia total (ver Figura 3), contrapone los alumnos que le dedican más de 4 horas de estudio diario a la carrera (representado en el semieje positivo), respecto a aquellos que le dedican menos (representado en el semieje negativo).
- El tercer eje factorial, que reúne el 14.83% de la inercia total, la variable más relevante es "Valor alimento" con una contribución 61.2.

EIGENVALUES
COMPUTATIONS PRECISION SUMMARY : TRACE BEFORE DIAGONALISATION.. 1.0000
SUM OF EIGENVALUES..... 1.0000
HISTOGRAM OF THE FIRST 6 EIGENVALUES

NUMBER	EIGENVALUE	PERCENTAGE	CUMULATED PERCENTAGE	
1	0.3970	39.70	39.70	**********
2	0.1888	18.88	58.59	**********
3	0.1483	14.83	73.42	********
4	0.1097	10.97	84.39	******
5	0.0916	9.16	93.54	*****
6	0.0646	6.46	100.00	****

Fig. 3. Caracterización de los ejes factoriales.

LOADINGS, CONTRIBUTIONS AND SQUARED COSINES OF ACTIVE CATEGORIES
AXES 1 TO 5

CATEGORIES			LOADINGS					CONTRIBUTIONS					SQUARED COSINES				
IDEN - LABEL	REL. WT.	DISTO	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
14 . Acceso a PC en vivienda																	
No - No	1.80	8.25	-1.93	-1.47	-0.86	-0.44	-0.07	17.0	20.7	9.0	3.2	0.1	0.45	0.26	0.09	0.02	0.00
Si - Si	14.86	0.12	0.23	0.18	0.10	0.05	0.01	2.1	2.5	1.1	0.4	0.0	0.45	0.26	0.09	0.02	0.00
			CUMULATED CONTRIBUTION =					19.0	23.2	10.1	3.5	0.1					
15 . Tienes Internet en Vivienda																	
No - No	4.50	2.70	-1.17	-0.47	0.46	-0.66	0.48	15.5	5.4	6.3	17.8	11.3	0.51	0.08	0.08	0.16	0.08
Si - Si	12.16	0.37	0.43	0.18	-0.17	0.24	-0.18	5.7	2.0	2.3	6.6	4.2	0.51	0.08	0.08	0.16	0.08
			CUMULATED CONTRIBUTION =					21.2	7.3	8.7	24.4	15.4					
25 . Valor Alimento																	
suf - Suficiente	14.86	0.12	0.19	-0.01	-0.26	-0.11	0.04	1.4	0.0	6.6	1.8	0.3	0.31	0.00	0.54	0.11	0.01
insu - Insuficiente	1.80	8.25	-1.59	0.07	2.12	0.95	-0.34	11.5	0.0	54.6	14.7	2.3	0.31	0.00	0.54	0.11	0.01
			CUMULATED CONTRIBUTION =					12.8	0.0	61.2	16.4	2.6					
38 . Cuantas hs a estudiar																	
>4 - Más de 4	12.84	0.30	0.18	-0.46	-0.01	0.18	-0.07	1.0	14.6	0.0	4.0	0.7	0.10	0.72	0.00	0.11	0.02
<4 - Menos de 4	3.83	3.35	-0.59	1.55	0.04	-0.62	0.24	3.4	48.8	0.0	13.4	2.5	0.10	0.72	0.00	0.11	0.02
			CUMULATED CONTRIBUTION =					4.4	63.4	0.0	17.4	3.2					
47 . Relación c/compañeros																	
B - Buena	14.19	0.17	0.28	-0.08	0.15	-0.21	-0.15	2.8	0.4	2.1	5.7	3.5	0.45	0.03	0.13	0.25	0.13
R - Reg-Mala-Inexistente	2.48	5.73	-1.61	0.43	-0.85	1.20	0.87	16.1	2.5	12.0	32.5	20.3	0.45	0.03	0.13	0.25	0.13
			CUMULATED CONTRIBUTION =					19.0	2.9	14.1	38.1	23.8					
57 . Tu Rendimiento académico es																	
B - Bueno	14.64	0.14	0.28	-0.07	0.09	0.01	0.20	2.9	0.4	0.7	0.0	6.7	0.56	0.04	0.05	0.00	0.30
Mal - Malo	2.03	7.22	-2.01	0.51	-0.62	-0.08	-1.47	20.7	2.8	5.2	0.1	48.1	0.56	0.04	0.05	0.00	0.30
			CUMULATED CONTRIBUTION =					23.6	3.2	5.9	0.1	54.8					

Fig.4. Contribuciones de las variables activas según los 5 primeros ejes factoriales.

Teniendo en cuenta las proyecciones de calidad en el primer plano factorial de las modalidades de las variables suplementarias, que se pueden identificar a través de los valores-test mayores a 1.96 (en valor absoluto) que conducen a rechazar la hipótesis aleatoriedad con un nivel de significación de 0.05, se observa que: las variables suplementarias relacionadas con el primer eje factorial son "Año que cursas", "Edad",

"Cómo costeaste estudios", "Promedio con aplazos", "Valor Estudio", "Importancia de los aplazos", "Cuantas horas le dedica a la práctica", "Material de Biblioteca" y "Relación con el docente". Los alumnos cuyo rendimiento académico es bueno, tienen PC e Internet en vivienda, suficiente dinero para alimento y estudio, le dedica más de 4 horas diarias a estudiar, tienen buena relación con compañeros y docentes, cursan tercer año, su edad es entre 18 a 24 años, costean los estudios sólo con aporte y el promedio con aplazos es mayor a 8. Se oponen de aquellos que poseen mal rendimiento, no tienen PC ni Internet en vivienda, es insuficiente el dinero para alimento y estudio, le dedican menos de 4 horas a la práctica y a estudiar, expresan que su relación con compañeros es regular, mala o inexistente, cursan cuarto año, la edad oscila entre 24 a 29 años, costean el estudio sólo con trabajo, sienten mucho fracaso cuando lo aplazan, consideran que el material de Biblioteca es de difícil acceso y su relación con docentes es regular.

Mediante el AC se clasifican los alumnos de acuerdo a su similitud, por clasificación jerárquica, utilizando el método del vecino más próximo, considerando los primeros tres ejes factoriales que reúnen una inercia total acumulada del 73,42% aproximadamente. A partir de la cual se obtiene la representación gráfica del proceso del agrupamiento mediante un Dendograma en la que se distinguen dos grupos (ver Figura 5).

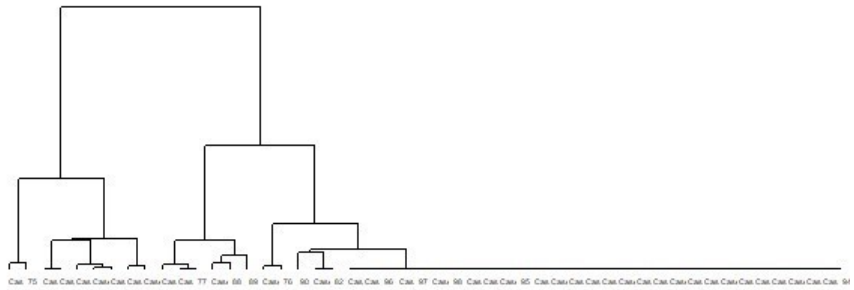


Fig.5. Dendograma.

La primera clase o grupo la componen 12 alumnos, que representan el 16,22% del total de los alumnos encuestados. Se destacan por: no poseer internet en vivienda (55% del total de alumnos de esta modalidad forman esta clase, y el 91.67% de los alumnos de la clase no tienen internet en su casa); no tienen acceso a PC en vivienda (87.50% del total de alumnos de esta modalidad forman esta clase, y el 58.33% de los alumnos de la clase no tienen acceso a PC en vivienda); su rendimiento académico es malo (77.78% del total de los alumnos que presentan esta modalidad; representa el 58.33% de los alumnos de la clase no tienen buen rendimiento); su relación con compañeros es regular, mala o inexistente (63.64% del total de los alumnos que presentan esta modalidad; representa el 58.33% de los alumnos de la clase tienen esta modalidad); el dinero para alimento y estudio considera que es insuficiente (75%, 28.57% del total de alumnos que presentan cada modalidad; 50%, 83.33% de los alumnos de la clase tienen la modalidad, respectivamente); cursan cuarto año (71.43% del total de los

alumnos que presentan esta modalidad; representa el 41.67% de los alumnos de la clase tienen esta modalidad); tienen de 24 a 29 años de edad (36% del total de los alumnos que presentan esta modalidad; representa el 75% de los alumnos de la clase tienen esta modalidad); sienten mucho fracaso con los aplazos (31.03% del total de los alumnos que presentan esta modalidad; representa el 75% de los alumnos de la clase tienen esta modalidad); y considera que su relación con los decentes es regular (36.84% del total de los alumnos que presentan esta modalidad; representa el 58.33% de los alumnos de la clase tienen esta modalidad). Gfg

La segunda clase reúne el 83.78% del total de los alumnos encuestados. Los alumnos de esta clase se caracterizan por: poseer Internet en vivienda (98.15% del total de alumnos de esta modalidad; 85.48% de los alumnos de la clase); tener acceso a PC en vivienda (92.42% del total de alumnos de esta modalidad; 98.39% de los alumnos de la clase); tener buen rendimiento (92.31% del total de alumnos de esta modalidad; 96.77% de los alumnos de la clase); su relación con compañeros es buena (92.06% del total de alumnos de esta modalidad; 93.55% de los alumnos de la clase); consideran que tienen suficiente dinero para alimento y estudio (90.91%, 94.87% del total de alumnos que presentan cada modalidad; 96.77%, 59.68% de los alumnos de la clase tienen la modalidad, respectivamente); costean los estudios sólo con aporte (100% del total de alumnos de esta modalidad; 43.55% de los alumnos de la clase); su edad es entre 18 a 24 años (95.12% del total de alumnos de esta modalidad; 62.90% de los alumnos de la clase); y su promedio con aplazo es mayor a 8 (100% del total de alumnos de esta modalidad; 35.48% de los alumnos de la clase).

En la segunda etapa de trabajo, para definir un modelo de discriminación mediante MRL, se dispone de la información proveniente de la encuesta, para las variables categóricas cuyas modalidades se definen en la Tabla 1.

Tabla 1. Variables consideradas para definir el MRL

Lista de variables	
V01 Rendimiento académico	V07 Dinero para Alimento
0 Bueno	0 Suficiente alimento
1 Malo	1 Insuficiente alimento
V05 Tiene PC en vivienda	V15 Horas que dedica a estudio
0 No PC vivienda	0 Más de 4 hs
1 Si PC vivienda	1 Menos de 4 hs
V06 Tiene internet vivienda	V20 Relación con Compañeros
0 No internet vivienda	0 Buena relación
1 Si internet vivienda	1 Regular-Mala-inexistente

Se confecciona un programa en R cuyos pasos se pueden resumir en los siguientes:

1) Lectura de la base de datos. 2) Categorización de las variables explicativas (V05, V06, V07, V15 y V20). 3) Categorización de la variable respuesta (V01). 4) Selección de una muestra de entrenamiento y una muestra test. 5) A partir de la muestra de entrenamiento se realiza la estimación de los parámetros, y las pruebas de ajuste para el modelo con todas las variables y el modelo de investigación. 6) Confección de tablas

de datos mal clasificados, para la muestra de entrenamiento y test. 7) Cálculo de tasas aparentes asociadas a la muestra de entrenamiento y test. 8) Cálculo de odds radio para el modelo reducido, para interpretar los coeficientes del modelo.

En la imagen de la Figura 6 se observa el resultado de realizar el ajuste por máxima verosimilitud con el software R, considerando todas las variables de la Tabla 1. Como la deviance es menor que el valor crítico $\chi^2_{(n-k)gl;\alpha} = \chi^2_{47gl;\alpha-0.05} = 64.00111$ residual $S_k = 23.966$ se puede asumir que el modelo ajusta bien, con un nivel de significación α=0.05. Luego se aplica el método paso a paso hacia atrás, para analizar si podemos hallar un modelo más simple que el antes fijado. Para ello se utiliza la rutina step de R. Inicialmente se tiene un AIC=33.97 para el modelo con todas las variables, y la función de R considera la eliminación de cada una de las variables para finalmente sacar del modelo la variable V06: "Tiene internet en su casa" por ser la que produce un AIC más bajo (32.16). En el siguiente paso considera la eliminación de alguna de las tres restantes variables, pero el algoritmo en R decide quedarse con ellas ya que su eliminación supone un aumento, en el mejor de los casos, del AIC último hallado. Como los valores de AIC no difieren mucho, entre el modelo con todas las variables y el modelo reducido, se analiza además la inclusión o no de la variable V06 realizando una prueba de significación mediante una prueba de razón de verosimilitudes. A partir del cual, bajo un nivel de significación del 5%, como G=Deviance (modelo reducido) – Deviance (modelo con todas las variables) =25.881-25.867=0.014 resulta menor que el valor crítico $\chi^2_{1gl} = 3.841459$, se puede considerar que la inclusión de la variable V06 no contribuye al modelo.

```
R Console

> log.glm=glm(V01~V05+V06+V07+V20,d,family="binomial",subset=train)
> summary(log.glm);

Call:
glm(formula = V01 ~ V05 + V06 + V07 + V20, family = "binomial",
    data = d, subset = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6790  -0.2926  -0.2158  -0.2158   2.7464

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                -0.6908     1.2829  -0.538   0.5903
V05Si PC vivienda          -2.4386     1.4714  -1.657   0.0975 .
V06Si internet vivienda    -0.6187     1.3984  -0.442   0.6582
V07 Insuficiente alimento   1.8916     1.3979   1.353   0.1760
V20Regular-Mala-inexistente 2.3674     1.1389   2.079   0.0377 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 41.087  on 51  degrees of freedom
Residual deviance: 23.966  on 47  degrees of freedom
AIC: 33.966
```

Fig.6. Resultados del ajuste del modelo con todas las variables utilizando la rutina glm de R.

En la imagen de la Figura 7 se pueden observar los parámetros estimados y su significación en el modelo reducido. El MRL reducido resulta:

$$\ln\left(\frac{p}{1-p}\right) = -0.9256 - 2.7748 \ \ d_{V05} + 2.2377 \ d_{V07} + 2.4573 \ \ d_{V20}$$

Donde p la probabilidad de que la variable V01: "Rendimiento Académico" tome la categoría 1 ("Mal Rendimiento") d_{V05}, d_{V07} y d_{V20} variables dummy que toman el valor 1 cuando la modalidad que se observa corresponde a la categoría 1; en caso contrario las variables toman el valor 0.

Sea $\hat{\beta}_j + X_i \cdot \hat{\beta}$ el segundo miembro del modelo reducido planteado, donde $X_i = (d_{V05}, d_{V07}, d_{V20})$, $\hat{\beta_0} = -0.9256$ y $\hat{\beta} = (-2.7748, 2.2377, 2.4573)^T$ vector de parámetros estimados. Si el "Alumno i" tiene como vector de observaciones asociadas a las variables del modelo, x_i

$$\widehat{p_{(x_i)}} = \frac{1}{1+e^{\hat{\beta_0}+X_i \cdot \hat{\beta}}} \Rightarrow \widehat{q_{(x_i)}} = 1 - \widehat{p_{(x_i)}} = \frac{e^{\hat{\beta_0}+X_i \cdot \hat{\beta}}}{1+e^{\hat{\beta_0}+X_i \cdot \hat{\beta}}}$$

Se definen como reglas para la clasificación: "Si $\frac{e^{\hat{\beta_0}+X_i \hat{\beta}}}{1+e^{\hat{\beta_0}+X_i \hat{\beta}}}$ >0.5, entonces el Alumno i pertenece a la clase de los de "Buen Rendimiento", caso contrario a los de "Mal Rendimiento".

```
R Console
> log.glm=glm(V01~V05+V07+V20,d,family="binomial",subset=train)
> summary(log.glm);

Call:
glm(formula = V01 ~ V05 + V07 + V20, family = "binomial", data = d,
    subset = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6182  -0.2210  -0.2210  -0.2210   2.7294

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)                -0.9256     1.1682  -0.792   0.4282
V05Si PC vivienda          -2.7748     1.2692  -2.186   0.0288 *
V07 Insuficiente alimento   2.2377     1.1644   1.922   0.0546 .
V20Regular-Mala-inexistente 2.4573     1.1104   2.213   0.0269 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 41.087  on 51  degrees of freedom
Residual deviance: 24.156  on 48  degrees of freedom
AIC: 32.156
```

Fig.7. Resultados del ajuste del modelo reducido utilizando la rutina glm de R.

La Tabla 2 representa la tabla de mal clasificados para la muestra de entrenamiento y muestra test, al aplicar el modelo reducido estimado con R. A partir de la tabla se

pueden calcular los errores aparentes correspondientes, resultando: 0.07692308 y 0.0909091 respectivamente. Por lo que el porcentaje de mal clasificados para la muestra de entrenamiento es de 7,69%, y para la muestra test de 9,09%. Ambos errores resultan similares, conservando un equilibrio en la proporción de errores, donde supera levemente el error aparente de la muestra test sobre el de la muestra de entrenamiento.

Tabla 2. Tabla de mal clasificados para la muestra de entrenamiento y muestra test.

Muestra Entrenamiento	Buen Rend.	Mal Rend.		Muestra Test	Buen Rend.	Mal Rend.
0:Buen Rend.	44	3		0:Buen Rend.	19	1
1:Mal Rend.	1	4		1:Mal Rend.	1	1

Los coeficientes estimados del modelo reducido se pueden interpretar mediante cocientes de chances. Para ello se calculan los exponenciales de los coeficientes estimados y sus valores recíprocos en caso de que los exponenciales resulten valores menores a uno. De lo que resulta: si las modalidades "posee suficiente dinero para alimento" y "tiene buena relación con sus compañeros" se mantienen fijas, la chance de que un alumno tenga mal rendimiento, respecto a que tenga buen rendimiento, es $1/e^{-2.7748} \cong 16$ veces mayor, si el alumno no tiene acceso a PC en casa, respecto al que si tiene acceso. Ahora si se mantienen fijas "no tiene acceso PC en su casa", y "tiene buena relación con sus compañeros", la chance de que un alumno no tenga buen rendimiento, respecto a que si lo tenga, es $e^{2.2377} \cong 9.371751$ veces mayor si "no tiene suficiente dinero para alimento" respecto al que tiene suficiente alimento. Por último, si se mantienen fijas "no tiene acceso a PC en su casa" y "tiene suficiente dinero para alimento", la chance de que el alumno no tenga buen rendimiento, respecto al que si lo tenga, es $e^{2.4573} \cong 11.67325$ veces mayor si el alumno tiene una relación regular, mala o no tener relación con sus compañeros, respecto al que tiene una relación buena.

4 Conclusiones

Mediante un análisis discriminante se pudo establecer el poder explicativo y discriminatorio de características que diferencian a los alumnos según su rendimiento, a partir de datos extraídos de una encuesta.

A través de la identificación de variables influyentes, mediante la aplicación de técnicas del AM, se pudo vislumbrar aspectos sociales, culturales y económicos fuertemente asociados al rendimiento del estudiante universitario, y con ello a la calidad educativa, tales como: el disponer de PC en vivienda, el tipo de relación con sus compañeros, tener suficiente dinero para alimento.

Esta información relevante de los alumnos puede contribuir en la formulación de políticas de mejoramiento o direccionamiento institucional. Los resultados contribuyen al aporte de herramientas que permitan realizar un diagnóstico válido para orientar de manera efectiva las intervenciones que realice la institución; en este sentido permite posteriormente obtener una medida de seguimiento año a año mediante la aplicación sistemática del instrumento con el objeto de evaluar el impacto de las acciones realizadas.

Referencias

1. Torrado Fonseca, M.y Berlonga-Silvente, V. (2013)- Análisis Discriminante mediante SPSS. REIRE -Revista d'Innovació i Recerca en Educación.
2. Peña, Daniel (2002), "Análisis de datos multivariantes"- McGRAW-HILL- ISBN84-4813610-1, Madrid, España.
3. Escudero Escorza. T (2000). La evaluación y mejora de la enseñanza universitaria: otra perspectiva. Revista de investigación educativa, 18(2) 405-416
4. Bertaut, M. B.(s.f.). Manual de introducción a los métodos factoriales y clasificación con SPAD. Servei d'Estadística Universitat Autónoma de Barcelona. Recuperado de: http://sct.uab.cat/estadistica/sites/sct.uab.cat.estadistica/files/manualSPAD.pdf
5. Díaz, M. P. & Demétrio, C. G. B. (1998)- Introducción a los modelos Lineales Generalizados – SCREM Editorial- ISBN 987-96970-1-4. Córdoba. Argentina.
6. Nelder, J. A. & Wedderburn, R. W. M. (1972) Generalized Linear Models. Journal of the Royal Statistics Society, A. 135: 370-384.
7. Chambers, J. M. & Hastie, T. J. (eds) (1992). Statistical Models in S., Chapman and Hall, New York.
8. Johnson, R. & Wichern, D. (1998) Applied Multivariate Statistical Analysis, Prentice_Hall, London.

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Data Sanitization in current hard disk

Fernando V. Sbampato, Marco A. T. Rojas, Fernando F. Redigolo, Tereza Cristina M. de Brito Carvalho

Laboratory of Architecture and Computer Network (LARC) – Polytechnic School of the University of São Paulo – Zip code 05508-10 – Brazil
{fsbampato, matrojas, fernando, carvalho}@larc.usp.br

Abstract. Data sanitization is a process traditionally applied using physical techniques, aiming at the completely destruction of the hard disk, however, there is an increasing interest in the use of logical techniques for data sanitization, that allow reusing the physical device. This work aims to evaluate the effectiveness of the standards (0, 1 and random) used in logical techniques to overwrite the data stored on hard disks. This evaluation is performed through the Magnetic Force Microscopy (MFM) technique, to verify if it is possible to "reconstruct" the original data stored on the hard disk after the data sanitization process. Results show the efficiency of using the overwriting standards in the logic techniques in the data sanitization process.

Keywords: Data Sanitization, Magnetic Force Microcopy, data override pattern, hard disk.

1 Introduction

Unauthorized access to confidential data is a continuous concern of users throughout computing history, due to the value assigned to these data [1], [2]. Data stored on hard drives used on computers always contains data from users, clients, and more. Most users believe that file removal commands completely erase the contents of the device. However, depending on the file system used in the operating system, only the logical address of the file stored in the file allocation table is overwritten with new metadata and the file with the original data are still stored on the hard disk [3].

Due to this fact, any user with access to this hard disk can try to recover this stored data, generating concerns to its owners. There are programs designed to recover data when they have not yet been physically removed from the hard disk, such as Recuva [4], causing more concern to users regarding the security of their data. One way to avoid this kind of data recovery attack is by applying a data sanitization technique on hard disks [5]. Data sanitization is the removal of data stored on a device in a way that prevents this data from being retrieved. Data sanitization can be performed by physical techniques, where the disk is physically destroyed, or by means of logical techniques, where the original data is overwritten with new data, so that the final binary string of the operation is the most different from the original binary string of the data. An example of a logical technique used for data removal is the DoD 5220.22-M technique,

from the US Department of Defense, used by Amazon Web Services (AWS) to sanitize hard drive data removed from its production environment at the end of its useful life [6].

The use of logical techniques, where the hard disk is not discarded at the end of the sanitization process, has been provided an increasing interest of users, since these techniques combine the increase of the life cycle of hard disks with security related to data confidentiality, allowing the reuse of the physical device, reducing costs and providing sustainability. However, there are discussions by the scientific community [7], [8] by standardization organizations, such as NIST [9] and ENIZA [10], and by companies such as Oracle [11] and Hewlett Packard [12], on the efficiency of the overwriting logic techniques to remove the data stored on the hard disk, as well as on the possibility of using the Magnetic Force Microscopy (MFM) to recover this data.

Aiming to present a contribution to this discussion, an evaluation with the overwriting patterns (0, 1 and random) used by the logical techniques was performed to verify the effectiveness of the logical techniques to sanitizing the data stored on hard disks. The remainder of this paper is organized as follows: Section 2 presents details about data sanitization and details about the MFM technique. Section 3 presents the work related to these topics. Section 4 presents details about the logical techniques and how the evaluation with the overwriting patterns used by the logical techniques was performed to overwrite the original data stored on the hard disk, and finally, Section 5 presents the conclusion and the future work intended for continuation of this research.

3 Related Works

In [7] a study is presented on the use of logical techniques to sanitize data in old models of hard disks, with the subsequent recovery of these data applying the MFM technique. The author presents arguments that allow the stored data to be retrieved on the hard disk, even after the device has been sanitized, among which are: I) the modulations used to store the data facilitate its reconstruction. II) during the process of writing the data on the hard disk tracks, data may "overflow" out of the disk path, allowing these data segments that are off the track to be recovered. However, such assumptions are no longer valid in current hard disks, since the current modulations compress the data more intensively, making it difficult to recover. The process of recording the data is more accurate due to the increase of the available space in the plate of the disk for recording, thus avoiding that the original sanitized data can be recovered.

In [13], it is presented the need for users to overwrite their original data, to ensure that no attempt to recover data will be successful. The authors created a 100 KB file and then sanitized the location on the hard disk where the file was stored. They contacted 20 companies specializing in data recovery, however, only one company set out to try to recover the file. However, the authors did not carry out the experimental evaluation with the company to verify the possibility of recovery, keeping the question open.

In [14], a theoretical study on data sanitization and the use of the MFM technique for data recovery on hard disk is presented. The authors relate arguments against the possibility of data recovery using MFM. However, no experimental evaluation was performed, keeping the discussion open.

In [15] it was presented a study that proved the possibility of reconstruction of data using MFM and processing techniques. However, this data reconstruction was performed only with original data recorded on the surface of the hard disk plate and not after the sanitization process occurred on the original data, keeping the discussion open. Data sanitization is a constant concern in computing, and currently in the cloud computing paradigm, where hard disks are also used for the storage of user data. Data sanitization is a problem faced and it is necessary solutions for it [16], [17], [18], [19]. That is, as presented through the related works where the hard disk is used as a means of data storage, the challenge of data sanitization is probably a difficulty to be faced, showing the importance of this work that aims to present a possible solution to this open data sanitation challenge.

4 Hard disk overwrite patterns and MFM evaluation

In this section we present the overriding patterns used by the logical techniques to sanitize the data stored on the hard disks and we show details of how MFM technique was evaluated with the overriding patterns used by the data sanitization techniques.

Data sanitization is performed when a hard disk is not more intended to be used [20]. However, regardless of the technique used, its main purpose is to avoid that any individual to access the data after the sanitization process [12]. In the evaluation performed in this work it was used the Dimension Icon microscope [21], having been chosen due to its sample port to hold the entire plate of the hard disk.

4.1 Overriding patterns used in logical techniques to overwrite data

The main goal of data sanitization techniques is to overwrite the original data stored on the hard disk, thus avoiding the destruction of the physical device, allowing it to be used again. Data sanitization is performed when a hard disk is not more intended to be used. However, regardless of the technique used, its main purpose is to avoid that any individual to access the data after the sanitization process. During the research carried out in this work, it was verified that the main overwriting patterns used by the logical techniques are the 0, 1 and random patterns [14], [22], [23], [24], [25], [26], [27].

They will be the standards used in the evaluation through the MFM technique to verify the possibility of data recovery after the data sanitization process be performed on the hard disk. During the research to relate the logical techniques cited in the literature, we observed that the main difference of each technique is the number of steps to perform the data sanitization process on the hard disk. Table 1 presents some examples of logical techniques cited in the literature. In Table 1, the countries of origin

of some logical techniques are highlighted, the number of steps, the step considered the main step by the author, and the reference of the technique. MFM is a technique discovered in the early 1980s, providing high resolution images of magnetic fields through a specialized microscope, named scanning microscope [28]. MFM allows analyzing the pattern of bits recorded on high density media, such as those used in current hard disks, where the bit becomes smaller than the wavelength of the light and therefore optical techniques cannot solve it.

In this work, as well as in other works that uses the MFM to recover data stored in hard disk, the main attention is given to the images of amplitude and phase, since they present variations of the magnetic fields related to the data bits recorded in the plate of the hard disk, so the next images are related to these images in the samples that were the target of the experimental evaluation. The remaining images are available at http://s1126.photobucket.com/user/FernandoSbampato/library/Data%20sanitization%20Techniques?sort=4&page=0.

Table 1 – Overview of Logical Techniques.

Technique	Country	N. de Steps	Verification	Reference
Gutmann	New Zealand	35	0	[7]
VSITR	Germany	7	0	[22]
RCMP TSSIT OPS-II	Canada	7	1 (step 7)	[23]
CSEG ITSG-06	Canada	3	1 (step 3)	[24]
DoD 5220.22-M	EUA	3	1 (step 3)	[25]
AR 38-19	EUA	3	0	[27]
GOST R 50739-95	Russia	3	0	[26]
ISM 6.9.92	Australia	1	1 (step 1)	[25]

As shown in Table 1, different countries and researchers have developed data sanitization techniques, but always using the same patterns of data overwriting (0, 1 and random), in order to ensure that the original data cannot be accessed after the end of the data sanitization process occurred on the hard disk, thus, presenting the importance given to the subject.

4.2 MFM Evaluation

The main goal of the evaluation is to verify, both through new hard disks and hard disks used in a production environment, that, after the data sanitization process occurred by one of the (0, 1 and random) patterns of overwritten by a logical technique, the stored data cannot be recovered by applying the MFM technique. This analysis was performed comparing the images generated by the scans performed on the plates of the hard disks used as samples. Figure 1 shows the steps performed in the evaluation. On the forth new hard drive, the 512-bytes string was written and the random pattern was applied to the entire disk. On the 5th and 6th hard drives (hard drives used) the same steps were applied on the 4th hard drive. The goal of the 1st and 2nd hard drives was to obtain the resulting MFM image for values of bits 0 and 1, magnetically recorded on the plates of

the hard disks. In an analogous way, the 3rd hard disk had the goal of obtain the image of the MFM with the string of 512 bytes. Finally, the images of disks 4, 5 and 6 were used to analyze the possibility of recovering the data stored on the hard disk after the application of the random pattern (the 4th disk was a new disk, the 5th disk was 2 years old, and the 6th had 3 years of use). A clean room was used to open and remove the plates from the hard disks, ensuring that the environment was controlled to avoid contamination in the samples both by particles present in the air and by liquid / solid residues.

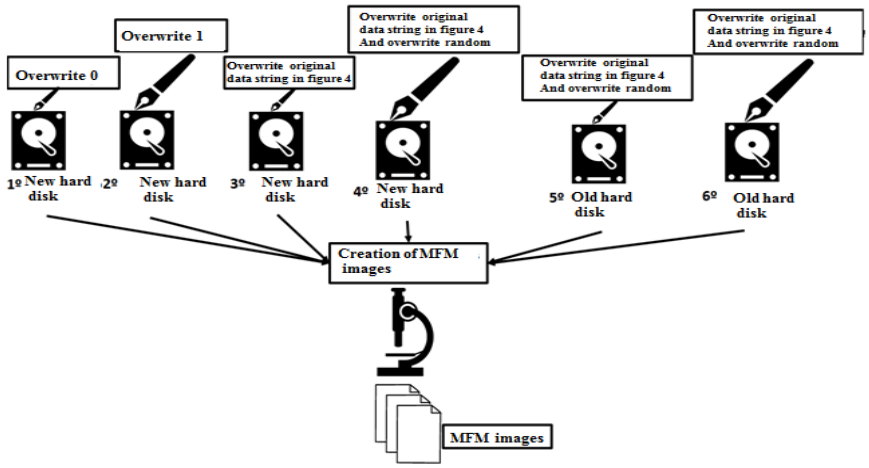


Fig. 1. Sequence of steps performed to perform the evaluation with the data override patterns.

On the first new hard disk, the binary value for the 0 standard was recorded on the entire hard disk, while on the second hard disk; the binary value for the standard 1 was recorded on the entire hard disk. On the third new hard disk, the 512-byte string shown in Figure 2 was written to all sectors of the hard disk.

As informações contidas neste arquivo são confidenciais e serão utilizadas para um teste experimental. Este teste será realizado por meio da utilização da técnica de microscopia de força magnética que visa possibilitar a leitura do campo magnético que esta na amostra do disco rígido. A amostra consiste em corta o prato do disco rígido de 1.5 diâmetro onde esta armazenada as informações que serão alvo da reconstrução. O objetivo e restaurar os dados quer foram sanitizados no disco rígido

Fig. 2. String in Portuguese recorded on the last 5 hard drives.

To perform the analysis accurately, it was necessary to estimate the size of the bit (physical space of the bit occupied in the hard disk disk) recorded on the hard disks used. As the 6 hard drives used in the experiment contained only one plate to record 500 GB of data, 250 GB (2147483648000 bits) are recorded for each side of the plate. Since each plate has a physical area of 38.4336 cm², it was possible to estimate the bit area by approximately 1.79 µm by a rule of three. This measure was necessary to configure the Scan Size used to quantify the amount of bits present in each image.

The 512-byte (4096-bit) string recorded on the hard drives has a total area of 7,331.84 mm², requiring a Scan Size of at most 86 µm, due to the fact that the image is configured in the geometric form of a square. However, images with this value of Scan Size for analysis of hard disk samples are not recommended, because this value is considered "large", not allowing the visualization of details of the data recorded in the samples. In this work, three Scan Size measurements (5, 10 and 20 µm) were used to observe the richness of the images that were analyzed to verify the possibility of data recovery using the MFM technique. Figure 3 shows the images related to the 1st sample where the 0 pattern was recorded on the entire hard disk, for Scan Size of 5 and 10 µm. In this sample, it was not possible to verify the presence of magnetic field, only disordered points were observed.

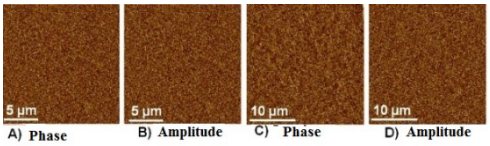


Fig. 3. Sample with pattern 0 recorded on the entire hard disk.

The goal of the MFM technique is to capture the variation of the magnetic field, not having this magnetic variation or this variation not being noticed during the sweep performed in the sample, only disordered points are observed. So, it is not possible to obtain an image of bit 0 recorded in the sample. Figure 4 shows the images of the second sample with the value of bit 1 recorded on the entire hard disk.

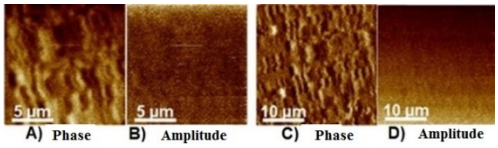


Fig. 4. Sample with pattern 1 recorded on the entire hard disk.

In this sample it was possible to verify the presence of the magnetic fields, to verify how the bit 1 is recorded in the disk plate and to obtain the corresponding image (Figure 4-A and C). Figure 5 presents a 3D image where it is possible to verify the topography of the region where pattern 1 was recorded on the hard disk plate of the phase image, showing the differences of the magnetic domains present in the sample. Figure 6 shows the images related to the 4th sample where the 512-byte string of Figure 2 was recorded. In Figure 6, it is possible to physically verify the separation between the tracks present in the hard disk plate, and the original data portions recorded in the sectors that compose the tracks of the disk, a segment containing information from the original string of Figure 2 is highlighted. Recorded portions that are different between the tracks and stretches that are similar have been verified. As a data sequence of 512 bytes was recorded in each sector of the hard disk, and since the tracks that make up the hard disk are concentric circles, the possibility of a segment as shown in Figure 6 was already expected to occur during the recording of the data.

According to [29], the value of bit 1 is greater than the value of bit 0, that is, the peaks present in Figure 6 correspond to bit 1 and the valleys correspond to bit 0. The mapped bit sequence found was at 00101101011001010110110110110101101001100 01001011100110101010101101011001 00, which corresponds to 70 bits. Using the ASCII table [30], to map the sequence of bits found, the stretch was mapped **in 1.5d** of the original string shown in Figure 2. Since for the last letter d, it was possible to identify two character possibilities, due to the fact that the last set of bits is composed of seven bits (011oo10), having two character possibilities which are: the character **d** corresponding to the bit sequence 01100100 and the character **e** corresponding to the bit sequence 01100101, as the sequence of data is known to identify the corresponding character occurred more easily. No other data was mapped due to the fact that the goal was only to verify the possibility of recovery of this recorded data and not the amount of data that could be recovered in the analysis, being this objective reached with this section mapped **in 1.5d**.

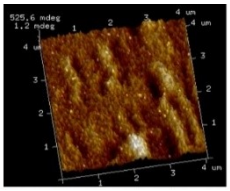


Fig. 5. Sample with pattern 1 recorded on the entire hard disk in 3D view.

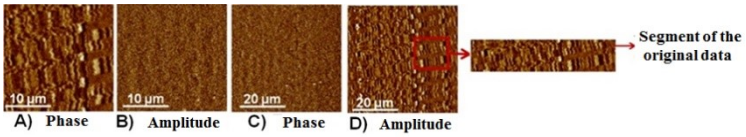


Fig. 6. Sample with original mapped data.

However, an attacker who does not know the sequence of the original data recorded on the hard disk would probably take a considerable amount of time trying to identify the corresponding sequence of characters. Another important fact is the type of data that is being retrieved, due to the fact that each type of file has a specific format, for example an .avi format file that corresponds to a video, this piece of recovered data would not make sense to the attacker. Another point is the amount of data that is being targeted for recovery. The images generated by the MFM technique present only bits of information, so to recover data from a file of, for example, 3.50 MB of .docx format, would require the recovery of 3,672,048 bytes. Only bits of recovered data are not sufficient to reconstruct a sequence of data that makes sense.

Several sequential images of the data recorded on the hard drive would be required to try "reconstructing" the data that was stored. For example, for the hard disks used in this research with its area of 38.4336 cm², would require 961 images using a Scan Size of 20 µm per image. The complexity involved in the entire process, from the creation of the images to the analysis and extraction of the data contained in the images, can

affect the final result, regarding the time taken to obtain the desired data. Another point that should be highlighted is the amount of files on the disks. In this research, a simple text file was used, but in a production environment, for example, a desktop of an office where different applications are executed, one would easily find several types of files stored, with different size and structures sharing the same disk.

Therefore, they would not be sorted in sequence in the sectors, but defragmented in the plate of the hard disk, making difficult its "reconstruction". For small bits of data, the use of the MFM technique can be justified. However, in the location required to recover data with larger amounts of bits, the time taken to extract the data could be a setback for the analysis, due to the lack of estimation of the time spent to reconstruct the data. Figure 6 shows the images generated by the scans performed in the 4th, 5th and 6th samples, where the 512-byte string of Figure 2 was recorded, and then applied to the random pattern throughout the hard disk. As seen in Figure 7, it was not possible to establish any bit sequence that is similar to some string bit sequence with the original data. Because the data is randomly overwritten, no recording pattern is respected, so values without logical meaning are simply overwritten. The result may even form words that express some meaning when a program is used to map the sequence of bits stored on the disk, however, these words are not enough to obtain any data related to the original data.

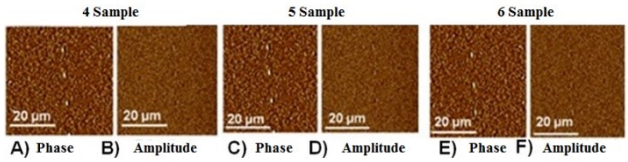


Fig. 7. Sample 4, 5, and 6 overwritten with the random pattern.

The images in Figure 7 prove that it is not possible to "reconstruct" the original data after they have been overwritten with the random pattern. Another fact is that the scan performed by the microscope that applies the MFM technique does not separate the "layers" of data previously recorded on the hard disk, but, it checks the variations of the magnetic fields of the last recording made in the plate of the device, that is, it is not possible to observe previous layers of recording in the images. For samples 5 and 6, where two hard disks used were used, the images were similar. Figure 8 presents the images resulting from the scans performed in these samples with a Scan Size of 20 µm, where it is possible to verify that for the hard disks in a production environment, the effect of the data sanitization is equivalent to those obtained in new hard disks, being possible to recover any remnant of the data. After this analysis, we observed the need to confirm again the veracity of the overwriting of the 0 pattern. In this way, the objective was to obtain new images for the 0 pattern to confirm the previous result. Thus, a 7th hard disk was prepared in the same manner as disks 4, 5 and 6 were prepared; Figure 8 shows the resulting images of this 7th sample, through Phase and Amplitude images.

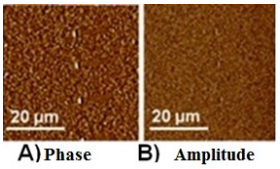


Fig. 8. Sample 7 sanitized with 0 pattern.

As observed in Figure 8, it was not possible to establish any relation with the original data string, similar to Figure 3, where it was also not possible to establish a relation that allows the reconstruction of the data with the original recorded string on the hard disk. The effect of overwriting with the random value on the seventh hard disk was the same as on disks 4, 5, and 6, that is, regardless of the logic technique used in this research, the overwriting effect is the same. The most important fact it was the data overwriting, in such a way that no part of the original data is left uncovered by the overwriting, thus, ensuring that it will not be reconstructed by the MFM technique. The effect of the overwriting is the same for new and used hard disks as shown in the samples used to carry out the experimental evaluation.

5 Conclusion and future Works

This work presented a study of the open problem on the possibility of restoring data using MFM after its sanitization, by a logic technique in current hard disks. Other works have already addressed the topic of data sanitization and the use of MFM for data recovery. However, this work has deepened on the subject, bringing both a logical evaluation of the related techniques and an experimental evaluation that evidences the safety in the use of the logical techniques in overwriting the data stored in hard disks.

In this way, a contribution was made regarding the security of the data stored in current hard disks, by means of the effectiveness verification of the logical techniques for data sanitization in current hard disks, ensuring that they will not be recovered through the use of the MFM (MFM images indicate that, when the original data is overwritten on the hard disk, no trace of the old data is retained). This result allows users to be more confident about the recovery of their data after the data sanitization process is successfully performed on the hard disk, since the use of logic techniques that use the 0, 1 and random patterns to sanitizing the data stored on the disks ensures the original data sanitization.

As future work we pretend to apply machine learning algorithms to support the process of identifying the bits presented in MFM images. The goal is to infer more complex patterns of bits representing the data recorded on the hard disk, making more complex and automated analyzes, thus enhancing the identification of the possible data sequences that are in MFM images. Such bit pattern identification mechanisms could then be used for an even more in-depth analysis of the data sanitization mechanisms. In the cloud computing scenario, the development of a framework for data sanitization is

proposed to demonstrate the use of the overwriting patterns used by the logical techniques, in an environment where data is being stored by several users at the same time, requiring higher security, due to the confidentiality that must be maintained to the data of the users throughout the life cycle of the data, including in its removal of the hard disks, when the user does not want to stop storing its data in this structure.

References

1. Fedders, John M., and Lauryn H. Guttenplan. "Document Retention and Destruction: Practical, Legal, and Ethical Considerations." Notre Dame Law.56, (1980)
2. Almorsy, Mohamed, John Grundy, and Ingo Müller. "An analysis of the cloud computing security problem." arXiv preprint arXiv:1609.01107, (2016)
3. Barreto, Gustavo Luis. "Use of anti-forensic techniques to ensure confidentiality" Curitiba, PUC.Available:http://www.ppgia.pucpr.br/~jamhour/Download/pub/RSS/MTC/referencias/TCC, (2009)
4. Data Recovery, https://datarecovery.wondershare.com/br/, (2017)
5. Thompson, David A., and John S. Best. "The future of magnetic data storage technology." IBM Journal of Research and Development 44.3 (2000)
6. Amazon, A. W. S. Amazon Web Services Overview of Security Processes, (2015)
7. Gutmann, Peter. "Secure deletion of data from magnetic and solid-state memory." Proceedings of the Sixth USENIX Security Symposium, San Jose, (1996)
8. Lindamood, J., Heatherly, R., Kantarcioglu, M., & Thuraisingham, B., Inferring private information using social network data. In Proceedings of the 18th international conference on World wide web (pp. 1145-1146). ACM. (2009)
9. NIST, U. Special Publication 800-88: Guidelines for Media Sanitization, (2006)
10. ENISA, Guidelines for SMEs on the security of personal data processing. https://www.enisa.europa.eu/publications/guidelines-for-smes-on-the-security-of-personal-data-processing/atdownload/fullReport, (2017)
11. Oracle Data masking best practice. http://www.oracle.com/us/products/database/data-masking-best-practices-161213.pdf, (2017)
12. Dong, Hao, She Kun, and Cheng Yu. "Research on secure destruction of digital information." Apperceiving Computing and Intelligence Analysis, 2009. ICACIA 2009. International Conference on. IEEE, (2009)
13. Bauer, Steven, and Nissanka Bodhi Priyantha. "Secure Data Deletion for Linux File Systems." Usenix Security Symposium. Vol. 174. (2001)
14. Wright, Craig, Dave Kleiman, and Shyaam Sundhar RS. "Overwriting hard drive data: The great wiping controversy." Information systems security, (2008)
15. Kanekal, Vasu. Data Reconstruction from a Hard Disk Drive using Magnetic Force Microscopy. University of California, San Diego, (2013)
16. Rojas, Marco Antonio Torrez, et al. "Inclusion of security requirements in sla lifecycle management for cloud computing." Evolving Security and Privacy Requirements Engineering (ESPRE), 2015 IEEE 2nd Workshop on. IEEE, (2015)
17. Lu, R., Zhu, H., Liu, X., Liu, J. K., & Shao, J., Toward efficient and privacy-preserving computing in big data era. IEEE Network, 28(4), 46-50. (2014)
18. Hashizume, Keiko, et al. "An analysis of security issues for cloud computing." Journal of Internet Services and Applications 4.1 (2013)

19. Brender, Nathalie, and Iliya Markov. "Risk perception and risk management in cloud computing: Results from a case study of Swiss companies." International journal of information management 33.5, (2013)
20. Lunsford, Dale L., Walter A. Robbins, and Pascal A. Bizarro. "Protecting information privacy when retiring old computers." The CPA Journal 74.7 (2004)
21. BRUKER, https://www.bruker.com/pt/products/surface-and-dimensional-analysis/atomic-force-microscopes/dimension-icon/overview.html, (2017)
22. Idaho, D. H. and W. (2017). Serves as successful test site for units. secure data sanitization becomes first in nation. http://www.securedatasanitization.com
23. Police, Royal Canadian Mounted. "Hard Drive Secure Information Removal and Destruction Guidelines." Retrieved September 25, (2007)
24. Government, O. C. Communications security establishment. https://www.csecst.gc.ca/en/system/files/pdfdocuments/itsg06-eng.pdf, (2017)
25. Hanson, Mark L. "The Regulation of Foreign Direct Investment in the United States Defense Industry." Nw. J. Int'l L. & Bus. (1988)
26. Dolak, John C., and Anne E. Dolak. "Information Systems Security and Privacy Issues in the Armed Forces." Computer L. Rev. & Tech. J. (2003)
27. Gonvindarajam, C. and Santhosh (2017). Communications security establishment. https://www.csecst.gc.ca/en/system/files/pdfdocuments/itsg06-eng.pdf
28. Grütter, P., Mamin, H. J., & Rugar, D. (1992). Magnetic force microscopy (MFM). In Scanning Tunneling Microscopy II (pp. 151-207). Springer Berlin Heidelberg.
29. Casey, Eoghan. Digital evidence and computer crime: Forensic science, computers, and the internet. Academic press, (2011)
30. Wessel, P., & Smith, W. H. Free software helps map and display data. Eos, Transactions American Geophysical Union, 72(41), 441-446. (1991)

DeSoftIn: Propuesta metodológica para desarrollo de software individual

Juan Sebastián Gonzaléz-Sanabria[1], Juan Antonio Morente-Molinera[2], Alexander Castro-Romero[1]

[1] Universidad Pedagógica y Tecnológica de Colombia, Tunja, Boyacá, Colombia
{juansebastian.gonzalez, alexander.castro01} @uptc.edu.co
[2] Universidad Internacional de La Rioja, Logroño, España

Abstract. Different Informatics Engineering undergraduate programs over the world are demanding to the students the presentation of works and, particularly, the presentation of a Grade project work that in the most of the cases is related to the development of software. However, in the moment of doing the projects, students find themselves with the problem of the methodology or model to use, as the existent software development methodologies includes team works and, for the case of Grade Works, these must be done individually. The above entails that the projects do not show as a result the proposed object, or take longer than the expected, among other difficulties. This work presents a methodological proposal for the development of individual software projects, named "DeSoftIn", principally on the academy, which allows accomplishment of the project objectives.

Keywords: agile methods; computer engineering; development methodology; software quality; software engineering.

1 Introducción

Desde la aparición de la ingeniería informática como disciplina se han planteado modelos, marcos y metodologías que presentan los "pasos básicos" o ideales para llevar adecuadamente los proyectos de desarrollo de software. Sin embargo, al no existir homogeneidad en factores como: los estilos de desarrollo, los grupos de trabajo, los recursos, entre otros, ha conllevado a que actualmente existan bastantes metodologías, en su mayoría enfocadas en equipos de trabajo (dos o más personas).

Adicionalmente, durante la formación académica de los estudiantes que se inclinan por esta disciplina, se ven continuamente en la obligación de realizar proyectos de manera individual, que en la gran mayoría de los casos se hacen sin seguir un conducto o una metodología. Por lo anterior, durante este proceso, tienen continuos tropiezos y errores que no son detectados sino hasta la fase de entrega del producto resultante.

Las metodologías usadas actualmente para desarrollo (eXtreme Programming - XP, Cascada, Iterativo, entre otros) proponen una conformación de equipos de mínimo 5 personas, radicando allí la mayor dificultad para aplicarlas en proyectos individuales.

Así mismo, en estas metodologías, cada persona del equipo cumple con una serie de funciones muy específicas, pero en varias de las fases no existe comunicación transversal entre las personas.

Por otra parte, los tiempos que se utilizan van en promedio de 15 a 30 días por entrega, es decir un lapso estimado de 90 a 120 días para disponer del producto final, tiempo del que no se dispone en el desarrollo de un proyecto en la academia, puesto que generalmente para estos proyectos se tiene de 30 a 60 días para el desarrollo completo. Igualmente, se requiere una inversión significativa a nivel de recursos físicos y financieros, los cuales no aplican en la mayoría de los desarrollos de software en proyectos de académicos.

Con base en las opiniones de expertos, se puede deducir que una metodología de desarrollo individual de software debe contar con todos los aspectos de calidad y de eficiencia propios de la puesta a punto de un producto desarrollado en equipos. Dicha metodología debe proveer las fases y los artefactos necesarios para brindar la versatilidad de las metodologías usadas tradicionalmente, cumpliendo con tiempos, objetivos y alcance definidos para el proyecto.

Con la motivación expuesta, en el presente trabajo se formula una propuesta metodológica de desarrollo de software, llamada "DeSoftIn" que permita, principalmente, a los estudiantes de la ingeniería informática tener un punto de referencia cuando deban trabajar de manera individual. Para lograrlo, inicialmente se dará una breve conceptualización sobre la teoría necesaria para el entendimiento del proyecto, posteriormente se presenta la metodología que se utilizó. Seguidamente se continua con una comparación de las principales metodologías de desarrollo, para finalizar con la metodología propuesta, y una serie de conclusiones y recomendaciones a tener en cuenta.

2 Metodología

Para el desarrollo de la investigación se definieron tres fases principales, en la primera se realizó una búsqueda de la literatura para la realización del estado del arte del tema de investigación. En la segunda, se seleccionaron las metodologías más utilizadas en la actualidad, para realizar una breve comparación de las mismas con base en casos de uso ampliamente divulgados y a experiencias de profesionales en el área. Por último, en la tercera fase, se plantea una propuesta de metodología para aplicar en proyectos individuales de desarrollo de software, presentándose los resultados obtenidos de su aplicación en un caso de estudio acompañados de una evaluación usando el método 4-DAT (4-Dimensional Analytical Tool) [1].

En la primera fase se realizó un estudio sistemático de investigaciones desarrolladas en los últimos cinco años en cuanto a metodologías de desarrollo de software y su aplicación en diferentes contextos, para lo cual se buscaron artículos en diversos índices, indizadores y bases de datos científicas como: Redalyc o IEEE Xplore. Posterior a tener un compendio de los artículos, se dio lectura a aquellos que tenían más

citaciones, con el objetivo de seleccionar los artículos mayormente relacionados con el objetivo del presente proyecto.

En la segunda fase, se seleccionaron y categorizaron, las metodologías de desarrollo de software, particularmente, las conocidas como ágiles, indagando en la literatura encontrada y con entrevistas a profesionales en el área. Lo anterior, con el fin de establecer una comparativa, que permita encontrar las falencias de dichas metodologías al momento de aplicarlas en proyectos desarrollados de manera individual.

Al haber identificado las ventajas y desventajas que presentan en proyectos individuales, las metodologías de desarrollo seleccionadas, se extrajeron las características más relevantes y con mayor acogida entre los desarrolladores. Y, gracias esto, se planteó una propuesta de metodología soportada en experiencias, casos de éxito y opiniones de expertos. Dicha propuesta metodológica se aplicó en un caso de estudio, con el fin de conocer su rendimiento frente a las metodologías de mayor trayectoria y reconocimiento. Adicionalmente, la propuesta metodológica fue evaluada haciendo uso del método 4-DAT.

3 Propuesta Metodológica DeSoftIn

En el presente apartado, se formula la propuesta metodológica, explicando las fases, roles, habilidades y destrezas necesarias para llevar a cabo de una mejor manera este tipo de proyectos.

3.1 Fases

1) Planificación y Análisis: La acción principal a realizar en esta fase es la definición del alcance del proyecto, esto debe ir acompañado del análisis de los requisitos, con el fin de establecer o estimar los tiempos, así como evaluar el conocimiento que se requiere sobre herramientas, técnicas y tecnologías a utilizar.

Para la planificación y el control de tiempos, es muy importante que en el análisis se diferencie entre lo que se debe y lo que se puede hacer, puesto que se es necesario tener en cuenta las limitaciones y restricciones del cliente, principalmente a nivel de recursos. Una vez se tenga claro este aspecto se procede a definir las actividades que se harán en cada uno de los sprints acorde a una priorización. Estas actividades se representarán en una línea de tiempo que, teniendo en cuenta el concepto de Last Planner System [2], se revisa desde el final hacía el inicio, con el fin de proveer y disponer de los recursos requeridos con antelación y no esperar hasta encontrarse con la necesidad para buscar el recurso.

Posterior a esto, se deben definir el plazo de realización del desarrollo incluyendo los tiempos necesarios para capacitarse o aprender sobre alguno de los aspectos requeridos y que no sean de dominio del desarrollador.

Es importante incluir, al igual que en todo desarrollo, la planificación de los riesgos, con el objetivo de identificar los responsables de aplicar las respuestas definidas para cada riesgo cuando la aplicación se encuentre en desarrollo.

Por último, se deben priorizar los requisitos, para empezar el diseño con base en los de mayor importancia y transversales a todo el proyecto, con el fin de presentar después de la primera entrega, un resultado funcional del mismo.

Es de resaltar que los requisitos, a diferencia de lo propuesto para la documentación de las demás metodologías, no se presentan con los formatos tradicionales, en cambio de esto, se propuso uso de una lista de chequeo de las funcionalidades requeridas por el cliente vinculándolas a los roles del sistema, como se presenta un ejemplo en la Fig. 1.

	Rol 1	Rol 2	…	Rol n
Requisito 1		X		X
Requisito 2	X			
…				
Requisito n	X			

Fig. 1. Lista de chequeo para la recolección de requisitos.

En el formato presentado se marca con una X aquellos usuarios que intervienen en una funcionalidad específica, lo que simplifica para el desarrollador el control unificado sobre permisos, roles y funcionalidades del proyecto.

2) Diseño: Definidos los requisitos en la anterior fase, en cada entrega del diseño que se realice se incluirán gradualmente, acorde a la prioridad de los mismos. Así mismo, en esta fase se debe recopilar y complementar la información necesaria para la óptima implementación de los requisitos.

Se deben realizar los diferentes diagramas, se sugiere el uso de BPMN (Business Process Model and Notation, llamado en español “Modelo y Notación de Procesos de Negocio”), sin embargo, se deja a libertad del diseñador la elaboración de un diagrama que otorgue una visión global del negocio. Adicionalmente, se deben efectuar los prototipos de interfaces, para luego ser validados con el cliente, y obtener su aprobación y perfeccionamiento para pasar a la siguiente fase.

3) Desarrollo: La fase de desarrollo es la que conlleva una mayor responsabilidad dentro de las fases iterativas, puesto que se deben “codificar” los requisitos con base en los prototipos aprobados, al fin de obtener un resultado funcional, y así mismo, es donde la autocrítica y experticia del desarrollador de vera a prueba.

Después de la primera entrega, o primer sprint, si se asocia con SCRUM, adicionalmente se incluyen las recomendaciones realizadas por el consultor y/o el cliente y de los requisitos propios a desarrollar en cada una de las entregas.

Tanto en el diseño como en el desarrollo se puede hacer uso del formato presentado en la Fig. 1, incluyendo una gama de colores para conocer el estado de avance del cumplimiento de cada requisito. Ejemplo de esto se presenta en la Fig. 2.

	Rol 1	Rol 2	…	Rol n
Requisito 1				
Requisito 2				

…				
Requisito n				

Fig. 2. Control de avance de requisitos.

Allí se usa el color verde cuando el requisito se encuentra desarrollado y aprobado por el cliente, naranja cuando está en fase de evaluación, amarillo cuando se encuentra en desarrollo y rojo para aquellas funcionalidades que no se han iniciado.

4) Implementación: Una vez superado el desarrollo, el prototipo debe ser implementado, validad y probado. En estos tres procesos se debe prestar atención, implementando recomendaciones y aspectos sugeridos en normas y estándares de modelos de calidad de software como ISO/IEC 15504 [3]. Igualmente, a nivel de las pruebas se sugiere guiarse por estándares y nomas de seguridad de la información, como la norma ISO 27000.

Se debe realizar la integración de las funcionalidades desarrolladas en cada entrega con las anteriores, y hacer las respectivas pruebas de calidad e integración de los mismos.

Es importante incluir en esta fase lo concerniente a la gestión de los riesgos, con el fin de ir estableciendo estrategias de monitoreo y control de los mismos. Igualmente, se deben tener en cuenta los impactos de cada uno de los riesgos establecidos, por lo que se requiere gran habilidad y conocimiento por parte del desarrollador, junto con una notable capacidad de comunicación para alertar al cliente de los mismos, sin generar alarma.

Por otra parte, en caso de que el software desarrollado deba integrarse con algún sistema, en esta fase se realizarán las correspondientes pruebas de integración durante cada una de las iteraciones con las respectivas funcionalidades que se desarrollen en la misma.

5) Evaluación: Al finalizar cada fase, que se sugiere sea de 15 días, debe realizarse una evaluación conjunta entre el "equipo" desarrollador y el cliente, con el fin de ir evaluando si lo desarrollado e implementado cumple con lo planteado, también se sugiere haya reunión con el consultor o asesor con el fin de que aporte opiniones que vean desde un punto de vista técnico la calidad del producto a entregar.

En la Fig. 3, se presenta el flujo entre las fases descritas.

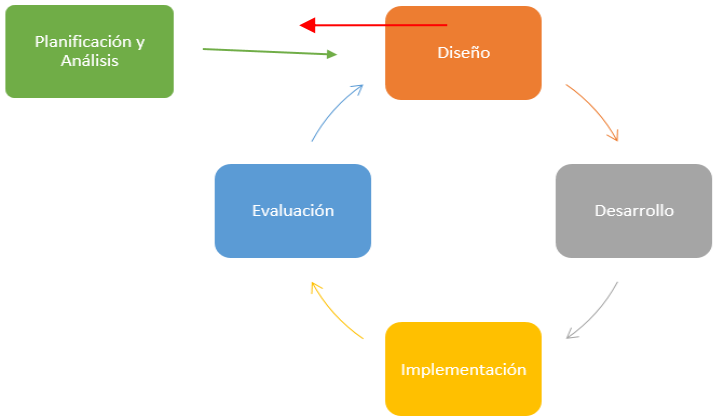


Fig. 3. Flujo de las fases de "DeSoftIn".

La duración de cada una de las iteraciones, o Sprint, estará dada por el desarrollador del proyecto. Sin embargo, se sugiere que no supere los 10 días, con el fin de que, al tratarse de un desarrollo individual, en este lapso se pueda realizar las observaciones por parte del asesor. Así mismo, para la duración de los sprints, se sugiere hacer uso de los tiempos definidos en la práctica de entregas continuas (de 3 a cinco días).

3.2 Roles

Al tratarse de proyectos de desarrollo de manera individual, es evidente hablar de roles, sin embargo, se proponen dos roles dentro de esta metodología, los cuales son el equipo de proyecto (siendo un equipo unipersonal) y el consultor.

1) Equipo del Proyecto: Para el caso del equipo del proyecto, en el desarrollo puntual de la metodología a proponer en el presente trabajo, se conforma puntualmente por una persona, en la cual recae la responsabilidad de las diferentes labores propuestas, como los son: jefe de proyecto, desarrollador, "tester", arquitecto, entre otros. Adicionalmente es la persona que tiene la responsabilidad para decidir cómo organizar el cumplimiento de los objetivos propuestos para cada iteración.

2) Consultor: Si bien, la metodología es propuesta de manera individual, se sugiere tener un colaborador externo con un conocimiento específico en el tema del proyecto. Lo anterior se incluye teniendo en cuenta, que la propuesta metodológica está orientada, en primera instancia, a desarrollos de pregrado, en la cual la labor de consultor, la puede realizar el profesor o tutor del trabajo. Dicho consultor, podrá aportar ideas y experiencias que contribuyan al producto. La labor así mismo, es menos activa, por ende, la dedicación es mínima.

3.3 Habilidades y destrezas

Es en este aspecto es donde mayor dificultad se presenta en la propuesta metodológica, puesto que, una misma persona debe contar con demasiadas habilidades, no solo técnicas, sino también personales. Entre las principales habilidades que deben resaltar en las personas que apliquen la metodología son:

- Comunicación activa y paciente para lograr abstraer y retener la información brindada por el cliente en la fase de requisitos. Así mismo debe estar en la capacidad de explicar cada una de las decisiones tomadas dentro del proceso con el fin de exponer de manera clara los resultados obtenidos.
- Capacidad de trabajo personal y sin supervisión, pues debe exigirse y controlar sus tiempos propios, obligando a tener un elevado nivel de disciplina.
- Excelente formación y conocimiento sobre las herramientas y técnicas a utilizar. Así mismo, debe poseer una "curva" o capacidad de aprendizaje rápida y efectiva, en caso de que se requieran realizar ajustes o cambios en alguna tecnología no contemplada en el proyecto inicialmente.
- Conocimiento sobre pruebas y tipos de pruebas de calidad y seguridad de software. Esta habilidad, es otro aspecto clave en esta propuesta metodológica, radicando su dificultad de aplicarla al ser el mismo desarrollador, el que en gran parte tiene que evaluar lo realizado para encontrar errores y falencias tanto a nivel de código como de funcionalidades, conllevando a generar conflictos de interés personal.
- Capacidad de determinar, gestionar y controlar riesgos en diferentes ambientes, incluyendo a nivel de código, desastres naturales, ataques, entre otros.

3.4 Artefactos y técnicas

A continuación, se presentan las técnicas y artefactos que complementan los formatos presentados en la Fig. 1 y Fig. 2.

1) Sprint de 5 a 10 días: Al hacer uso de Sprints cortos es posible realizar los cambios de una manera más pronta y así mismo, tener pequeñas versiones funcionales, al inicio, que permitan al cliente hacerse una idea del rumbo del producto a obtener. Adicionalmente, el tiempo de las entregas se define en este lapso, al ser tan cortos los plazos para los desarrollos individuales en la academia (menos de dos meses). Se considera oportuno y relevante hacer uso de los tiempos establecidos en la práctica de entregas continuas, que, permitirían un continuo control por parte del asesor y/o del cliente del proyecto.

2) Pausas entre Sprint: Teniendo en cuenta que el desarrollo estará trabajado por una sola persona, es conveniente definir un tiempo de pausa entre las entregas, con el fin de dar descanso a la persona y tenga la capacidad de mirar neutralmente el desarrollo que se hace, y de esta manera no sobrecargar de responsabilidades al desarrollador. En estas pausas se sugiere incluir o alternar el descanso con cursos que le permitan adquirir nuevas habilidades para aplicar en el proyecto.

3) Bitácora: Es necesario que el desarrollador lleve un diario o bitácora con las acciones que va realizando sobre la aplicación. Es importante que se realice esta acción, pues al no interactuar con otras personas, es la mejor manera de llevar un control eficiente sobre los cambios y las inclusiones que se vayan haciendo en el proyecto. Igualmente, es la manera de evaluar el cumplimiento y las acciones tomadas dentro del flujo del proyecto.

4) Tarjetas CRC: Las tarjetas CRC (Clase – Responsabilidad – Colaborador) permiten llevar un control sobre la asignación de responsabilidades y de cómo se presenta la colaboración con otros objetos. Se suele realizar una tarjeta por cada clase, resumiendo las responsabilidades de la clase y la lista de objetos con los que colaboran para poder funcionar [4].

5) Reuniones: Finalizada cada entrega, se debe realizar una reunión con el cliente y posteriormente con el consultor, con el fin de dar por terminado el Sprint, o planificar los ajustes necesarios. En las reuniones, así como en la planificación del proyecto, se debe tener en cuenta la estimación de la dedicación.

6) Estimación de la dedicación: Es de particular atención, pues al estar bajo la responsabilidad de una misma persona los roles de jefe de proyecto y de desarrollador, está sujeto a un elevado grado de disciplina. Es por ello que se debe tener clara la diferencia entre horas disponibles para trabajar y horas de dedicación, siendo la principal diferencia entre estas dos las horas o tiempo de distracción.
Para lo anterior, se sugiere tener en cuenta la fórmula (1) propuesta en [4]:

$$VE=DHD * FD \quad (1)$$

Donde DHD se refiere a los días-hombre disponible, FD al factor de dedicación y VE es la velocidad estimada de avance del proyecto. El factor de dedicación es una estimación, en el caso de desarrollos individuales, hace referencia al nivel de concentración del desarrollador en el proyecto, si este factor es bajo quiere decir que la persona es susceptible a distracciones e impedimentos (incluyendo distracciones familiares, personales, entre otras) que demorarían el tiempo de entrega del proyecto.

3.5 Herramientas

Si bien, en la mayoría de metodologías y propuestas metodológicas encontradas se propone el uso de determinadas herramientas para controlar y gestionar el proyecto, para la presente propuesta de metodología, se deja a libertad del desarrollador, con el fin de no sesgar el criterio del mismo, y no invertir tiempos en aprender otras herramientas con las que el desarrollador ya se sienta más cómodo.

4. Evaluación de "DeSoftIn" con 4-dat

El método 4-DAT evalúa, entre otros aspectos, que una metodología de desarrollo de software tenga en cuenta los principios del manifiesto ágil, es decir, que priorice las personas, es orientado a la comunicación, flexible (permita adaptarse fácilmente), rápida (rápido e iterativo con versiones funcionales del producto), eficiente (poco tiempo y buena calidad), adaptable (adecuada reacción a cambios), y aprende (mejoras durante y posterior al desarrollo) [5].

Pese a que 4-DAT propone 4 dimensiones, la evaluación se soportó principalmente en el análisis obtenido con los resultados de la dimensión 2, que evalúa la caracterización de la agilidad, mediante la comprobación de la existencia de agilidad en la metodología propuesta, tanto a nivel de proceso como a nivel de prácticas. En esta dimensión, se caracteriza la agilidad mediante el cálculo del grado de agilidad (DA) dependiendo de los términos flexibilidad (FY), velocidad (SD), eficiencia (LS), aprendizaje (LG) y adaptabilidad (RS), si alguna fase o práctica soporta una característica de agilidad particular, entonces se le asigna 1 punto en esa celda en particular, de lo contrario 0. Con base en los anterior, se obtienen los resultados presentados en la Tabla I.

TABLE I. EVALUACIÓN DE DESOFTIN CON 4-DAT

"DeSoftIn"	**FY**	**SD**	**LS**	**LG**	**RS**	**Total**
(i) Fases						
Planificación y Análisis	1	1	1	1	0	4
Diseño	1	1	1	1	1	5
Desarrollo	1	1	0	1	1	3
Implementación	1	0	0	1	1	4
Evaluación	1	0	0	1	1	4
Total	5	3	2	5	4	19
Grado Agilidad (DA)	5/5	3/5	2/5	5/5	4/5	19/25
(ii) Prácticas						
Desarrollo iterativo e incremental mediante iteraciones cortas	1	1	1	1	1	5
Programación en parejas	0	0	0	0	0	0
Pruebas	1	1	0	1	1	4
40 horas semanales	0	0	0	0	0	0
Rápida retroalimentación	0	1	0	1	1	3
Diseño Simple	1	1	0	1	0	3
Refactorización	1	1	1	1	1	5
Participación de todos los miembros del proyecto	1	1	0	1	1	4
Reutilización constante	1	1	0	1	1	4
Estándar de codificación	1	0	0	1	1	3
Reunión para examinar la iteración terminada y planificar la siguiente	1	1	1	1	0	4
Reporte de progreso	1	1	1	1	1	5
Disponibilidad del cliente	1	1	1	1	1	5
Uso de artefactos exclusivos	1	1	1	1	1	5
Total	11	11	6	12	10	50
Grado Agilidad (DA)	11/14	11/14	6/14	12/14	10/14	50/70

Al analizar los resultados obtenidos, se puede observar que una de las falencias de la metodología viene dada en lo correspondiente a eficiencia, resaltando que para el caso del estudio se dieron los valores más pesimistas en este sentido, es decir, asumiendo que la persona que realizará el proyecto debe adquirir conocimientos en algunas de las herramientas que se utilizarán en el proyecto.

En la Tabla II se presenta la comparación de los valores obtenidos para la propuesta de metodología respecto a XP y Scrum [3].

TABLE II. COMPARACIÓN DE XP, SCRUM Y DESOFTIN

	DeSoftIn	XP	SCRUM
Fases	0,76	0,70	0,60
Practicas	0,71	0,73	0,80
Promedio	0,74	0,72	0,70

Teniendo en cuenta la comparación realizada en la Tabla II, si bien DeSoftIn obtiene una calificación más relevante que SCRUM, es conveniente reiterar en el hecho de que SCRUM es un marco de trabajo, lo que le conlleva a tener una calificación baja en la parte de fases. En cuanto a las practicas, SCRUM sobresale notablemente, y XP es superior, lo que indica que deben incluirse o mejorarse las prácticas proyectadas para la propuesta.

Si bien, la calificación obtenida por la metodología propuesta, con base en la dimensión 2 del método 4-DAT, es llamativo el bajo nivel de eficiencia obtenido, lo cual se da por el panorama pesimista por el que se optó al evaluar la metodología, se complementan estas fallas, al presentarse una falta de control por un par y por el hecho de requerir mayor dedicación y compromiso a nivel de una sola persona.

5 Conclusiones

Pese a que solamente se presenta una propuesta metodológica de desarrollo de software, y se es consciente de que se requiere poner la misma a prueba bajo diferentes entornos con el fin de perfeccionarse, a continuación, se puede concluir lo siguiente respecto a la aplicación de "DeSoftIn":

- Menos, es más: es una de las premisas de la propuesta, pues se basa en tener exclusivamente la documentación necesaria, y sin el uso de formatos rígidos e innecesarios, en algunos casos. El uso de metodologías y modelos tradicionales se torna ineficiente al tener pequeños grupos de trabajo, o al trabajar proyectos a pequeña escala, por la cantidad de formatos, diseños, y demás artefactos que requieren bastante tiempo para su elaboración.
- Adicionalmente, el uso de lenguaje complejo, o incluso técnico dificulta en ocasiones la comprensión de la documentación por parte de terceros, o de personas del equipo que no hayan participado en la redacción. Por lo anterior, el

uso de una bitácora con las acciones que se vayan efectuando en el proyecto permitiría un control más simplificado.

- Cantidad no es calidad: en lo referente a los equipos de desarrollo, el tener equipos demasiado grandes genera dificultad en la comunicación, mientras que, en equipos pequeños, la comprensión es mayor y existe un mayor entendimiento ente los integrantes del mismo.
- ¿El cliente siempre tiene la razón?: Si bien es una concepción planteada durante bastante tiempo en el ámbito del comercio, para el caso de desarrollo de software, la opinión de expertos y profesionales en el área dice lo contrario, la mayoría de las clientes el cliente no sabe realmente que es lo que quiere, por lo que es necesario concientizarlo de lo que realmente se puede hacer y ayudarlo a ser realista con las expectativas.

  DeSoftIn responde a esta necesidad, y al no incluir la fase de Planificación y Análisis dentro del ciclo de desarrollo, se controla que no se presenten cambios abruptos en los requisitos. Sin embargo, es posible considerar la opinión y el criterio del cliente en cuanto a los diseños de las funcionalidades obtenidos en el análisis de los requisitos.
- Aprendizaje continuo: en los casos en los que las tecnologías y herramientas son impuestas por el cliente, y el desarrollador no posee los conocimientos de las mismas, requiere de un continuo interés por parte del mismo para aprenderlas en tiempos menores.

  Lo anterior, aparte de ayudar a ampliar el conocimiento en herramientas y técnicas, genera la necesidad de practicar y mantenerse al tanto de las novedades en los tiempos en los que no esté realizando algún proyecto.
- Desarrollo de cualidades: uno de los factores más criticados en el área de la computación es la insensibilidad de los profesionales en esta rama, sin embargo, con el uso de la propuesta metodológica presentada, se obtuvo, en el desarrollador, un sentido de responsabilidad y compromiso, al estar “en juego” su reputación, lo que adicionalmente, permite aprender a valorar el trabajo en equipo.
- Priorización de funcionalidades simples y cortas, y entregas continuas: esta premisa le brinda una sensación de aumento de la productividad, y le permite una aproximación constante al cliente con el sistema. Lo anterior, facilita que el cliente brinde aportes y opiniones que permitan el éxito del producto final.
- Disminución de riesgos: cuando se presentan entregas lejanas, la diferencia entre lo proyectado por el cliente y lo desarrollado puede llegar a distar bastante. Por lo cual, al primar entregas en lapsos cortos de tiempo se genera una mayor realimentación del producto en tiempo “real” y por ende el control y gestión rápido de los riesgos determinados.

  Si bien, los tiempos cortos son una ventaja de todas las metodologías ágiles, al hacerse uso de la práctica de entregas continuas, reduce el tiempo, pasando de tiempos de 15 días a dos o tres días e incluso menos, bajo la modalidad de lanzamientos.

Lo presentado, coadyuva a resaltar que DeSoftIn se convierte en un elemento de formación académica del estudiante del área de informática. Lo anterior, radica en que, a diferencia de las demás metodologías, modelos y marcos de trabajo propuestos para el desarrollo de software, se centra en un contexto académico.

Referencias

1. J. P. Gamboa y A. Rosado, «Diseño de un método ágil de desarrollo de software basado en XP, SCRUM, OPENUP y validado con la herramienta de análisis 4-DAT,» Ingenio, vol. 8, pp. 19-31, 2015.
2. L. F. Botero y M. E. Álvarez, «Last planner, un avance en la planificación y control de proyectos de construcción Estudio del caso de la ciudad de Medellín,» Ingeniería y Desarrollo, nº 17, pp. 148-159, 2005.
3. A. Alarcón-Aldana, J. S. González-Sanabria y S. L. Rodriguez-Torres, «Guía para pymes desarrolladoras de software, basada en la norma ISO/IEC 15504,» Revista Virtual Universidad Católica del Norte, nº 34, pp. 285-313, 2011.
4. E. Avila-Domenech, A. Abad y V. De la Cruz, «Delfdroid: metodología ágil de desarrollo de software para dispositivos móviles,» Revista INGENIERÍA UC, vol. 20, nº 3, pp. 59-70, 2013.
5. A. Qumer y B. Henderson-Sellers, «Measuring agility and adoptability of agile methods: a 4-Dimensional Analytical Tool,» de IADIS International Conference Applied Computing 2006, San Sebastián, España, 2006.
6. A. Qumer y B. Henderson-Sellers, «Comparative evaluation of XP and SCRUM using the 4d analytical tool (4-DAT),» de European and Mediterranean Conference on Information Systems (EMCIS) 2006, Costa Blanca, Alicante, España, 2006.

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Strategies for Conflict Resolution Among Multiple Norms in Multi-agent Systems

Eduardo Augusto Silvestre[1]and Viviane Torres da Silva[2]

[1]Federal Institute of Triângulo Mineiro, IFTM, Uberaba, Brazil
[2] IBM Research (on leave from UFF), Rio de Janeiro, Brazil
{eduardosilvestre@iftm.edu.br, vivianet@br.ibm.com}

Abstract. Norms are being used to regulate the behavior of the autonomous agents. Norms describe the behavior that the agent is obliged, permitted and forbidden to perform in the system. One of the main challenges on developing normative systems is that norms may conflict with each other. Norms are in conflict when the fulfillment of one norm violates the other and vice-versa. In previous works authors presented techniques do detect conflict among multiple norms. This paper complements previous works presenting strategies for resolving conflicts among multiple norms. The strategies were applied in a set of norms and the results discussed.

Keywords: multi-agent system; norm; conflict detection; conflict resolution

Introduction

Multi-agent systems (MAS) have been gaining great importance in the development of various applications. MAS are autonomous, and heterogeneous societies that can work to achieve common or different goals [1].

In order to deal with the autonomy and diversity of interests among different members, the behavior of agents is governed by a set of norms specified to regulate their actions[2]. The norms govern the behavior of agents by defining obligations (stating the actions that the agents must perform), prohibitions (stating the actions that the agents must not perform) or permissions (stating the actions that the agents can perform). An important challenge when implementing normative MAS is that the set of norms can be in conflict. Conflicts occur when norms regulating the same behavior are activated and are inconsistent [3]. In such cases, the agent is unable to fulfill all the activated norms.

There are several works that deal with normative conflicts, the approaches normally check for conflicts by analyzing the norms in pairs. Just recently[4–6] the detection of conflicts among multiple norms was addressed.For instance, let's consider a conflict that can only be detected if norms N1, N2 and N3 are analyzed together. N1 obliges agent A to dress red shirt. N2 forbids agent A to dress red pant. N3 obliges agent A to dress pant and shirt of the same color. There are no conflicts between the pairs N1-N2,

N2-N3 and N1-N3, but when the three norms are analyzed together, we can figure out the conflict.

There are several approaches that propose different mechanisms for the detection of normative conflicts. Besides detecting the conflicts, it is necessary to solve such conflicts in order to avoid unintentional violations, i.e., violation of norms that cannot be avoided by the agent. To the best of our knowledge, this approach was not addressed. This paper shows several conflict resolution techniques among multiple norms.

The remainder of this paper is organized as follows. In Section 0, a brief introduction is presented to the strategies of detecting conflicts among multiple norms Section 0 presents the various strategies developed for conflict resolution.Section 0 analyzes some of the main related works. Finally, Section 0 states some conclusions and future work.

Conflict Detection

The strategies used to detect conflicts were discussed in detail in previous works[4–6]. For contextualization, in this paper, only an introduction to the concepts used for conflict detectionwill be presented.

The norm definition is based on [7] but our representation is more complex and expressive. A norm n ∈Nrm is a tuple of the form: `(deoC, c, e, a, ac, dc)`, where deoC is a deontic concept from the set {O, F, P}, respectively, obliged, forbidden and permitted; c ∈ C is the context where the norm is defined; e ∈ E is the entity whose action is being regulated; a ∈ A is the action being regulated; ac ∈ Cd indicates the condition that activates the norm and dc ∈ Cd is the condition that deactivates the norm.

An action is defined by the name of the action and, optionally, its attributes and their values, an object where the action will beexecuted and a list of attributes (with their values). Thus, in this paper we define four different ways to represent the action: (i) action;(ii) action object; (iii) action (attribute1 = {value1}, attribute2 = {value2}, ...}) and (iv)action object (attribute1 = {value1}, attribute2 = {value2},...}).

In order to exemplify these four ways to describe a action, let's consider the following four prohibition norms: (i)Na = forbids agent A to get dressed;(ii)Nb = forbids agent A to dress pant; (iii)Nc = forbids agent A to dress red clothes and (iv)Nd forbids agent A to dress red pants.

The action of the norms can be represented as: (i)Na: {to dress}; (ii) Nb: {to dress pant}; (iii)Nc: {dress (color={red})} and (iv)Nd : {dress pant (color={red})}.

Our conflict checker algorithm is divided in three steps. First, all norms are transformed into permissions (details in[8]) in order to facilitate the analysis. Since all norms are permissions, the analysis made by the conflict checker is very simple, it checks if the norms intersect. Two or more norms intersects if there is at least one possible situation where the agent is able to fulfill all norms being analyzed.

In order to apply the transformation, we assume that if an agent is obliged to execute an action, it needs to be permitted. Therefore, the transformation from an obligation norm to a permission norm is direct. A prohibition is converted into a permission by (i)

negating the execution of the action being prohibited, (ii) negating the execution of the action over the object, (iii) and (iv) by inverting the values of the attributes. The second step of our conflict checker is responsible to filter the norms by including them into bags of similar norms. In order to do so, such step uses 3 filters. The filters separate into bags the norms that apply in the same context, govern the same entity and regulate the same action. After applying all filters, only the norms stored in the same bag are the ones that may be in conflict. Norms stored in different bags apply in different contexts, govern different entities or actions. The analyzes of the conflict is executed in the third step of the algorithm. The algorithm checks if the norms in each bag are in conflict. It starts by checking the norms by pairs of norms and then consider all possible group of k-norms until k be equal to the number of norms in the bag. At the end, the algorithm is checking for conflicts among all the norms of the bag at the same time.

For instance, let's consider the three norms described in Section 0. Since norm N3 is a complex norm (a norm applied over two objects), we have split it in two norms: N3a (obliges agent A to dress pants of color X) and N3b (obliges agent A to dress shirts of color X), where X is a generic color. In the first step of the algorithm, it transforms the three obligations into permissions. Then, in the second step, the algorithm groups all norms in the same bag since they are applied in the same context (we are omitting the context for simplicity), govern the same entity (agent A) and regulate the same action (to dress). In the third and last step, it verifies that there is a conflict among these three norm. The conflict exists because there is not an intersection among the norms, i.e., agent A is unable to dress a red shirt, a pant that is not red, and a shirt and a pant of the same color.

Strategies for Conflict Resolution

This section presents the strategies that have been developed to resolve the conflicts detected by the Conflict Checker. Section 0 presents the main strategies found in the literature used for conflict resolution among pairs of norms. Section 0 describes the strategies that have been created that can be used to resolve conflicts among multiple norms. Section 0 details the implementation of conflict resolution.

Famous Strategies for Conflict Resolution

Another important point in the study of normative conflicts is the resolution of the conflicts found. After the detection of the conflicts by the Conflict Checker a strategy of conflict resolution is necessary. To resolve the conflict two basic operations can be performed:

- Removal of one or more existing norms for conflict elimination
- Rewriting of one or more existing norms for conflict elimination

In summary, some of the main strategies found in the literature are

Removal of one or more existing norms for the elimination of conflicts.

Lex Posterior, Lex Superior and LexSpecialis[9][10][11]. In the case of Lex posterior, the newer norm takes precedence over older norms. In the case of the LexSuperior, the norm imposed by the greater power has priority. In the case of LexSpecialis, the more specific norm takes precedence.

Precedence of modality[12][13]. If the positive modalityis chosen then, in cases of conflicts, permission and obligation will overwrite the prohibition. If negative modalityis chosen then, in cases of conflict, the prohibition will overwrite the permission and the obligation.

Priority among the norms[14][15]. At design time, a priority is chosen for the norms arbitrarily. Norms with the lowest priority are excluded to resolve the conflict. In [14] authors define priority as the salience of the norm.

Rewritten one or more existing norms for the elimination of conflict.

Direct Conflict with Restrictions [3]. In this case, besides the normal definition of the norm, the representation of authors also has restrictions. The conflict is solved by manipulating these constraints; by manipulating the constraints, the conflicting intersections are eliminated.

Indirect Conflicts[3]. The strategy for resolving indirect conflicts is the same applied to direct conflicts described previously.

Strategies for Resolving Conflicts Developed

As the conflict among multiple norms was first addressed in the literaturerecently[4–6], no approach was also found for conflict resolution among multiple norms. Using the strategies presented previously as inspiration, the authors developed and implemented some conflict checking strategies among multiple norms. Strategies can be based on the removal of conflicting norms (Strategy 1, Strategy 2 and Strategy 3) or strategies can be based on changing conflicting norms (Strategy 4, Strategy 5 and Strategy 6).

Strategy 1. This strategy of conflict resolution is divided into two steps: removal of all norms with variables at a single time and application of the policy of conflict resolution among pairs of norms.

The first step is to remove all norms that have variables at a single time. Because conflicts among multiple norms only occurs if there is a variable, if these norms are excluded, there will automatically be no conflict among multiple norms. In this strategy, all norms are deleted at one time.

The second step of the strategy is to resolve existing conflictsamong pairs of norms. The strategyis applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Strategy 2. This strategy of conflict resolution is divided into two steps: removal of all norms with variables (one norm at a time) and application of the policy of conflict resolution among pairs of norms.

The first step is to remove all norms that have variables until conflicts among multiple norms are no longer found. Unlike Strategy 1, norms with variables are deleted in a process repeatedly, one at a time. A norm is chosen with variable that participates in more conflicts and this norm is removed. The Conflict Checker runs again until there is no more conflict among multiple norms. In this approach, it is not mandatory that all norms with variables be removed.

The second step of the strategy is to resolve existing conflicts among pairs of norms. The strategy is applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Strategy 3. This strategy of conflict resolution is divided into two steps: removal of all norms with greater participation in conflicts and application of the policy of conflict resolution among pairs of norms.

The first step is to remove norms that participate most in conflicts until there is no conflict among multiple norms. The norm that participates in the largest number of conflict combinations is chosen and is removed from the set. This process is repeated until there is no more conflict among multiple norms. The norm chosen may have variable or not.

The second step of the strategy is to resolve existing conflicts among pairs of norms. The strategy is applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Strategy 4. This strategy of conflict resolution is divided into two steps: change of all norms with variables (one norm at a time) and application of the policy of conflict resolution among pairs of norms.

The first step is to change the activation period for all norms that have variables at one time. Because the conflict among multiple norms only occurs if there is a variable, if these norms are changed, and no longer intercept, there will be no conflict automatically. In this strategy, all norms are changed at once.

The second step of the strategy is to resolve existing conflicts among pairs of norms. The strategy is applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Strategy 5. This strategy of conflict resolution is divided into two steps: change of all norms with variables (one norm at a time) and application of the policy of conflict resolution among pairs of norms.

The first step is to change the activation period of norms that have variables until no conflicts between multiple norms are found anymore. Unlike Strategy 4, norms with variables are changed in a process repeatedly, one at a time. It is chosen a norm with variable that participates of more conflicts and this norm has its period of activation changed. The Conflict Checker runs again until there is no more conflict among multiple norms. In this approach, it is not mandatory that all norms with variables be changed.

The second step of the strategy is to resolve existing conflicts among pairs of norms. The strategy is applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Strategy 6. This strategy of conflict resolution is divided into two steps: change of all norms with greater participation in conflicts and application of the policy of conflict resolution among pairs of norms.

The first step is to change the activation period of the norms that participate most in the conflicts until there is no conflict between multiple norms. The norm that participates in the largest number of conflict combinations is chosen and has its activation period changed. This process is repeated until there is no more conflict between multiple norms. The norm chosen may have variable or not.

The second step of the strategy is to resolve existing conflicts among pairs of norms. The strategy is applied by changing the activation period of the norms, so that the norms do not have more conflict. The norm that participates most in conflicts is chosen, its activation period is modified so that it does not intercept with any other norm and is

checked again if a new conflict still exists. This step is done repeatedly until there are no more conflicts among the input norms.

Implementation of Conflict Resolution.

In order to implement previously defined conflict resolution strategies, an input norms scenario was used. In this scenario the authors have manually defined 40 norms in a given domain (9 norms of TYPE (i), 9 norms of TYPE (ii), 9 norms of TYPE (iii) and 13 norms of type (iv)). The conflict checker detected 625 conflicts (6 conflicts among norms of TYPE (i), 6 conflicts among norms of TYPE (ii), 10 conflicts among norms of TYPE (iii), 510 conflicts among norms of TYPE (iv), 12 conflicts between norms of TYPE (i) and TYPE (ii), 12 conflicts between norms of TYPE (i) and TYPE (iii), 17 conflicts between norms of TYPE (i) and TYPE (iv), 0 conflict between norms of TYPE (ii) and TYPE (iii) (as expected), 17 conflicts between norms of TYPE (ii) and TYPE (iv) and 35 conflicts among norms of TYPE (iii) and TYPE (iv)). The program was able to detect these conflicts without identifying false positives and forgetting false negative conflicts. All the implementations were developed using the Java language and the project is available at https://goo.gl/QM5Sbe

The Table1presents a summary of the application of conflict resolution strategies implemented in Java. It is worth mentioning that the execution time of conflict resolution strategies was similar in all cases. In strategies that remove norms the total number of norms at the end is less than the number of norms at the beginning. Some strategies require a greater number of executions of Conflict Checker, in larger applications this requirement can be an impediment in terms of computational cost. The conflict resolution strategy to be used will depend on the need to maintain the number of original norms or not.

Table 7. Summary of implementation of conflict resolution strategies.

	Strategy 1	Strategy 2	Strategy 3	Strategy 4	Strategy 5	Strategy 6
Norms Initially	40	40	40	40	40	40
Norms TYPE(i)	9	9	9	9	9	9
Norms TYPE(ii)	9	9	9	9	9	9
Norms TYPE(iii)	9	9	9	9	9	9
Norms TYPE(iv)	13	13	13	13	13	13
Conflicts Initially Detected	625	625	625	625	625	625

Conflcits Detected at the end	0	0	0	0	0	0
Time for Resolution	5978ms	6124ms	6174ms	5279ms	6724ms	6250ms
Norms at the end	33	33	35	40	40	40
Executions of Conflict Checker	13	16	15	13	16	15

Related Work

There are many approaches to detect and solve normative conflicts. These approaches were nicely summarized in [16]. This section will focus in conflict resolution. Some of the approacheswere specified to be used at *design time* while others at *runtime*. In addition, some of them are able to solve only *direct* normative conflicts while others are also able to solve *indirect* conflicts.

The strategies used by the analyzed proposals can be divided into two kinds: norm prioritization (one norm overrides another in particular circumstances) and norm update (one of the norms in conflict is updated).

As an example of norm prioritization, [17] present two norms: (N1) a Christian ought not kill his neighbor; and (N2) if a soldier is ordered to kill an enemy, then he ought to kill him; and assume that there is an individual that is a Christian soldier who received the order to kill. Then, norms N1 and N2 are addressed to this individual and there is a normative conflict between N1 and N2. The strategy applied to resolve the conflict consists in deciding which norm is more relevant based on the role. In this example, the norms addressed to the role soldier are stated to be more relevant than norms addressed to the role Christian. Thus, norm N2 takes precedence on norm N1.

A situation of norm update is found in [3], where norms regulate parameterized actions. The authors present two conflicting norms: (N1) agent1 is forbidden to deploy (X), where $5 \leq X < 10$; and (N2) agent1 is obliged to deploy (X), where $8 < X < 10$. The mechanism used for resolving the conflict updates a norm by modifying the values of its constraints in order to reduce the scope of the norm and eliminate the conflict

The majority of proposals establish an order of prioritization between norms to specify which norm is more relevant. In this sense, there are three classic principles found in the literature [3] that have been used to solve deontic conflicts: *lex posterior* (it prioritizes the most recent norm), *lexspecialis*(it prioritizes the most specific norm), and *lex superior* (it prioritizes the norm imposed by the most important issuing authority). Other approaches reduce the scope of influence of the conflicting norms in order to eliminate the overlap between them. They do so by manipulating the components of one of the norms.

Conclusions and Future Work

Research in the area of MAS has strongly increased in recent years, in particular, research in normative systems. Despite significant, several research problems have not yet been addressed, there are still many challenges to be considered.

After an extensive literature search, several articles were found on the identification and resolution of normative conflicts. Such works focus on the identification and resolution of normative conflicts by considering pairs of norms. However, there are conflicts, as the ones exemplified in the text, which can only be detected and resolved when checking for conflicts among multiple norms.

The paper discusses and implements several strategies for conflict resolution. Two groups of strategies were applied. Strategies 1, 2 and 3 remove norms and apply the policy of conflict resolution among pairs of norms. Strategies 4, 5 and 6 change norms and apply the policy of conflict resolution among pairs of norms. After applying the strategies, the initial set of norms will be free of conflicts. The strategies used appear as a first alternative for further research on conflict resolution among multiple norms.

In this work, we have not considered indirect conflicts [18]. The algorithm presented in this paper does only check for direct conflicts, i.e., conflicts among norms that have the same entities, contexts and actions. An important and necessary extension of our work is the identification of indirect conflict among multiple norms.

References

1. Wooldridge, M.: An Introduction to MultiAgent Systems. Wiley Publishing (2009).
2. da Silva, V.T.: From the specification to the implementation of norms: an automatic approach to generate rules from norms to govern the behavior of agents. Auton. Agents Multi-Agent Syst. 17, 113–155 (2008).
3. Vasconcelos, W.W., Kollingbaum, M.J., Norman, T.J.: Normative conflict resolution in multi-agent systems. Auton. Agents Multi-Agent Syst. 19, 124–152 (2009).
4. Silvestre, E.A., da Silva, V.T.: Verifying Conflicts Among Multiple Norms in Multi-agent Systems: (Extended Abstract). In: Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems. pp. 1383–1384. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC (2016).
5. Silvestre, E.A., da Silva, V.T.: An Approach to Verify Conflicts among Multiple Norms in Multi-agent Systems. In: 2016 {IEEE/WIC/ACM} International Conference on Web Intelligence, {WI} 2016, Omaha, NE, USA, October 13-16, 2016. pp. 367–374 (2016).
6. Silvestre, E.A.: Verificaçãoo de Conflitos entre Múltiplas Normas em Sistemas Multiagentes, (2017).

7. da Silva, V.T., Braga, C., Figueiredo, K.: A Modeling Language to Model Norms. Coord. Organ. Inst. Norms Multi-Agent Syst. AAMAS2010. 25 (2010).
8. von Wright, G.H.: Deontic Logic. Mind. 60, 1–15 (1951).
9. Leite, J.A., Alferes, J.J., Pereira, L.M.: Multi-dimensional Dynamic Knowledge Representation. In: Proceedings of the 6th International Conference on Logic Programming and Nonmonotonic Reasoning. pp. 365–378. Springer-Verlag, London, UK, UK (2001).
10. Kollingbaum, M., Norman, T.: Strategies for resolving norm conflict in practical reasoning. ECAI Workshop Coord. Emergent Agent Soc. 2004. (2004).
11. Lopes Cardoso, H., Oliveira, E.: Norm Defeasibility in an Institutional Normative Framework. In: Proceedings of the 2008 Conference on ECAI 2008: 18th European Conference on Artificial Intelligence. pp. 468–472. IOS Press, Amsterdam, The Netherlands, The Netherlands (2008).
12. Moffett, J.D., Sloman, M.S.: Policy conflict analysis in distributed system management. J. Organ. Comput. Electron. Commer. 4, 1–22 (1994).
13. Kagal, L., Finin, T.: Modeling conversation policies using permissions and obligations. Auton. Agents Multi-Agent Syst. 14, 187–206 (2007).
14. García-Camino, A., Noriega, P., Rodríguez-Aguilar, J.-A.: An Algorithm for Conflict Resolution in Regulated Compound Activities. In: Proceedings of the 7th International Conference on Engineering Societies in the Agents World VII. pp. 193–208. Springer-Verlag, Berlin, Heidelberg (2007).
15. Governatori, G., Rotolo, A.: Justice Delayed Is Justice Denied: Logics for a Temporal Account of Reparations and Legal Compliance. In: Leite, J., Torroni, P., Ågotnes, T., Boella, G., and van der Torre, L. (eds.) Computational Logic in Multi-Agent Systems. pp. 364–382. Springer Berlin Heidelberg (2011).
16. Santos, J.S., Zahn, J.O., Silvestre, E.A., Silva, V.T., Vasconcelos, W.W.: Detection and resolution of normative conflicts in multi-agent systems: a literature survey. Auton. Agents Multi-Agent Syst. 1–47 (2017).
17. Cholvy, L., Cuppens, F.: Solving Normative Conflicts by Merging Roles. In: Proceedings of the 5th International Conference on Artificial Intelligence and Law. pp. 201–209. ACM, New York, NY, USA (1995).
18. da Silva, V.T., Zahn, J.: Normative Conflicts that Depend on the Domain. In: Coordination, Organizations, Institutions, and Norms in Agent Systems {IX} - {COIN} 2013 International Workshops, COIN@AAMAS, St. Paul, MN, USA, May 6, 2013, COIN@PRIMA, Dunedin, New Zealand, December 3, 2013, Revised Selected Papers. pp. 311–326 (2013).

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Evaluación de la Calidad de Servicios de Software y la Satisfacción del Cliente Interno

Elizabeth Jeinson, Carlos Salgado, Alberto Sanchez, Mario Peralta

Departamento de Informática - Facultad de Ciencias Físico-Matemáticas y Naturales
Universidad Nacional de San Luis
Ejército de los Andes 950 – C.P. 5700 – San Luis – Argentina
e-mail: ejeinson@gmail.com;
{csalgado, alfanego, mperalta}@unsl.edu.ar

Resumen. La satisfacción del cliente es una idea central en todas las definiciones de calidad. La medición de la satisfacción del cliente para evaluar la calidad de los servicios tiene un gran desarrollo en las décadas del ’80 y del ’90. La mayoría de los autores se enfocaron en el cliente externo desde la perspectiva de mejorar la oferta de los servicios. Se desarrollaron algunos modelos que siguen siendo muy aceptados y sobre ellos se basa este trabajo. El cliente interno tuvo mucho menos atención entre los investigadores. Sin embargo, el cliente interno es un gran consumidor de servicios de software en las organizaciones. Este trabajo propone una adaptación de estos modelos que permita evaluar la satisfacción del cliente interno en servicios de software y tecnologías de la información.

Palabras clave: Calidad de Servicio, Cliente interno, Servicios TIC [Tecnologías de la Información y la Comunicación]

Introducción

Las áreas de soporte dentro de las organizaciones tienen como objetivo proveer bienes y servicios a otras áreas para que puedan realizar los procesos productivos o aquellos que hacen a la esencia del negocio de la empresa. Las áreas de sistemas y tecnología son usualmente áreas de soporte que tienen como cliente interno a todos los empleados de la organización. En ocasiones, también a los clientes de la empresa, si éstos hacen uso de sistemas de autogestión en soporte web o móvil.

Se denomina cliente interno a los empleados que son consumidores de servicios que presta otra área de una organización. El cliente interno, a diferencia del cliente externo, no tiene opciones de seleccionar proveedores ya que son provistos por la organización. Esta situación lo convierte en un consumidor cautivo de la oferta del proveedor [1]. Ante una demanda creciente las áreas que administran sistemas y tecnología necesitan tener clientes internos satisfechos. En gran medida tener clientes satisfechos y buena imagen, habilita mejores presupuestos que pueden luego ser invertidos en actualización

tecnológica y entrenamiento. Esto se logra con una buena calidad de servicio. Uno de los criterios fundamentales por el cual las empresas eligen sus proveedores de tecnología ya sea internos o externos es por su calidad [2]. Desde este punto de vista, al analizar la literatura, se observó que la medición de la satisfacción del cliente para evaluar la calidad de los servicios tiene un gran desarrollo, sin embargo, la mayoría de los autores se enfocaron en el cliente externo desde la perspectiva de mejorar la oferta de los servicios. En este sentido, se desarrollaron algunos modelos que siguen siendo muy aceptados como SERVQUAL (Service Quality), SERVPERF (Service Quality Performance Based), RSQS (Retail Service Quality Scale), Service Quality Model, entre otros, pero que no prestan atención a la calidad de servicio hacia el cliente interno. Con esta motivación, se pretende revisar los modelos de calidad de servicios que son generales y proveer una adaptación a las particularidades de los servicios TIC para clientes internos. Para ello, en un primer paso, se tomó el modelo SERVQUAL y se realizó una encuesta que permitió detectar las debilidades de dicho modelo en cuanto a la calidad de los servicios prestados.

El presente trabajo se organiza de la siguiente manera: en la sección 2 se presenta una revisión de la literatura relacionada a los modelos de calidad de servicios software. La sección 3 se describe la propuesta de investigación presentada, la sección 4 delinea la metodología aplicada y en la 5 se presentan los resultados de un caso de estudio. Finalmente, en la sección 6 se delinean las conclusiones y trabajos futuros.

Revisión de la Literatura

Las primeras definiciones de calidad surgieron en torno a calidad de producto y de proceso, de autores vinculados a la industria como Juran [3], Deming [4], las normas ISO [5], donde la satisfacción del cliente fue un aspecto esencial para entender la calidad. El desarrollo de la calidad en los servicios planteó algunas diferencias intrínsecas a la naturaleza de los servicios [6]: la intangibilidad, la heterogeneidad y la inseparabilidad de la producción al consumo del servicio. Estas características hacen que la calidad de servicios sea más difícil de evaluar que la calidad de productos [7]. Los investigadores confluyeron en la idea que la calidad de servicios involucra una comparación entre las expectativas del cliente y el desempeño del servicio entregado. Este proceso comparativo se denomina el "paradigma de la desconformación" [8] donde el resultado de la experiencia del cliente se denomina "desconformación positiva" si su experiencia supera sus expectativas y "desconformación negativa" si sus expectativas no se ven colmadas. El modelo de Calidad de Servicio de Grönroos [9] compara el servicio que el cliente espera respecto de la percepción del servicio recibido. Esa comparación tiene dos aspectos: *la calidad técnica*, que se ocupa de los "qué" y *la calidad funcional*, referida a los "cómo". En esta misma línea se desarrolla el modelo SERVQUAL [10] que establece una serie de brechas entre las expectativas del cliente y las tareas de entregar el servicio, que determina la calidad del servicio a los ojos del cliente.

Pero entonces, ¿la calidad del servicio y la satisfacción del cliente es lo mismo? Se impone la idea que la calidad es una actitud. Esta idea es sostenida por los autores de SERVQUAL donde afirman que un servicio y sus proveedores deben tener características que proyecten una alta imagen de calidad [11]. Mientras que satisfacción se define como: estado psicológico resultante de la sensación de expectativas frustradas en combinación con sentimientos anteriores del consumidor sobre la experiencia de consumo. Esta definición, vincula la satisfacción a una transacción específica, mientras que la actitud del consumidor respecto de la calidad tiene que ver con una orientación afectiva acerca de un producto o servicio. Es decir, puede pasar que haya satisfacción con una transacción en particular, y aun así la percepción de la calidad del producto/servicio sea baja. Entonces, son dos construcciones relacionadas donde la incidencia de la satisfacción con el paso del tiempo resulta de la percepción de la calidad del servicio. Desde el punto de vista del cliente, hay autores que sostienen que son construcciones distintas [12] dependiendo de la dimensión y el ítem que está evaluando: la satisfacción se relaciona con el tiempo de respuesta, la recuperación del servicio y el entorno físico, mientras que la calidad del servicio la vinculan con el precio, la experiencia y el trasfondo del servicio. Por tratarse este trabajo de clientes internos se exploraron trabajos relacionan la calidad del servicio con los clientes internos. La idea de atender la satisfacción del cliente interno se expresa en un trabajo de Vandermerwe and Gilbert [13] donde surge con fuerza la idea del marketing interno. Un sistema de servicios para el cliente interno basado en un intercambio eficiente entre áreas y departamentos es clave para cualquier iniciativa de calidad total y mejora continua. También surge el concepto de "Cadena de beneficios del servicio" [14] que es un modelo causal donde se establecen relaciones entre la calidad del servicio interno y la satisfacción del empleado en la forma de retención y productividad. Estos valores derivan en un servicio externo de valor que traerán la satisfacción del cliente, su lealtad y, en consecuencia, crecimiento y beneficios a la empresa. La cadena de beneficios del servicio es un modelo muy referenciado en toda la literatura del marketing, cada vez que es útil vincular la calidad del servicio final con empleados satisfechos. Un modelo adicional que es pertinente citar, surgido de la escuela de negocios de Harvard [15], afirma la validez de un marco de trabajo, llamado "Capacidad del Servicio". Se define como capacidad de servicio a la percepción del empleado que tiene la habilidad necesaria para servir al cliente. El trabajo establece que la Calidad del servicio interno influye en la satisfacción del cliente a través la capacidad del servicio. Trabajos posteriores indican que la satisfacción del cliente interno tiene una influencia significativa en la creación de confianza positiva en una organización [16]. En el negocio del conocimiento y la información, la confianza, integridad y credibilidad son muy valorados.

Propuesta de la Investigación

SERVQUAL es uno de los modelos más utilizados y versionados con alternativas propuestas sobre la idea fundante que propone [17]. SERVQUAL desarrolla un marco

de 5 categorías con 22 ítems que permiten evaluar abarcativamente todo tipo de servicios, según se muestra en la Figura 1. Las dimensiones e ítems siguen siendo el pilar en que otros autores basan sus investigaciones, según lo indican trabajos de comparaciones de modelos y dimensiones [18]. Y en general es uno de los más utilizados por autores que investigan calidad de servicio y satisfacción del cliente en diversos tipos de servicios por ejemplo: Sector bancario de Nepal [19], servicio de aerolíneas en Reino Unido [20] y en servicios de educación privada en Pakistán [21], por citar algunos o autores que reconocen que por razones de disponibilidad de datos no lo utilizaron pero deberían haberlo hecho, como el caso de un trabajo sobre el sector bancario minorista en Vietnam [22].

Sobre las categorías/ítems/componentes que presenta este modelo, respecto de la determinación de calidad de servicio y satisfacción del cliente, se realizará un análisis exploratorio en una empresa de servicios financieros que realiza encuestas entre los empleados acerca de su satisfacción con el desempeño de dos áreas TIC. Este análisis está orientado a responder las siguientes preguntas:

1. *¿Qué entiende una organización que es importante conocer respecto de la satisfacción de sus clientes internos usuarios de los servicios TIC?*
2. *¿Qué aspectos esenciales considera el empleado/cliente interno deben ser tenidos en cuenta para evaluar su satisfacción con los servicios de software y tecnología que le son provistos por su organización?*
3. *¿Qué adaptaciones se puede proponer al modelo SERVQUAL para que refleje la satisfacción de los clientes internos respecto de los servicios de software y tecnología?*

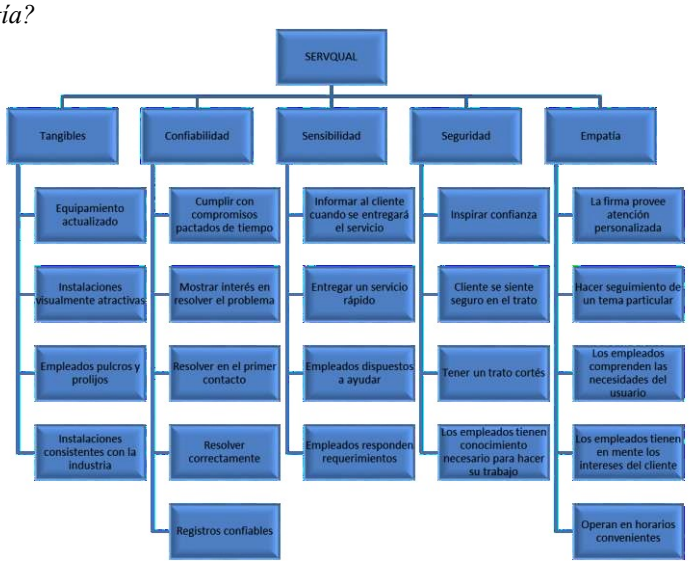


Fig. 1: Modelo SERVQUAL [10], [23]

Metodología

Para responder las preguntas se utilizó el instrumento de la encuesta. Esta encuesta tiene por finalidad evaluar los servicios internos que son prestados a sus empleados, con la idea subyacente de la mejora continua. La encuesta en cuestión se analizó contra el modelo SERVQUAL, determinando algunas adaptaciones por las características del cliente (interno) y del servicio (software y tecnologías de la información), aportadas desde la perspectiva de la organización.

La encuesta está compuesta por preguntas de evaluación y observaciones abiertas, como se muestra en el ejemplo de la Figura 2. Los participantes de ésta son empleados (clientes internos) que consumen servicios de software y tecnología. Se identificó el puesto de los empleados entre aquellos que tienen responsabilidad de gestión y personal a cargo de los que no. Los prestadores del servicio son áreas internas de la misma empresa. Las preguntas abiertas se analizaron cualitativamente mapeando el discurso de los clientes internos con el modelo SERVQUAL, respondiendo de esta manera a la segunda pregunta. El resultado de ambos análisis permitirá responder la tercera pregunta.

Personal de Contacto		Amabilidad
		Idoneidad
		Actitud de Servicio
		Rapidez
Proceso	Capacidad Operativa	Procedimiento
		Formularios útiles
		Formularios disponibles
		Soporte Técnico
		Atención de reclamos
	Producto Final	Calidad
		Cumplimiento del tiempo
		Comunicación
Percepción de la atención		Satisfacción

Fig. 2. Modelo de Encuesta

Caso de Estudio

El modelo propuesto, basado en SERVQUAL, tenía que ser validado. Para ello, se aplicó en una empresa financiera como caso de estudio. Esto era parte del plan de trabajo, puesto que los directivos de la misma tenían la necesidad de información acerca de la satisfacción de sus empleados respecto de las tecnologías e instrumentos que estaban utilizando. Y previendo que tal vez tenían que actualizarse tecnológicamente a la brevedad, por los constantes cambios actuales y, sabiendo que iban a tener que satisfacer o mantener la opinión de sus clientes, era también prudente tener capacitados

y actualizados a sus empleados, quienes debían estar conformes para poder brindar un buen servicio.

Para el caso de estudio, se analizó la estructura de encuesta y su realización en la organización financiera, de alrededor de 3200 empleados. La empresa fijó objetivos a cada área respecto de la nota a obtener en estas encuestas de satisfacción. Las encuestas son anónimas y aleatorias para todos aquellos que contactaron las áreas que dan los servicios de software y tecnología en el trimestre.

Los empleados debían evaluar cada aspecto con una escala del 1-10 [10 es la mejor puntuación y 1 la peor]. En cada categoría/ítem se puede agregar un comentario abierto. Se analizaron 198 encuestas de 3 trimestres para dos áreas prestadoras de servicios de software y tecnología de la información. El área A, por las características de su trabajo, fue evaluada por empleados de nivel de supervisión y jefes, mientras que el área B, por empleados sin responsabilidad de conducción. La empresa fijó como objetivo obtener un puntaje de 9.0 en las evaluaciones de satisfacción.

El análisis de los comentarios se realizó para un trimestre, referido a las dos áreas encuestadas. Los comentarios no son obligatorios, por lo tanto, sólo respondieron los empleados que se sintieron motivados a hacerlo.

5.1 Análisis de Resultados

Del análisis de la encuesta elaborada por la empresa vs. el modelo SERVQUAL surgen las siguientes observaciones:

- La organización no considera necesario evaluar la dimensión de “Tangibles” en ninguna de sus categorías.
- La dimensión “Empatía” es sólo evaluada en aquellas categorías que tienen que ver con el interés que demuestra el prestador de servicio en las necesidades del cliente.
- El modelo SERVQUAL no tiene ninguna categoría que evalúe las cualidades del proceso o procedimiento utilizado.

Los resultados y puntajes obtenidos de realizar la encuesta se muestran en la Tabla 1.

Tabla 1: Resumen de las respuestas a las encuestas de tres trimestres para dos áreas que prestan servicios de software y tecnología.

	Area A			Area B		
Categoría	Promedio	Moda	Varianza	Promedio	Moda	Varianza
Amabilidad	9,31	9	0,47	9,21	9	0,50
Idoneidad	8,83	9	1,01	8,97	9	0,83
Actitud de Servicio	8,94	9	0,74	9,02	9	0,73
Rapidez	8,29	9	1,18	8,64	9	1,10
Procedimiento	8,32	9	0,98	8,70	9	0,78
Utilidad de los Formularios	8,95	9	0,81	8,98	9	0,80
Disponibilidad de los Formularios	9,25	9	0,53	9,15	9	0,65
Soporte Tecnico	8,40	9	0,96	8,70	9	0,94
Solución de Pedidos y/o reclamos	8,33	9	0,84	8,58	9	0,86
Calidad	8,72	9	1,17	8,84	9	1,08
Conformidad con el Tiempo Pactado	8,15	8	0,89	8,58	9	0,79
Comunicaciòn con el Cliente	8,81	9	0,66	8,91	9	0,60
Satisfacción general	8,29	8	0,75	8,58	9	0,65

Con estos resultados se podría concluir que el usuario percibió el servicio que se presta con una valoración muy buena. La nota que más asignaron es el 9. Sin embargo, el tenor de los comentarios parece algo disociado del puntaje asignado. Del análisis cualitativo de los comentarios surgen las siguientes observaciones generales:

- Los empleados/usuarios no se atienen a la categoría consultada en forma estricta. Hay comentarios que claramente son de otras áreas.
- No logran diferenciar, en todos los casos, las responsabilidades de un área de tecnología y de otra.
- La gran mayoría de los comentarios son negativos.
- Los comentarios positivos principalmente se incluyen en los comentarios libres, mientras que en las categorías específicas se realizan evaluaciones negativas.

A continuación, una lista de categorías expresadas en los comentarios por orden de frecuencia de aparición: Entre empleados jefes y supervisores:

1. Falta de cumplimiento con el **tiempo**.
2. Disconformidad con los **procedimientos** que deben realizar (falta de simplicidad, burocracia y obstáculos).
3. Falta de "**orientación al negocio"**. Esto es, que el servicio de soporte termine afectando el desarrollo de las actividades productivas.
4. **Desconocimiento**, ineficiencia y falta de detalle, fundamentalmente en cómo funciona el negocio.
5. Falta de **compromiso con la solución**.
6. Fallas en el sistema de **prioridades**.
7. Falta de **comunicación**.

Entre empleados sin gente a cargo:

1. Fallas en **tiempos** de resolución.
2. Falta de "**orientación al negocio**". (Mayormente se afecta por falta de celeridad y, en menor medida, por la eficiencia de las soluciones).
3. Falta de **compromiso** de los ejecutantes con la solución.

4. Poca **eficiencia**, falta de comprensión de lo que se requiere, vinculado más a la solución técnica.
5. Falta de **empatía** y cordialidad.

A continuación, el mapeo del modelo SERVQUAL desde el punto de vista de lo que, según expresan los clientes internos, deberían satisfacer los servicios prestados:

- No hay ninguna referencia que indique necesario evaluar la dimensión de "Tangibles" en ninguna de sus categorías.
- La dimensión "Empatía" es sólo evaluada en aquellas categorías que tienen que ver con el interés que demuestra el prestador de servicio en las necesidades del cliente.
- No hay ninguna dimensión ni categoría que explicite la importancia de que el prestador de servicios conozca el negocio y los procesos de negocio a los que la tecnología da soporte.

Propuesta de Adaptación del Modelo SERVQUAL para Evaluar la Satisfacción del Cliente Interno con los Servicios de Software y Tecnología

A la luz de los resultados obtenidos y la necesidad de evaluar los clientes internos de la organización, se propone a continuación una adaptación al modelo SERVQUAL. Para evaluar la satisfacción del cliente interno para los servicios TIC brindados en una organización. En la Figura 3 se puede observar la adaptación a las nuevas necesidades y requerimiento de negocio que se ajustan a la realidad socio-económica actual. A continuación, se da una breve explicación de las modificaciones, tanto de los ítems que se excluyen como de aquellos que se agregan en base a los resultados del estudio y análisis de la encuesta realizada al personal interno de la empresa. En primer lugar, se optó por no evaluar la dimensión "Tangibles". Debido a que no aparecieron indicios serios de su necesidad desde la óptica de los empleados de la firma.

De la dimensión "Empatía" se optó por no evaluar los aspectos vinculados al trato personalizado (relacionado con clientes externos). Omitir aspectos a evaluar si ninguna de las partes le reconocer valor, redunda en tener un instrumento más sencillo. Este aspecto es fundamental, ya que los empleados son reticentes en general de completar encuestas y evaluaciones si no son obligatorias. Además de considerar la extensión de la misma puesto que reduce su entendibilidad y legibilidad.

Se consideró la necesidad de agregar un ítem en la dimensión "Confiabilidad" que evalúe el conocimiento del oferente del servicio respecto del negocio al que le da soporte el producto de software o el servicio tecnológico del que se ocupa. Este ítem surge con fuerza cada vez que se pide opinión a los empleados. Muchos lo entienden como compromiso con el negocio. Se demanda un servicio que supere su solvencia técnica, y conozca qué impacto tiene la degradación o falta del servicio en la organización.

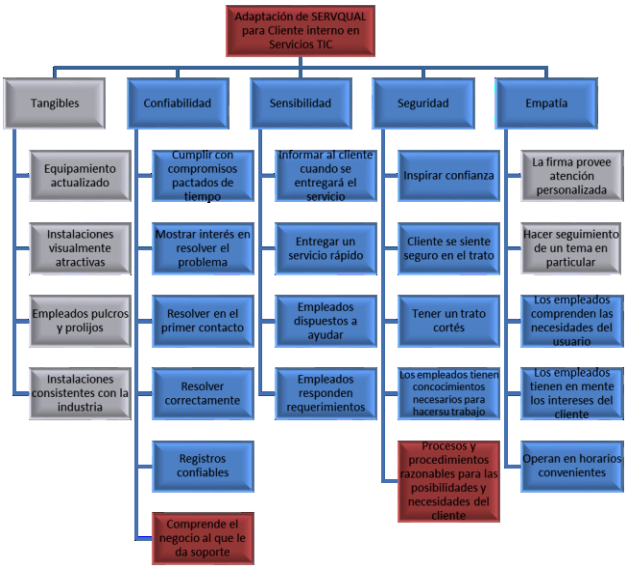


Fig. 3: Modelo Servqual adaptado para servicios TIC y clientes internos

Una buena calidad del servicio desde esta perspectiva requiere involucramiento y proactividad de acuerdo con la criticidad o sensibilidad del negocio que soporta. Finalmente, en la dimensión "Seguridad", se agrega un ítem que evalúe si, el proceso o procedimiento que se requiere del cliente para operar el servicio de software o para reclamarlo, es razonable con los conocimientos, urgencias y necesidades del cliente interno. Se propone agregar este ítem a esta dimensión ya que los procesos y procedimientos operan en una organización como la forma autorizada y reconocida de hacer las cosas. El cliente interno debería poder evaluar si la forma en la que se brinda el servicio tecnológico, los mecanismos de solicitudes, reclamos y resolución son a su criterio complejos, engorrosos, poco eficientes o pierden de vista el propósito esencial que es servir al negocio de la mejor manera.

El modelo propuesto de evaluación contempla, tanto la percepción de la organización, como la de los empleados. La organización desde su perspectiva de contar con una herramienta capaz de mejorar los servicios que presta internamente y los empleados desde la perspectiva de tener un soporte tecnológico más adecuado para hacer eficiente y eficaz su trabajo. Los servicios TIC en particular, son servicios complejos y compuestos por gran cantidad de participantes cuya virtuosa y correcta interacción hace posible una prestación de calidad. Un instrumento adecuado que ayude a optimizarlos agrega mucho valor en la búsqueda de la mejora continua.

Discusión y Conclusiones

En primer lugar, es importante hacer una referencia respecto de la disociación que parece haber entre las notas (altas) asignadas por los empleados respecto de los comentarios (negativos) realizados. Los resultados de la encuesta ponen de relieve una discusión necesaria y plantea una cuarta pregunta que será tema de nuevas investigaciones: ¿Este tipo de encuestas, vinculadas a los objetivos de las áreas, permiten realmente obtener un indicador de la satisfacción del cliente interno para este tipo de servicios?

Una de las razones de la distorsión, puede ser que, si bien hay disconformidad o insatisfacción respecto de los servicios prestados, a la hora de valorar a sus propios compañeros suben la nota para no perjudicar el cumplimiento de los objetivos de las áreas de tecnología. Esto se puede entender como solidaridad entre compañeros de trabajo, parte de la cultura organizacional. La idea de la distorsión en las evaluaciones entre grupos de trabajo es sostenida en estudios de cliente interno que muestran dificultades en la percepción de la realidad [24].

Respecto del modelo SERVQUAL y las particularidades observadas de este trabajo, la dimensión "Tangibles", que es fundamental en la evaluación de calidad de servicios con clientes externos, no tiene interés, ni por parte de la empresa, ni por parte de los empleados. Esto puede estar fundamentado, en que las instalaciones son compartidas y administradas por la misma empresa. El cliente interno, no siente que su proveedor de servicios es responsable ni debe ser evaluado por éstas. Por esta razón, la propuesta es no evaluarla. De igual manera, respecto de la dimensión "Empatía" - categoría

"Atención personalizada". Estos ítems, son valorados por el cliente externo, sin embargo, hacia adentro de una organización, no se espera que el servicio sea de mayor calidad por esta categoría.

Otra consideración importante es que el modelo SERVQUAL no contempla aspectos de procesos ni procedimientos, mientras que, tanto la empresa como los empleados, refieren necesario evaluarlos, y la propuesta lo contempla.

Todos los aspectos relacionados con la "orientación al negocio", tales como evaluar si hay buen trabajo en equipo, comprensión y conocimiento del negocio y el impacto que tiene su degradación o la falta de servicio, son fundamentales para la calidad del servicio desde el punto de vista del cliente interno, mientras que un cliente externo no estaría en condiciones de evaluarlo. Este ítem también es agregado en la propuesta. Finalmente, hay algunas consideraciones que están relacionadas con las particularidades del servicio de software y las tecnologías de la información a la hora de evaluar la satisfacción de los clientes. El hecho de que los encuestados no tengan en claro cómo funciona el servicio, donde empiezan y terminan las responsabilidades de las distintas áreas, no sepa con quién quejarse o a quién pedir ayuda, no contribuye con tener resultados que reflejen la realidad. Por otra parte, ¿es razonable pedirles a los clientes internos que conozcan los detalles y procesos tecnológicos?

Las evaluaciones para medir la satisfacción del cliente interno son muy importantes para las áreas TIC de las empresas. El primer paso fue obtener una idea de qué es importante evaluar.

Esto se reflejó en un modelo que es una variación de un modelo de calidad lo suficientemente probado como es SERVQUAL. Como validación del modelo propuesto se lo utiliza/aplica en una empresa nacional con casa central en la ciudad de Córdoba. Se pretende aplicar el modelo para la evaluación de una organización de otro rubro distinto al financiero. Este modelo debe permitir monitorear el estado de satisfacción y acusar los puntos débiles que es necesario mejorar. Para llevar a cabo los procesos y actividades de manera tal de lograr la satisfacción en todos los actores que intervienen en el trabajo diario dentro de las organizaciones e instituciones.

Referencias

[1] G. W. Marshall, J. Baker y D. W. Finn, «Exploring internal customer» JOURNAL OF BUSINESS & INDUSTRIAL MARKETING, vol. 13, nº 4/5, pp. 381-392, 1998.
[2] L. Grossi y J. A. Calvo - Manzano, «Análisis de Decisiones en la Selección de Proveedores de Tecnologías de la Información: Una revisión sistemática,» RISTI, pp. 67 - 79, 2011.
[3] J. M. Juran, Juran•s Quality Control Handbook. 4" ed., Nueva York: McGraw-Hill, 1988.
[4] W. E. Deming, Out of the Crisis, Cambridge: MIT Center for Advanced Engineering, 1986.
[5] ISO, ISO/IEC 20000-1:2005, Ginebra: ISO/IEC, 2005.
[6] L. Berry, «Services Marketing Is Different» vol. 30, nº 24-29, 1980.
[7] V. Zeithalm, «How consumer evaluation processes differ between goods and service» nº 186 -190, 1981.
[8] R. L. Oliver, «A cognitive model of the antecedens and consequences of Satisfaction Decisions» vol. XVII, nº 460-469, 1980.
[9] C. Grönroos, «A Service Quality Model and its Marketing Implications» vol. 18, nº 36 - 44, 1984.
[10] A. Parasuraman, V. A. Ziethalm, L. L. Berry, «SERVQUAL: A Multiple-item Scale for measuring consumer perceptions of service quality» Journal of Retailing, pp. 12 - 40, 1988.
[11] A. Parasuraman, V. A. Zeithalm, A. Malhotra, «E-S-QUAL A Multiple-Item Scale for Assessing Electronic Service Quality» Journal of Service Research, Vol. 7, pp. 1-21, 2005.
[12] D. Iacobucci, K. A. Grayson, A. L. Ostrom, «The Calculus of service quality and custormer satisfaction» Advances in Services Marketing and Management. V. 3, pp. 1-67, 1994.
[13] S. Vandermerwe y D. Gilbert, «Making internal services market driven» Business Horizons, pp. 83-89, 1989.
[14] J. L. Heskett, T. O. Jones, G. W. Loveman, W. E. Sasser y L. A. Schelsinger, «The ServiceProfit Chain» nº 164 -170, 1994.
[15] R. Hallowell, L. Schlesinger, J. Zornitzky, «Internal Service Quality, Customer and job satisfaction: Linkages and Implications for Management» Human Resource Planning, pp. 20 - 31, 2002.
[16] S. Rahayu, «Internal Customer Satisfaction and Service Quality Toward» ASEAN MARKETING JOURNAL, vol. III, nº 2, pp. 114-123, 2011.
[17] M. Adil, D. O. F. M. Al Ghaswyneh y A. M. Albkour, «SERVQUAL and SERVPERF: A Review of Measures in Services» Global Journal of Management and Business Research, vol. 13, nº 6, pp. 65-76, 2013.
[18] E. K. Yarimoglu, «A Review on Dimensions of Service Quality Models» Journal of Marketing Management, vol. 2, nº 2, pp. 79-93, 2014.

[19] D. K. Prof. Kundan y K. S. Sajeeb, «Measuring Service Quality and Customer Satisfaction: Empirical Evidence from Nepalese Commercial Banking Sector Undertakings,» Management Dynamics, vol. 16, nº 1, pp. 10-20, 2012.
[20] M. M. Alotaibi, Evaluation of "AIRQUAL" scale for measuring airlines service quality and its effect on customer satisfaction and loyalty, Cranfield: Cranfield University, 2015.
[21] K. B. Tariq, B. Dr. Mohsin y M. Mohsin, «Impact of service quality on customer's satisfaction: a study from service sector especially private colleges of Faisalabad, Punjab, Pakistan» International Journal of Scientific and Research Publications, vol. 3, nº 5, 2013.
[22] V. M. Ngo y H. H. Nguyen, «The Relationship between Service Quality, Customer Satisfaction and Customer Loyalty: An Investigation in Vietnamese Retail Banking Sector» Journal of Competitiveness, vol. 8, nº 2, pp. 103 -116, 2016.
[23] A. Parasuraman, V. A. Zeithalm y L. L. Berry, «Refinement and Reassessment of the SERVQUAL Scale» Journal of Retailing, vol. 67, nº 4, pp. 420 - 450, 1991.
[24] R. G. Gilbert, «Measuring internal customer satisfaction» Journal of Service Theory and Practice vol 10, pp. 178 - 186, 2000.

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Templado Simulado para problemas de muchos objetivos. Una comparativa con MOEA/D

Eduardo Morales[1], Benjamín Barán[1],

[1] Universidad Nacional de Asunción, Paraguay
eduardomoralesf@gmail.com,bbaran@pol.una.py

Resumen. Los problemas de optimización de muchos objetivos (MaOP-*Many objective optimization problems*) consideran 4 o más objetivos, lo que dificulta resolverlos con algoritmos tradicionales como los reconocidos Algoritmos Evolutivos Multi-Objetivos. Por lo tanto, el presente trabajo presenta una nueva alternativa, denominada TBSA, para resolver problemas MaOP, la cual se basa en el Templado Simulado (SA-*Simulated Annealing*) combinado con *clustering* y búsqueda tabú. El algoritmo TBSA propuesto es comparado con el algoritmo del estado del arte para problemas MaOP conocido como MOEA/D, usando el reconocido conjunto de pruebas DTLZ. Resultados experimentales prueban que el TBSA propuesto supera al MOEA/D en reconocidas métricas como IGD y Epsilon, y es competitivo en otras métricas como Hipervolumen.

Keywords: simulated annealing (SA), many-objective optimization, clustering, tabu search, hypervolume, coverage, inverted generational distance (IGD), epsilon metric, DTLZ test set, MOEA/D.

1 Introducción

Los problemas de múltiples objetivos (MOP -*Multiobjective Optimization Problems*) buscan optimizar dos o más objetivos simultáneamente. Estos objetivos pueden ser contradictorios, es decir, la mejora de un objetivo empeora algún otro objetivo. Típicamente, un MOP considera 2 o 3 objetivos. A los problemas de optimización de 4 o más objetivos se los conoce como problemas de optimización de muchos objetivos (MaOP - *Many-objective optimization problems*) [1].

Por sus particularidades, la resolución de problemas de múltiples objetivos no es trivial por lo que todavía es objeto de investigación en Ciencias de la Computación [2].

Este trabajo se centra en problemas MaOP que plantean nuevos desafíos relacionados al alto costo computacional al resolverlos, así como una limitada escalabilidad de los algoritmos evolutivos existentes [1]. En este contexto, se presenta

un análisis comparativo entre el algoritmo *Tabú Based Simulated Annealing*-TBSA propuesto por los autores [3] y el reconocido *Multiobjective Evolutionary Algorithm Based on Decomposition*-MOEA/D [4] especialmente diseñado para problemas MaOP, por lo que puede ser considerado como el más representativo del estado del arte cuando se optimizan 4 o más funciones objetivo.

2 Conceptos

Por lo general, un problema de múltiples objetivos se compone de un conjunto de n variables de decisión$(\vec{\mathrm{x}})$, un conjunto de m de funciones objetivo $\vec{\mathrm{y}} = f(\vec{\mathrm{x}})$, y un conjunto de r restricciones. Tanto las funciones objetivo como las restricciones se encuentran en función de las variables de decisión. Formalmente, se puede expresar como:

Optimizar
$$\vec{\mathrm{y}} = f(\vec{\mathrm{x}}) = [f_1(\vec{\mathrm{x}}), f_2(\vec{\mathrm{x}}), \dots, f_m(\vec{\mathrm{x}})]^T . \quad (1)$$

$$\vec{\mathrm{x}} = [x_1, x_2, \dots, x_n]^T \in X_f \subseteq \mathbb{R}^n . \quad (2)$$

$$\vec{\mathrm{y}} = [y_1, y_2, \dots, y_m]^T \in Y_f \subseteq \mathbb{R}^m . \quad (3)$$

sujeto a

$$\mathrm{g}(\vec{\mathrm{x}}) = [\mathrm{g}_1(\vec{\mathrm{x}}), \mathrm{g}_2(\vec{\mathrm{x}}), \dots, \mathrm{g}_r(\vec{\mathrm{x}})]^{\mathrm{T}} \leq 0 \quad (4)$$

donde $\vec{x}$ representa al vector solución compuesto por n variables de decisión, mientras que $\vec{y}$ representa al vector objetivo de dimensión m, ($m \geq 4$ en MaOP). El vector $\mathrm{g}(\vec{x})$ está compuesto por r restricciones que limitan el espacio factible de solución. Soluciones que cumplen con las restricciones (4) conforman el espacio factible de decisión X_f, cuyo mapeo al espacio objetivo define el espacio objetivo factible Y_f. Dependiendo de la naturaleza del problema estudiado, optimizar se puede entender como minimizar o maximizar cada función objetivo, simultáneamente.

Definición 2.1. Dominancia Pareto. Dadas las variables de decisión $\vec{x}$ y$\vec{\mathrm{x}}'$ tal que $\vec{x}, \vec{x}' \in X_f$, se dice que $\vec{x}$ domina a $\vec{\mathrm{x}}'$ si no es peor en ningún objetivo y es estrictamente mejor en al menos un objetivo.

Se utilizará la siguiente notación para indicar la relación de dominancia entre soluciones:

$\overrightarrow{x} \prec \vec{x}'$: $\vec{x}$ domina a $\vec{x}'$
$\overrightarrow{x} \succ \vec{x}'$: $\vec{x}$ es dominado por $\vec{x}'$
$\overrightarrow{x} \sim \vec{x}'$: $\vec{x}$ es no-comparable con $\vec{x}'$, i.e. ni $\overrightarrow{x}$ domina a $\vec{x}'$ ni $\vec{x}'$ domina a $\vec{x}$.

Definición 2.2. Conjunto Pareto. Se denomina conjunto Pareto (denotado como P^*) al conjunto de todas las soluciones factibles no dominadas, i.e. soluciones que no pueden ser mejoradas en ningún objetivo sin empeorar otro objetivo.

Como muchas veces no puede calcularse todo el conjunto Pareto P^*, en la práctica resulta suficiente con calcular una buena aproximación al conjunto Pareto, conocida como P_{known}.

Definición 2.3. Frente Pareto, (denotado PF^*) constituye el mapeo del conjunto Pareto P^* al espacio objetivo.

$$PF^* = \{f(\vec{x}): \vec{x} \in \mathrm{P}^*\}. \quad (5)$$

El mapeo del conjunto Pareto aproximado P_{known} al espacio objetivo es conocido como el frente Pareto aproximado PF_{known}.

3 Simulated Annealing

Templado Simulado (*Simulated Annealing* - SA) es una meta-heurística [5] de optimización inspirada en una variante de búsqueda local propuesta originalmente por Scott Kirkpatrick [6] enfocada a problemas mono-objetivo. Este algoritmo se basa en una analogía de la termodinámica referente al templado de metales, en el cual, si un metal con alta temperatura (cercana a la temperatura de fusión) es enfriado lo suficientemente lento, sus átomos se ordenan con una estructura de mínima energía. Por su sencillez y efectividad, el SA fue extendido para resolver problemas de optimización multiobjetivo [7], siendo denominado genéricamente como *Multi-Objective Simulated Annealing* (MOSA). A diferencia del SA mono-objetivo que retorna una única solución al problema, el resultado de un MOSA es un conjunto de soluciones denominado conjunto Pareto aproximado (P_{known}). Esquemas generales de SA mono y multi-objetivos son descritos en [3].

Según Farina y Amato [8], a medida que la cantidad de objetivos m aumenta, la probabilidad de que 2 soluciones sean no comparables tiende a 1, esto supone un inconveniente para algoritmos que no limitan el número máximo de soluciones del conjunto Pareto, como el *Dominance-based multiobjective simulated annealing*[9], por el enorme crecimiento en el número de soluciones no dominadas que se puede llegar a calcular y deben ser mantenidas en el conjunto P_{known}mientras el algoritmo va avanzando en su aproximación a P*.

Tabú Based Simulated Annealing (TBSA)

El *Tabú Based Simulated Annealing* (TBSA), es un algoritmo propuesto en [3]por los autores del presente trabajo, desarrollado especialmente para problemas *MaOP* con 4 o más objetivos, cuyas principales características son utilizar dominancia Pareto en las comparaciones entre soluciones y limitar el ingreso de una nueva solución no

dominada al conjunto Pareto aproximado mediante el uso de técnicas de *clustering* y zona tabú. El algoritmo 3.1 presenta el esquema general del algoritmo propuesto TBSA.

Algoritmo 3.1 TBSA. Esquema general del algoritmo TBSA.

```
1 Input: F(x⃗); Kt =cantidad total de evaluaciones de la función objetivo; HL; SL; σ; Ω; Nt; NL
2     K = 0,01 Kt;
3     Calcular T0;Calcular α;T = T0;dTG=0;
4     clustering_realizado = false;evaluaciones = 0;
5     while (evaluaciones < Kt )
6     for  i = 1 to K and (evaluaciones < Kt)
7         if (P_known> SL)
8             reducir(P_known, HL)
9             clustering_realizado = true
10        end if
11    x⃗'= Perturbar_Solución (x⃗,σ, Ω)
12    if   ( x⃗' ≺ x⃗  or   x⃗' ~ x⃗)
13    x⃗ = x⃗'
14        if ( clustering_realizado)
15          if (fuera_de_zona_tabú (x⃗',P_known))
16        P_known=Actualizar_Pareto (x⃗')
17            end if
18        else
19        P_known=Actualizar_Pareto ( x⃗')
20    end if
21    else if(uniforme(0,1)< exp(-ΔE/T))
22        x⃗ = x⃗'
23    end if
24            evaluaciones = evaluaciones +1;
25    end for
26    T= αT
27    end while
28    if (P_known> HL)
29        reducir(P_known, HL)
30    end if
31
32    Ouput: P_known
```

Al igual que el algoritmo AMOSA [10], el TBSA propuesto por los autores utiliza 2 parámetros: SL (*soft length*) y HL (*hard lenght*) (HL < SL) para mantener el conjunto Pareto aproximado. Cuando el conjunto P_{known} alcanza una cantidad SL de soluciones, el mismo es reducido a HL soluciones mediante *medoidclustering* [11]. El proceso de *clustering* genera HL soluciones representantes o *medoids*, las cuales tienen asociadas un conjunto de soluciones representadas. Durante el proceso de *clustering* el TBSA incorpora el cálculo de distancias Tabú para cada solución representativa, las cuales sirven para definir una zona Tabú, cuyo objetivo es evitar el ingreso de nuevas soluciones al conjunto Pareto aproximado cuando estas se encuentren dentro de la zona Tabú. A continuación, se detalla el proceso del cálculo de la distancia tabú para cada solución representativa.

Distancia tabú

La distancia $d$entre 2 soluciones se define como la distancia Euclidiana entre las soluciones en el espacio objetivo, conforme:

$$d(\vec{y},\vec{y}') = \sqrt{\sum_{i=1}^{m}\left(y_i - {y'}_i\right)^2}\,. \qquad (6)$$

Durante el proceso de reducción del conjunto Pareto, se calcula y asigna la distancia tabú (dT), para cada solución representante o *medoid* siempre y cuando la cantidad de soluciones representadas sea mayor a cero ($nr > 0$).

$$dT(\vec{r}, C) = \frac{\sum_{i=1}^{nr} d(\vec{r},C_i)}{nr}\,. \qquad (7)$$

donde $\vec{r}$ es una solución representativa o *medoid* y C el conjunto *cluster* de soluciones representadas por $\vec{r}$. nr indica la cantidad soluciones contenidas en C.

Luego, se calcula la distancia tabú global (dTG) como la media de las distancias dT de las soluciones representativas con $nr > 0$.

$$dTG = \frac{\sum_{i=1}^{h} dT_i}{h}\,. \qquad (8)$$

donde h indica la cantidad de soluciones representativas con $nr > 0$.

A las soluciones representativas con $nr = 0$ se les asigna $dT = dTG$.En consecuencia, cada solución representativa tendrá su propia distancia tabú calculada independientemente (a excepción de aquellas a las que se asignó la distancia tabú global dTG). Finalizado este proceso, se eliminan las soluciones representadas y P_{known} contiene las HL soluciones representativas.

Por razones de espacio, el presente trabajo se limita a describir el esquema de aceptación de nuevas soluciones en el TBSA. Para mayores detalles sobre los

parámetros σ; Ω; N_i; N_L, T_0 α, y el cálculo de la diferencia de energía entre las soluciones (ΔE) se sugiere el artículo original [3], donde se presenta el TBSA.

La actualización del conjunto P_{known} se realiza según el siguiente esquema:

a) si la nueva solución domina a alguna solución del conjunto P_{known}, se agrega la nueva solución a P_{known} y se eliminan las soluciones dominadas. En caso que ya haya sido calculada la distancia tabú global (dTG), esta es asignada a la nueva solución;

b) si la nueva solución es no comparable respecto a todas las soluciones del conjunto Pareto y además:

b.1) todavía no se ha realizado una reducción del conjunto Pareto, se agrega la nueva solución al conjunto P_{known};

b.2) si ya se ha realizado una reducción al conjunto P_{known}, se identifica la solución s del conjunto P_{known} que tenga la menor distancia d con la nueva solución $\vec{x}'$ en el espacio objetivo, dada por (6). Si la distancia d es menor a la distancia dT (distancia Tabú) asociada a s, entonces la solución no es ingresada al conjunto P_{known}. Toda región donde $d \leq dT$ es considerada *zona tabú,* por lo que no se aceptan soluciones en dicha zona (corresponde a la función `fuera_de_zona_tabú` del pseudocódigo). Si por el contrario la distancia d es mayor, se agrega la nueva solución al conjunto Pareto aproximado asignándole la distancia tabú global (dTG) y se eliminan del conjunto aquellas soluciones dominadas por $\vec{x}'$.

4 Comparación entre TBSA y MOEA/D

Considerando que el reconocido *Multiobjective Evolutionary Algorithm Based on Decomposition*-MOEA/D [4] fue especialmente diseñado para problemas MaOP, a continuación se lo compara con el TBSA propuesto por los autores con el fin de verificar el comportamiento del algoritmo propuesto con respecto a un algoritmo evolutivo que puede ser considerado como referente del estado del arte para problemas con 4 o más objetivos.

Se presentan a continuación experimentos utilizando el reconocido conjunto de funciones de prueba (*test set*) DTLZ propuesto por Deb et al. como un conjunto de problemas multiobjetivos que permiten comparar diferentes algoritmos en problemas de múltiples objetivos [12], En este trabajo se utilizaron los primeros 7 problemas denotamos como DTLZ1 a DTLZ7 considerando 3, 4, 6 y 8 objetivos. Tanto los algoritmos como las métricas de comparación fueron implementados en lenguaje Java utilizando el *framework jMetal (Metaheuristic Algorithms in Java)* [13] en su versión 4.5.

Todas las pruebas experimentales se ejecutaron en un ordenador Intel Core i7 de 2.0 GHz con 4 núcleos y una arquitectura de 64 bits con 8 GB RAM, bajo el sistema operativo Microsoft Windows 7.

Para ambos algoritmos, el criterio de parada utilizado fue la cantidad total de evaluaciones de la función objetivo. Se presentan resultados con 5.000 y 25.000 evaluaciones de la función objetivo. Dada la aleatoriedad de los algoritmos comparados (TBSA vs. MOEA/D), se han realizado 30 corridas para cada problema, con idéntica cantidad de objetivos y cantidad de evaluaciones de la función objetivo.

Siguiendo las recomendaciones de [2], en el presente trabajo se han utilizado las siguientes métricas de comparación: hipervolumen (HV) [14], distancia generacional invertida (IGD) [15], Cobertura (C) [16] y épsilon (ε) [17]. Notar que para las métricas C, IGD y ε, un menor valor representa un mejor resultado, mientas que para HV un mayor valor representa un mejor resultado.

El conjunto Pareto aproximado fue limitado a un máximo de 100 soluciones para ambos algoritmos.

Las tablas 1 y 2 detallan la configuración de parámetros utilizados para los algoritmos MOEA/D y TBSA respectivamente.

Tabla 1. Parametrización del MOEA/D.

Parámetro	Valor
Tamaño población	100
Probabilidad de cruzamiento	1
Índice de distribución de mutación	20
Índice de distribución de cruzamiento	20
Probabilidad de mutación	1/Cantidad de variables independientes
T	20

Tabla 2. Parametrización del TBSA.

Parámetro	Valor
SL (para 5.000 evaluaciones)	300
SL (para 25.000 evaluaciones)	2500
HL	100
Traversal scaling inicial	1
Local scaling inicial	0.2
Nt	50
NL	20

Resultados de la Comparación

Se presentan a continuación, cuadros comparativos representando los valores normalizados y promediados de las 30 ejecuciones independientes para cada algoritmo.

La normalización se realizó dividiendo cada valor por el mayor valor obtenido entre ambos algoritmos por cada problema DTLZ. Los resultados de las Figuras 1 al 6, corresponden al promedio de los valores normalizados considerando los problemas DTLZ 1 al 7 por número de objetivos. Los valores de las Figuras 7 y 8 corresponden a la métrica de Cobertura (C) por lo que no se requiere de normalización.

Según puede observarse en las Fig. 1 y 2, para problemas de hasta 4 objetivos resulta mejor (o igual) el TBSA mientras que para 6 y 8 objetivos el MOEA/D aventaja muy

ligeramente al TBSA considerando la métrica hipervolumen. Se hace notar que incluso cuando el MOEA/D fue mejor, la diferencia máxima entre los algoritmos fue solamente de un 2%.

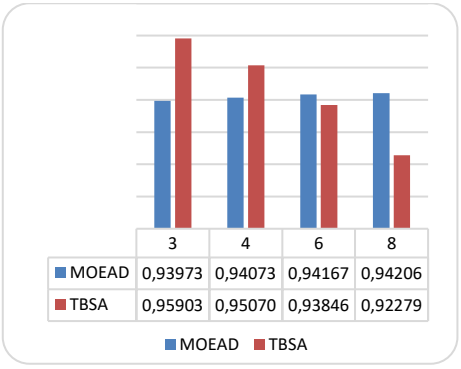


Fig. 1.HV para 5.000 evaluaciones.

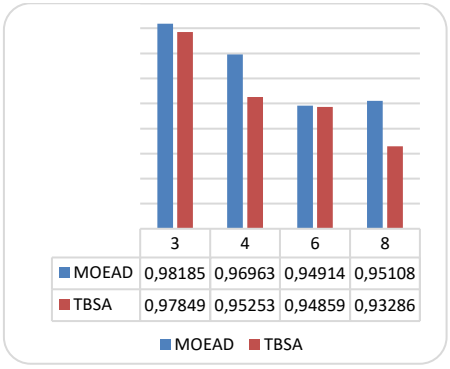


Fig. 2. HV para 25.000 evaluaciones

Considerando la métrica IGD (ver Fig. 3 y 4), el TBSA ha resultado con mejor desempeño que el MOEA/D, independientemente del número de objetivos considerados, logrando distanciarse considerablemente del MOEA/D en los experimentos con 8 objetivos y 25.000 evaluaciones, donde ha logrado valores aproximadamente 23% mejores que el MOEA/D.

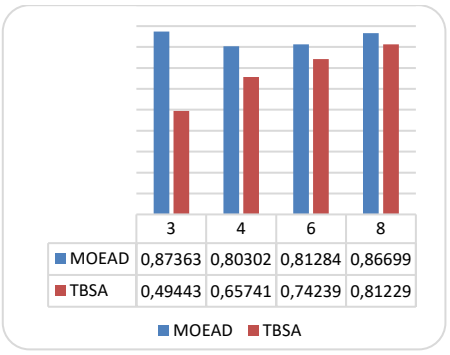


Fig. 3.IGD para 5.000 evaluaciones.

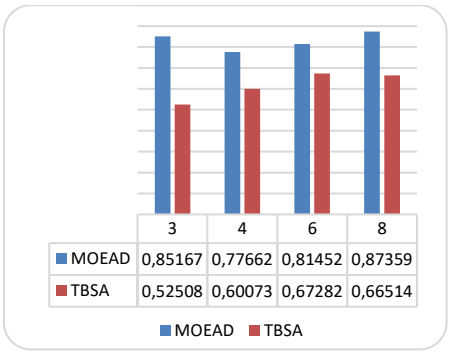


Fig. 4.IGD para 25.000 evaluaciones.

Considerando la métrica Épsilon (Fig. 5 y 6), también se evidencia que el TBSA ha logrado un mejor desempeño que el MOEA/D, independientemente de la cantidad de objetivos y número de evaluaciones.

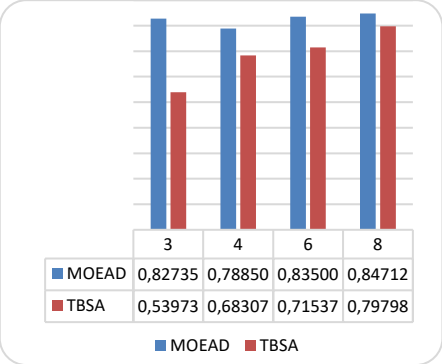


Fig. 5. Épsilon para 5.000 evaluaciones.

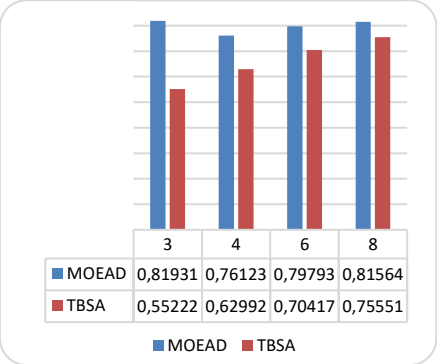


Fig. 6. Épsilon para 25.000 evaluaciones.

Finalmente, considerando la métrica de cobertura (Fig. 7 y 8), con 5.000 evaluaciones de la función objetivo, el TBSA ha igualado con el MOEA/D para 4 objetivos, mientras que el MOEA/D ha logrado aventajar al TBSA en los experimentos con 6 y 8 objetivos. Considerando 25.000 evaluaciones de la función objetivo, el TBSA ha aventajado al MOEA/D en los experimentos con 4 objetivos mientras que el MOEA/D ha logrado mejores resultados en los experimentos con 6 y 8 objetivos.

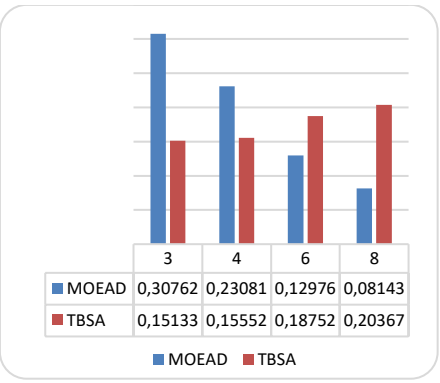


Fig. 8. Cobertura para 25.000 evaluaciones.

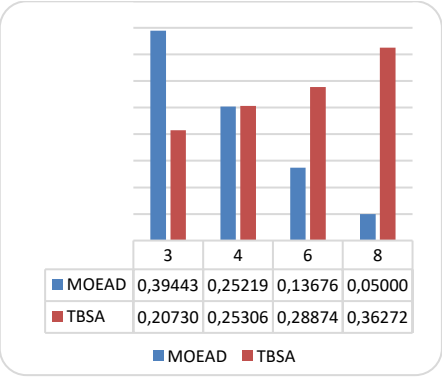


Fig. 7. Cobertura para 5.000 evaluaciones.

En resumen, los resultados experimentales arriba presentados demuestran que el algoritmo TBSA propuesto por los autores resulta altamente competitivo incluso al compararlo con un algoritmo del estado del arte especialmente diseñado para resolver problemas MaOP. En particular, se hace notar que si bien el TBSA propuesto no fue superior al considerar Hipervolumen o Cobertura, logró demostrar una ventaja considerable con métricas como IGD y Épsilon, lo que motiva a seguir trabajando con el TBSA propuesto.

5 Conclusiones.

El presente trabajo compara el algoritmo *Tabú Based Simulated Annealing*-TBSA propuesto por los autores [3] con un algoritmo del estado del arte para problemas MaOP con 4 o más objetivos, el reconocido *Multiobjective Evolutionary Algorithm Based on Decomposition*-MOEA/D [4].

Teniendo en cuenta los resultados experimentales presentados, se nota que el TBSA encuentra mejores resultados que el MOEA/D considerando las métricas IGD y Épsilon, independientemente de la cantidad de evaluaciones de la función objetivo y del número de funciones objetivo, logrando además valores muy cercanos al MOEA/D en la métrica de Hipervolumen. Solamente en la métrica de cobertura el MOEA/D ha logrado una ventaja apreciable respecto al TBSA considerando 6 y 8 objetivos.

En consecuencia, se puede afirmar que el TBSA se muestra como una opción competitiva para problemas *MaOP,* considerando que ha logrado obtener mejores desempeños en las métricas actualmente más utilizadas para problemas MaOP [2], especialmente en IGD, donde ha logrado superar al MOEA/D por hasta un 23%.

Como trabajo futuro, los autores se encuentran estudiando técnicas alternativas de reducción de la población (para reemplazar al *clustering*) y mejorar el tiempo de ejecución para cantidades crecientes de objetivos y variables independientes. Al mismo tiempo, se están haciendo las primeras pruebas comparativas con problemas del mundo comercial y de gran envergadura.

Referencias

1. von Lücken, C., Barán, B., & Brizuela, C.: A survey on multi-objective evolutionary algorithms for many-objective problems. Computational Optimization and Applications, vol. vol. 58(3), pp. 707-756 (2014)
2. Riquelme, N., Von Lücken, C., Barán, B.: Performance metrics in multi-objective optimization. In Computing Conference (CLEI), 2015 Latin American, IEEE, pp. 1-11 (2015)
3. Morales E., Barán, B.: Simulated annealing for many objective optimization problems. In: Computer Science Society (SCCC), 2016 35th International Conference of the Chilean. IEEE (2016)
4. Zhang, Q., & Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on evolutionary computation, vol. 11(6), pp. 712-731 (2007)
5. Bräysy, O., & Gendreau, M.: Vehicle routing problem with time windows, Part II: Metaheuristics. Transportation science, vol. 39(1), pp. 119-139 (2005)

6. Kirkpatrick, S.: Optimization by simulated annealing: Quantitative studies. Journal of statistical physics, 34(5), pp. 975-986 (1984)
7. Suman, B., Kumar, P.: A survey of simulated annealing as a tool for single and multiobjective optimization. Journal of the operational research society, vol. 57(10), pp. 1143-1160 (2006)
8. Farina, M., Amato, P.: On the optimal solution definition for many-criteria optimization problems. Fuzzy Information Processing Society-NAFIPS proceeding - IEEE, pp. 233-238 (2002).
9. Smith, K. I., Everson, R. M., Fieldsend, J. E., Murphy, C., Misra, R.: Dominance-based multiobjective simulated annealing. Evolutionary Computation, IEEE Transactions, vol. 12(3), pp. 323-342 (2008)
10. Bandyopadhyay, S., Saha, S., Maulik, U., & Deb, K.: A simulated annealing-based multiobjective optimization algorithm: AMOSA. IEEE transactions on evolutionary computation, vol. 12(3), pp. 269-283 (2008)
11. Sarle, W. S.: Finding Groups in Data: An Introduction to Cluster Analysis. Journal of the American Statistical Association, vol. 86(415), pp. 830-833 (1991)
12. Deb, K., Thiele, L., Laumanns, M., Zitzler, E.: Scalable Multi-Objective Optimization Test Problems. Evolutionary Computation, Proceedings of the IEEE 2002 Congress (CEC 2002), 1, pp. 825–830 (2002)
13. Durillo, J., Nebro, A., Alba, E. The jmetal framework for multi-objective optimization: Design and architecture. IEEE Congress on Evolutionary Computation (CEC), pp. 1- 8 (2010)
14. Coello, C. A. C, Lamont, G.,Van Veldhuizen, D.: Evolutionary algorithms for solving multi-objective problems. Springer Science & Business Media, New York (2007)
15. Li, H., Zhang, Q.: Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. Evolutionary Computation, IEEE Transactions, vol. 13(2), pp. 284-302 (2009)
16. Zitzler, E.: Evolutionary algorithms for multiobjective optimization: Methods and applications. Ph. D. thesis, Swiss Federal Institute of Technology (1999)
17. Zitzler, E., Thiele, L.: Multiobjective optimization using evolutionary algorithms—a comparative case study. International conference on parallel problem solving from nature, Springer, pp. 292-301 (1998)

Esquema de certificación tipo auditoría para el estándar ISO/IEC 29110-4-1 perfiles del ciclo de vida del software aplicable a la calidad de los procesos software de las VSE (very small entities)

Andrés Fabián Pino Anacona, Andrés Mauricio Caicedo Rendón, Dayner Felipe Ordóñez López, Alberto Bravo Buchely

Resumen. Las pequeñas organizaciones dedicadas al desarrollo de productos de software juegan un papel muy importante en el mercado nacional e internacional, sin embargo no logran un reconocimiento importante como organizaciones que desarrollan productos software de calidad debido a que los estándares internacionales contienen procesos muy tediosos y complejos de abordar para estas organizaciones limitadas en recursos, además de sus elevados costos de implementación. Este trabajo presenta un esquema de certificación por conformidad de requisitos presentados en el estándar ISO/IEC 29110-4-1 que puede ser usado por las pequeñas organizaciones desarrolladoras de software para (i) iniciar un enfoque orientado a procesos, (ii) garantizar que sus productos son de calidad y cumplen requisitos establecidos en estándares internacionales y (iii) sentar las bases para alcanzar certificaciones en estándares más robustos como ISO/IEC 15504 o CMMI.

Palabras clave: CMMI, ISO/IEC 15504, ISO/IEC 12207, certificación, estándar.

1. Introducción

En la actualidad existen múltiples empresas que pertenecen a la industria del software cuyo aporte a la actividad económica del país es de suma importancia debido a que ofrece múltiples fuentes de negocio [1] y apoya el crecimiento de otras organizaciones al proporcionar métodos más eficientes en la administración de la información que estos manejan. En este sentido, la creciente industria de desarrollo de software hace

aportes significativos con miras al desarrollo económico y tecnológico de los países. De acuerdo con el informe presentado [2] hoy en día se habla que el porcentaje de pequeñas empresas (que se denominarán VSEs en el presente documento por sus siglas en inglés de Very Small Entities) que desarrollan y/o mantienen software superan en un alto grado a las grandes industrias de desarrollo (siendo a nivel nacional el 80% pequeñas y medianas empresas y el 19% grandes empresas).

En efecto, la industria del software reconoce la importancia que tienen las pequeñas organizaciones desarrolladoras de software en la construcción de sus productos y servicios [3], viéndolas como una red de procesos interconectados que involucran desde los procesos llevados a cabo por la alta dirección hasta aquellos seguidos por la parte técnica de la misma. Estos procesos deben considerar aspectos que permitan garantizar

la trazabilidad y repetitividad en el desarrollo de productos, razón por la cual se viene evidenciando a través de estudios la importancia que tiene la implementación de programas de Mejora de Procesos de Software (SPI, por sus siglas en ingles de Software ProcessImprovement).

La mejora de procesos software se define como un esfuerzo planeado, gestionado y controlado cuyo objetivo es incrementar la capacidad de los procesos de desarrollo de software que están implementados en este tipo de organizaciones [4], y que les permite ubicarse en niveles de madurez especificados por los estándares.

Idealmente, la mejora de procesos es impulsada por los objetivos de negocio tales como incrementar la productividad, la satisfacción del cliente o incrementar la cuota de mercado. Varios enfoques inician con la identificación de los objetivos de negocio, seguida por la identificación de problemas potenciales que puedan impedir la realización de dichos objetivos de negocio. Concluyendo con un diagnóstico, en el que se identifican y se implementan las acciones correctivas necesarias.

Un trabajo de investigación previo muestra una revisión sistemática sobre estudios realizados a pequeñas y medianas empresas dedicadas a la industria de software que implementaron iniciativas SPI [5]. En este estudio se revelan los procesos de mayor interés que las organizaciones deben mejorar, algunos factores de éxito para su contexto, el esfuerzo realizado por las organizaciones para llevar a cabo iniciativas de SPI, resultados estadísticos sobre los estándares más utilizados y los países que más han llevado a cabo iniciativas en la temática, entre otros.

Este trabajo de investigación encuentra algunos factores que dificultan llevar a cabo las actividades del ciclo de vida del software y concluye con la importancia de implementar mejora de procesos en las organizaciones. Dichos hallazgos permiten establecer que entre los procesos de mayor interés para mejorar en las VSEs se tienen:

- E-licitación de requisitos.

Definiendo este término como proceso determinado en la primera etapa del ciclo de vida del software que permite abstraer una comprensión de un problema que se quiere resolver a través de un producto software. Se trata, esencialmente de una actividad humana donde se identifican las partes interesadas y se establecen las relaciones entre el cliente, los usuarios y el equipo de desarrollo [6].

- Gestión de proyectos.
- Documentación.
- Gestión de cambios en los requisitos.
- Establecimiento de procesos.
- Gestión de la configuración.

Por mencionar algunos, también se resaltan los beneficios que tienen los programas de implementación de SPI en las pequeñas organizaciones desarrolladoras de software, tales como, la reducción del esfuerzo de trabajo mejorando los procesos de e-licitación de requisitos software y el aumento en la capacidad general mediante la implementación de procesos de gestión para el código fuente y defectos [7]. Otros estudios de SPI en pequeñas organizaciones mencionan que es posible implementar procesos software de manera que se pueda tener beneficios relacionados con el costo-

eficiencia considerando sus objetivos de negocio específicos, características y limitaciones en recursos[8].

Estos estudios concluyen con la importancia de continuar aportando en el área de SPI en VSEs, ya que este tipo de organizaciones son importantes dentro de la economía del país, el alto impacto que tiene la mejora de procesos software en el desarrollo de productos de calidad y la consecución de los objetivos de estas organizaciones, entre otros.

Existen muchas diferencias entre organizaciones dedicadas a la industria del software que tienen certificados de calidad y las que no lo tienen, entre las cuales podemos mencionar [9]:

- La potenciación de su competitividad y productividad tanto a nivel nacional como internacional.
- El reconocimiento como empresas que producen software de calidad.
- El establecimiento de una cultura de calidad dentro de las organizaciones.
- El incremento en la satisfacción de sus clientes.

Estas características suponen uno de los escenarios más preocupantes para la industria del software en Colombia, de acuerdo con estudios realizados, ya que muchas de las pequeñas organizaciones no poseen certificaciones de calidad, y a su vez se establece que una de las estrategias propuestas por empresas para estimular a esta industria es desarrollar mecanismos de acceso a certificaciones internacionales [10].

En Colombia sólo el 14.46%, 162 empresas de desarrollo de software de 1120 encuestadas, tienen algún tipo de certificación (donde la mayoría son certificaciones ISO 9001) [11]y en ocasiones es difícil que estos tipos de certificación se adecuen a sus características y objetivos organizacionales. Como conclusión del VI Encuentro Latinoamericano de Mejora de Procesos de Software (SEPGLA 2010) realizado en Medellín Colombia, se estableció que las empresas deben abordar esquemas de certificación de calidad de software adecuadas a su contexto [12], que les permita desarrollar software de calidad, ser más competitivas a nivel nacional y poder expandirse a mercados internacionales.

En este orden de ideas, existen estudios que revelan cómo adaptar normas internacionales y de elevados costos a características presentadas por las organizaciones existentes en los países en vía de desarrollo, que en su mayoría están representadas por VSEs[1].

Estos estudios permiten a las VSEs definir y gestionar sus procesos, establecer que el conocimiento está dentro de la organización al tenerlos bien especificados y documentados, entregando productos software con la calidad esperada, respetando y cumpliendo con los plazos establecidos [13]. Estos aspectos conllevan a la consecución de certificados de calidad orientados a procesos y a ubicarse en niveles de madurez estipulados por los estándares.

Dichas organizaciones que están encaminadas a promover y dirigir esfuerzos en esta área de investigación ofrecen alternativas a las oportunidades y desafíos que presenta la creciente industria de software en países de América Central y América del Sur [10].

Continuando con los esfuerzos realizados y analizando las características de las empresas del sector TIC, especialmente las relacionadas con el desarrollo de software, el Ministerio de Tecnologías de la Información y las Comunicaciones de Colombia elaboró un informe presentado en 2015 [2] donde se caracteriza a las empresas ubicándolas de acuerdo a diferentes criterios de investigación y presentando análisis estadísticos concernientes a las regiones del país, flujo de activos, número de empleados, entre otros, que permiten conocer a nivel nacional el comportamiento y tipo de actividades de las organizaciones del sector. El estudio concluye en la necesidad de una alianza entre la academia y el sector productivo que permita apoyar a las empresas de la industria del software para que sean más sólidas y competitivas, lo cual se puede lograr mediante el ofrecimiento de nuevos esquemas de certificación de calidad.

Con base en lo anterior se puede afirmar que en las dos últimas décadas la comunidad de la ingeniería de software ha centrado sus intereses en la mejora de procesos software con el fin de aumentar la calidad en el producto así como la productividad en el desarrollo del mismo, sin embargo se tiene una idea generalizada de que el éxito en SPI solo está orientado hacia las grandes organizaciones que tienen la capacidad para optar por certificados de calidad propuestos por organizaciones como el Software Engineering Institute (SEI) o la International Organization for Standarization (ISO) [14], poniéndose en evidencia la carencia de oportunidades que las pequeñas organizaciones tienen, ya que los actuales estándares representan elevados costos no solo a nivel financiero, sino también en tiempo y esfuerzo requerido para su implementación, si se habla de los estándares más respetados y de uso extendido por las grandes empresas a nivel mundial como CMMI [13], mencionando también que estos estándares no se adecúan a sus características organizacionales, tales como el tamaño de la organización y sus objetivos de negocio[5].

Resulta importante mencionar otras de las razones por las cuales modelos como CMMI no pueden implementarse en las VSEs (a parte de las resaltadas anteriormente, y que se relacionan con aspectos económicos y de tiempo en su implementación), como lo es la cantidad y definición de los procesos establecidos para cada nivel de madurez con los que cuenta el modelo ya que este está diseñado para organizaciones que desarrollan y/o mantienen software de alto impacto (organizaciones de numerosos sectores, incluyendo aeroespacial, la banca, hardware, software, defensa, automoción y telecomunicaciones) [15], que cuentan con dependencias especializadas para las etapas del ciclo de vida del software y no para las pequeñas organizaciones donde existen características particulares que las definen, sus procesos no están claramente definidos o incluso estos pueden llegar a ser caóticos.

Sin embargo, en la actualidad las pequeñas organizaciones desarrolladoras de software también pueden contar con oportunidades para poder adquirir certificaciones que se puedan adecuar a su contexto y recursos, tales que, se pueda garantizar la calidad de los productos y/o servicios que dichas organizaciones ofrecen a sus clientes.

Por tanto, las características específicas de las pequeñas organizaciones desarrolladoras de software deben ser analizadas desde un punto de vista distinto al de las grandes empresas, para que estas puedan optar por programas de mejora de procesos software.

En efecto, surge la necesidad de abordar trabajos de investigación relacionados con el fortalecimiento de las VSEs desarrolladoras de software a través de una serie de estándares que se ajusten a este tipo de organizaciones los cuales consideren un conjunto de elementos que las caractericen donde se tenga en cuenta aspectos como número de empleados, presupuesto, tiempo y beneficio neto en el establecimiento de procesos de ciclo de vida del software, con el fin de brindar mayor productividad y competitividad en los mercados nacionales e internacionales.

En este artículo se presenta inicialmente (i) la estrategia de investigación llevada a cabo, (ii) el mapeo realizado entre los estándares ISO/IEC2 29110-4-1 e ISO/IEC 29110-5-1-2, (iii) el refinamiento y validación para los procesos de Gestión de Proyectos e Implementación de Software, (iv) El esquema de certificación ISO/IEC 29110-4-1 Perfil Básico y (v) conclusiones y trabajos futuros.

2. Estrategia de investigación:

La estrategia de investigación llevada a cabo para el desarrollo del trabajo se basa en cuatro etapas que giran en torno al esquema de certificación para el estándar ISO/IEC 29110-4-1.

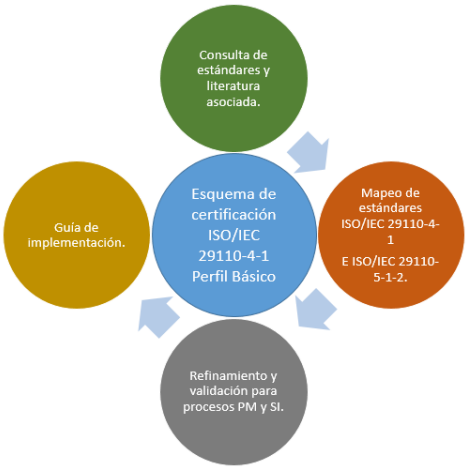


Figura 1 Estrategia de investigación

La Figura 1 muestra las cuatro etapas llevadas a cabo para el desarrollo del esquema de certificación tipo auditoría del estándar ISO/IEC 29110-4-1. Esta estrategia se basa en, (i) Consulta de estándares y literatura asociada, esta primera etapa tiene por finalidad introducir a los desarrolladores del trabajo de investigación en el contexto permitiendo afianzar conocimiento y ser la base para dar inicio al trabajo propuesto, (ii) Mapeo de estándares ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2, en esta etapa se realiza un mapeo entre los requisitos definidos en el estándar ISO/IEC 29110-4-1 y las

actividades y tareas definidas en el estándar ISO/IEC 29110-5-1-2 para determinar cómo los requisitos se pueden cumplir mediante un conjunto de tareas manteniendo un enfoque orientado a procesos que permita optar a la certificación, (iii) Refinamiento y validación para procesos PM (Project Management) y SI(Software Implementation). Una vez mapeado estos estándares se realizó un proceso de refinamiento y validación a partir del criterio del grupo de trabajo desarrollador de la tesis soportado por los elementos extraídos del estándar ISO/IEC 29110-3-1, el criterio obtenido desde académicos expertos en el área de gestión de proyectos y calidad de software, y los resultados de un ejercicio de comparación de los estándares realizado por grupos de estudiantes que colaboraron en el proceso, los cuales permitieron establecer una relación definitiva entre los estándares ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2 y (iv) Guía de implementación, la cual proporciona una guía que les permitiría a las pequeñas organizaciones poder seguir los pasos para lograr a satisfacción el cumplimiento de los requisitos definidos por el estándar.

3. Mapeo y validación

Partiendo de la importancia que tienen las VSE´s desarrolladoras de software en el mercado y de la necesidad de estas en adquirir certificaciones que les permitan garantizar que sus productos cumplen con especificaciones propuestas por estándares internacionales, se plantea un esquema de certificación que permita a una organización de normalización y certificación llevar a cabo procesos de auditoría dentro de las VSE con el objetivo de que estas se puedan certificar cumpliendo con una serie de requisitos planteados en la norma ISO/IEC 29110-4-1.

Para apoyar el cumplimiento de las VSE´s con los requisitos mencionados en el estándar ISO/IEC 29110-4-1, en esta investigación se relacionan dichos requisitos con las tareas propuestas por el estándar ISO/IEC 29110-5-1-2. El estándar ISO/IEC 29110-5-1-2 establece una serie de tareas y elementos que deben ser implementados por las pequeñas organizaciones para garantizar que sus procesos cumplen con lo especificado en esta parte de la norma y la empresa se encamine al logro de sus objetivos de negocio.

El objetivo de esta relación (requisitos de 29110-4-1 vs tareas de 29110-5-1-2) es ofrecer a las pequeñas empresas desarrolladoras de software un conjunto de tareas y productos de trabajo traídos desde la norma ISO/IEC 29110-5-1-2 que les permita cumplir los requisitos establecidos por la norma ISO/IEC29110-4-1 ya que esto permite que las organizaciones inicien un enfoque orientado a procesos en una organización para llevar a cabo el proceso de certificación.

Este proceso en el que se relacionan dos estándares es conocido como “mapeo”. El mapeo de normas es una de las estrategias más ampliamente usadas para relacionar modelos [16]. En este sentido, se pretende llevar a cabo el proceso de mapeo entre las partes 4-1 y 5-1-2 del estándar ISO/IEC29110 realizando las actividades descritas en el proceso de comparación y mapeo propuesta por [17]:

- Análisis de los modelos.
- Diseño del mapeo.
- Refinamiento y validación del mapeo.

En el Análisis de los modelos se estudia el estándar ISO/IEC 29110-4-1 el cual contiene los requisitos necesarios para cumplir con los procesos de PM (Project Management) y SI (Software Implementation). Es decir, esta parte de la norma describe

en términos de requisitos como llevar a cabo la gestión de proyectos y la implementación de software.

Por su parte, la norma ISO/IEC 29110-5-1-2 describe un conjunto de actividades, tareas, roles y productos de trabajo tanto del proceso de Gestión de Proyectos como del proceso de Implementación de Software.

Analizando la descripción tanto de requisitos como de actividades descritas por estas partes del modelo, cada uno de los requisitos de la ISO/IEC 29110-4-1 debe ser contrastado con el conjunto de tareas establecidas en el estándar ISO/IEC 29110-5-1-2 referente a los procesos mencionados.

Con base en lo anterior, el diseño del mapeo es desarrollado mediante la definición de una plantilla que muestra en el eje horizontal la definición de los requisitos de procesos especificados en el estándar ISO/IEC 29110-4-1 y en el eje vertical el conjunto relacionado de tareas con sus respectivos roles y productos de trabajo especificados en el estándar ISO/IEC 29110-5-1-2. Siguiendo el proceso descrito en el estándar ISO/IEC 29110-3-1 [18], se llevó a cabo la relación entre estos dos modelos teniendo en cuenta el análisis de los estándares y el criterio propio del grupo de trabajo dando como resultado el esquema que se muestra en la Tabla 1.Para este propósito el equipo de trabajo decidió conservar los nombres de tareas, productos de trabajo y roles en su idioma original.

					ISO/IEC 29110-5-1-2		
					Task	**Work product**	**Rol**
ISO/IEC 29110-4-1	**Process Requeriments**	**Project Management Process (PM)**	a	PM.1.1	Statement of work	PM, TL
				PM.1.2	Statement of work (Reviewed)	CUS
				PM.1.12	Statement of work (Scope, Objectives, Deliverables)	
			b	PM.1.3	Project plan	PM, TL
		Software Implementation Process (SI)	a	SI.1.1	Project Plan (Revisado)	PM,
				SI.3.2	Requirement Specification (Validated, Baselined)	TL, WT
			b	SI.2.1	Project Plan (Revisado)	TL, WT
				SI.3.1	Project Plan (Revisado)-Task	AN,
						DES,
				SI.4.1	Project Plan (Revisado)-Task	PR
				SI.5.1	Project Plan (Revisado)-Task	
				SI.6.1	Project Plan (Revisado)-Task	

Tabla 1Mapeo entre ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2

Finalmente el refinamiento y validación se desarrolló mediante una actividad realizada gracias al apoyo de dos expertos en gestión de proyectos y desarrollo de software del departamento de Sistemas de la Universidad del Cauca y grupos de estudiantes de pregrado de las asignaturas de gestión de proyectos informáticos y calidad de software del programa de Ingeniería de Sistemas de la Universidad del Cauca.

El refinamiento se llevó a cabo por medio de una actividad académica, un documento con el contenido y la definición de todos los requisitos especificados en el estándar ISO/IEC 29110-4-1 para el proceso de Gestión de Proyectos (PM - Project Management) e Implementación de Software (SI Software Implementation) y las actividades definidas para los mismos procesos establecidas en el estándar ISO/IEC 29110-5-1-2, con el fin de que los grupos de estudiantes y los expertos relacionaran, a partir de su conocimiento sobre el tema, los requisitos con cada una de las tareas de las normas mencionadas anteriormente.

Esta actividad dio como resultado sintetizado el esquema mostrado en la **¡Error! No s e encuentra el origen de la referencia.** y Tabla 3. Estas tablas muestran en el eje horizontal la definición de las tareas para el proceso de Gestión de Proyectos del estándar ISO/IEC 29110-5-1-2 y el eje vertical reúne los criterios de los grupos de estudiantes (Grp1, Grp2) y los expertos (Exp1, Exp2) para el proceso de Gestión de Proyectos, además de los criterios de los grupos de estudiantes (Grp1, Grp2, Grp3, Grp4) y los expertos (Exp1, Exp2) para el proceso de Implementación de Software. También se incluye la relación realizada por el equipo de trabajo desarrollador de la tesis (Est-Tesis) mostrado previamente en la Tabla 1.

Cada grupo relacionó los requisitos de la parte de la norma 4-1 con las tareas descritas por la parte 5-1-2 de la norma.

					PERSONAL INVOLUCRADO EN EL REFINAMIENTO					
					Requeriments ISO/IEC 29110-4-1					
				Task	Grp.1	Grp.2	Exp.1	Exp.2	Est-Tesís	Def
ISO/IEC 29110-5-1-2	Process Requeriments	Project Management Process (PM)	PM 1	PM 1.1	a	a,b,d	a	a,b	a	a
				PM 1.2	a	m	a	a	a	a
				PM 4.2	o	-	o	-	l,j	l,j

Tabla 2Mapeo definitivo entre ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2, Project Management (PM)

					PERSONAL INVOLUCRADO EN EL REFINAMIENTO							
					Requeriments ISO/IEC 29110-4-1							
				Task	Grp.1	Grp.2	Grp.3	Grp.4	Exp.1	Exp.2	Est-Tesís	Def
ISO/IEC 29110-5-1-2	Process Requeriments	Software Implementation Process (SI)	SI 1	SI 1.1	a	-	a	-	a	a	a	a
				SI 1.2	-	-	a,g,j	-	a(A)	-	-	a(A)

Tabla 3Mapeo definitivo entre ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2, Software Implementation (SI)

4. Esquema de certificación

Con base en el proceso desempeñado en etapas anteriores, se propone un esquema de certificación que relaciona los requisitos definidos en el estándar ISO/IEC 29110-4-1 con las tareas y elementos definidos en el estándar ISO/IEC 29110-5-1-2 a manera de plantilla de apoyo a la certificación para los procesos de Gestión de Proyectos (PM) e Implementación de Software (SI) (ver Tabla 4).

Durante el desarrollo del esquema de certificación se logró evidenciar la necesidad de llevar a cabo un esquema de apoyo a la certificación con el fin de que este se ajustara a un enfoque orientado hacia los organismos certificadores así como el desarrollo de una guía de implementación que nos permitiera abordar el enfoque desde la perspectiva de una pequeña organización.

Con base en lo anterior, se describe a continuación los dos enfoques que fueron llevados a cabo:

4.1 Esquema de certificación ISO/IEC 29110-4-1

En esta sección, se presenta el esquema de certificación para los requisitos definidos en el estándar ISO/IEC 29110-4-1, correspondientes con los procesos de Gestión de Proyectos (PM – Project Management) e Implementación de Software (SI – Software Implementation), con el objetivo de apoyar la certificación de pequeñas organizaciones en el perfil básico dentro del grupo de perfiles genérico presentados en el estándar.

Esta plantilla va orientada hacia los organismos certificadores con la intención de que esta sea utilizada como herramienta de apoyo a la certificación, la plantilla es un documento que contiene columnas que le permiten al personal encargado de realizar la auditoría marcar los requisitos satisfechos por la VSE con base en los indicadores de cumplimiento de requisitos, representados por las tareas y evidencias definidas en los modelos.

ISO/IEC 29110-4-1		ISO/IEC 29110-5-1-2			Evidencia		
Gestión Proyectos PM / Check		Actividad	Tareas / Check		PS	D	I
x	☐	PM.1	PM 1.1	☐	Declaración del trabajo.	Documento que describe el trabajo que se va a realizar relacionado con la implementación de Sw.	Actas de reuniones para la elaboración de la oferta inicial del proyecto.
			PM 1.2	☐	Plan de proyecto – Definir instrucciones de entrega.	Documento que especifica los elementos y componentes sw que se van a entregar al cliente.	Actas de reunión o correos que evidencian los componentes que se van a entregar al cliente. Acta de reunión o correo con la aceptación de los acuerdos.

Tabla 4Plantilla de apoyo a la certificación ISO/IEC 29110-4-1 para el proceso de PM

Esta plantilla nos muestra de izquierda a derecha, los requerimientos definidos en el estándar ISO/IEC 29110-4-1, las actividades y tareas definidas en el estándar ISO/IEC 29110-5-1-2 y las evidencias asociadas a cada una de las tareas, actividades y requerimientos. Es decir que a través de las evidencias encontradas ya se por medio de Productos de Salida (PS), evidencias Directas (D) o evidencias Indirectas (I) se puede determinar si una tarea que pertenece a una respectiva actividad es realizada por la organización, lo que permitirá garantizar el cumplimiento del requisito asociado a esa tarea.

Resulta importante resaltar que el desarrollo de la plantilla para el proceso de Implementación de Software es similar a la plantilla mostrada en la Tabla 4.

4.2 Guía de implementación

En esta sección, se presenta una guía de implementación para los requisitos definidos en el estándar ISO/IEC 29110-4-1, correspondientes con los procesos de Gestión de Proyectos (PM – Project Management) e Implementación de Software (SI – Software Implementation), con el objetivo de apoyar la implantación en las pequeñas organizaciones del perfil básico dentro del grupo de perfiles genérico presentados en el estándar.

Con el fin de facilitar la comprensión del proceso de auditoría y el cumplimiento de los requisitos establecidos en el estándar ISO/IEC 29110-4-1, la guía propuesta presenta cinco componentes: (i) las fases que las pequeñas organizaciones deben seguir para trabajar en un proceso de certificación, (ii) la definición de los indicadores de cumplimiento de requisitos, (iii) la definición de los indicadores de cumplimiento de requisitos a partir de tareas, (iv) la definición de indicadores de cumplimiento de requisitos a partir de productos de trabajo y (v) la relación de los elementos del esquema de certificación.

(i) Proceso de Auditoría

Para llevar a cabo la auditoría de las pequeñas organizaciones desarrolladoras de software que desean cumplir con los requisitos establecidos en el estándar ISO/IEC 29110-4-1, se debe seguir el proceso que se muestra en la Figura 2 [9]. Este proceso es conforme con las directrices establecidas en el estándar ISO/IEC 17021 e ISO/IEC 17065.

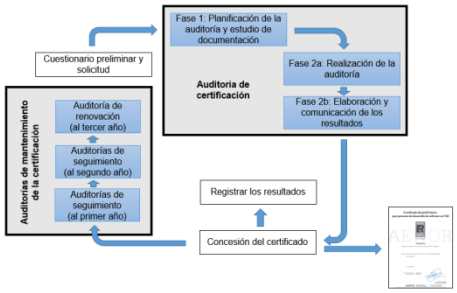


Figura 2. Proceso de auditoría

(ii) Definición de indicadores de cumplimiento de requisitos

Para que una pequeña organización cumpla con los requisitos establecidos en el estándar ISO/IEC 29110-4-1, se puede apoyar de una serie de indicadores de satisfacción de los requisitos logrando satisfacer un conjunto de tareas y productos de trabajo presentados en el estándar ISO/IEC 29110-5-1-2. La relación de las tareas descritas por el estándar ISO/IEC 29110-5-1-2 que apoyan el cumplimiento de los requisitos del estándar ISO/IEC 29110-4-1 se puede ver en las columnas tareas y evidencias de la Tabla 4 Plantilla de apoyo a la certificación ISO/IEC 29110-4-1.

La Tabla 5 presenta un ejemplo tomado de la tabla original con la descripción de los componentes deseados y las características de los productos de trabajo que la organización debe demostrar para satisfacer una tarea definida en el estándar ISO/IEC 29110-5-1-2.

El cumplimiento de las tareas y los productos de trabajo, permiten a la organización satisfacer los requisitos del estándar ISO/IEC 29110-4-1. Si la organización logra evidenciar a través de los productos de trabajo establecidos en la Tabla 5 el cumplimiento de los requisitos presentados en la Tabla 4 para el proceso de Gestión de Proyectos (PM – Project Management) e Implementación de Software (SI – Software Implementation), se podría garantizar la conformidad de los procesos de la organización con respecto a los requisitos establecidos en el perfil básico del estándar ISO/IEC 29110-4-1.

ID Producto de Trabajo	Producto de trabajo	Características	Tarea
PT1	Declaración del trabajo	Descripción del trabajo realizado relacionado con la implementación de sw. - Descripción del producto. - Propósito. - Requisitos generales del cliente. - Descripción del alcance. - Objetivos del proyecto. - Lista de productos que se van a entregar al cliente.	PM1.1

Tabla 5Indicadores de cumplimiento de requisitos

(iii) Definición de indicadores de cumplimiento de requisitos a partir de tareas

Continuando con la dinámica que se viene manejando para el cumplimiento de los requisitos la Tabla 6 muestra la relación entre los requisitos presentados en el estándar ISO/IEC 29110-4-1 y las tareas presentadas en el estándar ISO/IEC 29110-5-1-2. Cuando la VSE logre demostrar a partir de la ejecución de sus tareas, que se generan los productos de trabajo necesarios, se logra satisfacer los requisitos del estándar ISO/IEC 29110-4-1 y la pequeña organización puede optar por la certificación.

Requisito ISO/IEC 29110-4-1	Tarea
a	PM1.1, PM1.2
b	PM1.3
c	PM1.5
d	PM1.4, PM1.5, PM1.7, PM1.8

Tabla 6Indicadores de cumplimiento de requisitos a partir de tareas para el proceso de PM

(iv) Definición de indicadores de cumplimiento a partir de productos de trabajo

De igual manera la Tabla 7 muestra la relación entre los requisitos presentados en el estándar ISO/IEC 29110-4-1 y los productos de trabajo presentados en el estándar ISO/IEC 29110-5-1-2. Cuando la VSE logre demostrar que sus procesos generan las evidencias requeridas representadas por los productos de trabajo, se logra satisfacer los requisitos del estándar ISO/IEC 29110-4-1 y la pequeña organización puede optar por la certificación.

Requisito ISO/IEC 29110-4-1	ID Producto de Trabajo
a	PT1, PT2
b	PT3
c	PT3

Tabla 7Indicadores de cumplimiento de requisitos a partir de productos de trabajo para el proceso de PM

(v) Relación de los elementos del esquema de certificación

Finalmente este apartado muestra como un requisito presentado en el estándar ISO/IEC 29110-4-1 puede ser satisfecho a partir del cumplimiento de las tareas relacionadas presentadas en el estándar ISO/IEC 29110-5-1-2 y demostrando las evidencias o productos de trabajo correspondientes descritos en el mismo estándar.

En este sentido, los requisitos del proceso de Gestión de Proyectos (PM – Project Management) y del proceso de Implementación de Software (SI - Software Implementation) pueden ser satisfechos a partir del cumplimiento de las tareas asociadas, como se indica a continuación:

Requisito a) "Se deberá definir el alcance de trabajo para el proyecto, incluidos sus entregables", puede ser satisfecho a partir de la ejecución de las tareas:

- PM1.1 Revisar la declaración del trabajo.
- PM1.2 Definir con el cliente las instrucciones de entrega de cada uno de los entregables especificados en la declaración de trabajo.

Y se puede evidenciar con los productos de trabajo:

- PT1 Declaración del trabajo

- PT2 Instrucciones de entrega.

5. Conclusiones y trabajos futuros

5.1 Conclusiones

Del trabajo de investigación desarrollado se puede concluir que además del desarrollo del esquema de certificación para el estándar ISO/IEC 29110-4-1 es importante valorar el trabajo investigativo realizado para el desarrollo de dicho esquema. Es decir que el producto generado en este trabajo debe estar soportado por un proceso de investigación sistemático y coherente.

- Conceptos, generando conocimiento necesario para enfrentar de manera adecuada la propuesta establecida para el trabajo de grado con el fin de poder llevarlo a cabo de manera satisfactoria y logrando el cumplimiento total del alcance planteado.
- La diferencia existente entre una certificación por conformidad y una certificación por evaluación de capacidad de procesos para madurez organizacional. Una certificación por conformidad hace uso de un esquema de apoyo a la certificación con el fin de poder determinar si una organización cumple o no con una serie de requisitos específicos que le permitan lograr la certificación, por el contrario al evaluar por capacidad de procesos, se evalúa la capacidad que tiene el proceso orientado a la consecución de los objetivos de negocio de la organización, y de acuerdo a la evidencia obtenida se determina el grado de madurez que tiene una organización para lograr su certificación.
- La relación que existe entre las normas ISO/IEC 15504, ISO/IEC 12207 e ISO/IEC 29110, siendo este último estándar una base a la cual una VSE puede optar en primera instancia con el fin de poder encaminarse hacia una evolución organizacional que le permita a futuro implementar procesos definidos en estándares más robustos como lo es ISO/IEC 15504 y CMMI.

En segundo lugar, llevar a cabo un mapeo entre los estándares ISO/IEC 29110-4-1 e ISO/IEC 29110-5-1-2 permitió concluir que:

- Establecer una relación entre los requerimientos definidos en el estándar ISO/IEC 29110-4-1 y las tareas definidas en el estándar ISO/IEC 29110-5-1-2 para los procesos de Gestión de Proyectos e Implementación de Software, permite ofrecer a las pequeñas empresas desarrolladoras de software un conjunto de tareas y productos de trabajo traídos desde la norma ISO/IEC 29110-5-1-2 con el objetivo de cumplir los requisitos establecidos en la norma ISO/IEC29110-4-1 ya que esto conlleva que las organizaciones inicien un enfoque orientado a procesos en una organización para llevar a cabo el proceso de certificación.

Finalmente, el desarrollo de un esquema de certificación y su guía de implementación para el estándar ISO/IEC 29110-4-1 puede ofrecer:

- Una orientación para llevar a cabo un proceso de auditoría para optar por una certificación, con el fin de que la VSE comprenda las fases que son llevadas a cabo en el proceso y así poder ejecutarlas de manera adecuada con el objetivo de iniciar la conformidad en el estándar de calidad.
- La capacidad para establecer una cultura organizativa orientada a procesos, el reconocimiento de las VSE's como organizaciones que son capaces de desarrollar

productos software de calidad y la posibilidad de participar en mercados competitivos a nivel nacional e internacional.

- Una herramienta de apoyo para los organismos certificadores que les permita evaluar a través de las actividades y evidencias necesarias, los procesos que son llevados a cabo en las VSEs, determinando su conformidad con los requisitos del estándar ISO/IEC 29110-4-1 con el propósito de otorgar la certificación requerida por dichas organizaciones.

5.2 Trabajos futuros

- Hacer uso del esquema de apoyo a la certificación presentado en este trabajo de grado para verificar la conformidad de los requisitos definidos en el estándar ISO/IEC 29110-4-1 contra los procesos llevados a cabo en las pequeñas organizaciones desarrolladoras de software del departamento del Cauca.
- El presente trabajo puede servir de base para desarrollar esquemas de certificación por evaluación de capacidad de procesos y madurez organizacional para todos los perfiles descritos en el grupo de perfiles genéricos y otros propuestos por la Organización Internacional de Normalización (ISO), que permita a las pequeñas organizaciones desarrolladoras de software ubicarse en un perfil específico de acuerdo al nivel de madurez organizacional alcanzado.

Referencias

[1] F. J. Pino, F. García, F. Ruiz, and M. Piattini, "Adaptación de las normas ISO/IEC 12207: 2002 e ISO/IEC 15504: 2003 para la evaluación de la madurez de procesos software en países en desarrollo," in *JISBD*, 2005, pp. 187-194.

[2] S. d. sociedades, "Desempeño del sector software 2012-2014," 2015.

[3] ISO/IEC, "ISO/IEC TR 29110-1, Software engineering — Lifecycle Profiles for Very Small Entities (VSEs) — Part 1: Overview," ed, 2016.

[4] H. Oktaba, M. Piattini, F. J. Pino, and M. J. Orozco, *Competisoft: mejora de procesos software para pequeąs y medianas empresas y proyectos*: Alfaomega., 2009.

[5] F. J. Pino, F. García, and M. Piattini, "Software process improvement in small and medium software enterprises: a systematic review," *Software Quality Journal,* vol. 16, pp. 237-261, 2008.

[6] M. Manies and U. Nikual, "La Elicitación de Requisitos en el contexto de un proyecto software," *Revista Ingenierías USBMed,* vol. 2, pp. 25-29, 2011.

[7] P. Clarke and R. V. O'Connor, "The influence of SPI on business success in software SMEs: An empirical study," *Journal of Systems and Software,* vol. 85, pp. 2356-2367, 2012.

[8] C. G. von Wangenheim, S. Weber, J. C. R. Hauck, and G. Trentin, "Experiences on establishing software processes in small companies," *Information and software technology,* vol. 48, pp. 890-900, 2006.

[9] J. Garzás, F. J. Pino, M. Piattini, and C. M. Fernández, "A maturity model for the Spanish software industry based on ISO standards," *Computer Standards & Interfaces,* vol. 35, pp. 616-628, 2013.

[10] P. Bastos Tigre and F. Silveira Marques, *Desafíos y oportunidades de la industria del software en América Latina*: Cepal, 2009.

[11] Fedesoft, "Estudio de la caracterización de los productos y servicios de la industria de software y asociados," 2012.

[12] Intersoftware. (2010, Mayo). *VI Edición SEPG Latin America 2010*. Available: http://www.intersoftware.org.co/content/vi-edicion-sepg-latin-america-2010

[13] M. P. F. J. PINO, and C. M. FERNÁNDEZ, "Modelo de Madurez de la Ingeniería del Software," 2014.

[14] F. Pino, F. García, and M. Piattini, "Revisión sistemática de mejora de procesos software en micro, pequeñas y medianas empresas," *Revista Española de Innovación, Calidad e Ingeniería del Software,* vol. 2, pp. 6-23, 2006.

[15] C. P. Team, "CMMI for Development, version 1.2," 2006.

[16] SEI, "Process improvement in multimodel environments (PrIME Project). Available from: http://www.sei.cmu.edu/library/abstracts/webinars/18Jul2008.cfm. Accessed date: September 2009.," 2009.

[17] F. J. Pino, M. T. Baldassarre, M. Piattini, and G. Visaggio, "Harmonizing maturity levels from CMMI-DEV and ISO/IEC 15504," *Journal of software maintenance and evolution: Research and practice,* vol. 22, pp. 279-296, 2010.

[18] ISO/IEC, "ISO/IEC 29110-3-1. Systems and software engineering. Lifecycle profiles for Very Small Entities (VSEs). Part 3-1: Assessment guide," ed, 2015.

Identificación de competencias profesionales de Ingeniería de Software en el medio local y regional

Silvina Migani[1], María Inés Lund[1], Susana Chavez[1], Laura Zeligueta[2], María Isabel Masanet[1], Sandra Rodriguez[3], Adriana Martín[1]

1 Universidad Nacional de San Juan, San Juan, Argentina mlund@iinfo.unsj.edu.ar, {silvina.migani, susana.chvz, mimasanet, arianamartinsj}@gmail.com
2 Universidad Champagnat, Mendoza, Argentina zeliguetalaura@uch.edu.ar
3 Universidad Nacional de La Rioja, La Rioja, Argentina sandraorona27@yahoo.com.ar

Resumen. Existen muchos estudios relacionados a las competencias profesionales en diferentes áreas de conocimiento, en el ámbito laboral y académico. Sin embargo, es bastante más reducida la información respecto a las competencias profesionales vinculadas a la Ingeniería de Software en particular. Esto se debe, en gran medida, a que es una disciplina relativamente joven y además, completamente dinámica debido a la evolución vertiginosa de las Tecnologías de la Información y Comunicación. Y, en relación a las competencias requeridas en la región (San Juan, Mendoza y La Rioja), la información es prácticamente nula. Este trabajo describe el proceso de investigación realizado en torno a las competencias profesionales en Ingeniería de Software de la región. Consecuentemente, se elaboró una encuesta dirigida a empresas y organizaciones desarrolladoras de software, indagando sobre las competencias requeridas por ellas y los niveles de satisfacción percibidos en los profesionales informáticos que ingresan a las mismas. Este conocimiento permitirá, a futuro, identificar fortalezas y debilidades en la formación de los graduados de las carreras afines.

Palabras claves: Ingeniería de Software, Competencias Profesionales, Encuesta.

1 Introducción

La Ingeniería de Software (IS) es una disciplina relativamente joven y además es completamente dinámica [1], [2]. A esto se suma la globalización que supone nuevos retos para los trabajadores, quienes deben actualizarse continuamente, adquirir nuevas habilidades tecnológicas, pensamiento crítico y capacidad de síntesis. A su vez, el creciente nivel de complejidad de las actividades productivas en las que se inserta el profesional, provoca un mayor nivel de exigencia en sus capacidades.

Este contexto ha sido propiciado por el crecimiento vertiginoso de las Tecnologías de la Información y Comunicación (TIC), generando nuevos requerimientos, modelos y prácticas que necesariamente motivan la modificación del panorama laboral. En esta transformación se observa una tendencia progresiva hacia la colaboración entre personas para alcanzar un objetivo común. La tarea se organiza en equipos y cada integrante interactúa con el resto del grupo, muchas veces en forma distribuida, para

obtener una mejor productividad. La industria del software no es ajena a esta tendencia, los trabajos y las formas de trabajar han ido transformándose paulatina y constantemente.

Si bien existen muchos estudios relacionados a las competencias profesionales en el ámbito laboral y académico, es escasa la información respecto a las competencias profesionales para la IS, más aún si se consideran los enfoques innovadores que surgen de esta evolución, que sin dudas, conlleva a la generación de nuevas competencias.

Por lo tanto, es un desafío enfrentar la realidad del mercado local y regional en busca de información que permita enunciar y definir las competencias o habilidades, e incluso determinar el conocimiento o experiencia, que el medio requiere del profesional del software. Con este fin, se elaboró una encuesta dirigida a los responsables de las áreas de desarrollo de software de las empresas y organizaciones de software que existen en el medio regional, tanto públicas como privadas.

El artículo está organizado de la siguiente manera: en la sección 2 se presenta el marco teórico sobre el que se fundamenta el trabajo, en la siguiente se expone el trabajo realizado para obtener la encuesta que permitirá, a partir de los datos relevados, identificar las competencias profesionales requeridas en la IS, en la sección 4 se describen las conclusiones y el trabajo a futuro y finalmente las referencias citadas.

2 Marco Teórico

2.1. Ingeniería de Software

La IS es una disciplina emergente y está sometida a una evolución continua [1]. Depende fuertemente de la tecnología y de los métodos actuales y debe dar respuesta a los requerimientos de la realidad del mercado local [2]. Desde su concepción ha sido y es una disciplina ampliamente estudiada y aplicada tanto en ambientes académicos como en los productivos.

Existen numerosos procesos, técnicas, metodologías y modelos que dan fe de ello, muchos ya convertidos en estándares, tendientes a mejorar la calidad del proceso de desarrollo y en consecuencia mejorar la calidad del producto final. Entre ellos se encuentra el cuerpo de conocimientos básicos de IS conocido como SWEBOK [1].

El proceso de IS se define como "un conjunto de etapas parcialmente ordenadas con la intención de lograr un objetivo, en este caso, la obtención de un producto de software de calidad" [3].

2.2. Competencias Profesionales

El campo profesional se transforma y genera nuevos nichos de tareas. La mayor parte de los estudios señalan que una persona cambiará varias veces de empleo durante su etapa laboral activa. Por lo tanto, la versatilidad es una característica fundamental para desarrollar en la formación profesional. Es decir que la flexibilidad mental, la capacidad para adaptarse a nuevos desafíos, el saber cómo resolver problemas y situaciones problemáticas, la preparación para la incertidumbre, entre otras, son nuevas habilidades mentales que requerirán los profesionales del mañana y en las que se debe preparar [4].

El perfil de un profesional no sólo debe satisfacer los requerimientos de la sociedad, sino también debe proyectarlos de acuerdo a las necesidades de una región y de un país. En este sentido, es recomendable que su definición se realice a través de competencias [4]. Este concepto se comenzó a utilizar a partir de 1960, para identificar las capacidades con las que debía contar un trabajador para desempeñarse adecuadamente en el mercado de trabajo [5].

Las instituciones académicas deben ser conscientes de que su misión está en permanente transformación, su visión en constante efervescencia, y que su liderazgo (en el campo de la elaboración y transmisión del conocimiento) requiere de una nueva sensibilidad hacia los cambios sociales. Por ello, se vuelve imprescindible el contacto y el intercambio regular de opiniones con otros actores interesados como empresarios, referentes de la sociedad civil y gobiernos [4].

En este sentido, las competencias no se limitan a tener un conjunto de conocimientos teóricos y empíricos, y pueden no estar definidas como una tarea específica. Para Zarifian [6] la competencia se define como la inteligencia práctica para ejecutar situaciones que puedan apoyarse en conocimientos adquiridos y posibles de ser transformados de acuerdo con la complejidad de las mismas. Eso significa que las competencias de una persona no representan un estado o conocimiento invariable, sino que se transforman, evolucionan. En el proyecto Tuning Educational Structure in Europe [7], "las competencias representan una combinación de atributos (con respecto al conocimiento y sus aplicaciones, aptitudes, destrezas y responsabilidades) que describen el nivel o grado de suficiencia con que una persona es capaz de desempeñarse".

La Organización para la Cooperación y el Desarrollo Económicos (OCDE) ha hecho importantes aportes en la materia y define competencias como "la capacidad de poner en práctica de manera integrada habilidades, conocimientos y actitudes para enfrentar y resolver problemas y situaciones" [8].

Silva [5] las define como “un sistema de acción complejo que integra habilidades intelectuales, las actitudes y otros factores no cognitivos como la motivación, valores y emociones que son adquiridos y desarrollados por los individuos a lo largo de la vida y son indispensables para participar eficazmente en diferentes contextos sociales”.

Cruz [9], entiende las competencias como: Conocer y comprender: conocimiento teórico de un campo académico. Saber cómo actuar: aplicación práctica y operativa del conocimiento. Saber cómo ser: integrar los valores en la forma de percibir a los otros y de vivir en un contexto social.

En general, una competencia tiene que ver con una “combinación integrada de conocimientos, habilidades y actitudes conducentes a un desempeño adecuado y oportuno en diversos contextos”. La flexibilidad y capacidad de adaptación resultan claves para el mercado laboral actual y la educación como desarrollo general para que el profesional haga con lo que sabe [3].

En forma sintética, de acuerdo a los trabajos de [4], [7], [9], [10], las competencias pueden ser descritas y clasificadas en: Competencias Genéricas: son comunes a un profesional de cualquier titulación, estas competencias describen fundamentalmente conocimientos, habilidades, actitudes y valores, que son indispensables desarrollar para enfrentarse al mundo laboral y a una sociedad cambiante, donde las demandas tienden a hallarse en constante reformulación. Competencias Específicas: son las habilidades o destrezas relacionadas con el conocimiento de determinada disciplina o área de estudio,

son los conocimientos teóricos, prácticos o experimentales, los métodos y técnicas relevantes que atañen a las diversas áreas de la disciplina.

3. Trabajo Realizado

El trabajo realizado fue un proceso en el que se recorrieron varias etapas, desarrolladas en forma colaborativa por diferentes grupos de investigadores. En [11] se presentó la propuesta de identificación de competencias profesionales aplicada en este trabajo. Este proceso, si bien da la impresión de ser lineal, hay muchas tareas desarrolladas en forma paralela, y otras que necesitaron volver a etapas anteriores para aclarar y/o redefinir algunos conceptos.

3.1 Búsqueda y selección bibliográfica de trabajos de investigación relevantes

El objetivo de esta fase fue la identificación de artículos destacados en el área [12], [13], [14], [15], [16], a partir de los cuales se obtuvo el fundamento teórico. Cabe mencionar que, si bien no se siguió todo el rigor que exige, esta etapa fue orientada por el proceso propio de las revisiones sistemáticas de literatura [17].

3.2 Recopilación, análisis y definición de las competencias

Fueron múltiples las definiciones de competencias profesionales encontradas en la bibliografía, tal como se describe en el marco teórico. Por consiguiente, considerando las características esenciales de cada una de ellas, se acordó una definición ajustada al criterio del grupo de trabajo. Así, se llegó al siguiente concepto: “Las competencias profesionales constituyen un conjunto de capacidades, comportamientos, valores, habilidades y conocimientos que posibilitan a una persona desenvolverse efectivamente en una tarea específica”.

Como ya se mencionó en 2.2, se encontraron distintas formas de clasificación y denominación, en [7] y [12]se las clasifica en genéricas y específicas; en [13] en habilidades y atributos de comportamiento, habilidades cognitivas y habilidades técnicas; en [15] en competencias transversales y técnicas y por último, en [14] y [16], que sólo contemplan las genéricas, las denominan genéricas y transversales, respectivamente. Es decir, se observan fundamentalmente dos grupos de competencias, aquellas que no tienen una relación directa con la disciplina y otras que se encuentran directamente vinculadas a la misma, comúnmente denominadas como genéricas y específicas respectivamente.

A continuación se trabajó en particular sobre las competencias dentro del área de la IS, es decir, se inició el proceso de identificar el conjunto de competencias que un profesional debe contar para desempeñarse eficientemente en tareas propias al desarrollo de software.

Con el propósito de identificar un conjunto válido y completo de competencias, el proceso se encaró a través de dos métodos bien diferentes, llevados a cabo en forma paralela por dos grupos de trabajo:

- Basado en los trabajos científicos: se recopilaron todas las competencias enunciadas, tanto genéricas [12], [13], [14], [15], [16] como específicas [12] y [13], en una planilla donde se registró la denominación, la clasificación o característica a la que se vincula (denominada dimensión), y el artículo donde se la propone (fuente). Luego, se procedió al debate y al análisis minucioso de cada una de ellas, hasta consensuar y alcanzar un conjunto sintético y representativo de cada concepto plasmado en el superconjunto inicial, ya que éste incluía competencias semánticamente solapadas en forma total o parcial. Asimismo se analizó y estableció, para cada competencia, la dimensión o dimensiones correspondientes.
- Basado en las ofertas del mercado laboral: en [15] se estudian las competencias demandadas a los egresados en Informática a partir de datos obtenidos de ofertas de empleo publicadas en los principales periódicos de España. A partir de ese trabajo, se decidió considerar las ofertas laborales del área de la IS como otra fuente de datos. En consecuencia, se identificaron sitios digitales, tanto en el ámbito local como nacional, desde donde se recogió otro conjunto de competencias. Cabe mencionar que en este caso, los requerimientos aludían fundamentalmente a competencias específicas y conocimientos concretos.

En la Tabla1 se muestra el conjunto obtenido de competencias, distinguidas por tipo y vinculadas a la dimensión o clasificación correspondiente, después de analizar y consensuar los resultados obtenidos por ambos grupos de trabajo.

Tabla 1. Resultado consensuado de las competencias identificadas.

C.	Competencia	Dimensión
G	Habilidad para trabajar en forma autónoma	Autonomía
E	Tener iniciativa y ser resolutivo	Autonomía
N	Capacidad para tomar decisiones	Autonomía y liderazgo
	Capacidad para tomar decisiones	Autonomía y liderazgo
E	Capacidad para Investigar	Capacidad de aprender e
R		investigar
I	Capacidad para Aprender	Capacidad de aprender e
C		investigar
A	Capacidad de comunicación oral	Comunicación
S	Capacidad de comunicación escrita	Comunicación
	Tener capacidad para aportar soluciones alternativas o novedosas a los problemas o situaciones	Creatividad
	Capacidad para adaptarse a nuevas situaciones (organizativas, tecnológicas, etc)	Flexibilidad/adaptación
	Capacidad de motivar y conducir hacia metas comunes	Liderazgo
	Capacidad para la dirección de equipos	Liderazgo
	Capacidad para argumentar y justificar lógicamente las decisiones	Liderazgo y trabajo en

tomadas y las opiniones	equipo
Capacidad de comunicación en un segundo idioma.	Manejo de otro idioma
Capacidad de comunicación efectiva en inglés	Manejo de otro idioma
Capacidad de trabajar en contextos internacionales	Multiculturalidad
Capacidad para organizar y planificar	Planificación
Capacidad para formular y gestionar proyectos	Planificación y organización
Capacidad de iniciativa	Proactividad
Capacidad de ser activo y emprendedor	Proactividad
Capacidad para identificar, plantear y resolver problemas	Resolución de problemas
Capacidad de aplicar los conocimientos en la práctica	Resolución de problemas
Actuar en el desarrollo profesional con responsabilidad	Responsabilidad en el trabajo
Actuar en el desarrollo profesional con ética profesional	Responsabilidad en el trabajo
Ser una persona confiable	Responsabilidad en el trabajo
Compromiso con la calidad	Responsabilidad en el trabajo
Compromiso con la preservación del medio ambiente	Responsabilidad social
Compromiso ciudadano	Responsabilidad social
Capacidad de desempeñarse de manera efectiva en equipos de trabajo.	Trabajo en equipo
Capacidad de crítica y autocrítica	Trabajo en equipo
Habilidades interpersonales	Trabajo en equipo
Habilidad de Negociación	Trabajo en equipo y liderazgo
Habilidad en el uso de las tecnologías de la información y comunicación	Uso de tecnologías
E S Gestionar proyectos de software (planificar, organizar, guiar).	Gestión de proyectos
Aplicar metodologías/métodos/estrategias de desarrollo de software.	Metodologías desarrollo SW

P E C I F I C A S	Utilizar métodos de elicitación de requisitos.	Requerimientos
	Utilizar métodos/técnicas de análisis y especificación de requisitos.	Requerimientos
	Utilizar herramientas de software para la especificación de requisitos. Usar técnicas de verificación y validación de requisitos.	Requerimientos
	Diseñar y especificar la arquitectura de software (componentes de software).	Requerimientos Diseño
	Reconocer oportunidades para refactorizar/reutilizar código. Usar patrones de diseño para el diseño de módulos y componentes de software.	Diseño Diseño
	Conocer y aplicar lenguajes y herramientas de programación.	Construcción
	Conocer y aplicar frameworks y plataformas de desarrollo.	Construcción
	Diseñar e implementar bases de datos.	Diseño y construcción
	Aplicar criterios de seguridad, privacidad e integridad de los datos.	Seguridad
	Diseñar e implementar redes.	Diseño y construcción
	Diseñar pruebas de testing de software.	Testing
	Ejecutar planes de prueba (unitarios y de integración) e identificar acciones correctivas.	Testing
	Realizar auditoria de sistemas y peritaje informático.	Auditoria

3.3. Elaboración de la encuesta

Una vez obtenido el conjunto final de competencias, se procedió al diseño y elaboración de la encuesta, como así también a la definición de los destinatarios y a la selección del método de recolección de datos.

Diseño y elaboración de la encuesta. El núcleo de la encuesta está dado por las competencias identificadas en la etapa anterior, pero, para interpretar con mayor precisión la información recogida, se incluyen también ítems que permiten capturar, en cierta medida, el tipo de organización a la que pertenece el encuestado (en cuanto a tamaño, tipo de actividad, tipo de industria de sus clientes, antigüedad en el mercado, etc.), es decir, una serie de datos descriptivos que permitirá contextualizar las respuestas. También se incluyeron algunas preguntas de respuesta abierta que permiten a los encuestados incorporar competencias y/o aspectos relevantes que no hayan sido considerados en la misma.

En relación a cada competencia evaluada, los criterios de valoración son los siguientes:

Nivel de Importancia: Valora cuán significativa es la competencia para el desempeño profesional. Los posibles valores considerados son:0- Desconoce/No aplica, 1- Sin importancia, 2- Poco importante, 3- Importante, 4- Muy Importante, 5- Imprescindible.

Nivel de Cumplimiento/Satisfacción: Indica la medida en que las competencias/habilidades son satisfechas por el personal cuando ingresa a la Organización/Empresa. Los valores posibles son:0- Desconoce/No aplica, 1- Nada, 2- Poco, 3- Suficiente.

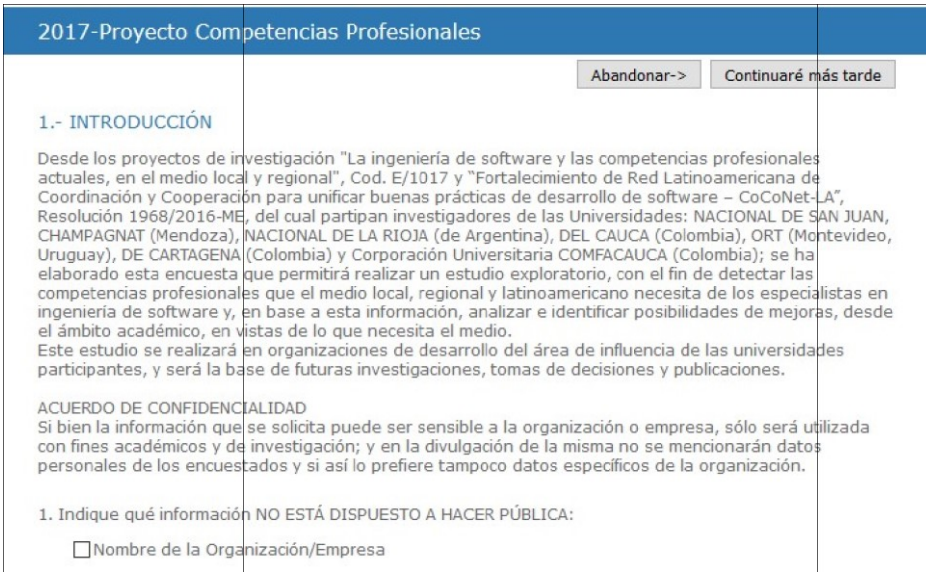
Destinatarios. Se dispuso dirigir la encuesta a los responsables (líderes de equipos, jefes, encargados o freelances, etc.) de Organizaciones/Empresas dedicadas al desarrollo de software, tanto del ámbito público como privado, del medio local y regional.

Método de recolección. El método elegido fue una encuesta on-line, por medio de ENCUESTAFACIL [18]. Esta es una herramienta web, que permite diseñar y elaborar encuestas en forma rápida y sencilla. Ofrece un informe preliminar de los datos recolectados y también una planilla de cálculo para que se procese la información por medio de otros paquetes estadísticos, si se desea. Este paquete tiene una interfaz sencilla para su uso y el Instituto de Informática de la FCEFN de la UNSJ tiene licencia Bono Oro Universia. El diseño estuvo limitado a las posibilidades que la herramienta brinda.

3.4. Revisión de la encuesta

En el proceso de definición de las competencias y elaboración de la encuesta, se trabajó en forma colaborativa con los investigadores que participan en el proyecto "La ingeniería de software y las competencias profesionales actuales en el medio local y regional" Financiado por la UNSJ (U.N. de San Juan, U. Champagnat y U.N. de La Rioja), y para la etapa de revisión se sumaron los investigadores del proyecto financiado por la SPU "Fortalecimiento de Red Latinoamericana de Coordinación y Cooperación para unificar buenas prácticas de desarrollo de software. CoCoNet-LA" (U. de Cartagena, U. del Cauca y Unicomfacauca de Colombia, Universidad ORT de Montevideo Uruguay y, U. Champagnat y U.N. de San Juan de Argentina). Las vías de comunicación fueron videoconferencias, correos electrónicos, mensajería instantánea y como propuesta de trabajo, realizar todo bajo una misma plataforma colaborativa (GoogleDrive).

A continuación, en la figura 1, se presenta parcialmente la encuesta definitiva, presentando las capturas de pantalla de la misma en la herramienta encuestafacil.com. La misma está dividida en 5 partes 1- Introducción, 2- Datos Descriptivos, 3- Competencias genéricas, 4- Competencias específicas y 5- Observaciones finales, para que el encuestado realice cualquier comentario que crea pertinente.



2017-Proyecto Competencias Profesionales

Abandonar-> Continuaré más tarde

1.- INTRODUCCIÓN

Desde los proyectos de investigación "La ingeniería de software y las competencias profesionales actuales, en el medio local y regional", Cod. E/1017 y "Fortalecimiento de Red Latinoamericana de Coordinación y Cooperación para unificar buenas prácticas de desarrollo de software – CoCoNet-LA", Resolución 1968/2016-ME, del cual partipan investigadores de las Universidades: NACIONAL DE SAN JUAN, CHAMPAGNAT (Mendoza), NACIONAL DE LA RIOJA (de Argentina), DEL CAUCA (Colombia), ORT (Montevideo, Uruguay), DE CARTAGENA (Colombia) y Corporación Universitaria COMFACAUCA (Colombia); se ha elaborado esta encuesta que permitirá realizar un estudio exploratorio, con el fin de detectar las competencias profesionales que el medio local, regional y latinoamericano necesita de los especialistas en ingeniería de software y, en base a esta información, analizar e identificar posibilidades de mejoras, desde el ámbito académico, en vistas de lo que necesita el medio.
Este estudio se realizará en organizaciones de desarrollo del área de influencia de las universidades participantes, y será la base de futuras investigaciones, tomas de decisiones y publicaciones.

ACUERDO DE CONFIDENCIALIDAD
Si bien la información que se solicita puede ser sensible a la organización o empresa, sólo será utilizada con fines académicos y de investigación; y en la divulgación de la misma no se mencionarán datos personales de los encuestados y si así lo prefiere tampoco datos específicos de la organización.

1. Indique qué información NO ESTÁ DISPUESTO A HACER PÚBLICA:

☐ Nombre de la Organización/Empresa

2.- DATOS DESCRIPTIVOS

*2. Nombre del Encuestado:

*3. Rol en la Organización/Empresa:

*4. Nombre de la Organización/Empresa:

*5. País donde reside la Organización/Empresa:

○ Argentina ○ Colombia ○ Uruguay ○ Otro

*16. Indique el Dominio/Industria de las aplicaciones que desarrollan (Bancario, Petrolero, Facturación y Ventas, etc.):

3.- COMPETENCIAS GENÉRICAS

Este cuestionario contiene enunciados relacionados a competencias/habilidades requeridas o esperadas en los desarrolladores de software que Ud. debe valorar en base a dos criterios, que se exponen a continuación:
NIVEL DE IMPORTANCIA: Valorar cuán significativa es la competencia para el Desempeño Profesional en la Organización/Empresa. Los niveles de importancia considerados son:
0- Desconoce/No aplica
1- Sin importancia
2- Poco importante
3- Importante
4- Muy importante
5- Imprescindible
NIVEL DE CUMPLIMIENTO/SATISFACCIÓN: Indicar en qué medida las competencias/habilidades son satisfechas por el personal cuando ingresa a trabajar en la Organizacion/Empresa.
0- Desconoce/No aplica
1- Nada
2- Poco
3- Suficiente

*17. Valore las competencias según lo indicado anteriormente:

	NIVEL DE IMPORTANCIA	NIVEL DE SATISFACCION
Compromiso ciudadano (actuar solidaria y responsablemente en la sociedad)	Elija una	Elija una
Compromiso con la preservación del medio ambiente.	Elija una	Elija una
Actuar en el desarrollo profesional con responsabilidad	Elija una	Elija una
Actuar en el desarrollo profesional con ética profesional	Elija una	Elija una
Ser una persona confiable	Elija una	Elija una

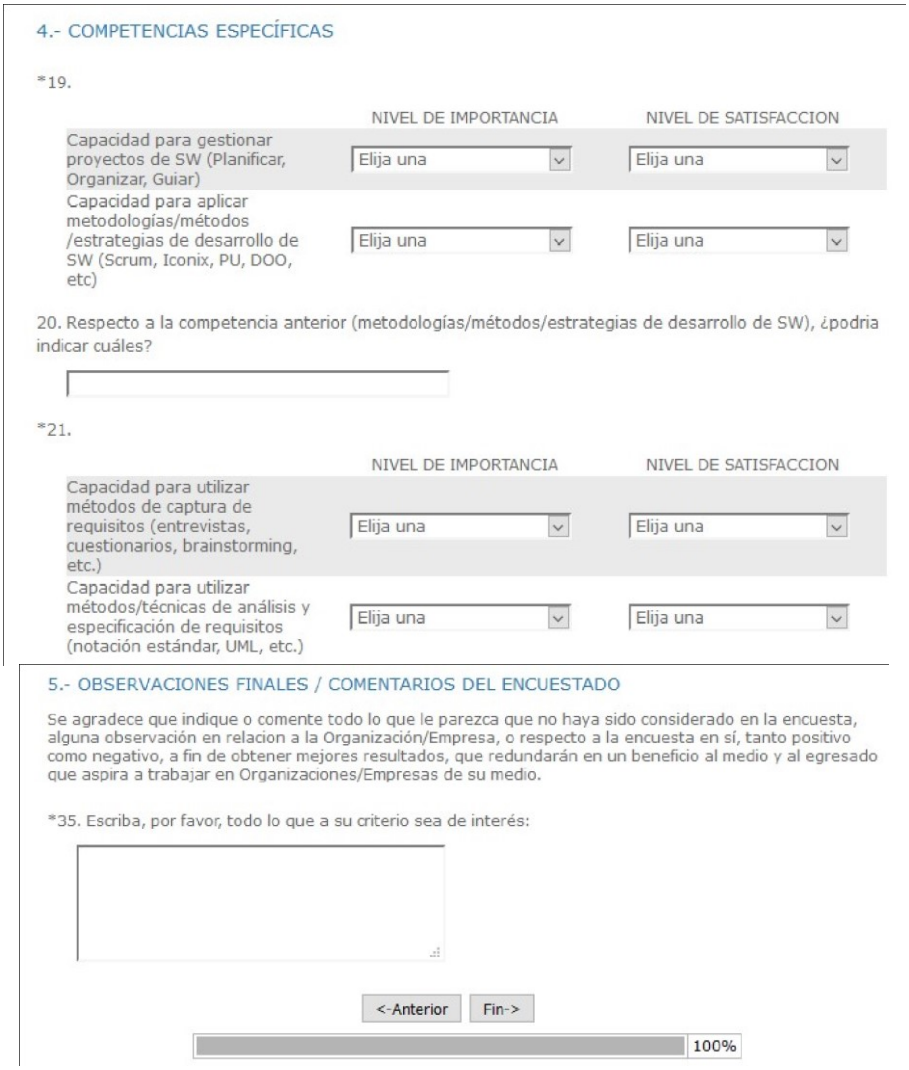


Fig. 1. Vistas parciales de la encuesta definitiva (encuestafacil.com)

3.5. Búsqueda y selección de encuestados

La búsqueda de los potenciales encuestados se llevó a cabo a través de distintos medios.

En San Juan se tomó en parte el registro de egresados del Colegio de Informáticos de la provincia, de los socios de la Cámara de Empresas de Tecnologías y Comunicaciones de San Juan (CASETIC) y profesionales registrados en la comunidad social LinkedIn. Se realizó un primer contacto a través de correo electrónico, se

explicaron los objetivos del proyecto y se consultó respecto a la predisposición a colaborar respondiendo la encuesta. En algunos casos no se obtuvo respuesta inmediata, por lo que fue necesario una segunda comunicación y en otros se debió establecer un contacto vía telefónica o personal.

En La Rioja se tomó como base el relevamiento realizado por los alumnos integrantes del proyecto con el acompañamiento de los docentes. La primera fuente fueron los docentes de las carreras del área de Informática y de Sistemas de la Universidad, esto derivó a una cadena de contactos y así se pudo aumentar la base de datos de posibles encuestados. La mayoría de los docentes pertenecen a las organizaciones/empresas donde los mismos son partícipes directos y en otros casos sólo son unipersonales y/o contratados por proyectos. Un primer contacto fue a través de sus teléfonos particulares, correos electrónicos y en ocasiones personalmente, donde se explicaron los objetivos del proyecto y se los comunicó de la encuesta que a posteriori recibirían.

La provincia de Mendoza cuenta con un polo tecnológico desde donde se tomaron las empresas asociadas (www.poloticmendoza.org), y a su vez un relevamiento para identificar organizaciones estatales o públicas para enviarles la encuesta.

3.6. Estado actual

El 27/09/2017 se abrió la encuesta al público, el link a la misma ha sido enviado por correo electrónico, desde cada universidad a su medio de influencia y se están comenzando a recibir respuestas.

4. Conclusiones y trabajo a futuro

El trabajo realizado ha permitido analizar y profundizar el conocimiento de las competencias profesionales en general, y en particular en el área de la IS; además de advertir su importancia dada la fuerte influencia que provocan en el desarrollo profesional.

Actualmente, el grupo de trabajo se encuentra en la etapa de acompañar y asistir a los encuestados y a la espera de obtener el mayor número posible de respuestas. Esto permitirá efectuar no sólo un análisis descriptivo por provincia y región que manifieste la visión del mercado laboral de los profesionales del software, sino también un estudio exploratorio, identificando fortalezas y debilidades y plantear soluciones, para que los profesionales egresen no sólo con la preparación necesaria, sino con las competencias desarrolladas para su mejor inserción y desempeño laboral. Este último aspecto incluye la revisión de los planes de estudios de las carreras afines de las universidades involucradas, que sin duda será por demás beneficiosa. Además, permitirá conocer las tecnologías, metodologías, estándares, etc. que son elegidos y utilizados en la industria; información por demás valiosa tanto en el ámbito académico como en el de la investigación.

Agradecimientos. Se agradece a los investigadores de la Universidad de Cauca: José L. Arciniegas, de ORT Uruguay: Gerardo Matturro, de Corporación Universitaria Comfacauca: Alex A. Torres, de U. Champagnat: Fernando Pinciroli y el equipo de investigación de UNLaR, por su colaboración en la revisión de la encuesta.

Referencias

[1] P. Bourque and R. E. Fairley, Guide to the Software Engineering - Body of Knowledge. 2014.
[2] E. Bareiša, E. Karčiauskas, E. Mačikėnas, and K. Motiejūnas, "Research and development of teaching software engineering processes," 2007, p. 75:1--75:6.
[3] I. Jacobson, G. Booch, and J. Rumbaugh, The Unified Software Development Process, 1 edition. New Jersey: Addison-Wesley Professional, 1999.
[4] P. Beneitone, C. Esquetini, J. Gonzalez, M. Maleta, G. Siufi, and R. Wagenaar, Eds., Reflexiones y perspectivas de la Educación Superior en América Latina. Informe final Proyecto Tuning- (2004-2007). Universidad de Deusto, Universidad de Groningen, 2007.
[5] M. Silva Laya, "¿Contribuye la Universidad Tecnológica a formar las competencias necesarias para el desempeño profesional? Un estudio de caso," Rev. Mex. Investig. Educ., vol. 13, no. 38, pp. 773–800, Sep. 2008.
[6] P. Zarifian, "Objectif Compétence. Pour une nouvelle logique," 1999.
[7] González, J. and Wagenaar, R., Eds., Tuning Educational Structures in Europe. Informe Final Fase 1. U. of Deusto - U. of Groningen, 2003.
[8] INEE, PISA para docentes. La evaluación como oportunidad de aprendizaje, Mexico. Instituto Nacional para la Evaluación de la Educación, 2005.
[9] M. A. Cruz, "Taller sobre el proceso de aprendizaje-enseñanza de competencia." Instituto de Ciencias de la Educacion de Zaragoza, 2005.
[10] A. Escalona O. and B. Loscertales P., "Buenas prácticas en la enseñanza y el aprendizaje de competencias genéricas," 2012.
[11] M. I. Lund, J. L. Arciniegas, G. Matturro, M. E. Torres Bermúdez, A.A. Monroy Ríos, L. Zeligueta, and S. Rodriguez, "Propuesta para la Identificación de Competencias profesionales en la Ingeniería de Software," in XI Congreso Colombiano de Computación, 2016, p. 4.
[12] J. L. Contreras Véliz et al., Proyecto Tuning América Latina. Educación Superior en América Latina: reflexiones y perspectivas en Informática. Bilbao: Universidad de Deusto, 2013.
[13] IEEE Computer Society, Software Engineering Competency Model (SWECOM) • IEEE Computer Society, V1.0. IEEE Computer Society, 2014.
[14] M. J. García García, L. Fernández Sanz, M. J. López Terrón, and Y. Blanco Archilla, "Demanded Key Skills in Technical Companies. Assessment Procedures in Higher Education," in TICAI (TICs para a Aprendizagem da Engenharía), C. Vaz de Carvalho, M. Llamas Nistal, and R. Silveira, Eds. 2008, pp. 141–146.
[15] F. Sánchez et al., "Competencias Profesionales del Grado en Ingeniería Informática," in TICAI (TICs para a Aprendizagem da Engenharía), C. Vaz de Carvalho, M. Llamas Nistal, and R. Silveira, Eds. 2008, pp. 147–154.
[16] Confedi, Competencias en Ingeniería. "Declaración de Valparaíso" Buenos AIres: Universidad de Fasta MdP, 2014.

[17] K. Petersen, S. Vakkalanka, and L. Kuzniarz, “Guidelines for conducting systematic mapping studies in software engineering: An update,” Inf. Softw. Technol., vol. 64, pp. 1–18, 2015.

[18] S. L. Copyright © 2005-2017 Encuesta Fácil, “encuestas online - software encuesta - crea y envia cuestionarios facilmente - Encuesta Facil.” [Online]. Available: http://www.encuestafacil.com/Default.aspx.

Herramienta de Accesibilidad Web: ARWeb

Pablo Pandolfo[1], Adrián De Armas[1], Bibiana Rossi[1]

[1]Instituto de Tecnología (INTEC) de la Universidad Argentina de la Empresa, UADE, Buenos Aires, Argentina
{ppandolfo, adearmas, birossi}@uade.edu.ar

Resumen. En 2010, se sancionó en Argentina, la ley 26.653 de Accesibilidad de la Información, que requiere que los sitios web del sector público (entes públicos estatales y no estatales, empresas privadas concesionarios de servicios públicos y empresas prestadoras o contratistas de bienes y servicios) respeten las normas y requisitos de accesibilidad recomendados por la ONTI (Oficina Nacional de Tecnologías de la Información). Este artículo presenta el desarrollo de la Herramienta ARWeb, que permite identificar problemas de Accesibilidad Web en forma automática de acuerdo con la norma vigente en Argentina desde agosto de 2014.

Palabras clave: accesibilidad web, pautas de accesibilidad, WCAG 2.0, HTML, CSS, ley 26.653, evaluación automática de accesibilidad.

1 Introducción

La accesibilidad web es la facilidad para que cualquier persona pueda acceder al contenido de un sitio en diferentes condiciones [1], de considerarse como el diseño universal, un diseño para todas las personas, cualesquiera sean sus condiciones físicas y ambientales. Todas las personas son diferentes y acceden a Internet de forma diferente, y la accesibilidad pretende que las interfaces de usuario se adapten y acomoden a esas diferencias, de forma que cualquiera pueda utilizarlas y acceder a la información. La Accesibilidad Web considera a las personas en su diversidad [2], alcanza a las personas con discapacidad, personas mayores y niños, personas con equipos anticuados, personas con bajos recursos económicos, conexiones lentas [3].

Desarrollar una página web accesible no supone tener que dejar de usar imágenes y colores. Algunos sitios web desarrollan una versión normal de la página y otra versión accesible para personas con algún tipo de discapacidad. No se debe considerar la accesibilidad como una opción que discrimine a las personas que no puedan acceder a la versión normal [4]. En ocasiones, la versión accesible es una versión reducida en la que no está disponible todo el contenido de la web original. No tiene sentido crear una versión poco accesible y otra accesible. Si se va a realizar un sitio web, lo mejor es realizar un único sitio.

Para hacer una web accesible existen razones éticas, técnicas y económicas que aportan ventajas tanto a los responsables del sitio web como a los usuarios [2]:

- Beneficia a todos los usuarios.
- Incrementa la audiencia de la web.
- Mejora el posicionamiento en buscadores.

- Maximiza la reutilización de contenidos.
- Brinda mayor compatibilidad con navegadores y dispositivos.
- Reduce costos en mantenimiento.
- Demuestra la responsabilidad social y atención hacia todos los usuarios sin discriminación.
- Cumple con las leyes y normativas que la regulan.
- Mejora la usabilidad.

Existen distintos motivos que dificultan la implementación de las pautas de accesibilidad web [5]:

- Desconocer la composición del público de un sitio.
- Desconocer cómo ofrecer la información de forma accesible.
- Considerar que la accesibilidad web implica más trabajo o mayores costos.
- Considerar incompatible un buen diseño y un sitio accesible.

El artículo se organiza de la siguiente manera: la sección 2 expone las normativas y estándares sobre accesibilidad web. En la sección 3 se presentan los organismos regulatorios de la accesibilidad web. En la sección 4 se introducen los niveles de orientación de las WCAG 2.0. En la sección 5 y 6 se describe el desarrollo de la herramienta ARWeb y la prueba de un sitio web respectivamente. Finalmente, la sección 7 se reseñan las conclusiones del trabajo.

2 Normativas y Estándares sobre Accesibilidad Web

En los orígenes de Internet no existía ningún tipo de regulación, legal o técnica, sobre accesibilidad, pero en los últimos años surgieron diversas iniciativas con la intención de normalizar la accesibilidad. En Argentina, la legislación vigente comprende:

- *Decreto 1172/2003* de Acceso a la Información Pública que establece que toda persona tiene derecho a requerir, consultar y recibir información de cualquier organismo público. El decreto señala como información: documentos escritos, fotográficos, grabaciones, soporte magnético, digital y que haya sido creada u obtenida por organismos públicos o que obre en su poder o bajo su control. En ese sentido se prevé que la información debe estar adecuadamente organizada, sistematizada y disponible para garantizar un amplio y fácil acceso. [6]
- *Decreto 378/2005* del Plan Nacional de Gobierno Electrónico en el que se impulsa el uso intensivo de las Tecnologías de la Información y las Comunicaciones (TICs) para mejorar la relación del gobierno con los ciudadanos, e incrementar eficiencia de la gestión y la transparencia y la participación. [7]
- *Ley 26.653* que garantiza y amplía el acceso a la información pública a las personas con discapacidad. Mediante esta ley, el Estado Nacional debe respetar en los diseños de sus páginas Web las normas y requisitos sobre accesibilidad de la información que faciliten el acceso a sus contenidos para todas las personas con discapacidad, con el objeto de garantizarles la igualdad real de oportunidades y trato, evitando así todo tipo de discriminación. [8]

- *Decreto 355/2013* que designa a la ONTI como la responsable de emitir las normas técnicas sobre accesibilidad web. [9]
- *Disposición ONTI 02/2014* que establece las normas de accesibilidad web basadas en las WCAG 2.0, define como obligatorio el nivel A y un sistema de puntos para establecer si una página es suficientemente accesible o no. Durante el primer período evaluatorio (aún vigente) el umbral de puntos se establece en 50 puntos. [10]

3 Organismos Regulatorios de la Accesibilidad Web

Existen diversos organismos que regulan la accesibilidad web. Algunos son reconocidos a nivel mundial y otros son propios de los países. [4]

A nivel internacional, el W3C es una organización fundada en el año 1994 y formada por empresas y organismos de distintos países y sectores que trabajan para desarrollar estándares web. La misión del W3C es la de guiar la web hacia su máximo potencial a través del desarrollo de protocolos de uso común y pautas que propicien el crecimiento y evolución, asegurando la interoperabilidad de la web para fomentar su universalidad. [11]

En el año 1998, la W3C presentó la Iniciativa de Accesibilidad Web (WAI, Web Accesibility Initiative) que es un grupo interno de trabajo permanente de la W3C, y que se enfoca en hacer la web accesible. La WAI colabora con organizaciones de todo el mundo, mejorando la accesibilidad en la web a través de cinco áreas de trabajo: tecnología, pautas, herramientas de validación y reparación, educación y difusión e investigación y desarrollo. [12]

4 Niveles de Orientación de las WCAG 2.0

Las WCAG 2.0 se han desarrollado con el fin de proporcionar un estándar para la accesibilidad del contenido web que satisfaga las necesidades de personas, organizaciones y gobiernos a nivel internacional. Las WCAG 2.0 se diseñaron para ser aplicadas a una amplia gama de tecnologías web ahora y en el futuro, y para ser verificables con una combinación de pruebas automatizadas y evaluación humana. [4]

Los individuos y organizaciones que emplean las WCAG son un grupo amplio que incluye diseñadores y desarrolladores web, profesores y estudiantes. Para poder satisfacer las necesidades de esta audiencia, se proporcionan varios niveles de orientación: principios, pautas, criterios de conformidad verificables y una amplia colección de técnicas suficientes, técnicas recomendables y fallos comunes documentados con ejemplos, enlaces a recursos adicionales y código. [3]

- *Principios*: en el nivel más alto se sitúan cuatro principios que proporcionan los fundamentos de la accesibilidad web: perceptible, operable, comprensible y robusto.
- *Pautas*: por debajo de los principios están las pautas que proporcionan los objetivos básicos que los autores deben lograr. Las pautas proporcionan el marco y los objetivos generales que ayudan a comprender los criterios de conformidad y a implementar mejor las técnicas.

- *Criterios de Conformidad*: para cada pauta se proporcionan los criterios de conformidad verificables que permiten emplear las WCAG 2.0 en aquellas situaciones en las que existan requisitos y necesidad de evaluación de conformidad como: especificaciones de diseño, regulación o acuerdos contractuales. Con el fin de cumplir con las necesidades de los diferentes grupos y situaciones, se definen tres niveles de conformidad: A (el más bajo), AA y AAA (el más alto).
- *Técnicas suficientes y recomendables*: las técnicas son informativas y se agrupan en dos categorías: aquellas que son suficientes para satisfacer los criterios de conformidad, y aquellas que son recomendables. Las técnicas recomendables van más allá de los requisitos de cada criterio de conformidad individual y permiten afrontar mejor las pautas. Algunas de las técnicas recomendables tratan sobre barreras de accesibilidad que no han sido cubiertas por los criterios de conformidad verificables.

Todos estos niveles de orientación (principios, pautas, criterios de conformidad y técnicas suficientes y recomendables) actúan en conjunto para proporcionar una orientación sobre cómo crear un contenido más accesible. [4]

5 Desarrollo de ARWeb

La evaluación, revisión o análisis de la accesibilidad web tiene por finalidad analizar, estudiar y validar las páginas web con el objetivo de que las páginas no presenten problemas de accesibilidad y cumplan las pautas y directrices de accesibilidad existente.

La disposición ONTI 2/2014 vigente en Argentina a partir de agosto de 2014 cuyos artículos más relevantes se encuentran en la sección 2.4, establece para las tecnologías HTML y CSS el nivel A como nivel mínimo de conformidad. [10]

Existen herramientas [13] [14] que evalúan el nivel A, sin embargo, se desconocen los algoritmos que realizan la evaluación y la forma en que determinan el puntaje de accesibilidad. La norma argentina vigente establece como puntaje mínimo de aprobación CINCUENTA (50) puntos.

ARWeb fue desarrollada para realizar una evaluación transparente y que sus resultados correspondan con los requerimientos de la normativa argentina.

5.1 Funcionalidades de ARWeb

ARWeb es una herramienta libre y abierta, que cuenta con toda la documentación para que otros desarrolladores puedan continuar extendiendo el framework que se proporciona. La herramienta está disponible para ser usada. [15]

ARWeb incluye funcionalidades que presentan otras herramientas, tales como Test de Accesibilidad Web (TAW) [13] y eXaminator [14]:

- Lectura del código HTML procedente de una URL de un sitio Web, de un archivo e inclusión directa de código.
- Evaluación del código HTML y CSS de la página con el fin de comprobar las pautas de accesibilidad Web citadas por el W3C. [11]
- Reporte de problemas hallados, resaltados en el mismo código verificado.

Por otro lado, se incorporaron otras funcionalidades, que no se encuentran en otros aplicativos disponibles:

- Reporte de problemas simplificado.
- Registro histórico de las verificaciones realizadas.
- Recomendaciones de desarrollo en cada una de las verificaciones.
- Exportación de documentos PDF y planilla XLS.

5.2 Secuencia de Objetos

Para comprender de qué manera interactúan los objetos principales del sistema cada que vez que la herramienta debe realizar una evaluación de accesibilidad web, se presenta en la Figura 1, la secuencia de objetos donde se puede observar los mensajes que se envían entre sí.

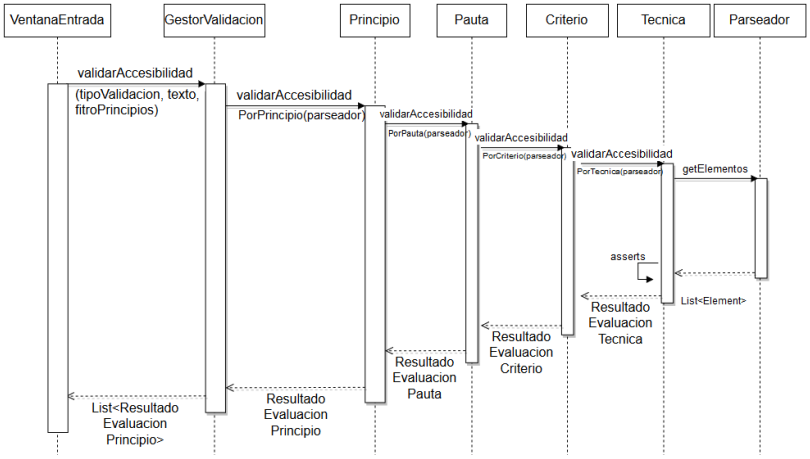


Fig. 1. Diagrama de Secuencia del Sistema ARWeb.

La ventana de entrada (donde se carga el elemento a validar) invoca al método *validarAccesibilidad*, llega al *GestorValidacion* con los parámetros del tipo de validación (página web, archivo o código), el texto asociado al tipo de validación (si es página web, su URL; si es un archivo, su nombre completo con la ruta incluida; y si es código; su código HTML) y los principios a evaluar, una vez que se analizan todos estos parámetros se invoca a la clase Principio mediante el método *validarAccesibilidadPorPrincipio*. Cada Principio invoca a la clase Pauta mediante el

método *validarAccesibilidadPorPauta*. Cada Pauta invoca a la clase Criterio mediante el método *validarAccesibilidadPorCriterio* y por último cada Criterio invoca a la clase *Tecnica* mediante el método *validarAccesibilidadPorTecnica*. Es en este método que se hacen las validaciones de accesibilidad correspondientes y devuelve un objeto de la clase *ResultadoEvaluacionTecnica* que encapsula:

- *Tipología*: si es una imagen, un formulario, una tabla, etc.
- *Nombre de la Técnica*.
- *Resultado de evaluación*: si el resultado fue OK, un ERROR, requiere revisión MANUAL o IMPOSIBLE realizar comprobación automática.
- *Descripción de la Técnica*.
- *Recomendación:* detalles técnicos de cómo resolver el error.
- *Incidencias*: cantidad de errores que se encontraron.

5.3 Interfaces Gráficas y Reportes de ARWeb

ARWeb es una aplicación de escritorio desarrollada con el framework Swing del lenguaje Java. En la Figura 2 puede observarse la ventana principal de la herramienta que presenta cuatro solapas:

- Página Web, donde se solicita la dirección URL del recurso a verificarse.
- Archivo, donde se solicita examinar en el sistema de archivos, un archivo de extensión html a evaluar.
- Código, donde se solicita la edición directa del código HTML.
- URLs., se habilita una lista con todos los links identificados en la dirección URL solicitada en la solapa Página Web.

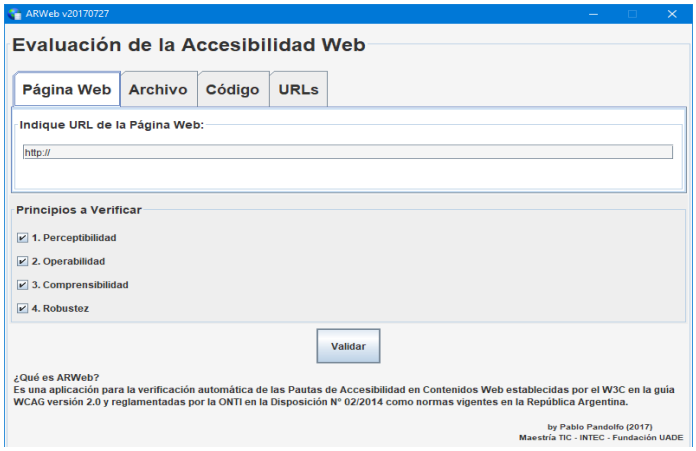


Fig. 2. Ventana principal del Sistema ARWeb.

Al realizar el análisis del recurso introducido, se invoca al controlador GestorValidacion quien devuelve el resultado del análisis. El controlador redirige el resultado hacia la ventana de resultados (ver Figura 3) donde se muestra información

de análisis de accesibilidad web, histórico de verificación del recurso y los resultados generales.

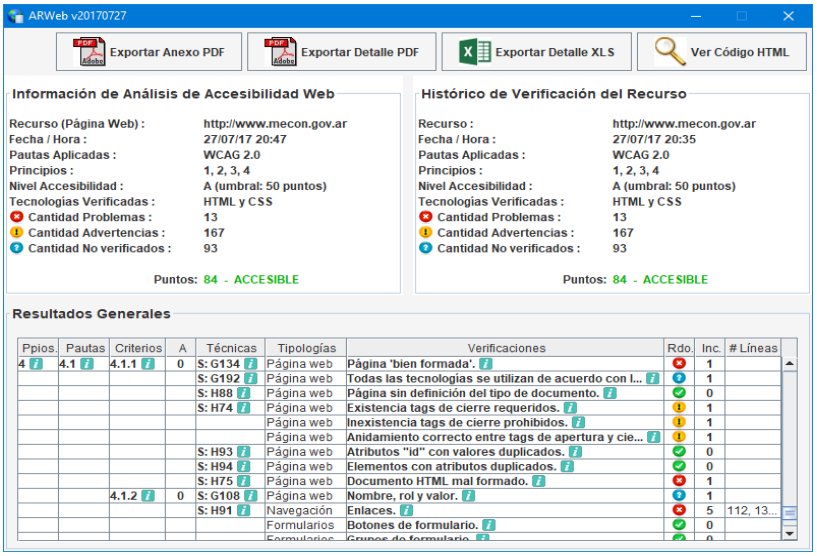


Fig. 3. Ventana de resultados del Sistema ARWeb.

Los Puntos de Accesibilidad se calculan siguiendo la siguiente regla: si todas las verificaciones de todas las técnicas suficientes de un criterio son superadas, entonces suma 4 (cuatro) puntos, caso contrario no suma nada. El usuario que desea verificar qué tan accesible puede llegar a ser una página HTML en particular, puede guiarse con el valor que arroje los Puntos de Accesibilidad. Si el valor es igual o supera los 50 puntos, el recurso es Accesible y el color de texto se cambia a verde, caso contrario, el recurso es No Accesible y el color del texto se cambia a rojo. [10]

Al código HTML del recurso se accede a través del botón “Ver Código HTML”, que muestra al usuario el código fuente analizado destacando las incidencias de Problemas, Advertencias y No verificados por cada línea.

Los reportes que se pueden generar desde ARWeb son:

- *Resultados de Niveles Mínimos de Conformidad en formato PDF* (resultados conforme el modelo del Anexo II de la Disposición 2/2014 de la ONTI).
- *Resultados del Detalle de Análisis en formato PDF* (resultados detallados del análisis de Accesibilidad Web; es decir, en el caso de falla de alguna técnica, muestra el código HTML donde falló y las recomendaciones de accesibilidad para arreglarlo).
- *Resultados del Detalle de Análisis en formato XLS* (el archivo contiene puntajes y fórmulas para que el usuario ajuste el análisis web y la planilla los recalcule automáticamente).

6 Prueba de un Sitio Web

Con el fin de comprobar el funcionamiento de la herramienta con un sitio real, se utilizó la página principal del Sitio Web del Ministerio de Hacienda de la Nación Argentina (www.mecon.gov.ar). La página tiene 612 líneas de código (LDC) y 430 tags HTML (TAGS). La evaluación se realizó durante el mes de agosto del año 2017.

En la Tabla 1 se muestra la cantidad de técnicas y verificaciones analizadas con la herramienta y el resultado correspondiente a cada criterio de conformidad:

Tabla 1. Cantidades evaluadas y Resultados de Accesibilidad Web.

Principios	Pautas	Criterios	Técnicas	Verificaciones	Puntajes	Resultados
1	1.1	1.1.1	27	54	4	OK.
	1.2	1.2.1	4	4	4	OK.
		1.2.2	3	3	4	OK.
		1.2.3	6	7	4	OK.
	1.3	1.3.1	20	46	0	Falló 1 técnica suficiente.
		1.3.2	6	22	4	OK.
		1.3.3	1	1	4	OK.
	1.4	1.4.1	6	11	4	OK.
		1.4.2	3	3	4	OK.
2	2.1	2.1.1	2	7	0	Fallaron 5 técnicas suficientes.
		2.1.2	1	1	4	OK.
	2.2	2.2.1	4	7	4	OK.
		2.2.2	6	7	4	OK.
	2.3	2.3.1	3	3	4	OK.
	2.4	2.4.1	7	17	4	OK.
		2.4.2	3	4	4	OK.
		2.4.3	3	8	4	OK.
		2.4.4	12	20	4	OK.
3	3.1	3.1.1	1	2	4	OK.
	3.2	3.2.1	3	4	4	OK.
		3.2.2	5	5	4	OK.
	3.3	3.3.1	5	5	4	OK
		3.3.2	11	16	4	OK.
4	4.1	4.1.1	7	9	0	Fallaron 2 técnicas suficientes.
		4.1.2	6	16	0	Fallaron 5 técnicas suficientes.
TOTALES	**12**	**25**	**155**	**282**	**84**	

La herramienta ARWeb efectuó 282 verificaciones agrupadas en 155 técnicas correspondientes a los 25 criterios de conformidad de Nivel A. Las pruebas automáticas revisaron tanto los tags HTML como los atributos y valores CSS especificados en las técnicas H y C respectivamente. El tiempo de evaluación no superó los diez segundos.

Los fallos en las verificaciones de las técnicas suficientes producen el fallo del criterio de conformidad y no suma puntos.

El resultado OK significa que todas las verificaciones de todas las técnicas suficientes tuvieron un resultado exitoso; por lo tanto, el criterio de conformidad suma 4 puntos. Para la página web analizada, hubo 21 criterios que resultaron exitosos, obteniendo un puntaje de 84 puntos. Se concluye que la página es ACCESIBLE.

La herramienta ARWeb es utilizada en el análisis de sitios llevados a cabo en los trabajos finales de carrera por estudiantes de grado de Ingeniería en Informática y estudiantes de posgrado de la Maestría Tecnología Informática y de Comunicaciones de la Universidad UADE [16] [17] [18].

7 Conclusiones

El objetivo de la accesibilidad web es que las páginas web sean utilizadas por la mayor cantidad de personas. El interés por el tema generó una serie de proyectos de investigación en el que se evaluaba la accesibilidad web en distintos segmentos del mercado (universidades, bancos, etc.).

Para realizar el análisis de todos los sitios se utilizó como referencia las normas y requisitos de accesibilidad que deben cumplir los sitios web del sector público. Los sitios web del sector público deben respetar las normas y requisitos de accesibilidad recomendados por la ONTI (Oficina Nacional de Tecnología Informática).

La experiencia lograda a lo largo de los distintos proyectos de investigación ponía en evidencia la necesidad de una herramienta que facilite la evaluación de accesibilidad web de acuerdo con la norma vigente en Argentina desde agosto de 2014. Como respuesta a esta necesidad, se desarrolló la herramienta ARWeb, para que sus resultados faciliten la evaluación directa de los que pide la norma argentina.

En la Tabla 2, se compara ARWeb con otras herramientas de revisión de la Accesibilidad Web.

Tabla 2. Cuadro comparativo con otras herramientas de revisión de la Accesibilidad Web.

Caracteristicas	**TAW**	**eXaminator**	**ARWeb**
Técnicas HTML.	Parcial	Parcial	Completa
Opciones.	URL. No permite validación de un sitio completo.	URL, upload de archivo e inclusión directa de código.	URL, upload de archivo e inclusión directa de código.
Selección de Principios de Accesibilidad	No	No	Si
Agrupación de resultados por Principios.	Si	No	Si

Posibilidad de extender herramienta.	No	No	Si
Consejos de desarrollo.	No	No	Si
Indicación del Grado de Accesibilidad.	No	Si, otorga un puntaje de 0 a 10.	Si, en función de la Legislación Argentina.
Indicación del problema en el código fuente.	Si	Si	Si
Visualización en formato PDF y XLS.	No	No	Si
Registro histórico de verificaciones.	No	No	Si

ARWeb contribuye a la detección eficiente de los problemas de accesibilidad para que un sitio web pueda ser evaluado y que recomendaciones son necesarias para que sea Accesible.

Lograr la accesibilidad en páginas web beneficia a todos los usuarios, y logra una mejor aceptación a los sitios web, mejorando el acceso web en general.

Este trabajo puede ser continuado contemplando en su totalidad los Criterios de Conformidad; es decir, los de Nivel AA y AAA e incorporando validaciones de elementos de la tecnología JavaScript dentro de la página web que atenten contra la percepción visual o auditiva del usuario.

Referencias

1. Honorable Congreso de la Nación Argentina. Ley 26.653 de Accesibilidad de la Información en Páginas Web, http://www.infoleg.gob.ar/infolegInternet/anexos/175000-179999/175694/norma.htm
2. Segovia, Claudio. Accesibilidad en Internet. 1a. ed. California: Creative Commons, 212 p. (2006)
3. WCAG. Web Content Accessibility Guidelines 2.0, https://www.w3.org/TR/WCAG20
4. Revilla Muñoz, Olga. WCAG 2.0 de forma sencilla. 1a. ed. Madrid: Itákora Press, 2013. 170 p. ISBN 978-84-614-6410-4.
5. Carreras Montoto, Olga. Usable y Accesible, http://olgacarreras.blogspot.com.ar/
6. Poder Ejecutivo Nacional. Decreto 1172/2003: Acceso a la Información Pública, http://www.infoleg.gov.ar/infolegInternet/anexos/90000-94999/90763/norma.htm
7. Poder Ejecutivo Nacional. Decreto 378/2005: Plan Nacional de Gobierno Electrónico, http://www.infoleg.gov.ar/infolegInternet/anexos/105000-109999/105829/norma.htm
8. Honorable Congreso de la Nación Argentina. Ley 26.653 de Accesibilidad de la Información en Páginas Web, http://www.infoleg.gob.ar/infolegInternet/anexos/175000-179999/175694/norma.htm
9. Poder Ejecutivo Nacional. Decreto 355/2013: Acceso a la Información Pública, http://servicios.infoleg.gob.ar/infolegInternet/anexos/210000-214999/210143/norma.htm
10. Oficina Nacional de Tecnologías de Información. Decreto 2/2014: Norma de Accesibilidad Web 2.0, http://servicios.infoleg.gob.ar/infolegInternet/anexos/230000-234999/233667/norma.htm

11. W3C. World Wide Web Consortium, https://www.w3.org/
12. WAI. Web Accessibility Initiative, https://www.w3.org/WAI/
13. TAW. Servicios de Accesibilidad y Movilidad Web, http://www.tawdis.net/index.html?lang=es
14. Examinator. Servicio en línea para evaluar de modo automático la Accesibilidad de una Página Web, http://examinator.ws/
15. ARWeb. Herramienta de Accesibilidad Web, https://sites.google.com/site/uadeticarweb/arweb-deploy
16. Rossi, Bibiana; Ortiz, Claudia; Chapetto, Viviana. Accesibilidad de la información en Sitios Web argentinos. XVIII Workshop de Investigadores en Ciencias de la Computación, WICC, 2016. Área Ingeniería de Software pp 443-447, 2016. ISBN 978-950-698-377-2.
17. Castro Valeria; Ortiz, Claudia; Chapetto, Viviana; Balleto Carmen; Rossi Bibiana. ¿Las Redes Sociales cumplen con los criterios de Accesibilidad? XIX Workshop de Investigadores en Ciencias de la Computación, WICC, 2017. Área Ingeniería de Software pp 580-584, 2017. ISBN 978-987-42-5143-5.
18. Rossi Bibiana, Ortiz Claudia, Chapetto, Viviana, Passarelli Alberto. Accesibilidad de los Sitios Web de las Universidades Argentinas. Simposio de Informática en el Estado 2017. Proceedi-ngs de las 46° JAIIO, Jornadas Argentinas de Informática

Arquitectura de Software en el Proceso de Desarrollo Ágil. Una perspectiva de Integración de MVC en el Proceso ICONIX

Juan Aranda[1], Mirta Navarro[1], Marcelo Moreno[1], Lorena Parra[1], José Rueda[1], Juan Pantano[1]

[1]Departamento de Informática - F.C.E.F. y N. - U.N.S.J.
Complejo Islas Malvinas. Cereceto y Meglioli. 5400. Rivadavia. San Juan. Argentina

juanaranda@live.com, mirtaenavarro@yahoo.com.ar, mpmoren@gmail.com, lorenaparra152@yahoo.com.ar, josericardorueda@hotmail.com, juancruz871@hotmail.com

Abstract. En Ingeniería de Software existe la inquietud de integrar proceso de desarrollo de software ágil y arquitectura que se ajuste a entornos cambiantes, que cumplan con las necesidades de los usuarios y aseguren un producto de calidad en la construcción de Sistemas de Información (SI).En este escenario, contar con un modelo de la arquitectura del sistema en etapas tempranas se hace necesario, dado que anticiparse a la especificación detallada del sistema permitirá contar con un modelo de alto nivel de una alternativa de solución a los requerimientos planteados, que en sucesivos refinamientos conducirán al producto final. El presente trabajo propone aplicar la arquitectura MVC dentro del ciclo de vida iterativo incremental del proceso ICONIX, a fin de identificarlas actividades propias de la arquitectura de software, en las distintas fases del proceso y analizar cómo MVC favorece el aseguramiento de atributos de calidad esperados para el Sistema de Información.

Keywords: Software Architecture, Agile methodologies, Information Systems.

Introducción

ICONIX se define como un proceso de desarrollo de software práctico. ICONIX está entre la complejidad de Rational Unified Process (RUP) y la simplicidad y pragmatismo de Extreme Programming (XP), sin eliminar las tareas de análisis y diseño que XP no contempla [1].

Modelo-Vista-Controlador (MVC) es un patrón de arquitectura de software que separa los datos, la lógica de negocio y la interfaz de usuario de una aplicación, en componentes distintos [4]. MVC favorece determinados atributos de calidad; incrementa la reutilización, la flexibilidad y la evolución por separado de los diferentes aspectos [5].

En el presente trabajo, se plantean dos líneas principales de abordaje. La primera consiste en identificar las actividades de la arquitectura del software en las diferentes fases del dentro del ciclo del proceso ICONIX. La segunda, se refiere a aplicar el modelo MVC dentro de las fases del proceso de desarrollo iterativo e incremental propuesto por ICONIX, evaluando el impacto de MVC sobre los atributos de calidad.

Contexto

El presente trabajo se encuadra dentro del área de I+D de la Ingeniería de Software (IS) y de los Sistemas de Información (SI). Otorga continuidad a publicaciones anteriores [6],[7],[8] desarrolladas por el equipo de investigación, referidas al análisis, comparación y selección de metodologías ágiles, uno los dos ejes centrales del proyecto de investigación: "Integración de Metodologías Ágiles y Arquitecturas de Software en el Desarrollo de Sistemas de Información", iniciado en enero de 2016, con una duración de dos años y que tiene como unidades ejecutoras al Departamento de Informática de la Facultad de Ciencias Exactas, Físicas y Naturales (FCEFN) de la Universidad Nacional de San Juan (UNSJ).

El grupo de investigación tiene una trayectoria de varios años en diferentes proyectos [10], [12], [14] y anteriores, vinculados a Metodologías de Desarrollo y Sistemas de Información, con numerosas publicaciones en diferentes ámbitos, y con la formación de recursos humanos en el área de interés.

ICONIX

ICONIX es un proceso simplificado, que unifica un conjunto de métodos de orientación a objetos con el objetivo de abarcar todo el ciclo de vida de un proyecto. Sigue un proceso de desarrollo iterativo e incremental, dirigido por casos de uso, compuesto de cuatro fases y adaptado a los patrones de UML. Fue elaborado por Doug Rosenberg y Kendall Scott a partir de una síntesis del proceso unificado de los "tres amigos" Booch, Rumbaugh y Jacobson [2].

La Fig.1 representa la visión del sistema desde dos puntos de vista: vista estática y vista dinámica. Las vistas, también denominadas modelos, pueden ser desarrollados en forma paralela y recursiva. La vista estática, compuesta por el modelo del dominio y el diagrama de clases, modela el funcionamiento del sistema sin interacción con el usuario. La vista dinámica, representa al sistema en interacción con el usuario, a través de solicitudes (acciones) a las que el sistema presenta alguna respuesta en tiempo de ejecución.

El desarrollo de Sistemas de Información en ICONIX, propone un proceso iterativo e incremental de cuatro fases, dirigido por casos de uso. Esta última característica, significa que los casos de uso integran el trabajo de desarrollo durante todo el ciclo de vida del software [3]. Por este motivo, adquiere importancia la elaboración del documento de requisitos, práctica que refuerza la noción fundamental de que un sistema debe satisfacer las necesidades de los usuarios, y no a la inversa.

El manejo de casos de uso produce argumentos concretos específicos y fácilmente comprensibles por los diferentes actores involucrados. La trazabilidad permitirá establecer relaciones bidireccionales entre los requerimientos y las funcionalidades implementadas por medio de los casos de uso.

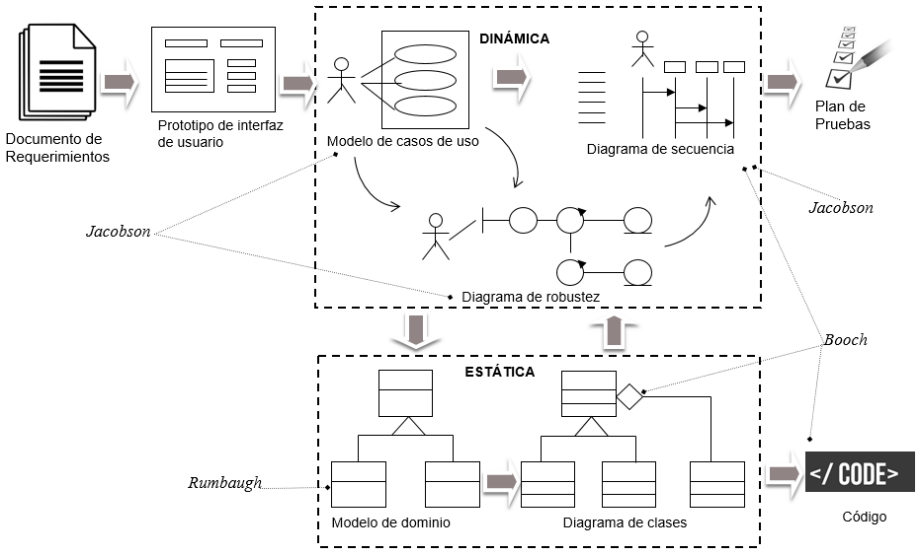


Fig. 4. **Fig. 2.**Vistas en ICONIX[3]

En el proceso iterativo incremental de ICONIX, la vista estática se refina incrementalmente durante las iteraciones sucesivas a través de la vista o modelo dinámico (compuesto de CU, análisis de robustez y diagramas de secuencia). En cada fase, se obtienen artefactos que se van refinando a medida que avanza el proceso de desarrollo. En la Fig. 2 se describen las fases del proceso y los diferentes productos o artefactos resultantes en cada fase (parte inferior).

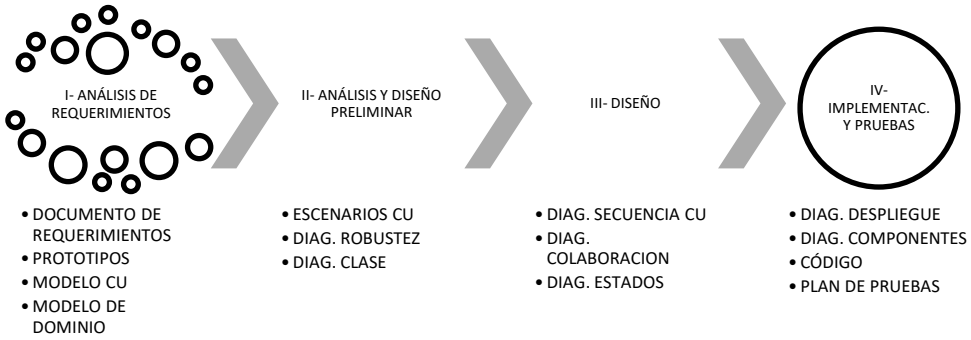


Fig. 5. **Fig. 2.**Fases y Artefactos en ICONIX[4].

[3] Fuente: elaboración propia en base al original [1]
[4]Fuente: Elaboración propia en base al original [1]

MVC

Arquitectura de Software

La Arquitectura de Software es el conjunto de decisiones de estructura de diseño de alto nivel que reúne los requerimientos técnicos y operacionales del sistema. Lo mismos resultan difíciles de cambiar durante el proceso de desarrollo. Esa estructura se compone de elementos de software, las relaciones entre ellos, las propiedades de ambos (elementos y relaciones) y de patrones arquitectónicos que guían a esta estructura. [6].

La arquitectura es importante para satisfacer los requisitos no funcionales, que están relacionados a los atributos de calidad como el rendimiento, seguridad y escalabilidad [9]. Todo sistema posee arquitectura, si no se desarrolla la arquitectura explícitamente, se obtendrá una de todas formas, pero puede no ser la adecuada para favorecer los atributos de calidad deseados. Tener una arquitectura, no es lo mismo a tener una arquitectura cuyo comportamiento es conocido.

La medida en que un sistema alcanza sus requisitos de calidad depende de las decisiones de arquitectura, ya que es un factor crítico para alcanzar los atributos de calidad. El modelado de la arquitectura del sistema, sólo puede permitir, pero no garantizar, que cualquier requisito de calidad se alcance.

El Proceso de Modelado de una Arquitectura de Software, representado gráficamente en la Fig. 3, se compone de tres fases principales:

- Definir los requerimientos: Involucra crear un modelo desde los requerimientos que guiarán el diseño de la arquitectura basado en los atributos de calidad esperados
- Diseño de la Arquitectura: Involucra definir la estructura y las responsabilidades de los componentes que comprenderán la Arquitectura de Software
- Validación: Significa "probar" la arquitectura, típicamente pasando a través del diseño contra los requerimientos actuales y cualquier posible requerimiento a futuro.

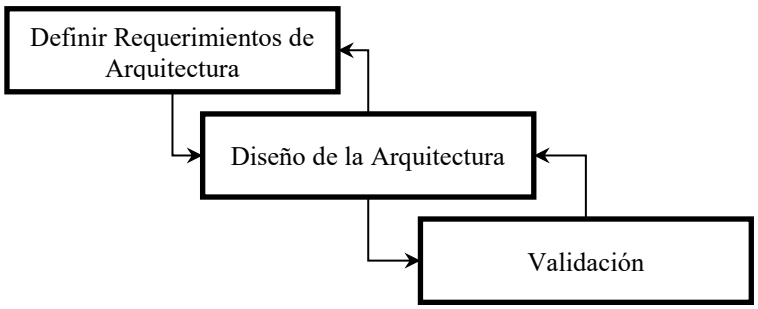


Fig. 3. Proceso de Modelado de una Arquitectura de Software

Modelo Vista Controlador (MVC)

Modelo Vista Controlador (MVC), representado gráficamente en la Fig.5, es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Se trata de un modelo maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo [04].

El Modelo, contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia. Gestiona el acceso a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento. Define las reglas de negocio (la funcionalidad del sistema). Lleva un registro de las vistas y controladores del sistema.

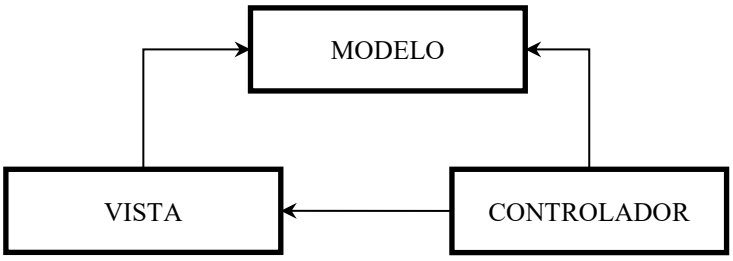


Fig. 6. **Fig. 4.**Modelo Vista Controlador[5]

La vista, o interfaz de usuario, se forma con la información que se envía al cliente y los mecanismos interacción con éste. La vista recibe los datos del modelo y los visualiza al usuario.

El controlador, actúa como intermediario entre el modelo y la vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. Recibe los eventos de entrada y contiene reglas de gestión de eventos, del tipo "SI Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas.

La arquitectura de software en el ciclo de desarrollo ágil de ICONIX

La AS interviene en las decisiones tempranas del desarrollo de un sistema destacando la importancia de sus tareas (esquematizadas en Fig. 3) antes del diseño [07]. Para ubicar estas tareas dentro del proceso de desarrollo de ICONIX, se describen previamente en la **Fig. 5**, las cuatro fases que conforman el proceso de desarrollo y las actividades correspondientes a cada fase.

La concepción de la arquitectura, comienza desde la fase “I-Análisis de Requerimientos”, en la que se identifican los requerimientos, los objetos del dominio del problema y los casos de uso (CU). En esta etapa, el arquitecto de software debe

[5] Fuente: Versión actualizada del original en inglés Trygve Reenskaug (autor MVC) [18]

interpretar, conciliar y proponer alternativas de solución en función del conocimiento de los expertos del dominio y de los planificadores del proyecto [17]. La **Fig. 6**, representa la interacción entre estos actores. El propósito de esta interacción es llegar a establecer una integridad conceptual del modelo a desarrollar expresada mediante la arquitectura.

"La integridad conceptual es la clave para el diseño de sistemas con éxito y tal integridad conceptual puede ser tenida por un pequeño número de mentes llegando juntos a diseñar la arquitectura del sistema" [15]

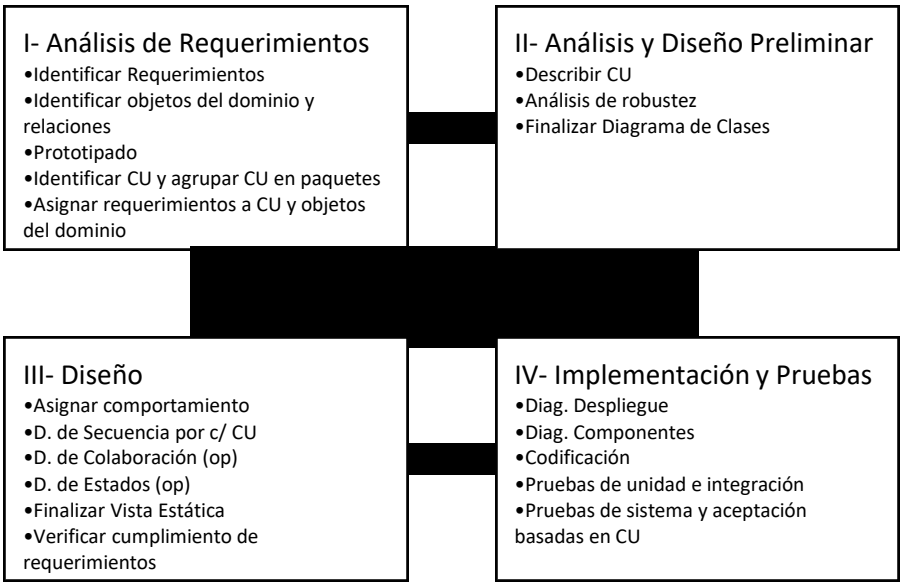


Fig. 7. **Fig. 5.**Fases y actividades por cada iteración ICONIX.

En la fase siguiente "II-Análisis y diseño preliminar", se realiza la descripción de los CU en escenarios, el análisis de robustez y se obtiene el Diagrama de Clases refinado. Respecto de la arquitectura, en esta etapa deben identificarse los requerimientos significativos arquitectónicamente que son aquellos que tienen un impacto esencial sobre la arquitectura [11], esto consolida la integridad conceptual antes mencionada y refina la primera arquitectura que expresa principalmente a elementos genéricos que representan a los requerimientos funcionales.

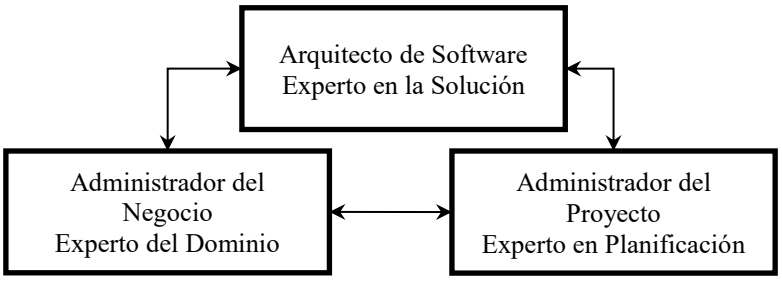


Fig. 8. **Fig. 6.**Interacción de la AS con los actores involucrados en el ciclo de requerimientos[6].

Hasta este momento, se ha identificado un conjunto de objetos que interactúa con otro conjunto de objetos en los casos de uso, es decir las relaciones estáticas, pero no se ha especificado sobre el comportamiento de estos. Esto proceso se denomina modelado de la interacción. Aquí, se visualiza cómo el nuevo sistema realiza un comportamiento útil. Esta es la Fase "III-Diseño", y cuyo hito consiste en verificar cumplimiento de requerimientos. Respecto a la arquitectura, en este paso se toman decisiones arquitectónicas que involucran a los actores del diseño del sistema relacionadas con aspectos de implementación de requisitos [13]. Conforme se avanza por este ciclo los requerimientos se van incorporando en forma iterativa a la arquitectura relacionándose entre sí. Aquí, los requisitos están al mismo nivel que la arquitectura enfatizando la igualdad en importancia entre la arquitectura y los requisitos, en el sentido de que un entendimiento oportuno de los requerimientos y una elección adecuada de la arquitectura son claves para el desarrollo.

Los requisitos que son derivados de las necesidades de los usuarios son conocidos como funcionales o también como características operacionales del sistema, y son útiles para formar los elementos básicos de cada estructura que integra la AS, mientras que los requisitos que capturan la mayoría de las facetas de cómo estos requisitos funcionales deben llevarse a cabo se les conoce como requisitos no funcionales, los cuales pueden ser expresados en términos de requisitos de calidad que tienen su origen en los servicios de calidad [07].

Los requisitos de calidad están conformados por un conjunto de atributos de calidad, los más comunes de estos son escalabilidad, seguridad, desempeño y fiabilidad, estos sirven para dirigir el diseño de una AS, como se muestra en el método de diseño propuesto por Bass [19], llamado "diseño dirigido por atributos", conocido técnicamente por sus siglas en inglés como ADD (Attibute Driven Design). Mediante estos atributos de calidad se llega a refinar las estructuras diseñadas con los requisitos funcionales llegando a producir la especificación de la arquitectura, que constituye el producto más importante generado en este ciclo de desarrollo del software.

Al momento de iniciar la fase "IV – Implementación y Pruebas", ya se ha realizado la revisión crítica del diseño, consistente en el análisis de trazabilidad entre: requerimientos y casos de uso, requerimientos y clases, casos de uso y diagramas de secuencia. En esta instancia, se está en condiciones de realizar las tareas propias de esta fase, consistente en construir los diagramas necesarios de despliegue y componentes,

[6]Fuente: Elaboración propia en base a versión original [15]

proceder a la codificación, elaborar el plan de pruebas de unidad, integración, pruebas de sistema y aceptación basadas en CU.

Esta etapa es la de mayor impacto de la AS, dado que aquí se implementa el diseño de la arquitectura del sistema, los productos de software y se realiza la validación y la verificación del mismo, se ejecutan los planes de pruebas y se realizan las correcciones necesarias.

ICONIX permite mantener constantemente, mediante la matriz de trazabilidad, la relación entre los elementos de grano grueso, que son los elementos arquitectónicos, con los elementos de grano fino que corresponden al diseño para asegurar el cumplimiento de la totalidad de los requerimientos de sistema.

Una Propuesta de Implementación de MVC en el ciclo de ICONIX

Según lo expuesto en las secciones anteriores, el proceso de desarrollo en ICONIX propone un desarrollo iterativo e incremental donde los casos de uso dirigen el proceso de desarrollo en cada iteración.

La alternativa propuesta comienza obteniendo en las etapas iniciales una primera versión de la AS. Esto considerando desde el inicio los requisitos de arquitectura del sistema. Para ello se realiza un análisis profundo, no solamente de los requerimientos funcionales, sino también de los no funcionales, teniendo en cuenta los atributos de calidad deseados.

En cuanto a las funcionalidades del sistema, ICONIX proporciona una herramienta que facilita el agrupamiento de CU en paquetes. Esto se corresponde con una estrategia en AS de separación, que permite reducir la complejidad inherente que el software representa, puesto que al separar los diferentes aspectos que intervienen, se logra una mejor compresión de sus elementos y de las relaciones que se establecen entre ellos [15]. En MVC, esto sucede desde la concepción del patrón, dado que justamente propone la separación en Modelo, Vista y Controlador.

Esta separación, se puede concretar dividiendo al sistema en funciones representadas por módulos que se relacionan entre sí de manera jerárquica o se puede descomponer al sistema en clases, donde cada clase representa una abstracción y estas se relacionan entre sí de diversas formas. El proceso parte del dominio del problema de donde se obtiene la especificación de los requisitos, dicha especificación sirve de entrada para iniciar el proceso de descomposición. Como resultado de este proceso de descomposición se obtienen los primeros elementos arquitectónicos. Los arquitectos de software junto con los expertos en el dominio serán quienes tengan gran responsabilidad en el proceso de obtener esta primera versión de la AS.

A partir de esta primera versión de la AS los responsables de la administración del proyecto identificaran cuales son los requisitos prioritarios para avanzar en el plan.

De acuerdo a las prioridades establecidas para los requisitos, se evalúan las distintas alternativas de solución, teniendo presentes los requerimientos funcionales y no funcionales, las restricciones y los atributos de calidad deseados.

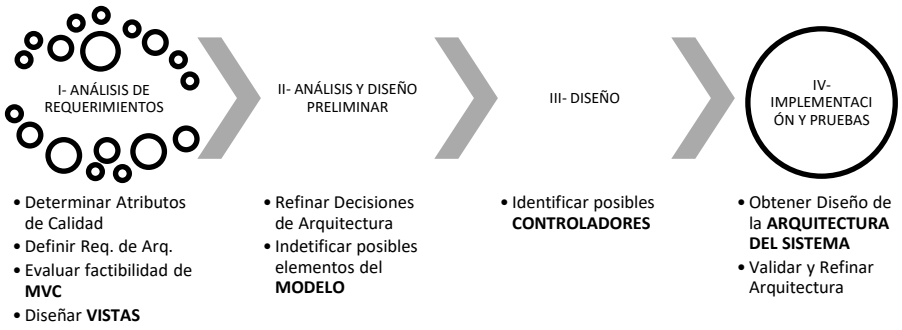


Fig. 9. **Fig. 7.**Alternativa de Implementación de MVC en el ciclo de ICONIX.

El desarrollo de los prototipos de interfaz de usuario en ICONIX, se corresponderá casi en su totalidad las vistas en MVC. Esta tarea se realiza durante la primera fase del proceso ICONIX, lo que facilita la identificación de aspectos relevantes de la AS desde las primeras etapas del desarrollo del sistema.

Además de la identificación de requerimientos y el prototipado, según se expuso en secciones anteriores, durante la primera fase del proceso se desarrollará también un modelo que especifique qué debe realizar la aplicación. Este modelo, libre de detalles de implementación, se completa con los diagramas de casos de uso y el modelo de dominio. Los artefactos resultantes de la primera fase, serán refinados en la fase siguiente obteniendo una descripción más precisa de los casos de uso y el diagrama de clases definitivo. Al finalizar estas actividades, se completa la vista estática que, relacionado con las actividades de la AS, favorece la realización del Modelo en MVC.

En la tercera fase del proceso ICONIX, se modela cómo el nuevo sistema realiza un comportamiento útil, obteniendo los diagramas correspondientes a los casos de uso, los diagramas de colaboración y de estados. Con estos artefactos, el arquitecto de software está habilitado para refinar en las decisiones de arquitectura. El comportamiento del sistema modelado en esta fase, facilitará la visualización del Controlador en MVC.

Las primeras actividades de la fase final del proceso ICONIX, se refieren a la realización del diagrama de despliegue y diagrama de componentes. En estos diagramas es donde se materializa la AS, resultante de las actividades realizadas en las fases anteriores.

Conclusiones

En el presente trabajo, se presentó una propuesta de integración en la que se situaron las tareas específicas de la arquitectura de software, en las distintas fases del proceso de desarrollo ágil ICONIX, aplicando el patrón de arquitectura MVC.

Los beneficios de la integración, resultan de las propias ventajas de utilizar un proceso de desarrollo de software ágil y un patrón de arquitectura conocido, que se ajuste a entornos cambiantes, que cumplan con las necesidades de los usuarios y aseguren un producto de calidad en la construcción de Sistemas de Información (SI).

Cabe destacar que, contar con un modelo de la arquitectura del sistema en etapas tempranas del proceso, anticipándose a la especificación detallada, permitirá contar con un modelo de alto nivel de una alternativa de solución a los requerimientos planteados, que en sucesivos refinamientos conducirán al producto final, favoreciendo el aseguramiento de atributos de calidad esperados, debido a la aplicación del patrón MVC, para el Sistema de Información.

Referencias

01. Rosenberg, D. & Stephens, M. (2007). Use Case Driven Object Modeling with UML: Theory and Practice. Apress. (ISBN 1590597745)
02. Rosenberg, D., Stephens, M. & Collins-Cope, M. (2005). Agile Development with ICONIX Process. Apress. (ISBN 1590594649)
03. Rosenberg D., Scott K. (1999) Use Case Driven Object Modeling with UML: A Practical Approach. Addison-Wesley Professional (ISBN 0201432897)
04. Galloway J., Wilson B. et al. Professional ASP.Net MVC 5. Willey 2014 ISBN: 978-1-118-79475-3
05. Bass, Clements, Kazman. Software Architecture in Practice (2nd ed.). Addison-Wesley 2003.
06. Navarro M., Moreno M., Aranda J., et al. Integración de Metodologías Ágiles y Arquitecturas de Software en el desarrollo de Sistemas de Información. XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016) ISBN: 978-950-698-377-2
07. Navarro M., Moreno M., Aranda J., et al. Integración de Arquitectura de Software en el Ciclo de Vida de las Metodologías Ágiles. Una Perspectiva Basada en Requisitos. XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017). (566-569). ISBN 978-987-42-5143-5
08. Navarro M., Moreno M., Aranda J., et al. Selección de Metodologías Ágiles e Integración de Arquitecturas de Software en el Desarrollo de Sistemas de Información. XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017). (632-636). ISBN 978-987-42-5143-5
09. ISO/IEC/IEEE42010. IEEE Std 1471:2000, "Recommended Practice for Architectural Description of Software Intensive Systems".
10. Navarro M., Moreno M., Aranda J., et al. Proyecto de Investigación "Aplicabilidad de Metodologías y Tecnologías Informáticas en el Desarrollo de Sistemas de Información" Código: 21/E 979 –FCEFN- UNSJ – CICTCA- SIGEVA 2014-2015
11. Martin Fowler. "Patterns of Enterprise Application Architecture" Addison- Wesley. 2003 1st Edition. ISBN-13: 007-6092019909.
12. Navarro M., Moreno M., Aranda J., et al. Convergencia de Tecnologías Informáticas y Metodologías para la Implementación de Sistemas de Información" Cod: 21/E/871 FCEFN. 2012-2013. – CICTCA- SIGEVA.
13. Kendall Scott, D. R. (2001). Applying Use Case Driven Object Modeling with UML: An Annotated e-Commerce Example. Addison Wesley.
14. Navarro M., Moreno M., Aranda J., et al. "Integración de Metodologías Ágiles y Arquitecturas de software en el Desarrollo de Sistemas de Información" Cod: 21/E 1027. FCEFN. et al. 2016-2017
15. Clements, P., F. Bachmann, L. Bass, D. Garlan, J. Ivers, R. Little, R. Nord, J. Stafford. Documenting Software Architectures: Views and Beyond (2nd ed.). Addison-Wesley Professional, 2010.

16. Cordero R., Las vistas arquitectónicas de software y sus correspondencias mediante la gestión de modelos. Tesis Doctoral. Univ. Pol. de Valencia
17. McBride Matthew R. The Software Architect, ACM Communications (2007), V.50, N5.
18. Trygve Reenskaug. Model View Controller http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html (Octubre 2017)
19. Bass L., Kazman R. Architecture-Based Development. Technical Report CMU-SEI-99-TR-007,1999

Desafíos de la Informática Forense en el Análisis de Dispositivos Móviles

Fabián Talio1, Mariela Asensio1, Lucas García Alonso1

1Grupo de Investigación de Informática Forense
Facultad de Ingeniería de la Universidad de Mendoza
Sede San Rafael
{fabian.talio, mariela.asensio, lucas.garcia}@um.edu.ar

Abstract. El creciente auge de dispositivos móviles con Sistema Operativo Android se debe a la multifuncionalidad y portabilidad de estos dispositivos. Enfocándose en el mundo de los negocios, mejorando la productividad en las organizaciones y favoreciendo la conectividad entre personas, estos dispositivos se han convertido en una herramienta poderosa.

Debido a esto, ha aumentado su vulnerabilidad y son el foco de ataques o se encuentran involucrados en investigaciones. Se presenta entonces un nuevo desafió para la Informática Forense, que si bien ya tiene sentadas bases en el análisis sobre PC de escritorio y notebooks, su aplicación sobre dispositivos móviles aún no está tan desarrollada.

Este trabajo muestra pues los avances realizados por el Grupo de Investigación de Informática Forense de la UM (G.I.I.F.) en cuanto a los mecanismos y las herramientas necesarias para la adquisición de evidencia digital.

Keywords: Análisis Forense, Evidencia Digital, Dispositivos Móviles, Android, Técnicas de Adquisición.

1. Contexto

El presente trabajo resulta de los informes de avance elaborados en el marco del Proyecto de Investigación *"Estudio, Evaluación y Clasificación de Herramientas para el Análisis Forense"*, que se desarrolla en la Facultad de Ingeniería de la Universidad de Mendoza en su Sede San Rafael. El objetivo de este proyecto consiste en analizar una serie de herramientas forenses orientadas a dispositivos móviles, basados en Android, existentes en la actualidad, para concluir -a posteriori- con la elaboración de un protocolo adecuado para realizar el análisis forense sobre este tipo de dispositivos.

2. Introducción

Las investigaciones forenses digitales casi siempre implican un teléfono inteligente. Cada vez con mayor frecuencia, el teléfono inteligente es la única forma de evidencia digital en relación con la investigación y es el dispositivo más personal que un individuo posee. La importancia de la evidencia obtenida de teléfonos inteligentes y otros dispositivos móviles se ha convertido en crucial para todo tipo de investigaciones y sobre todo en las delictivas.

La investigación forense en los teléfonos inteligentes es más compleja que "encontrar evidencia" y obtener respuestas. El perito o investigador forense, no puede confiar únicamente en las herramientas de su laboratorio. Es muy importante entender cómo usar las herramientas en forma correcta. Es muy difícil que las herramientas actuales analicen todo desde los teléfonos inteligentes y entiendan cómo se pusieron los datos en el dispositivo. Examinar e interpretar los datos es parte del trabajo a

realizar. Las tecnologías de los teléfonos inteligentes cambian constantemente y la mayoría de los profesionales forenses no están familiarizados con los formatos de datos de cada tecnología.

Es por ello que partimos de la base de realizar un análisis de las herramientas forenses existentes enmarcadas dentro de las GLPI. Si bien no son muchas las existentes en el mercado, y siempre hablando de herramientas de software libre, podemos decir, que la utilidad es muy limitada y a veces compleja al momento de crear imágenes de las particiones de los dispositivos analizados, para a posteriori ser analizadas, no alterando el dispositivo en su forma original.

3. Análisis Forense sobre dispositivos móviles

Como sabemos la Informática Forense es una ciencia que mediante un conjunto de técnicas especializadas ayuda a detectar pistas sobre ataques informáticos y posibilita la recuperación de información perdida o eliminada después que ha ocurrido algún tipo de incidente. Según la Oficina Federal de Investigaciones de Estados Unidos (FBI), la Informática Forense es "la ciencia de adquirir, preservar, obtener y presentar datos que han sido procesados electrónicamente y guardados en un medio informático".

Arellano González y Darahue expresan que "si bien uno de los propósitos de la Informática Forense consiste en determinar los responsables de los delitos informáticos, también permite esclarecer la causa original de un ilícito o evento particular para que no vuelva a repetirse". Es por ello que la Informática Forense ya no sólo se aplica a casos judicializados sino que hoy en día muchas empresas, organismos e incluso particulares solicitan los servicios de un especialista en informática forense.

La evidencia digital es el elemento fundamental para el análisis forense y sus rasgos distintivos la convierten en un constante desafío. Jeimy Cano le otorga las siguientes características: es volátil, es anónima, es duplicable, es alterable y modificable y es eliminable. Esto nos demuestra la importante tarea que tienen por delante los especialistas en el área forense para obtener, custodiar, revisar y presentar la evidencia hallada en una escena del delito; dado que es su responsabilidad la de asegurar la confiabilidad e integridad de los datos recogidos; sin obviar la necesidad de una presentación idónea de los resultados.

Actualmente, y desde hace aproximadamente diez años, el uso de dispositivos móviles se ha incrementado notablemente ya que se han convertido en una herramienta indispensable tanto en lo personal como en el ámbito empresarial. Hoy en día nuestros dispositivos móviles (tablets y celulares) son pequeños ordenadores que llevamos a todos lados, donde no sólo guardamos nuestras fotos personales, listas de contacto y archivos de trabajo sino también el correo, todas nuestras contraseñas y hasta le instalamos aplicaciones para poder interactuar con nuestra cuenta bancaria. Todo esto convierte a los dispositivos móviles en una gran fuente de rastros digitales y si bien la informática forense es una disciplina que se viene aplicando a las PC y a las notebooks su aplicación a los dispositivos móviles aún no se encuentra tan desarrollada.

En el mercado existe gran variedad de gamas de dispositivos móviles, como también diferentes versiones del Sistema Operativo que los componen. Según un informe de la empresa de investigación International Data Corporation (IDC), las ventas de móviles inteligentes seguirían creciendo si bien no al ritmo de años anteriores. Para el quinquenio 2015-2020, establece un aumento en tasa combinada del 5 % y se estima

que el total de móviles inteligentes vendidos a comienzos de la próxima década sería de 1.839 millones de unidades. De acuerdo a las estadísticas, los dispositivos móviles con Sistema Operativo Android encabezan el mercado a nivel mundial, gracias a su uso en organizaciones y al uso de las personas en su vida cotidiana.

En consecuencia uno de los desafíos que enfrenta actualmente la informática forense es realizar análisis forense sobre dispositivos móviles y en el caso de esta investigación específicamente sobre un Smartphone con Sistema Operativo Android.

Si bien el análisis forense se rige por ciertos principios básicos, existen algunas diferencias que es necesario resaltar con respecto al análisis forense en dispositivos móviles: la arquitectura de hardware de una PC o notebook no es la misma que la de un dispositivo móvil, existen determinadas aplicaciones que sólo están disponibles para un único sistema operativos e incluso para diferentes versiones de dicho sistema operativo y por último, dada la diversidad de modelos y tecnología de los dispositivos que impiden el uso de las mismas herramientas forenses.

3.1. Consideraciones Iniciales

Como punto de partida de nuestra investigación, hicimos un estudio exhaustivo del Sistema Operativo Android que nos permitiera conocer sus características, hicimos un repaso por las distintas versiones partiendo desde Apple Pie (1.0) hasta Nougat (7.0), siendo conscientes que a medida que avancemos podrán surgir nuevas versiones que presenten nuevas funcionalidades o modifiquen las existentes, como es el caso de la nueva versión de Android en la que aparecen nuevas medidas de seguridad importantes. Se cambia el cifrado de disco entero, que se implementaba hasta el momento, por el cifrado basado en archivos para proporcionar una mayor protección y mayor aislamiento de los usuarios y perfiles en un dispositivo

Analizamos la arquitectura interna de este Sistema Operativo y las distintas capas que lo conforman haciendo hincapié en el Kernel de Linux. Android depende de Linux para los servicios base del sistema como son seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. También actúa como capa de abstracción entre el hardware y el resto de la pila de software. Además de proporcionar controladores hardware, el kernel se encarga de gestionar los diferentes recursos del teléfono (energía, memoria,…) y del sistema operativo en sí: procesos, elementos de comunicación (networking), etc.

Otro tema inicial de estudio fue el almacenamiento en Android, la memoria de un Smartphone o Tablet, al igual que el disco duro de una computadora, puede dividirse en varias partes denominadas “particiones”, cada una de las cuales tiene un propósito específico. La mayoría de estas particiones se crean en la memoria interna del dispositivo, una memoria de estado sólido (flash), conocida como NAND. Si un dispositivo tiene tarjeta SD para extender su espacio de almacenamiento, la tarjeta en cuestión va a representar otra partición.

Para finalizar esta etapa inicial en el estudio de Android nos enfocamos en el Sistema de Archivos. La mayoría de dispositivos con Android utilizan un sistema de ficheros llamado YAFFS, un desarrollo ligero optimizado para almacenamiento Flash y que ya se usaba en otros dispositivos móviles, pero surge un problema, y es que el sistema YAFFS es un sistema orientado a sistemas con un único hilo de ejecución, lo que supondría la aparición de cuellos de botella en sistemas dual-core. Luego Android comenzó a utilizar el sistema de archivos llamado Ext4.

El Ext4 es el sistema de archivos que implementan la mayoría de distribuciones de Linux, y sobre todo es bastante estable y confiable como la mayoría de los sistemas basados en Linux, con el mínimo riesgo de pérdida de información.

3.2. Elementos a Analizar

Los elementos a analizar dentro del dispositivo son: la tarjeta SIM, la memoria interna del teléfono, la memoria RAM y las unidades de almacenamiento externo (SD).

La información que debemos ser capaces de encontrar es la siguiente: contactos de teléfono y redes sociales, datos de llamadas, SMS, mensajería instantánea, etc., información de geolocalización como GPS y WiFi, documentos electrónicos y Backups del dispositivo.

Para llevar a cabo el análisis forense, existen algunas guías de buenas prácticas que pueden tomarse como referencia como el RFC 3227 "Guía Para Recolectar y Archivar Evidencia" escrito en febrero de 2002 por Dominique Brezinski y Tom Killalea, ingenieros del Network WorkingGroup. Es un documento que provee una guía de alto nivel para recolectar y archivar datos relacionados con intrusiones. Muestra las mejores prácticas para determinar la volatilidad de los datos, decidir qué recolectar, desarrollar la recolección y determinar cómo almacenar y documentar los datos. Otro documento a tener en cuenta es el ISO/IEC 27037:2012 "Tecnología de la información. Técnicas de seguridad. Directrices para la identificación, recogida, adquisición y preservación de evidencias electrónicas". Esta norma pretende orientar a los responsables de la identificación, recolección, adquisición y preservación de potencial evidencia digital, tomando como premisas fundamentales: la relevancia, la confiabilidad y la suficiencia lo que reviste de formalidad cualquier investigación basada en evidencia digital.

3.3. Acceso al dispositivo

Antes de comenzar con el trabajo, el primer paso es preservar las evidencias para lo que es necesario aislar el dispositivo evitando que tenga conectividad. Generalmente se recomienda la utilización de una Jaula de Faraday que anula los efectos de los campos electromagnéticos externos para evitar cualquier tipo de comunicación con el dispositivo desde el exterior. En el caso que el dispositivo se encuentre encendido lo más aconsejable sería ponerlo en modo avión y luego colocarlo en la jaula de Faraday.

Hoy en día la mayoría de los dispositivos poseen un código de bloqueo y es el primer escollo con el que nos encontraremos. Existen tres tipos de bloqueo en un dispositivo Android: el bloqueo por Patrón en el cual el usuario para acceder debe dibujar un patrón en la pantalla, el bloqueo por PIN en el que el usuario debe introducir su número de identificación personal formado por cuatro dígitos y por último el bloqueo con la utilización de Password mediante el cual el usuario debe introducir su contraseña generalmente formada por letras mayúsculas, minúsculas, números y símbolos.

Existen tres métodos para acceder al dispositivo bloqueado que son evadir, deshabilitar y crackear. En todos los casos es necesario que la opción USB Debugging esté activada. El modo de Depuración USB también conocida como Modo de Desarrollador es un método transparente de conectar un dispositivo con Android a una PC con Android SDK lo que permite tener acceso y control del dispositivo posibilitándonos tomar instantáneas o screenshots, ejecutar comandos de terminal con ADB, realizar backup de datos e inclusive rootear un dispositivo.

En el párrafo anterior hacíamos mención a ADB, Android Debug Bridge, la cual es una herramienta que nos permite interactuar con un dispositivo Android físico o virtual desde cualquier PC. Mediante la utilización de comandos podemos ejecutar una consola

Shell dentro del dispositivo lo que nos permite listar los directorios y sus contenidos y copiar o mover archivos.

3.4. Técnicas de adquisición

Existen fundamentalmente dos técnicas de adquisición: lógica y física.

La extracción lógica permite obtener datos del dispositivo mediante el acceso al sistema de archivos con la ayuda del sistema operativo, para ello es necesario la utilización de herramientas que nos permitan comunicarnos con el SO solicitando la información directamente al sistema. De este modo podemos obtener datos como registro de llamadas, mensajes, contactos, imágenes, videos, archivos de audio entre otros. Es importante aclarar que estos datos sólo pueden recuperarse siempre y cuando no hayan sido borrados, la única excepción es la base de datos SQLite. Un método de adquisición lógica consiste en utilizar cualquier *customrecovery*, como por ejemplo CWM. El cual nos permite realizar un backup del dispositivo aunque se encuentre protegido con contraseña, patrón o PIN o incluso no tenga habilitado el USB Debugging. Todos los dispositivos con Android ya sean Smartphones o Tablets, cuentan con un Recovery, conocido como consola de Recuperación. Básicamente es una partición en la memoria interna del dispositivo, y que se puede iniciar a esta partición para acceder a una serie de opciones tales como recuperar el sistema operativo, pero sus opciones son pobres. Es por eso que existen una serie de formatos de recuperación a los que normalmente se les llaman CustomRecovery que sustituyen el sistema de recuperación que viene de fábrica por otro que hace lo mismo pero con más opciones que dan más control del dispositivo. CWM Recovery (ClockworkModRecovery) es uno de los CustomsRecovery más usados en los sistemas de recuperación de Android modificados, ya que está disponible para casi todos los dispositivos Android. El motivo por el cual se tiene acceso a la información en modo recovery a pesar de estar bloqueado, se debe a que Android no llega a arrancar en su modo de funcionamiento normal lo que hace que no se carguen las rutinas de protección. Sin embargo, en el caso que el dispositivo estuviera cifrado, se generaría el backup pero este no contendrá una copia de la partición /data. A partir de la versión 4.0, Android incorpora una utilidad para realizar copias de seguridad del dispositivo e incluso de la tarjeta SD. La principal ventaja es que se puede tener acceso al volcado, aplicaciones y bases de datos que se encuentren en el dispositivo sin necesidad de ser root, y luego mediante una aplicación se puede acceder y analizar el backup.

La extracción física accede directamente al sistema de almacenamiento sin tener en cuenta el sistema de archivos. Permite realizar una copia en bruto del dispositivo lo que no sólo nos da acceso a los datos intactos, sino también a los eliminados u ocultos. Este método nos permite además obtener contraseñas, información de ubicación, correos electrónicos, etc. Uno de los aspectos de la extracción física es el análisis de la memoria RAM conocido también como volcado de la memoria RAM. Existen distintas herramientas que nos permiten realizar este trabajo como es el caso de la aplicación Android Device Monitor incluida dentro de Android SDK. También podemos realizar la adquisición usando adb y el comando dd (dataset definition) que permite copiar y convertir datos de archivos. Para poder trabajar con este comando será necesario tener privilegios root dado que el acceso a determinadas particiones sólo estará disponible si contamos con permisos de superusuario.

Podemos decir que una de las ventajas de la extracción lógica con respecto a la física es que resulta más sencillo y rápido de realizar contando con las herramientas específicas de extracción.

4. Herramientas para el análisis

A la hora de realizar el proceso de extracción existe un número notable de herramientas de análisis forense que se deben tener en consideración. Dependiendo de su funcionamiento se pueden clasificar de acuerdo a diferentes criterios como: complejidad, tiempo de análisis requerido, riesgo de pérdida o destrucción de evidencias, nivel invasivo y nivel de fiabilidad. Evidentemente, nos encontraremos con herramientas comerciales que permiten la extracción de evidencias de un modo más rápido, sin tener que preocuparnos si el dispositivo se encuentra bloqueado.

La gran desventaja que poseen es el alto costo de adquisición. Dado que nuestro proyecto se centra en la utilización de herramientas forenses GLPI, se han analizado algunas distribuciones y herramientas disponibles con el fin de encontrar aquellas que permitan un mejor análisis forense.

4.1 Distribuciones y Herramientas de Software Libre

Las distribuciones analizadas hasta el momento son:

CAINE(*ComputerAidedINvestigativeEnvironment*): Creada como un proyecto forense digital

Los principales objetivos de diseño que CAINE pretende garantizar comprende, un entorno interoperable que apoya al investigador digital durante las cuatro fases de la investigación digital y una Interfaz gráfica con selección de herramientas forenses para varios tipos de dispositivos.

SANS INVESTIGATIVE FORENSIC TOOLKIT (SIFT): Es una aplicación VMware preparada para realizar análisis forense compatible con la mayoría de los formatos. Las características de esta distribución, la cual se basa en Ubuntu LTS 16.04, incluye un conjunto herramientas forenses de software libre como así también soporte de sistema de archivos expandidos. Se destacan operaciones como el montado de imágenes, creación de líneas de tiempo, recopilación de memoria volátil o efímera y el uso de herramientas como Sleuthkit o Autopsy.

DEFT (Digital Evidence & Forensics Toolkit)

Es una personalización de Ubuntu con una colección de programas forenses y documentos creados por un conjunto de personas, equipos y empresas. Cada una de estas obras puede estar bajo una licencia diferente. La distribución DEFT Linux puede realizar análisis forense en dispositivos **Android, iPhone y BlackBerry**, además de poder extraer datos de SQLite. Incluso se puede rastrear la red local y la información que pasa a través de ella.

SANTOKU

Santoku Linux, es un ISO Linux de arranque que se puede ejecutar como Live CD o instalar en una PC. Distribución gratuita y de código abierto con herramientas enfocadas en el trabajo forense de dispositivos móviles. Está disponible como una edición gratuita, es una variante de la distribución MobiSec Ubuntu, con una interfaz gráfica de escritorio Gnome. Se basa en OWASP's MobiSec especializada en pruebas de seguridad, análisis de malware y análisis forenses para teléfonos móviles, válida para dispositivos con Android, BlackBerry, iOS y Windows Phone. También se encuentra

disponible, la versión Santoku Community Edition, un proyecto colaborativo para proveer un entorno Linux preconfigurado con utilidades, drivers y guías para este campo.

Santoku Linux cuenta con buena cantidad de software para llevar a cabo todas estas tareas, por ejemplo las herramientas de desarrollo son Android SDK Manager, Android Studio, Eclipse, FastBoot, SBF Flash, ACML Printer2, Heimdall y Google Play API. Para penetration testing incluyeZenmap, Nmap, Burp Suite, w3af Console/GUI, Ettercap, Zap, y SSLStrip. Las de análisis forense son Android Brute Force Encryption, Scalpel, Yaffey, ExifTool, Sleuth Kit, Iphone Backup Analuzers y AFLogical Open Source Edition. Las herramientas para ingeniería inversa son Jasmin, AndroGuard, radare2, Bulb Security SPF, JD-UI, Smali, Drozer, Baksmali, APKTool, Procyon, dex2jar y AntiVL.

HUEMUL: - Basada en Debian

Una característica de esta distribución es que está separado su menú en Etapas, en donde aparecen la Etapa de Adquisición, Clonación y Análisis. También cuenta con la opción de Borrado seguro, Celulares y Cifrado. Huemul es una distribución basada en Debian y concebida dentro del proyecto Centrux. Esto le brinda un mayor grado de confiabilidad y asegura su continuidad a lo largo del tiempo. Cuenta con un conjunto de herramientas sumamente útiles a la hora de realizar una pericia informática, desarrollada en el marco del Proyecto Centrux con el apoyo y respaldo académico de la Universidad Tecnológica Nacional Facultad Regional Avellaneda.

MOBILEDIT FORENSIC

Ofrece la posibilidad de investigar a fondo el contenido de un teléfono móvil, desde la versión del software, hasta las notas o mensajes SMS archivados. Todas las operaciones se realizan desde una interfaz similar a la de cualquier gestor de dispositivos móviles, que nos muestra toda la información detallada, a la que se puede acceder a través de una serie de cómodos menús

Dentro de todas estas distribuciones encontramos herramientas como:

ADB.

Las siglas ADB significan Android Debug Bridge y se corresponden con una herramienta que nos permite controlar el estado de nuestro smartphone Android. Podemos actualizar el sistema, ejecutar comandos *shell*, administrar el direccionamiento de puertos o copiar archivos.

Básicamente, en ADB encontramos una colección de herramientas y comandos útiles que nos ayudarán a comunicar nuestro dispositivo directamente con una computadora para, entre otros, acceder al modo recovery. Por supuesto, para que esto sea posible necesitamos un cable y conectar el smartphone vía USB con una estación de trabajo.

BITPIM.

BitPim es un programa que permite ver y manipular datos en muchos teléfonos del tipo CDMA de LG, Samsung, Sanyo y otros fabricantes. Esto incluye el PhoneBook, Calendar, WallPapers, RingTones (la funcionalidad varía según el teléfono) y el sistema de archivos para la mayoría de los teléfonos con chipset Qualcomm CDMA.

4.1.1 Las principales Ventajas del uso de estas Herramientas

Al ser de Software libre permite la modificación de las distribuciones acorde a necesidades puntuales, sin costo de licenciamiento y con mayor seguridad en las aplicaciones y con una completa libertad en el manejo del Sistema Operativo.

4. 1.2 Las principales Desventajas

Las mismas herramientas van quedando obsoletas ante los cambios de las versiones de los sistemas operativos de Android, que a su vez, con los nuevos requerimientos de seguridad, en cada una de las versiones del sistema operativo, hace obsolescencia prematura de las mismas. A su vez, el cifrado de las aplicaciones más utilizadas, como ser Whatsapp y Telegram entre otras, complican el trabajo forense de los dispositivos. Otra de las desventajas, todos los dispositivos Android de los principales fabricantes, se distribuyen sin el usuario root habilitado y sin la posibilidad de acceder al sistema de archivos que contiene el sistema operativo. La escasez de documentación influye notablemente a la hora de realizar pericias apremiadas por el tiempo.

5. Algunas dificultades

La primera dificultad importante detectada, ha sido el hecho de conseguir acceso al móvil sin conocer la contraseña o patrón de desbloqueo del mismo. Se han dedicado muchas horas de búsqueda y prueba de diferentes métodos, y se ha podido constatar que con versiones de Android 6.0 y superiores no existe forma, al menos gratuita, de conseguirlo hasta el momento.

Otro aspecto relevante, es la necesidad de contar con acceso root para poder realizar algunas técnicas y sobre todo para poder acceder a la información importante. Además se ha comprobado que para poder garantizar el acceso al dispositivo se debe contar con la USB Debugging activada. Muchas veces el uso de algunas técnicas implica la alteración del dispositivo lo que vulnera el principio de la ciencia forense.

5. Conclusiones

Como se ha expresado anteriormente, este proyecto de investigación aún se encuentra en fase de desarrollo por lo que a pesar de las dificultades encontradas se sigue trabajando en el análisis y depuración de las herramientas necesarias para el análisis forense en dispositivos móviles.

Existe una realidad que es que tanto las medidas de seguridad propias de Android como las propietarias de los diferentes fabricantes de teléfonos móviles, que se han tomado en los últimos tiempos para evitar que los usuarios seamos víctimas de los nuevos ataques, son de gran ayuda para proporcionar un nivel de seguridad mucho mayor. Sin embargo, esto conlleva un incremento en las dificultades que debe afrontar la investigación digital forense de allí los futuros desafíos serán la búsqueda de soluciones.

6. Referencias

1. *Android*. (s.f.). Obtenido de https://www.android.com/

2. Cano, J. J. (2009). *Computación Forense. Descubriendo los Rastros Informáticos.* México: Alfaomega.

3. *Deft*. (s.f.). Obtenido de http://www.deftlinux.net/

4. Heather Mahalik, R. T. (2016). *Practical Mobile Forensics.* Birmingham: Packt Publishing Ltd.

5. *International Organization for Standardization*. (s.f.). Obtenido de https://www.iso.org/standard/44381.html

6. Darahue M. E. y Arellano González, L. E. (2011). *Manual de Informática Forense.* Buenos Aires: ERREPAR S.A.

7. Darahue M. E. y Arellano González, L. E. (2015). *Manual de Informática Forense II.* Buenos Aires: ERREPAR S.A.

8. Darahue M. E. y Arellano González, L. E. (2016). *Manual de Informática Forense III.* Buenos Aires: ERREPAR S.A.

9. *RequestforComments: 3227*. (s.f.). Obtenido de http://www.ietf.org/rfc/rfc3227.txt

10. Tahiri, S. (2016). *Mastering Mobile Forensics.* Birmingham: Packt Publishing Ltd

Una propuesta para manejar datos masivos en bases de datos NewSQL

Pablo Gomez, Maria Murazzo, Nelson Rodriguez

[1]Universidad Nacional de San Juan, Facultad de Ciencias Exactas, Físicas y Naturales, Departamento de Informática
Complejo Universitario Islas Malvinas (CUIM), Rivadavia, San Juan, Argentina
{pablo.gomez.allende@gmail.com, marite@unsj-cuim.edu.ar, nelson@iinfo.unsj.edu.ar}

Abstract. La generación de grandes cantidades de datos en conjunción con las soluciones basadas en Cloud permiten el acceso a importantes recursos computacionales a menores costos. En este contexto las bases de datos NewSQL aparecen como una solución que combina la capacidad de almacenamiento de las bases de datos NoSQL con las garantías y facilidades provistas por las bases de datos SQL. Google Spanner se presenta como un representante del NewSQL inserto en una potente plataforma de cloud, lo que permite brindar a sus clientes un servicio de almacenamiento y procesamiento de datos eficiente, eficaz, confiable y con alta disponibilidad.

Keywords: Big Data, Cloud Computing, NewSQL, Google Spanner, GCP, Google Cloud Platform

1 Introducción

Debido a la aparición de nuevas tecnologías, dispositivos, medios de comunicación y aplicaciones, convergiendo hacia IoT (Internet of Thing); la cantidad de datos que se produce aumenta exponencialmente. Este aumento en la cantidad de datos demanda nuevas estrategias que permitan su almacenamiento, procesamiento y análisis de manera eficiente; esto conlleva un cambio de paradigma en las arquitecturas de cómputo, los algoritmos y los mecanismos de procesamiento.

Frente a esta problemática se ha popularizado el término Big Data [1], el cual se usa para describir grandes conjuntos de datos, los cuales exhiben las propiedades de: variedad, volumen, velocidad, variabilidad, valor y complejidad; todas ellas denotan a datos multidimensionales. Estas propiedades hacen que los sistemas de cómputo convencionales sean muchas veces inapropiados para lograr un procesamiento adecuado [2].

Para realizar operaciones de cómputo intensivo y mejorar la velocidad de procesamiento, los entornos de computación distribuida [3] son los ideales para big data debido a la creciente demanda y complejidad computacional involucrados. Esto se debe a que este tipo de entornos de computación brindan un modelo destinado a resolver

problemas de cómputo masivo utilizando un gran número de computadoras organizadas sobre una infraestructura de comunicaciones distribuida. De esta manera es posible compartir recursos heterogéneos, basados en distintas plataformas, arquitecturas y lenguajes de programación, situados en distintos lugares y pertenecientes a diferentes dominios de administración sobre una red que utiliza estándares abiertos.

Ejemplos de este tipo de arquitecturas son los cluster y el cloud [4], los cuales proveen una infraestructura que favorece de manera eficiente y escalable el procesamiento sobre grandes cantidades de datos. El gran problema de los clusters radica en el alto costo de adquisición y mantenimiento de los mismos.

Otra solución, es usar como plataforma el Cloud Computing [5], entendiendo como tal a un modelo que permite el acceso a un pool de recursos computacionales configurable, donde los usuarios podrán acceder a dichos recursos a demanda, debido a que el cloud ofrece elasticidad en sus servicios, con lo cual desde el punto de vista de los usuarios los servicios pueden crecer o disminuir de manera sencilla y con mínimo esfuerzo. El cloud ofrece varios modelos de servicio tales como:

- *Software as a Service (SaaS) o Software como Servicio*: consiste en permitir al cliente el uso de aplicaciones corriendo en una estructura cloud del proveedor.
- *Platform as a Service (PaaS) o Plataforma como Servicio*: es la capacidad otorgada al cliente de poder desplegar en la plataforma del proveedor diferentes aplicaciones construidas utilizando lenguajes, librerías, servicios y herramientas soportados por dicha plataforma.
- *Infrastructure as a Service (IaaS) o Infraestructura como Servicio*: en este caso se le provee al cliente una infraestructura que le provee capacidad de procesamiento, almacenamiento y otros recursos computacionales fundamentales que le permiten desplegar y correr software arbitrario.

También es posible contar con los beneficios del Big Data en el cloud, esto es lo que se llama *Big Data as a Service* o *BDaaS [6]*. El *BDaaS* permite contar con servicios relativos al Big Data que permiten mejorar el procesamiento y análisis de datos al mismo tiempo que se reducen los costos de dicha actividad. Típicamente el *BDaaS* consta de tres capas:

- *Big Data Infrastucture as-a-Service o Infraestructura de Big Data como Servicio:* permite el almacenamiento y procesamiento de grandes volúmenes de datos a través de *Computing-as-a-Service* y *Storage-as-a-Service*.
- *Big Data Platform as-a-Service o Plataforma de Big Data como Servicio:* permite a los usuarios acceder, analizar y construir aplicaciones analíticas sobre grandes conjuntos de datos. Incluye diferentes formas de almacenamiento y manejo de datos, tales como: *Cloud Storage, Data-as-a-Service* y *Database-as-a-Service.*
- *Big Data Analytics Software-as-a-Service o Software Analítico de Big Data como Servicio:* permite explotar cantidades masivas de datos estructurados y

no estructurados para entregar a los usuarios resultados inteligentes en tiempo real. Utiliza interfaces Web, Hadoop y NoSQL.

En este caso nos interesa trabajar con *Big Data as a Service* a través de *Google Cloud Platform -GCP* (cloud.google.com), más particularmente nos centraremos en una herramienta perteneciente a los servicios de almacenamiento o *Storage Services* de GCP llamada Google Spanner, la cual es una base de datos NewSQL.

Este trabajo se organizará de la siguiente manera: en la siguiente sección se introducirá brevemente a las bases de datos NewSQL, distinguiéndolas de las soluciones NoSQL. En la Sección 3 se hará referencia a Google Spanner. En la sección final se abordarán las conclusiones.

2- NoSQL vs NewSQL

Las soluciones de tipo *NoSQL [7]* consisten en bases de datos no relacionales. Sin embargo el término NoSQL, también referido como "No solo SQL" agrupa un conjunto de soluciones con características muy diferentes. De acuerdo con [8] algunos atributos propios de estas soluciones son: modelos de datos flexibles y no relacionales, capacidad de escalar horizontalmente, alta disponibilidad, no soportan transacciones de tipo *ACID* (generalmente los sistemas NoSQL solamente consideran una consistencia eventual).

NewSQL [9] hace referencia a soluciones que proveen almacenamiento de datos y a la vez tienen como objetivo principal lograr que el modelo de base de datos relacional pueda incorporar escalabilidad horizontal y tolerancia a fallos de manera similar a como lo hacen las soluciones NoSQL.

Algunas de las soluciones más representativas de las bases de datos NewSQL son *Google Spanner [10]*, *VoltDB [11]*, *Clustrix* [12] y *NuoDB [13]*.

Una de las características fundamentales de las soluciones NewSQL es el uso del lenguaje SQL para realizar consultas, aunque la implementación del lenguaje para cada solución posee distintas restricciones.

El uso de este tipo de base de datos se considera apropiado en casos donde se ha utilizado anteriormente un sistema de base de datos relacional tradicional, pero existe la necesidad de escalar el sistema y mejorar la performance. Estas soluciones también son apropiadas para sistemas que poseen fuertes requisitos de consistencia, o en aplicaciones donde la estructura de los datos es conocida y es poco probable que la misma sufra cambios. Una ventaja de estas soluciones es que permiten reducir costos en entrenamiento de personal dado que la muchas de las herramientas propias de los sistemas de base de datos relacionales son compatibles con las bases de datos NewSQL y utilizan el lenguaje SQL.

2.1 Aspectos importantes a comparar

Ciertos atributos son importantes para estos dos tipos de soluciones, tales como:

- *Los modelos de datos*. Donde las soluciones NoSQL utilizan modelos *clave-valor*, *column-family*, *document stores* o *bases de datos de grafos*. En cambio las soluciones NewSQL se basan mayormente en el modelo relacional, y en un modelo semi-relacional en el caso específico de Google Spanner.
- *Las consultas*, las bases de datos NoSQL utilizan mayormente *MapReduce* y lenguajes similares a SQL. Las bases de datos NewSQL utilizan el lenguaje SQL para realizar consultas, aunque como ya se mencionó anteriormente, cada solución implementa el lenguaje de distinta manera y con limitaciones propias.
- En ambas soluciones es importante la capacidad de *escalar horizontalmente*. En general hay tres dimensiones que se buscan escalar: las peticiones de lectura, las peticiones de escritura y el almacenamiento de datos.
- En las soluciones NoSQL y NewSQL es también importante el *particionado* de los datos, es decir, almacenar distintas particiones o fragmentos del conjunto de datos en distintos servidores.
- Otro atributo importante es la *replicación o copia de los datos*, donde generalmente se utilizan dos esquemas: replicación maestro-esclavo o replicación multimaestro. En el caso de las bases de datos NewSQL se considera que utilizan replicación multimaestro o sin maestro, debido a que cualquier nodo puede recibir actualizaciones de datos.
- El *atributo de consistencia*, es decir, asegurar que una transacción lleve a la base de datos de un estado consistente a otro estado consistente, se trabaja de distintas maneras en las soluciones NoSQL y NewSQL. En general, las soluciones NoSQL utilizan una consistencia eventual. Mientras que las soluciones NewSQL aseguran una consistencia fuerte o inmediata.
- El *control de concurrencia* es importante en ambas soluciones, ya que trabajan con un gran número de usuarios concurrentes y gran volumen de operaciones de lectura y escritura.

3 GCP y Google Spanner

Google Cloud Platform es un conjunto de *recursos físicos* tales como discos duros y computadoras, y también de *recursos virtuales*, tales como máquinas virtuales, contenidos en los datacenters de Google alrededor del mundo. Esta distribución geográfica ofrece ciertos beneficios tales como redundancia de datos en caso de fallos y latencia reducida debido a la proximidad de los datacenters a los clientes. En la figura 1 se puede ver la pantalla inicial de la consola de GCP.

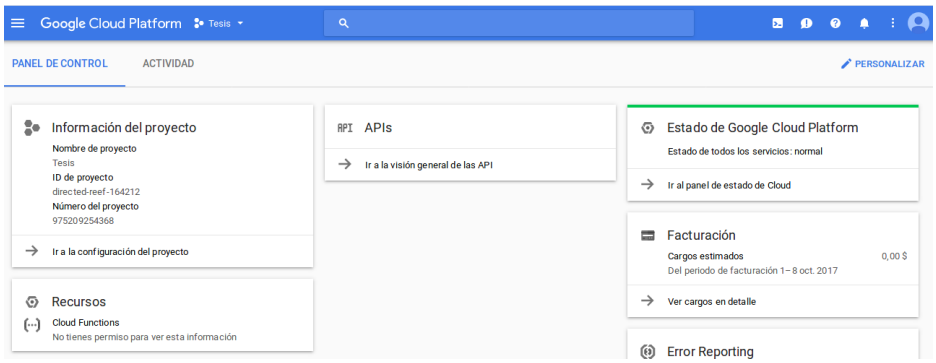


Figura 1. Consola de Google Cloud Platform

Spanner es una base de datos escalable, globalmente distribuida, diseñada y construida por Google. En el más alto nivel de abstracción "es una base de datos que distribuye o fragmenta datos entre varios conjuntos de máquinas de estado de Paxos, en centros de datos ubicados en distintos lugares del mundo" [10].

Spanner utiliza la duplicación de datos para lograr una disponibilidad global de manera tal que los clientes automáticamente cambian la réplica que están utilizando en caso de que la misma falle. Un aspecto importante a destacar es que Spanner automáticamente redistribuye los datos en distintas máquinas mientras la cantidad de datos y el número de servidores cambian, y migra los datos entre distintas máquinas (e incluso entre distintos centros de datos o datacenters) para lograr un balanceo de carga y responder mejor a las fallas.

El foco de Spanner es el manejo de datos replicados en distintos datacenters. Los datos se almacenan en tablas semi-relacionales; los mismos son versionados y a cada versión se le asigna una marca de tiempo (timestamp) con su tiempo de commit.

Al tratarse de una base de datos NewSQL, Spanner soporta transacciones de propósito general y provee un lenguaje de consulta basado en SQL.

3.1 Implementación de Spanner

Spanner está organizado como un conjunto de zonas, las cuales se consideran una unidad administrativa o un conjunto de ubicaciones a través de las cuales los datos se pueden replicar. Las zonas se pueden agregar o quitar de un sistema que está en funcionamiento a medida que nuevos datacenters entran en servicio y otros se cierran. Las zonas también son la unidad de aislamiento físico: pueden existir una o más zonas en un centro de datos.

Una zona tiene un "maestro" (zonemaster) y entre cien y miles de servidores de Spanner (spanservers). Los zonemasters asignan datos a los spanservers, y estos últimos entregan los datos a los clientes. Existen proxies ubicados por zona que son utilizados por los clientes para localizar los spanservers designados para entregarles los datos.

El driver de ubicación maneja el movimiento de datos a través de las zonas en cuestión de minutos. El mismo se comunica periódicamente con los spanservers para encontrar datos que necesitan ser movidos, ya sea para satisfacer restricciones actualizadas de replicación o para balanceo de carga.

Cada spanserver es responsable de entre cien y mil instancias de una estructura de datos llamada "tableta" (tablet). Cada tablet se almacena en un sistema de archivos distribuido denominado Colossus.

Para poder comenzar a trabajar con Spanner necesitaremos crear una instancia, la cual funciona como un contenedor para las bases de datos que vayamos a crear. La creación de una instancia es muy sencilla y puede realizarse a través del entorno gráfico que ofrece GCP tal como lo muestra la figura 2. También es posible crear una instancia utilizando la consola de Linux a través de una herramienta de línea de comandos llamada gcloud. Si bien mostraremos algunos de sus comandos, la instalación y configuración de la herramienta gcloud no es relevante a los fines del presente trabajo. El comando necesario para crear la instancia creada en la figura 2 es el siguiente:

```
gcloud spanner instances create instancia-suac --config=regional-us-central1 --description="Instancia SUAC" --nodes=1
```

Como se puede ver en ambos casos es necesario especificar el nombre de la instancia, la región y la cantidad de nodos. La figura 3 muestra la instancia de Spanner ya creada.

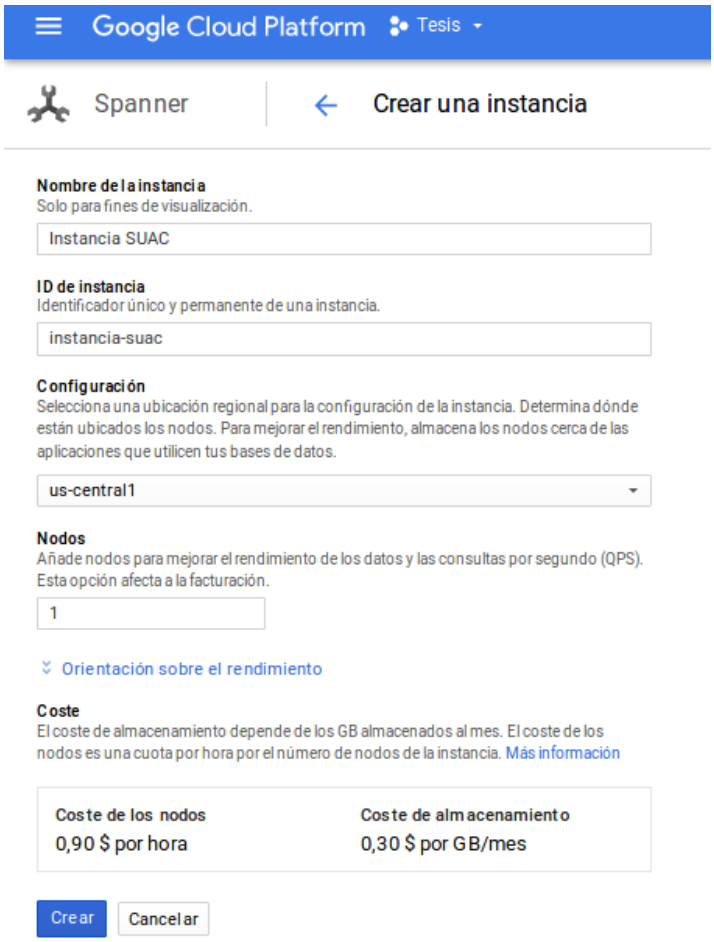


Figura 2. Creación de una instancia en Spanner

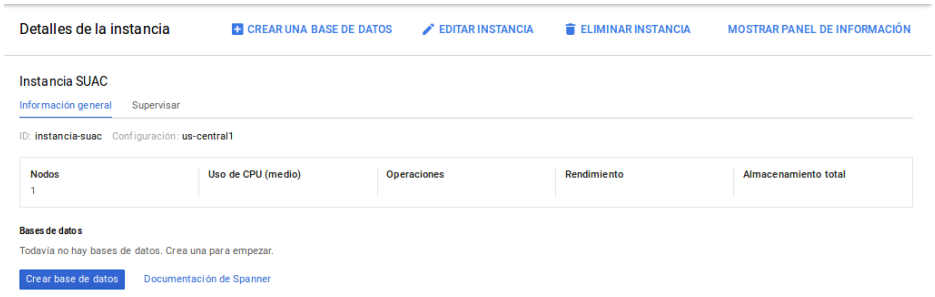


Figura 3. Instancia de Spanner creada

3.2 Caso práctico

Ante la necesidad de trabajar con grandes cantidades de datos, se recurre a datasets públicos, específicamente en este caso se utilizan los datasets provistos por el gobierno de la Ciudad Autónoma de Buenos Aires [14], dado que algunos de ellos presentan gran cantidad de datos. El dataset utilizado presenta datos del Sistema Único de Atención Ciudadana, el cual se encarga de atender las necesidades y reclamos de los vecinos de la CABA.

La documentación de Spanner [15] muestra cómo crear una nueva base de datos dentro de una instancia, con lo cual será también posible crear tablas para insertar datos y realizar consultas.

En la figura 4 podemos ver la base de datos creada y también una tabla propia.

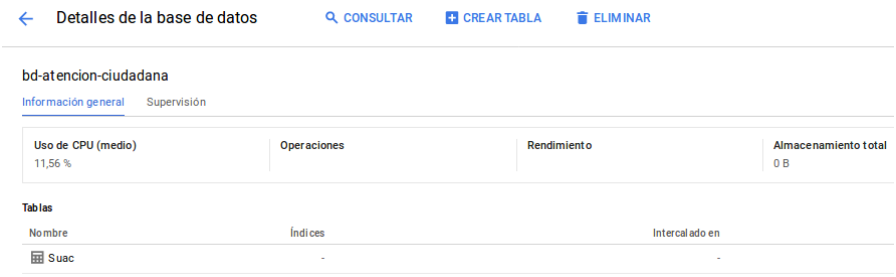


Figura 4. Base de datos “bd-atencion-ciudadana” con su tabla “Suac”

En la figura 5 podemos observar los detalles de la tabla “Suac” con sus nombres de columnas y tipos de datos. Spanner permite modificar las tablas desde el entorno gráfico de GCP.

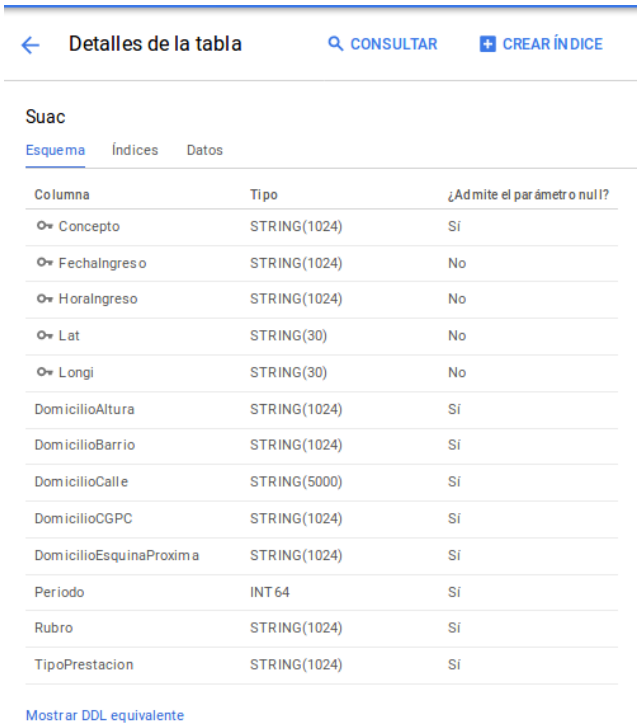


Figura 5. Detalles de la tabla "Suac"

Spanner también provee una API para crear clientes en los lenguajes Python, REST, GO, Ruby, PHP, Java, C# y Node.js. A través de un cliente hecho en cualquiera de estos lenguajes, podemos realizar también la creación y manipulación de bases de datos y tablas en Spanner. En el caso de la creación de la tabla mostrada en la figura 5, el código en python sería similar al siguiente:

```
def create_table(instance_id):
spanner_client = spanner.Client()
instance = spanner_client.instance(instance_id)
database_id = "bd-atencion-ciudadana"
database = instance.database(database_id)

operation = database.update_ddl([
"""CREATE TABLE Suac (
FechaIngreso STRING(1024) NOT NULL,
HoraIngreso STRING(1024) NOT NULL,
Longi STRING(30) NOT NULL,
Lat STRING(30) NOT NULL,
Concepto STRING(1024),
DomicilioAltura STRING(1024),
DomicilioBarrio STRING(1024),
DomicilioCalle STRING(5000),
DomicilioCGPC STRING(1024),
DomicilioEsquinaProxima STRING(1024),
Periodo INT64,
Rubro STRING(1024),
```

```
TipoPrestacion STRING(1024)
) PRIMARY KEY (FechaIngreso, HoraIngreso, Concepto, Longi, Lat)"""
])

print('Waiting for operation to complete...')
operation.result()

print('Tabla Creada')
```

Hasta el momento no se ha encontrado una forma transparente de insertar datos en las tablas creadas en Spanner, debido a ello se recurre a un cliente hecho en Python que toma los datos de un archivo y los inserta en la tabla. Una vez que los datos fueron insertados podemos realizar consultas a las tablas utilizando lenguaje SQL. Para ello la consola de Spanner provee una herramienta, que permite realizar consultas. Sin embargo la herramienta no permite insertar ni eliminar datos. En la figura 6 podemos ver una consulta a la tabla "Suac", que consiste en seleccionar los distintos tipos de prestación que se brindan a los vecinos de la CABA. Es posible ver cómo además del resultado de la consulta, la herramienta muestra el tiempo que le llevó a la misma ejecutarse, y si fuera necesario podemos acceder a una explicación del funcionamiento de la consulta realizada entrando en la pestaña "Explicación", tal como se muestra en la figura 7.

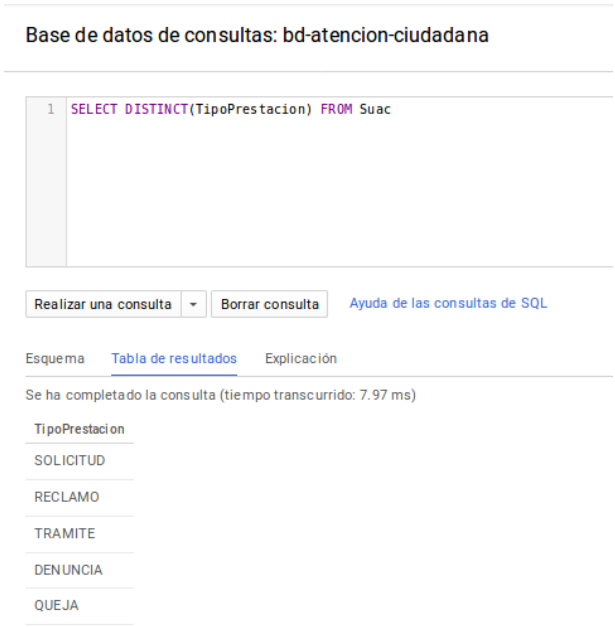


Figura 6. Consultando los tipos de prestación almacenados en la tabla

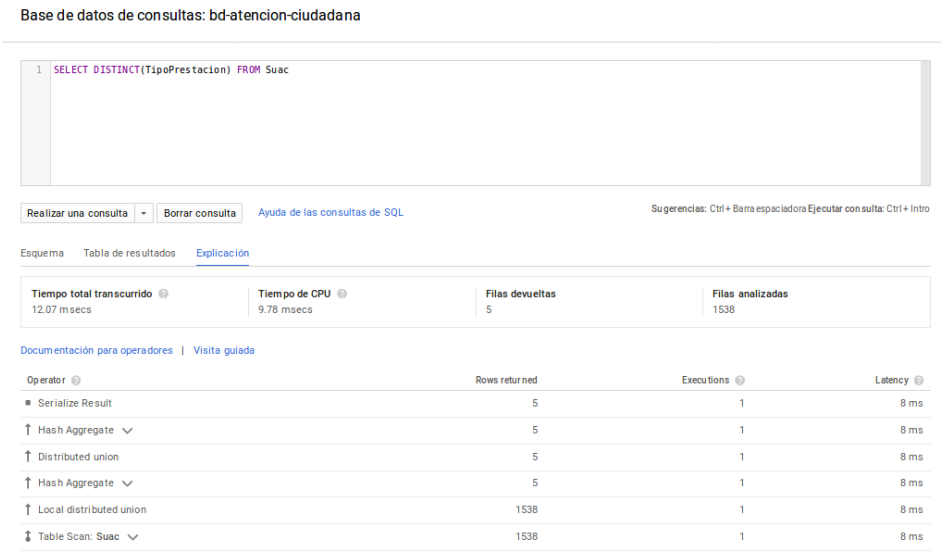


Figura 7. Explicación de la consulta realizada

4 Conclusiones

Las cantidades masivas de datos que se generan y almacenan actualmente acentúan la necesidad de contar con herramientas que permitan procesarlas y aprovechar su utilidad de manera eficiente. El cloud surge como una respuesta a esta necesidad incorporando también herramientas potentes tales como las bases de datos NewSQL. Dentro de esta categoría Spanner surge como parte de una plataforma completa y poderosa que brinda enormes posibilidades a sus usuarios permitiéndoles la creación de distintas aplicaciones, el almacenamiento y el procesamiento de grandes volúmenes de datos.

Por el momento la investigación se ha centrado en el manejo de la consola web de Spanner, logrando la creación de bases de datos y su posterior consulta. También se ha logrado la inserción de datos a través de un cliente construido en lenguaje Python, así como la manipulación de las bases de datos a través de este mismo cliente que está en una fase muy temprana de desarrollo.

A futuro se plantea como próximos objetivos investigar una forma más transparente para la inserción de los datos en Spanner y la comparación del rendimiento de Spanner contra soluciones de tipo SQL y NoSQL.

5 Bibliografía

[1] Y. Zhai, Y.-S. Ong, and I. W. Tsang, “The Emerging ‘Big Dimensionality,’” 2014.

[2] C. Kacfah Emani, N. Cullot, and C. Nicolle, "Understandable Big Data: A survey," *Comput. Sci. Rev.*, vol. 17, pp. 70–81, 2015.

[3] A. S. Tanenbaum and M. Van Steen, *Distributed Systems: Principles and Paradigms, 2/E*. 2007.

[4] D. B. Kahanwal and D. T. P. Singh, "The Distributed Computing Paradigms: P2P, Grid, Cluster, Cloud, and Jungle," Nov. 2013.

[5] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," p. 7, 2011.

[6] Z. Zheng, J. Zhu, and M. R. Lyu, "Service-Generated Big Data and Big Data-as-a-Service: An Overview," in *2013 IEEE International Congress on Big Data*, 2013, pp. 403–410.

[7] R. Cattell, "Scalable SQL and NoSQL data stores," *ACM SIGMOD Rec.*, vol. 39, no. 4, p. 12, 2011.

[8] K. Grolinger, W. A. Higashino, A. Tiwari, and M. A. M. Capretz, "Data management in cloud environments: NoSQL and NewSQL data stores," *J. Cloud Comput.*, vol. 2, no. 1, 2013.

[9] Mh. will the database incumbents respond to N. and N. Aslett, "How will the database incumbents respond to NoSQL and NewSQL?," *San Fr.*, vol. 4 April 20, no. February 2009, pp. 1–5, 2011.

[10] J. C. Corbett *et al.*, "Spanner : Google's Globally-Distributed Database," *Proc. OSDI'12 Tenth Symp. Oper. Syst. Des. Implement.*, pp. 251–264, 2012.

[11] VoltDB.inc, "VoltDB Technical Overview," 2017. [Online]. Available: https://www.voltdb.com/wp-content/uploads/2017/04/VoltDB_Technical_Overview_14Sept17.pdf. [Accessed: 07-Oct-2017].

[12] CluxtrixDB, "A NEW APPROACH TO SCALE-OUT RDBMS," 2017. [Online]. Available: https://www.clustrix.com/wp-content/uploads/2017/01/Whitepaper-ANewApproachtoScaleOutRDBMS.pdf. [Accessed: 07-Oct-2017].

[13] inc NuoBD, "NuoDB ® Emergent Architecture," 2013. [Online]. Available: http://go.nuodb.com/rs/nuodb/images/Greenbook_Final.pdf. [Accessed: 07-Oct-2017].

[14] "Buenos Aires Data." [Online]. Available: https://data.buenosaires.gob.ar/.

[15] Google, “Cloud Spanner Documentation,” 2017. [Online]. Available: https://cloud.google.com/spanner/docs/quickstart-console?hl=es.

Árboles Arraigados una Herramienta en la Visualización de Elementos de Gramáticas y Palabras en Lenguajes

Nancy Alonso[1] , Elisa Oliva[1]

[1] Departamento de Informática, Facultad de Ciencias Exactas,Físicas y Naturales, Universidad Nacional de San Juan,Argentina
{alonsonancye, elisaoliva65}@gmail.com

Resumen. Un progreso en el arte de programar del alumno, se puede lograr mejorando la calidad y fiabilidad de su trabajo, para lo cual se requiere que el estudiante se introduzca en el conocimiento de la perspectiva formal sobre la programación: sintaxis, semántica, corrección y eficiencia del lenguaje, todo esto permite mayor nivel de abstracción. El aporte desde matemática discreta es brindar los fundamentos matemáticos de teoría de la computación y aplicar herramientas de visualización gráfica, por ejemplo para agilizar la deducción en gramáticas de símbolos alcanzables, generadores, y en su lenguaje generado, de cadenas sintácticamente correctas, por aplicación de árboles de búsqueda y árboles de derivación.

Palabras Claves: Árbol dirigido, Gramática, Símbolos alcanzables, Símbolos generadores.

1 Introducción

Los avances tecnológicos de los últimos años y el proceso de acreditación de carreras de grado han producido cambios en la currícula de las Licenciaturas en Ciencias de la Computación y en Sistemas en Información, apoyando el desarrollo de muchos temas de aplicación, entre los que se encuentran los Métodos Discretos.

La asignatura Matemática Discreta tiene como objetivo proporcionar contenidos matemáticos que fundamenten al estudiante el uso estructuras finitas, tales como grafos y sus aplicaciones, base en Teoría de la Computación.

Los árboles tienen aplicaciones en: diseño de compiladores, sistemas expertos, sistemas evolutivos, sistemas conscientes, manejo de directorios de la PC. Sus aplicaciones son muchas, aunque su implementación no es tan usada como las bases de datos tradicionales.

En este trabajo se aborda el uso de árboles en gramáticas: para detectar símbolos generadores, alcanzables y por medio de árboles de derivación, visualizar que el uso de ciertas combinaciones sintácticas permite generar palabras en un lenguaje determinado, mientras que otras combinaciones resultan ser expresiones agramaticales o sólo formas sentenciales que no tendrán todos sus elementos del alfabeto (aun cuando en otro lenguaje sí pudieran ser correctas). Se basa su construcción en aplicación de un número finito de veces de las reglas formales de la gramática (reglas de producción).

2 Metodología

La metodología aplicada, está enfocada en que los alumnos aprendan a desarrollar su capacidad lógico-matemática , lograr una participación activa en cada uno de los procesos de enseñanza-aprendizaje y fortalecer el pensamiento gráfico. El desarrollo del pensamiento visual autónomo [1], es una vía para mejorar el pensamiento matemático, logrando un equilibrio entre los contextos gráficos, numéricos y algebraicos. Las gráficas han sido usadas desde la antigüedad hasta nuestros días como una forma de describir el movimiento de situaciones, la variación de fenómenos para caracterizar comportamientos, como una forma de interpretación de soluciones, como una forma de comunicar resultados para entender problemas o generar habilidades para resolverlos [2].

Los contenidos teóricos se desarrollan con métodos inductivos, deductivos, de análisis y síntesis. Las aplicaciones de práctica se trabajan en forma grupal bajo la supervisión del equipo docente y cada guía se acompaña de un anexo con ejercicios propuestos.

3 Contenidos Teóricos de la Propuesta

3.1 Árboles Arraigados

Un árbol arraigado [3],[4], de raíz v_0 es un grafo dirigido $G = (V, R)$ en el cual existe un vértice v_0, denominado raíz del árbol, que no admite trayectorias en sí mismo y para cada vértice $v_i \neq v_0$, existe una única trayectoria $T_{v_o v_i}$ en el grafo G.

Elementos de un Árbol Arraigado

Raíz - Vértice v_0 que destaca la definición.
Vástago - El vértice v_j es un vástago, o hijo del vértice v_i, si $(v_i,v_j) \in R$
Hoja (vértice terminal) - Vértice que no tiene vástagos.
Vértice interno - Vértice que no es raíz ni hoja.
Nivel 0 - Conjunto unitario cuyo elemento es la raíz del árbol.
Nivel 1 - Conjunto de vástagos de la raíz.
Nivel k - Conjunto de vástagos de los vértices del nivel k−1
Altura de G: máx {long. $T_{v_o v_i}$, $v_i \in V$} (cantidad de niveles no nulos)

Tipos de árboles

Etiquetado: Tienen vértices y/o arcos con etiquetas o nombres.
n-ario: Los vértices, que no son hojas, tienen a lo sumo n vástagos.
n-ario completo: Todos los vértices, que no son hojas, tienen n vástagos.
Ordenado: Cuándo se ha prefijado un orden en su conjunto de vértices, por niveles o por ramas.
Árbol de expresión algebraica: En este árbol las etiquetas son: raíz y vértices internos operaciones del álgebra y hojas elementos del portador del álgebra.

3.2 Métodos de Búsqueda con Arboles.

Un problema general, en la teoría de árboles arraigados, implica la exploración de un grafo en busca de cierto vértice que tenga asociada alguna propiedad sobre lo que modela el grafo. Se trabajan grafos conexos y se realizará la búsqueda por medio de un árbol arraigado [4], [5], [6], [7]. Existen dos formas de búsqueda, la búsqueda en amplitud (BFS) y la búsqueda en profundidad (DFS).

Tabla 8. Lista de adyacencia de un grafo G

Vértice	Vértices adyacentes
v1	v2- v4- v5
v2	v1- v8
v3	v4
v4	v1
v5	v2- v4- v9
v6	v3- v7
v7	v6
v8	v2
v9	v5- v6- v8

Búsqueda en amplitud (BFS)

La búsqueda en amplitud, o de expansión, se puede utilizar para determinar la distancia más corta, mínimo número de arcos que hay que recorrer para pasar, del vértice inicial al vértice seleccionado y se realiza por niveles del árbol.

Procedimiento

Se selecciona un vértice inicial, raíz del árbol de búsqueda. Se visitan los vértices a distancia 1 del inicial (adyacentes), en el orden de la lista y se escribe una secuencia de vértices visitados. En una etapa (k-1), se visitan los vértices a distancia k-1 del inicial que no fueron visitados y se visitan los vértices adyacentes a éstos, en el orden de la lista, y que no fueron visitados. El algoritmo finaliza cuando se visitan todos los vértices del grafo. Notar que en cada paso de búsqueda, se seleccionan caminos más cortos desde la raíz, es conveniente agregar una etiqueta a los arcos que indique el orden de lectura de la búsqueda.

Ejemplo: La siguiente figura presenta el árbol BFS del grafo cuya lista de adyacencia se dio en la tabla anterior, tomando como raíz al vértice v_2.

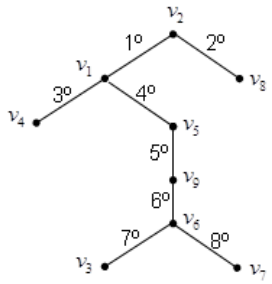


Fig. 1. Árbol de la búsqueda en amplitud del grafo G con raíz v_2.

Visualmente, se puede concluir que la secuencia de vértices visitados en la búsqueda en amplitud es: $v_2\ v_1\ v_8\ v_4\ v_5\ v_9\ v_6\ v_3\ v_7$.

Búsqueda en profundidad (DFS)

Esta búsqueda se realiza por ramas del árbol.

Procedimiento

Se marca el vértice inicial y se visita y marca su primer adyacente. Se visita el primer adyacente del marcado en el nivel anterior.

- si no fue visitado, se marca y se pasa a su adyacente
- si ya fue visitado, se vuelve al adyacente marcado en el nivel anterior y se repite el proceso.
- si se encuentra un vértice que no tiene adyacentes no visitados, se regresa al vértice del nivel anterior que tenga vértices adyacentes no marcados.

El algoritmo finaliza cuando se visitan a todos los vértices del grafo.

El árbol de Búsqueda en profundidad (DFS), con raíz en el vértice inicial, se construye con los pasos de búsqueda, sus arcos se etiquetan con el orden de lectura de la búsqueda.

Ejemplo: La figura presenta el árbol de la búsqueda en profundidad del grafo cuya lista de adyacencia se dio en Tabla 1, seleccionando al vértice v_2, como raíz.

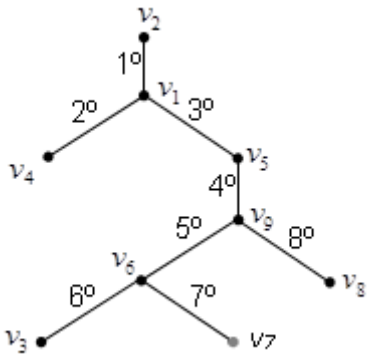


Fig. 2. Presenta el árbol de la búsqueda en profundidad del grafo G, tomando como raíz al vértice v_2.

Visualmente, se puede concluir que la secuencia de vértices visitados en la búsqueda en profundidad es: $v_2\, v_1\, v_4 v_5 v_9 v_6 v_3 v_7 v_8$.

3.3 Examen con Arboles de Símbolos Alcanzables y Generadores en Gramáticas

Una gramática G, es una estructura algebraica formada por cuatro elementos [4], [8], [9]:

$$G=(N,\Sigma,S,P). \quad (1)$$

N: Conjunto finito de símbolos no terminales
Σ: Conjunto finito de símbolos terminales (alfabeto del lenguaje)
S: Símbolo no terminal denominado símbolo inicial o axioma ($S \in N$).
P: Relación de producción.

$$P \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* x (N \cup \Sigma)^*. \quad (2)$$

$$(\alpha,\delta) \in P. \quad (3)$$

Determina una "regla de producción" que se lee "la cadena α produce la cadena δ" (α se denomina "cabeza" y δ "cuerpo" de la regla), y se simboliza con:

$$\alpha \mapsto \delta\,. \qquad \textbf{(4)}$$

Relación deriva

La relación deriva se simboliza con $\underset{\text{k)}}{\Rightarrow}$ y está definida en $(N \cup \Sigma)^*$. Si existe la regla de producción k) definida en **(4)**, se obtiene:

$$\beta\alpha\sigma \underset{k)}{\Rightarrow} \beta\delta\sigma. \qquad \textbf{(5)}$$

Se lee "la cadena βδσ, se deriva de la cadena βασ, por la aplicación de regla k) ".
Si la cadena βδσ, se deriva de la cadena βασ, por aplicación de las reglas $k_1)$, $k_2)$,…,$k_n)$ se escribe :

$$\beta\alpha\sigma \underset{k_1)\ldots k_n)}{\Rightarrow} \beta\delta\sigma. \qquad \textbf{(6)}$$

O bien se indica:

$$\beta\alpha\sigma \Rightarrow^{*} \beta\delta\sigma\,. \qquad \textbf{(7)}$$

Símbolos accesibles y generadores de una gramática

Un símbolo $X \in (N \cup \Sigma)$, es alcanzable si existe una derivación de la forma definida en **(7)**, que se hay iniciado desde el axioma, es decir donde: βασ = S y βδσ=βXσ

Un símbolo $X \in N$, es generador si existe una derivación como la definida en **(7)**, para $w \in \Sigma^*$, dada por:

$$X \Rightarrow^{*} w. \qquad \textbf{(8)}$$

El axioma de un lenguaje no vacío siempre se lo considera "útil", es generador y aunque puede ser o no alcanzable.

Árbol arraigado para determinar símbolos alcanzables.

- La raíz del árbol se etiqueta con S, axioma de la Gramática.
- La raíz tendrá tantos hijos como reglas de producción tengan como cabeza el axioma.
- Cada vértice interno tendrá tantos hijos como reglas de producción puedan aplicarse a través de la relación deriva.
- Un vértice será hoja cuando no sea posible aplicar la relación deriva.
- Todo símbolo perteneciente a una cadena vértice del árbol, será alcanzable.

- Todo símbolo que no pertenece a cadena alguna de los vértices del árbol, es no alcanzable.

Las hojas de estos árboles de raíz S, son cadenas que contienen sólo símbolos terminales, pertenecientes al lenguaje de la gramática.

Al considerar la siguiente gramática: G=(N,Σ,S,P), N={A,B,C,D,E,S}, Σ={a,b,c,d}, con reglas de producción: 1)S ⊢→ Db; 2) S ⊢→ aBC; 3)S ⊢→ baC; 4) BC ⊢→ d; 5)B ⊢→ dBC; 6)AaS ⊢→ da; 7) C ⊢→ a; 8)E ⊢→ Bac

La siguiente figura presenta el árbol para determinar símbolos alcanzables, tomando siempre para su desarrollo el axioma como raíz.

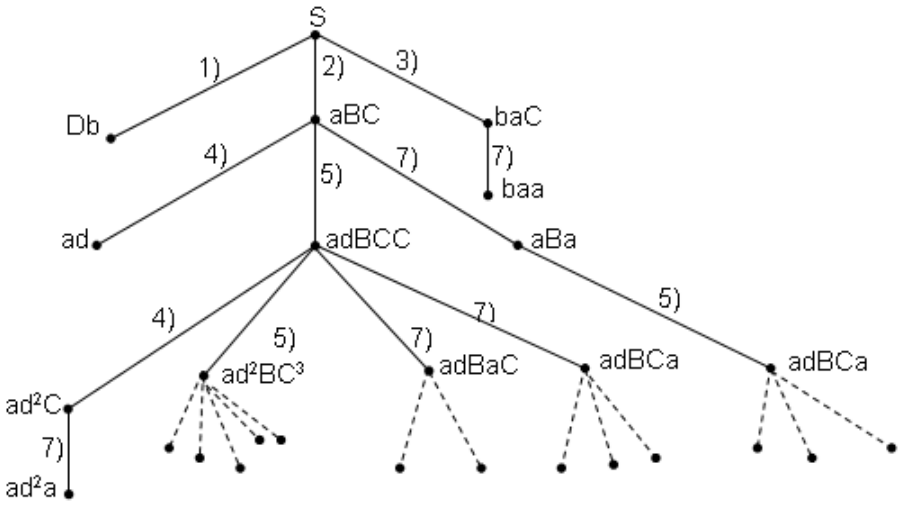


Fig. 3. Arbol para determinar símbolos alcanzables, pudiéndose leer hasta el momento de este desarrollo que son símbolos alcanzables: {B,C,D,a,b,d}

Árbol arraigado para determinar símbolos generadores.

- La raíz del árbol se etiqueta con el símbolo no terminal *X*.
- La raíz tendrá tantos hijos como reglas de producción tengan como cabeza el símbolo *X*.
- Cada vértice interno tendrá tantos hijos como reglas de producción puedan aplicarse a través de la relación deriva.
- Un vértice será hoja cuando no sea posible aplicar la relación deriva.
- Si el árbol tiene, por lo menos, una hoja etiquetada con una cadena sólo de símbolos terminales, *X* es generador.
- Si no es posible, encontrar una hoja cuya cadena contenga sólo símbolos terminales, X es no generador.

Al considerar la gramática: G=(N,Σ,S,P), N={A,B,C,D,E,S}, Σ={a,b,c,d}, con reglas de producción: 1)S⊢→ Db; 2) S⊢→ aBC; 3)S ⊢→ baC; 4) BC⊢→ d; 5)B⊢→ dBC; 6)AaS⊢→ da; 7) C⊢→ a; 8)E⊢→ Bac

La siguiente figura presenta los árboles para determinar símbolos generadores, tomando siempre para su desarrollo los símbolos no alcanzables como raíz.

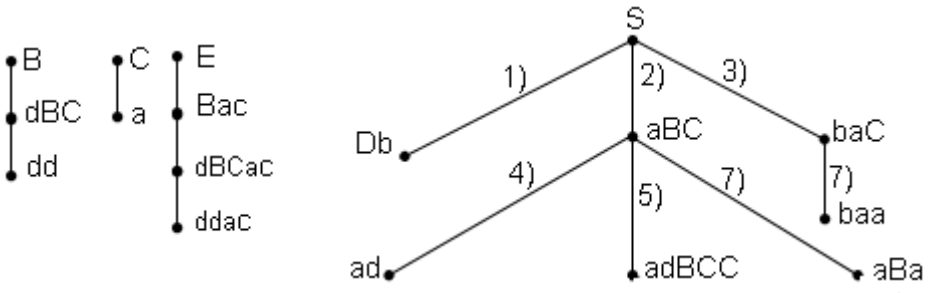


Fig. 4. Árboles para determinar símbolos generadores, pudiéndose observar que en la gramática G son símbolos generadores: {B, C, E, S}. Los símbolos no terminales A y D, no son cabeza de regla no se puede aplicar la relación deriva, por lo tanto A y D son no generadores.

3.4 Árbol de Derivación

El árbol de derivación [8] de una cadena aceptada por el lenguaje correspondiente a una gramática de Tipo 2 ó Tipo 3 es útil en el análisis sintáctico de Lenguajes de Programación. Tiene raíz y vértices internos etiquetados con símbolos no terminales y hojas etiquetadas con símbolos terminales ó con la cadena vacía.

De cada árbol de derivación se extrae una palabra del lenguaje de la gramática G, L(G), recíprocamente para cada palabra de L(G) se construye un árbol de derivación, cuando este árbol no es único la Gramática es ambigua.

Reglas de construcción del árbol de derivación

- Se etiqueta la raíz del árbol con el axioma S de la gramática.
- Se aplica una regla con cabeza S, se dibujan tantos vástagos de S como símbolos tiene el cuerpo y se etiquetan de izquierda a derecha en el orden en que están en el cuerpo. Los vástagos etiquetados con símbolos terminales o con la cadena vacía ε son hojas del árbol (no tienen vástagos).
- En cada vértice con símbolo no terminal se aplica una regla que lo tiene como cabeza y se dibujan tantos vástagos como símbolos tiene el cuerpo, los vástagos se etiquetan

de izquierda a derecha en el orden que están en el cuerpo. Los vástagos etiquetados con símbolos terminales o con la cadena vacía ε son las hojas del árbol.

- El proceso continúa hasta obtener sólo vértices hojas.

Para encontrar la sucesión de símbolos que forman la palabra de L(G) se realiza en el árbol una búsqueda en profundidad. Se selecciona el primer vástago de la raíz y se recorre la rama, considerando los primeros vástagos de cada nodo, hasta encontrar la hoja, se continúa la búsqueda en profundidad. La palabra está formada con símbolos que etiquetan las hojas encontradas en la búsqueda. La cadena que forma la palabra se denomina "frontera" ó "resultado" del árbol. Para la lectura de un árbol de derivación se lo considera ordenado por niveles y los vértices de cada nivel se preceden de izquierda a derecha

Al considerar la gramática: G=(N,Σ,S,P), N={A,B,S}, Σ={a,o,d,l,h,p,s}, con reglas de producción: 1)S⟼ hB; 2) S⟼ sBo; 3)A ⟼ p; 4) A⟼ l; 5)B⟼ aA; 6)B⟼ aBa; 7) B⟼ d

La siguiente figura presenta los árboles de derivación de las secuencias: "hada", "sapo" y "saalalo", palabras de L(G).

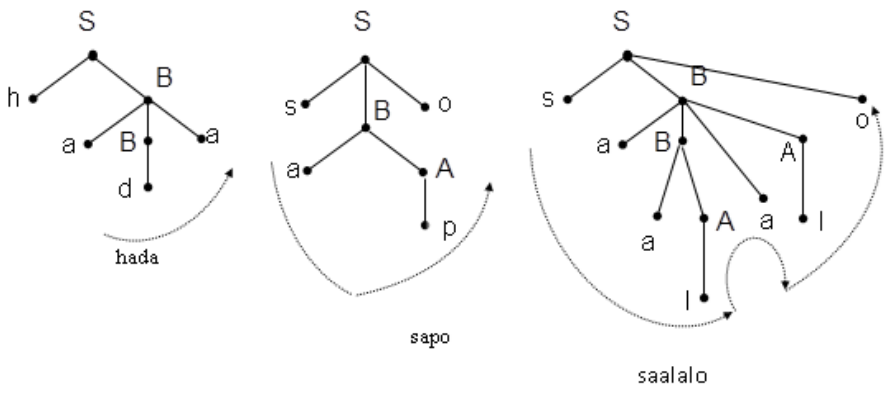


Fig. 5. Árboles de derivación de las palabras: "hada", "sapo", "saalalo" de L(G)

4 Conclusiones

En las carreras Licenciaturas en Sistemas de Información y Ciencias de la Computación, los diagramas de árbol, se utilizan con frecuencia, siendo gran ayuda en la organización de datos y en la visualización de relaciones. Los diagramas arborescentes, constituyen una

de las subclases más útiles de grafos, importantes en el estudio de estructuras de datos, ordenaciones, codificación y problemas de optimización.

El uso de este tipo de prácticas con fuerte base visual, en el aula, permite mejorar la calidad de los resultados del aprendizaje, y de los procesos del quehacer académico en carreras no matemáticas. Las gráficas son una herramienta a sumar, para que el alumno desarrolle razonamiento matemático, ayudan a visualizar situaciones, contribuyen a una forma de interpretar de soluciones, son una forma de comunicar resultados después de entender un problema, son un registro que favorece la resignificación de conocimiento.

El trabajo con árboles se aplicará en lo inmediato a la detección de símbolos inútiles de una gramática, pero esta estrategia didáctica de visualización gráfica se orienta a un modo de exploración que guía al alumno a hacer matemática: en el análisis de procesos y búsqueda de resultados.

Bibliografía

1. Buendía, G.: Una revisión socio-epistemológica acerca del uso de las gráficas. En G. Buendía (Ed.) A diez años del posgrado en línea en Matemática Educativa en el IPN México: Colegio Mexicano de Matemática Educativa AC,21-- 40(2010)
2. Guétmanova, A.: Lógica. Serie: Biblioteca del estudiante. Moscú: Progreso (1989).
3. Abellanas, M.: Análisis de algoritmos y teoría de grafos (1°Ed.). Madrid: RA-MA. (1990).
4. Chillemi, A., Alonso, N.: Matemática discreta para informáticos (1°Ed). Argentina: UNSJ (2013).
5. Grimaldi, R.: Matemáticas discreta y combinatoria (3°Ed). México: Prentice Hall (1998).
6. Korfhage R.: Lógica y Algoritmos (1°Ed). México: Limusa (1974).
7. Johnsonbaugh R.: Matemáticas Discretas (1°Ed). México: Iberoamérica (1990).
8. Cubero, E., Moreno, M., Salomon, R.: Teoría de Autómatas y lenguajes Formales (3°Ed). Madrid: Mc Graw Hill. (2007).
9. Hopcroft J.-Motwani R.-Ullman J.: Teoría de autómatas Lenguajes y computación (3°Ed). España: Pearson (2008).

Juez Imparcial: infraestructura para compilación, verificación y originalidad de programas en entornos E-Learning

Marco Aponte[1], Eduardo Florencio[1] y Cristian Cappo[1]

[1] Facultad Politécnica, Universidad Nacional de Asunción, San Lorenzo, Paraguay

Resumen. El proceso de corrección manual de tareas de programación es un proceso tedioso y extenso. En busca de una mejora en dicho proceso surge la alternativa de que la corrección se realice de forma automática. En la Facultad Politécnica de la Universidad Nacional de Asunción, así como en otras facultades de la Universidad existen carreras que cuentan con materias referentes a programación de computadoras; para la gestión de las mismas frecuentemente se utiliza un sistema de gestión de cursos, también conocido como entorno *E-learning*. Este artículo presenta una solución integrada al entorno de e-learning *Moodle* que de forma automática realiza la compilación, ejecución de casos de prueba, calificación y posibilita la generación de un informe de originalidad de las tareas entregadas por los alumnos. Esta propuesta fue implementada y sometida a experimentos en un ambiente real de aula, que mostró su utilidad y un ahorro de tiempo y esfuerzo por parte del docente con respecto a la corrección manual.

Palabras clave: Juez en línea, Sistema de Gestión de Cursos, E-learning, Verificación de Originalidad, Automatización, Moodle.

Introducción

Desde la invención del motor de diferencias de Charles Babbage en 1822, las computadoras han requerido un medio de instrucción para poder realizar una tarea específica. Este medio de instrucción es lo que se conoce como lenguaje de programación. Los mismos han ido evolucionando desde entonces hasta la actualidad. En sus comienzos eran solamente utilizados en el campo científico, para luego ir migrando hacia propósitos más generales. Debido a la creciente necesidad de programas en diferentes ámbitos, los lenguajes de programación comenzaron a ser enseñados en las universidades del mundo. La programación es una de las principales capacidades que forman parte de las carreras de informática y ésta requiere que sea adquirida a través de prácticas, las que son realizadas con asignaciones de programación en algún lenguaje particular [1].

En el contexto de las universidades del Paraguay, específicamente en la Universidad Nacional de Asunción (UNA), varias son las carreras que incluyen como parte de su malla curricular asignaturas relacionadas a programación de computadoras. Para la gestión de estas materias es frecuente la utilización de sistemas de gestión de cursos, como entornos

E-learning o plataformas de educación virtual, frecuentemente *Moodle* [10]. En dichas materias luego del desarrollo del contenido teórico, es usual que se realicen ejercicios prácticos de programación, donde el código escrito por los alumnos debe ser analizado por los instructores, es decir ser compilado y ejecutado para verificar su correcto funcionamiento. Con frecuencia las tareas son entregadas a través del sistema de gestión de cursos y deben ser luego evaluadas por el docente. Este trabajo actualmente se realiza de forma manual, ya sea por parte del profesor titular de la cátedra o del auxiliar. Dentro de esta actividad se encuentra la verificación de originalidad de los códigos entregados. El proceso de corrección (compilar, ejecutar, corroborar que la salida sea correcta y comparar códigos para verificar la originalidad) es tedioso y extenso; el tiempo requeridopodría ser utilizado de forma más eficiente en otras tareas académicas.

Teniendo en consideración la problemática descrita, el proceso de corrección podría automatizarse utilizando lo que se conoce como Juez en línea, un programa que de forma automática realiza tanto la compilación y ejecución de códigos, así como la verificación del resultado y la originalidad de los mismos [1].

Este artículo presenta una solución incorporada a un entorno *E-learning* que facilita el proceso de corrección de las tareas de programación e incorporara facilidades para la verificación de originalidad de los trabajos entregados. Específicamente integra el entorno de compilación de programas y ejecución de casos de prueba definidos por el docente a la plataforma de e-learning *Moodle*, con un mecanismo de control de originalidad para asistir al docente en la verificación de los trabajos enviados.

El resto del artículo se encuentra organizado de la siguiente manera. Los trabajos relacionados a esta propuesta se describen en la Sección 2. La Sección 3 contiene los fundamentos teóricos más importantes que guardan relación con los sistemas de gestión de cursos, jueces en línea y control de originalidad. En la Sección 4 se presenta el desarrollo de la propuesta. En la Sección 5 se detallan los experimentos realizados y los resultados obtenidos en dichos experimentos. Por último, la Sección 6 corresponde a las conclusiones y posibles trabajos futuros.

Trabajos Relacionados

Un Juez en línea es una plataforma que de manera automática compila y ejecuta código según casos de prueba predefinidos y utilizado generalmente en competencias de programación [1]. En estas competencias se forman grupos de participantes que resuelven variados problemas y las soluciones propuestas por los concursantes son compiladas y evaluadas a fin de determinar los puntajes. Dicha actividad tiene requerimientos similares a los de la corrección de tareas de programación de alumnos. Entre las herramientas utilizadas en competencias de programación como ACM se tiene a *DOMjudge* [13], que así como la solución propuesta en este trabajo posee incorporado un juez en línea, brinda soporte a lenguajes como C/C++ y Java, ofrece entorno seguro de ejecución (*sandbox*) y se encuentra implementado principalmente en PHP; difiere en que está orientado a competencias de programación y no a cursos de programación, no posee integración con *Moodle* ni verificación de originalidad; y a *SharifJudge* [12], que posee las mismas semejanzas que *DOMjudge* con la solución propuesta y de forma adicional posee verificación de originalidad con un servicio remoto de verificación, denominado MOSS

[2], al igual que la solución propuesta en este trabajo se encuentra orientado a cursos de programación, mas no posee integración con *Moodle* u otra plataforma de educación virtual. Por lo mencionado anteriormente no son las soluciones más aptas para ser implementadas en la institución mencionada, ya que en la misma se dictan cursos de programación y se utiliza una plataforma de educación virtual basada en *Moodle* denominada *Educa*[7].

La corrección automatizada de tareas no es una idea nueva, sino que desde años atrás se viene implementando soluciones que responden parcialmente a la necesidad de eficiencia existente. En 2012 fue propuesto por Zhigang et al. [16] un plugin para *Moodle* [11,10] que es semejante a lo propuesto en este trabajo, sin embargo no es funcional en la actualidad debido a los cambios estructurales realizados a *Moodle*; el plugin brinda soporte a los lenguajes C, C++ de forma local y varios otros de forma remota utilizando un servicio pagado de compilación de programas vía web [15], tales como C#, JavaScript, Perl y PHP. Chaves et al. [4] proponen una herramienta similar integrada con *Moodle* con soporte a lenguajes tales como C, C# y PHP, más al momento de la redacción de su artículo no fue puesto a disposición del público una versión estable de la herramienta.

En cuanto a control de originalidad se pueden mencionar varias herramientas propuestas en la literatura. El trabajo presentado por Bowyer et al. [3] se basa en su experiencia relativa a la utilización del sistema MOSS para obtener los niveles de similitud de los trabajos presentados por alumnos durante cursos de programación. En el trabajo mencionado se destaca el buen desempeño de MOSS, el sistema determinó que el nivel de originalidad del primer curso fue de 87%, y el del siguiente semestre el nivel detectado fue de un 93,6%. Dicho nivel fue verificado de forma manual por los docentes, concluyendo que la mayoría de los casos de alto grado de similitud fueron copias parciales o completas de tareas de otros alumnos. La principal competencia de MOSS es Jplag [15], que posee las mismas funcionalidades y un rendimiento similar. Otras herramientas no tan populares son: *CodeMatch* [6], una solución propietaria que compara archivos de códigos fuente en múltiples directorios para hallar el grado de similitud entre los mismos, soporta 40 lenguajes de programación; *YAP3* [9], un sistema para detectar plagio tanto en programas de computadoras como en otros textos entregados por alumnos; y *Sherlock* [14], un programa para encontrar similitudes en documentos de texto.

Fundamentos

En esta sección se explica brevemente las bases teóricas correspondientes a los sistemas de gestión de cursos, jueces en línea y control de originalidad.

Sistemas de Gestión de Cursos

Un sistema de gestión de cursos es una aplicación de software que automatiza la administración, seguimiento y reportes de programas de capacitación, tales como carreras de grado, postgrados, cursos, seminarios [7]. Lo más básico que ofrecen los sistemas de

[7]http://www.educa.una.py

gestión de cursos son herramientas que permiten al educador crear el sitio web para un curso en particular y un control de acceso para que sólo los estudiantes inscriptos en el mismo puedan ver los contenidos.

También ofrecen una amplia variedad de herramientas que permiten al curso tener mayor efectividad, proveyendo métodos fáciles para: subir y compartir materiales, tener conversaciones y/o discusiones en línea, dar exámenes, completar cuestionarios, dar y corregir tareas, así como registrar notas.

Existen varios sistemas de gestión de cursos como *Blackboard*[8], *Moodle*[9], *Desire2Learn*[10], *Instructure*[11], entre otros; algunos de ellos son sistemas propietarios y otros de código abierto.

El que mayor soporte posee entre los de código abierto es *Moodle*, con una cuota de mercado del 23% [8]. *Moodle* es utilizado en la Universidad Nacional de Asunción. Teniendo en cuenta el soporte y su uso actual en la UNA, el juez en línea propuesto en este trabajo se integra a este sistema de gestión de cursos.

Jueces en línea

Un juez en línea es un programa encargado de validar el correcto funcionamiento de un código de programación, ya sea en un entorno didáctico o de competencia de programación [1]. Algunos jueces en línea existentes son: *DOM Judge* [13], *SharifJudge* [12], *Online Judge* [16] entre otros.

Las ventajas que provee un juez en línea a través de la corrección automatizada de códigos fuentes en comparación a la corrección manual son las siguientes: la corrección se hace de manera precisa, no da lugar a la subjetividad ni errores humanos, provee un entorno seguro de compilación y ejecución de códigos, menor período de tiempo de corrección, consistencia en las correcciones.

Control de Originalidad

Existen trabajos que son el resultado de la creatividad humana fruto del razonamiento en múltiples ámbitos tal como en la música, la literatura y así también en la informática. Cuando el trabajo de alguien es reproducido sin reconocer la fuente, se produce lo que se conoce como plagio. Probablemente los casos más frecuentes ocurren en instituciones académicas donde los estudiantes copian el contenido de libros, periódicos, Internet, sus compañeros, etc. sin citar como referencia. Aunque a veces es intencional, hay muchos casos en donde estudiantes copian sin intención, simplemente a causa de no estar al tanto de la forma en que las fuentes deben ser referenciadas en su trabajo. Este problema no está limitado sólo a texto escrito, también ocurre en código de software en donde fragmentos son copiados y reutilizados sin referenciar al autor original.

[8] http://www.blackboard.com
[9] https://moodle.org
[10] https://www.d2l.com
[11] https://www.instructure.com/

La mayoría de los estudiantes en la actualidad entregan sus tareas de texto escrito y/o código fuente de forma digital. Aunque sea más conveniente y práctico para estudiantes y profesores, la versión digital permite al estudiante copiar con mayor facilidad. Es por ello que es necesaria la utilización de una herramienta que permita detectar el grado de similitud existente entre las tareas entregadas; es posible utilizar un programa que realice de forma automática la comparación y brinde como resultado un reporte de originalidad referente a las tareas de los alumnos. Algunos de ellos existentes en la actualidad son: *MOSS* [2], *JPlag* [15] y *YAP3* [9].

Propuesta

La Figura 1 presenta la arquitectura de la solución propuesta. Basada en la plataforma Moodle se construyó un plugin del tipo *Assign Submission* que a través del API (*JudgeLib*) interactúa con los procesos de corrección acorde al lenguaje de programación seleccionado. Estos procesos son ejecutados de manera segura en el servidor de compilación en formato de *Sandbox*.

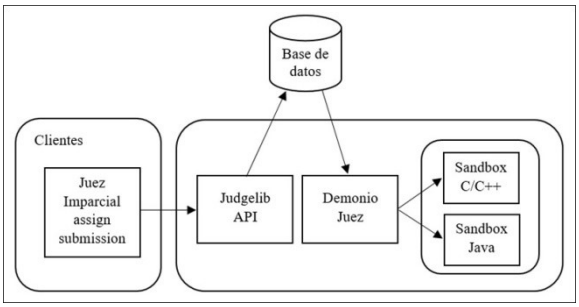


Fig. 1. Arquitectura general del Juez en línea

El proceso detallado de interacción entre plataforma y los usuarios de la misma, que serían profesores y alumnos, se muestra en la Figura 2. En una primera fase el profesor procede a la creación de la tarea del tipo "*Juez Imparcial*", configurando los parámetros y los casos de prueba del ejercicio. La interfaz se comunica con la API de Moodle encargada de procesar los datos introducidos por el profesor y mandarlos a la base de datos para ser guardados. Una vez guardada la información, la base de datos retorna un mensaje de éxito a la API de Moodle que luego es reenviado a la interfaz web; que a su vez despliega el mensaje de éxito al profesor. En caso de producirse algún error en toda la fase, dicho error es capturado y desplegado al profesor.

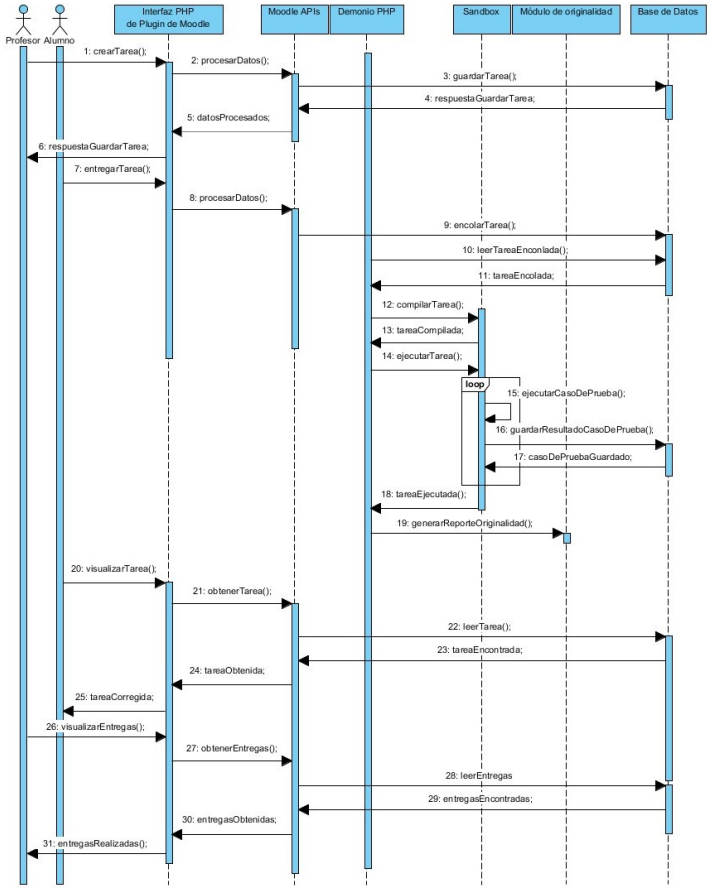


Fig2. Diagrama de Interacción del Juez en Línea

En una segunda fase el alumno entrega la tarea asignada dentro de la plataforma de gestión de cursos Moodle. La interfaz se comunica con la API de Moodle pasándole la tarea entregada para su corrección. La API de Moodle solicita a la base de datos encolar la tarea recibida para ser procesada por el demonio PHP. El demonio detecta la presencia de una nueva tarea pendiente de corrección en la cola; por lo que toma dicha tarea de la base de datos y la procesa. Para la corrección se delega el control al *sandbox* que compila y posteriormente ejecuta el código de la tarea entregada dentro del entorno seguro pasando los parámetros de entrada ya definidos con anterioridad y comparando la salida de la ejecución con la definida de forma previa por el profesor para cada caso de prueba, guardando en la base de datos los resultados para cada uno de ellos acumulando el puntaje obtenido para la nota. La base de datos retorna un mensaje por cada resultado correspondiente a un caso de prueba. Al finalizar la corrección, el demonio solicita al módulo de verificación de originalidad el reporte de similitud correspondiente a todas las

tareas entregadas hasta el momento. En caso de producirse algún error en toda la fase, dicho error es atrapado y se despliega un mensaje de fallo al alumno.

En una tercera fase el alumno solicita la visualización del estado de la tarea entregada a través de la interfaz web del plugin. La interfaz traspasa la solicitud a la API de Moodle de la información correspondiente a la tarea. La API de Moodle lee de la base de datos la información de la tarea para retornar la tarea obtenida a la interfaz web para su visualización al estudiante. En el estado de la tarea se visualiza el puntaje obtenido para cada caso de prueba, así como los parámetros de entrada y salida utilizados para la corrección. En la Figura 3 puede observarse un ejemplo de la interfaz disponible para el estudiante sobre el estado de entrega de una tarea. Ésta corresponde a una tarea en lenguaje C que se encuentra para calificar, con dos casos de prueba definidos que fueron aceptados.

En una última fase el profesor visualiza las entregas que desee a través de la interfaz web. La interfaz web se comunica con la API de Moodle, que a su vez obtiene de la base de datos las tareas solicitadas. Una vez obtenidas las tareas de la base de datos, la API realiza el traspaso de las mismas a la interfaz web. La interfaz web despliega en pantalla las tareas solicitadas, junto con los puntajes obtenidos para cada caso de prueba y así también el enlace correspondiente al reporte de originalidad para todas las tareas entregadas hasta el momento. La propuesta integra a *JPlag* [15] como verificador de similitud entre las tareas de los alumnos.

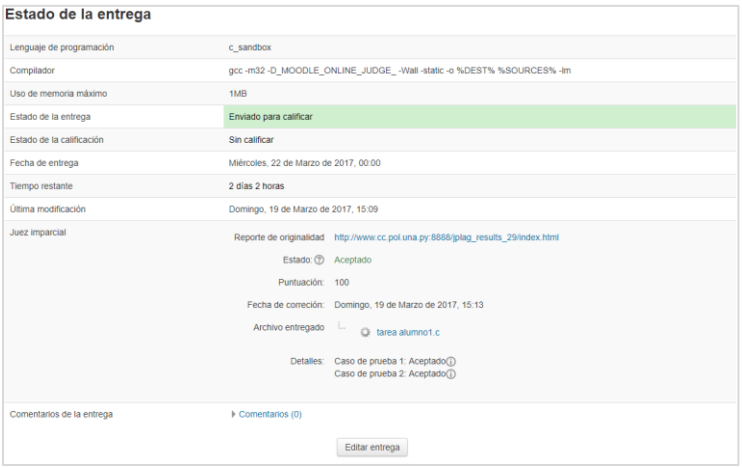


Fig 3. Interfaz del estudiante sobre el estado de entrega de una tarea.

Aspectos técnicos

La herramienta está formada por varios componentes tal como se visualiza en el diagrama de secuencia, Figura 2. El componente encargado de la comunicación con el cliente es el plugin de Moodle programado en PHP, el mismo se encarga de la creación de tareas y casos de prueba por parte del profesor y así también del manejo de la entrega de las tareas por parte de los estudiantes. Las tareas entregadas por los estudiantes son encoladas en la base de datos para ser corregidas por el juez en línea. El juez en línea se encuentra construido en los lenguajes PHP y C. El juez en línea es un demonio que se encuentra constantemente verificando la cola, para que al momento de encolarse las tareas, proceda a la corrección de las mismas. La corrección implica la llamada al sandbox correspondiente, ya sea al de C/C++ o al de Java, pasándole el código que será compilado y ejecutado, el sandbox retorna la salida del programa al demonio, que realiza la comparación de la salida del programa con la salida esperada previamente definida por el profesor en los casos de prueba. Una vez realizada la corrección, el juez en línea escribe en la base de datos el resultado de los casos de prueba, junto con las salidas obtenidas de los programas entregados. Posteriormente el plugin lee de la base de datos los resultados guardados por el juez en línea para desplegar las notas de los estudiantes. Automáticamente después de la corrección de las tareas, el plugin realiza la llamada al módulo de control de originalidad programado en Java que a su vez realiza una comparación entre todas las tareas entregadas hasta el momento, dando como resultado un reporte de originalidad.

Alcance y limitaciones

La infraestructura desarrollada permite la corrección y verificación de originalidad de tareas de programación basadas en la entrada/salida estándar. Estas deben ser predefinidas por el responsable del curso. Actualmente los lenguajes soportados son C, C++ y Java. Las tareas deberán contar con un solo archivo de código fuente. La corrección de las tareas se realiza de forma secuencial por el juez. La solución requiere de un servidor Moodle versión mayor a 2.3. Para el *sandbox* de Java no se analizan los comandos utilizados en las tareas entregadas para verificar si son maliciosos.

Experimentos y resultados

Las pruebas se realizaron en el entorno de desarrollo. El mismo consistió en un servidor Linux de la distribución Ubuntu Server 14.04.5 LTS (64 bits) con 8GB de memoria RAM y un procesador Intel de 8 núcleos con frecuencia de 2,4 GHz. En él se instaló un servidor *Moodle* versión 3.0.4, así como las librerías necesarias para el funcionamiento de la herramienta, tal como un compilador de C/C++, una máquina virtual de Java; y herramientas de líneas de comando a ser utilizadas por el *sandbox* para brindar el entorno seguro de ejecución de código.

Se realizó la capacitación necesaria a los profesores y ayudantes de cátedra por medio de tutoriales y clases presenciales de demostración. A partir de la formación recibida, los

encargados de las clases de laboratorio explicaron a los alumnos la forma de utilizar la solución propuesta.

Para las pruebas en el ambiente de desarrollo fue necesario el registro de los alumnos a través de la interfaz web de *Moodle*. Ellos se registraron en los cursos creados por los profesores. Una vez habilitadas las tareas, pudieron ser visualizadas por los alumnos para su posterior entrega.

Se realizó la prueba de la infraestructura desarrollada en un ambiente real y en un ambiente de pruebas. El ambiente real consistió en la utilización de la plataforma propuesta por parte del curso Lenguajes de Programación I de la carrera Ingeniería en Informática de la Facultad Politécnica de la Universidad Nacional de Asunción. Se dieron en total 8 clases de laboratorio en donde los docentes subían las tareas y los alumnos las entregaban. Las mismas fueron compiladas, ejecutadas y corregidas por la plataforma. La misma procedió a realizar el reporte de originalidad comparando las tareas entregadas. La cantidad de alumnos varió entre 20 y 30. La cantidad de casos de prueba para cada tarea varió entre 1 y 3.

La solución propuesta logró la corrección automática de tareas y generó los reportes de originalidad correspondientes a las tareas entregadas por los alumnos. El proceso completo de compilación, ejecución, evaluación de resultados y generación de reportes no incurrió en tiempos mayores a 1 segundo.

Con estos resultados quedaron demostradas las ventajas de la corrección automática, volviendo el proceso de corrección una secuencia mecánica y eficiente.

Para los casos donde la duración de la ejecución máxima fue superada, la herramienta cortó exitosamente la ejecución y guardó el motivo de terminación correspondiente. Así también el sistema implementado para limitar la entrega de tareas en un tiempo definido resultó exitoso. Los límites establecidos para uso de memoria RAM y archivos escritos en disco se vieron efectivos.

Las notas se calcularon y guardaron de forma correcta en la base de datos basándose en los puntajes obtenidos en los casos de prueba definidos a priori por los profesores. El soporte al lenguaje español resultó útil, ya que es el lenguaje nativo utilizado por el alumnado y profesorado en su mayoría.

Las pruebas tanto en el ambiente real, como de desarrollo fueron exitosas tanto para los lenguajes C/C++ como para Java.

De acuerdo a los resultados de la evaluación de usabilidad (SUMI report[12]) realizada a profesores y alumnos, se considera a la herramienta como: intuitiva, robusta y rápida.

Conclusiones y trabajos futuros

Esta propuesta presenta un entorno de compilación y ejecución de casos de prueba integrado con la plataforma de E-Learning *Moodle* a través de un plugin para la gestión de tareas de programación. Así también como parte de la infraestructura se provee un sistema de comparación de tareas que permite analizar información sobre el grado de similitud u originalidad entre las tareas entregadas. Los lenguajes soportados actualmente son C, C++

[12]http://sumi.uxp.ie

y Java. Se establecieron los requerimientos y la información necesaria para incorporar en un futuro otros lenguajes utilizados usualmente en las aulas de la universidad.

Se realizaron pruebas y el monitoreo de rendimiento que permitieron definir los parámetros de funcionamiento de la plataforma. Así también se realizó la evaluación de usabilidad de la plataforma construida desde la perspectiva del estudiante y del docente.

Como trabajos futuros se propone agregar soporte para otros lenguajes de programación como Python, Ruby, JavaScript entre otros o utilizados a nivel nacional para enseñanza básica como SL[13]. Otro aspecto a considerar es la paralelización del módulo de corrección de tareas de forma a aprovechar mejor la infraestructura base e incluso poder distribuir en diferentes servidores para alta disponibilidad ya sea de forma local o en la nube. También se está trabajando en integrar un sistema de base de datos centralizada de tareas y casos de prueba, disponible para los docentes de las asignaturas relacionadas a programación de computadoras.

Referencias

1. Kurnia, A., Lim, A., Cheang, B.: Online judge. Computers & Education, 36(4), 299-315 (2001).
2. Aiken, A.: MOSS. https://theory.stanford.edu/~aiken/moss (2014), [Online; accedido 20-Junio-2017]
3. Bowyer, K., Hall, L.: Experience using “MOSS”to detect cheating on programmingassignments. In: In Frontiers in Education Conference on. vol. 3, pp. 13–18. IEEE (1999)
4. Chaves, J., Castro, A.: Mojo: Umaferramenta para integrar júizes online aoMoodleno apoioaoensino e aprendizagem de programação. HOLOS 5, 246–260 (2014)
5. Clough, P.: Plagiarism in natural and programming languages: an overview ofcurrent tools and technologies (2000)
6. Corporation, S.: CodeMatch. http://www.safe-corp.com/products_codematch.htm (2017), [Online; accedido 20-Junio-2017]
7. Ellis, R.K.: Field guide to learning management systems. ASTD Learning Circuitspp. 1–8 (2009)
8. Green, K.C.: A profile of the LMS market. The Campus Computing Project p. 23 (2013)
9. Kim, Y.T.: Contrast enhancement using brightness preserving bi-histogram equalization. ConsumerElectronics, IEEE Transactionson 43(1), 1–8 (1997)
10. Ltd, M.P.: About Moodle. https://docs.moodle.org/33/en/About_Moodle (2017), [Online; accessed 20-June-2017]
11. Ltd, M.P.: Assign Submission Plugins. https://docs.moodle.org/dev/Assign_ submission_plugins (2017), [Online; accessed 20-June-2017]
12. Naderi, M.: Sharif-Judge. https://github.com/mjnaderi/Sharif-Judge (2015), [Online; accessed 20-June-2017]
13. Org, D.: DOMjudge. http://www.domjudge.org (2017), [Online; accessed 20-June2017]
14. Pike, R.: Sherlock. http://www.cs.usyd.edu.au/~scilect/sherlock/ (2017), [Online; accessed 20-June-2017]
15. Prechelt, L., Malpohl, G., Philippsen, M.: Finding plagiarisms among a set of programs with JPlag. J. UCS, 8(11), 1016. (2002).

[13]http://www.cnc.una.py/sl

16. Zhigang, S.: Online Judge. https://github.com/hit-moodle/moodle-local_onlinejudge (2015), [Online; accessed 20-June-2017]
17. Zhigang, S., Xiaohong, S.: Moodle plugins for highly efficient programming courses.1st Moodle Research Conference 9, 157–163 (2012).

Estudio del rendimiento académico en Algoritmos y Estructuras de Datos

Cristian A. Martínez, Eduardo Xamena, Ismael Orozco, Diego Rodríguez, Carlos Nocera y Adriana Binda

Departamento de Informática, Universidad Nacional de Salta
Av. Bolivia 5150 Salta Capital CP 4400, Argentina
cmartinez@unsa.edu.ar, {examena, iorozco, drodriguez, cnocera, abinda}@di.unsa.edu.ar

Abstract. La acreditación de la carrera de Licenciatura en Análisis de Sistemas de la Universidad Nacional de Salta generó cambios positivos a nivel curricular, como así también en formación de posgrado, tecnológico, de infraestructura, entre otros. Entre las recomendaciones efectuadas por la CONEAU para que se mantenga dicha acreditación, se incluye las de mejorar los porcentajes de regularidad y de abandono en las asignaturas del plan de estudios vigente. En este trabajo se aplican técnicas de Redes neuronales y Arboles de decisión sobre el rendimiento académico en la asignatura Algoritmos y Estructuras de Datos, con el objeto de adquirir conocimiento que permita tomar decisiones que mejoren el rendimiento de los estudiantes.

Keywords: Acreditación, Arboles de decisión, Deserción, Redes neuronales, Regularidad.

1 Introducción

La Comisión Nacional de Evaluación y Acreditación Universitaria (CONEAU), tiene como misión asegurar y mejorar la calidad de las carreras universitarias argentinas mediante actividades de evaluación y acreditación.
En ese sentido, la carrera de Licenciatura en Análisis de Sistemas (plan 1997) de la Universidad Nacional de Salta (UNSa) fue evaluada por la CONEAU. Como resultado de dicha evaluación y a los efectos de cumplir con las recomendaciones y compromisos informados por dicha institución para alcanzar la acreditación de la misma, el Departamento de Informática junto a la Comisión de Carrera presentaron un nuevo plan de estudios (LAS plan 2010[14]). Sobre el nuevo plan de estudios, la CONEAU eleva un informe al Departamento de Informática para que subsane algunas falencias detectadas. Entre ellas, recomienda mejorar los porcentajes de regularidad, de deserción y de prolongada permanencia de los estudiantes de la carrera de grado.

La Minería de datos es un campo de Estadística y de las Ciencias de la Computación dedicado a descubrir patrones (hasta ahora desconocidos) en volúmenes de datos de gran

[14] http://bo.unsa.edu.ar/cs/R2010/R-CS-2010-0135.pdf

tamaño. El objetivo del proceso de la Minería de datos consiste en extraer información desde un conjunto de datos y transformarla para su posterior uso.
En este trabajo, nos centraremos en el rendimiento académico de los estudiantes que cursan Algoritmos y Estructuras de Datos, perteneciente al nuevo plan de estudios. Nuestra motivación consiste en detectar aspectos que afectan el cursado del estudiante. Para ello, se aplicarán técnicas de Data Mining que nos permitan obtener información aún desconocida por la Cátedra, para luego tomar decisiones a partir de un estudio científico y objetivo.
El trabajo se organiza de la siguiente manera: la sección 2 revisa algunos trabajos sobre rendimiento académico universitario. La sección 3 introduce aspectos de interés de la asignatura. En la sección 4 se describirán los datos, las técnicas utilizadas y se analizarán los resultados referidos al rendimiento académico de los estudiantes que cursan la asignatura. Por último, las conclusiones del trabajo, las acciones remediales que se están aplicando y el trabajo futuro, aparecen en la sección 5.

2 Estado del Arte sobre rendimiento académico

En las siguientes subsecciones se enumeran distintos trabajos que analizan el desempeño académico agregado a distintos niveles, en diversos ámbitos geográficos.

2.1 Estudios en instituciones extranjeras

Heredia et al. [4] intentaron predecir la deserción mediante Arboles de Decisión, en la Universidad Simón Bolívar de Colombia. Por su parte, Schumacher et al. [10] evaluaron datos sobre la deserción o finalización de estudios de posgrado en una universidad de Estados Unidos. Es destacable que los resultados para Redes Neuronales en este trabajo muestran mayor precisión predictiva, respecto a un dataset evaluado previamente con otras herramientas.
Chen y DesJardins [1] analizaron la realidad en cuanto a políticas de ayuda económica y oportunidades académicas, a nivel nacional para Estados Unidos. Tuvieron en cuenta la deserción académica, y lograron concluir que el problema de la desigualdad en EE.UU. sigue siendo muy profundo. Utilizaron la técnica de Regresión Logística.
Xenos et al. [12] estudiaron las causas de deserción en un curso a distancia de la Universidad Abierta de Grecia, mostrando que el abandono está correlacionado con el uso de recursos tecnológicos, la edad, el sexo y el conocimiento previo en informática.

2.2 Estudios en universidades nacionales

2.2.1. Estudios en la Universidad Nacional del Litoral (UNL)

Vaira et al. [11] realizaron un estudio sobre egreso y no egreso considerando 2 carreras de la UNL. En particular, analizaron la cohorte 1997, correspondiente a ingresantes de las carreras de Bioquímica y de Arquitectura.

El estudio correspondió al análisis de la información provista por el SIU y fichas de inscripción. Respecto a Bioquímica, egresaron un 22% de los alumnos (hasta diciembre del 2008). En cuanto a Arquitectura, egresaron un 34% (hasta diciembre del 2008). En Bioquímica, el nivel de escolaridad de los padres influyó en la graduación de los hijos. Respecto a Arquitectura, no se informó sobre dicho estudio. El estudiante que no egresa en Bioquímica es generalmente varón. En Arquitectura, el género no condiciona a la no graduación. Sobre los estudios secundarios, para Bioquímica no condicionó a la graduación, si provienen de escuelas públicas o privadas. Esto último no fue informado sobre los estudiantes de Arquitectura.

2.2.2 Estudios en la Universidad Nacional de Misiones (UNaM)

Kuna et al. [6] se enfocaron en analizar causas de abandono en primer y segundo año en 2003 sobre carreras de pregrado y grado de una Facultad de la UNaM.

Las variables de estudio fueron la forma de costear los estudios, cantidad de materias aprobadas y desaprobadas durante el primer año, cantidad de años entre la finalización de la secundaria y el ingreso a la Universidad, cantidad de finales desaprobados y ausentes durante el primer año, cantidad de materias regularizadas durante el primer año de estudios, título secundario y si debían viajar más de 10 km hacia la Universidad. El estudio arroja 2 resultados contundentes: el 62% de los alumnos que regularizaron 1 o ninguna materia, no tuvieron actividad en el segundo año; de ellos, el 59% que trabajaba, abandonó los estudios. A su vez, del 62% de alumnos que regularizaron 1 o ninguna materia, un 31% que costea sus estudios con el aporte de sus padres, no continuó sus estudios.

2.2.3 Estudios en la UTN Regional Resistencia (Chaco)

La Red Martínez et al. [7] realizaron un estudio sobre rendimiento académico de los estudiantes de Algoritmos y Estructuras de Datos, de la carrera Ingeniería en Sistemas de Información de la UTN Regional Resistencia.

El trabajo consistió en obtener perfiles según el rendimiento académico. Para ello, analizaron variables tales como situación laboral del estudiante, del padre, de la madre, horas dedicadas al estudio, residencia actual, lugar de residencia y dependencia de la escuela secundaria. Algunos de los resultados listados indican que el porcentaje de no-regularización es del 74,04%; los estudiantes que regularizan dedican más de 10 horas de estudio, y presentan un mejor rendimiento aquellos con madre con estudios secundarios o superior y padre con título universitario.

2.2.4 Estudios en la Universidad Nacional de Salta (UNSa)

Romero et al. [9] estudiaron la deserción en la Facultad de Ciencias Exactas de la UNSa. La información provino del SIU Guaraní y de una encuesta específica realizada a desertores de las cohortes 2005-2012.

Algunos resultados: del total de la muestra, el 48% abandonó sus estudios durante el primer año de estudios. Por otra parte, el 39% pertenece a la Licenciatura en Análisis de Sistemas y el 66% son varones. El 69% inició sus estudios antes de los 20 y el 12% eran

mayores de 25 años. El 92% era soltero y el 68% dependía económicamente de sus familiares, mientras que el 26% trabajaba. El 47% eligió su carrera por su preferencia a las Ciencias Exactas y un 28% por las expectativas laborales. El 59% tuvo una adecuada carga horaria en Matemática durante los 2 últimos años del secundario. El estudio concluye que las 2 principales causas de deserción son por problemas económicos y falta de tiempo para los estudios.

3 Algoritmos y Estructuras de Datos (AyED)

Se dicta[15] para la Licenciatura en Análisis de Sistemas (LAS) Plan 2010[16] y la Tecnicatura Universitaria en Programación (TUP)[17]. En ambas, está ubicada en el primer cuatrimestre del segundo año, con una carga semanal de 8 horas (4 horas de teoría y 4 horas de práctica). A nivel de cátedra, desde el año 2013 se compone de 1 Profesor Adjunto a cargo de clases teóricas y 2 Jefes de Trabajos Prácticos a cargo de clases prácticas, en sus respectivas comisiones. Respecto al número de inscriptos por año, en promedio 53 pertenecen a LAS y 22 a TUP.

Según las correlatividades de ambos planes, la asignatura se relaciona con:

- LAS Plan 2010
    - 1er. Año: Elementos de Programación, Matemática para Informática, Análisis Matemático I y Programación.
    - 2do. Año: Paradigmas y Lenguajes, y Teoría de la Computación II.
    - 3er. Año: Sistemas Operativos.
- TUP Plan 2011
    - 1er. Año: Elementos de Programación, Matemática para Informática, Análisis Matemático I y Programación.
    - 2do. Año: Paradigmas y Lenguajes.
    - 3er. Año: Programación de Aplicaciones Web.

Los contenidos mínimos de la asignatura son los siguientes:

Teoría de las Estructuras Discretas. Definiciones y pruebas estructurales. Teoría de Números. Aritmética modular. Estructuras de control. Recursividad. Eventos. Excepciones. Concurrencia. Tipos abstractos de datos: definiciones. Especificación abstracta. Operaciones. Isomorfismo. Contenedores lineales.

Estructuras de datos. Tipos de datos recursivos. Representación de datos en memoria. Estrategias de implementación. Manejo de memoria en ejecución. Algoritmos en grafos: Algoritmos de análisis y manipulación de grafos. Costos. Aplicación. Estructuras arbóreas: árboles generales y n-arios, binarios, balanceados. Estrategia de diseño de algoritmos.

De acuerdo a lo anterior, la asignatura tiene una fuerte base teórica en Matemática y Programación. Por otra parte, es muy fuerte la relación que tiene con asignaturas de segundo y tercer año de ambas carreras.

[15] Para la carrera de grado se dicta desde 2011, y para la de pregrado a partir de 2013.
[16] http://bo.unsa.edu.ar/cs/R2010/R-CS-2010-0135.pdf
[17] http://bo.unsa.edu.ar/cs/R2011/R-CS-2011-0596.pdf

4 Estudio del rendimiento académico

Esta sección describe el proceso de Minería de datos aplicado para obtener información de relevancia sobre datos académicos, socio-económicos de estudiantes (y su grupo de familiar) que cursan Algoritmos y Estructuras de Datos.

4.1 Dataset de estudio

Consiste en 469 registros referidos a estudiantes de LAS y TUP que cursaron la asignatura entre 2011 y 2017. Los datos provienen del SIU Guaraní (datos personales, de exámenes de asignaturas previas e información censal), de la Secretaría de Bienestar Universitario de la UNSa (datos de becas) y de la asignatura (notas de parciales). Los datos considerados para el estudio son:

- Edad al inicio de la cursada
- Sexo
- Tipo de colegio secundario en que se graduó (privado o público)
- Nivel de estudios de los padres
- Situación laboral del estudiante y de los padres
- Si cobraba alguna beca mientras cursaba la asignatura
- Condición en la materia: primer cursado o recursante
- Si aprobó la asignatura "Matemática para Informática"
- Si aprobó la asignatura "Programación"
- Si aprobó el primer parcial de la asignatura durante el cursado
- Situación al finalizar el cuatrimestre: regular, libre o abandonó

Algunos resultados a nivel univariado/bivariado: el promedio de edad es 21.2 años, de los cuales el 80% son varones. Respecto al establecimiento secundario, el 58% provienen de instituciones públicas. El 83% de los estudiantes aprobaron "Matemática para Informática", y el 66% "Programación"[18]. Del total de inscriptos, el 40% regularizó la asignatura, el 35% no regularizó y el 25% abandonó el cursado (no se presentó a alguno de los 2 exámenes recuperatorios). Finalmente, el 70% de los alumnos no trabaja, aunque el 22% admitió estar buscando trabajo. Además, el 8.5% fue beneficiario de alguna beca mientras cursaba la asignatura. Sobre sus progenitores, el 64% de los padres y el 60% de las madres completaron el secundario; sólo completaron estudios universitarios el 2.6% de los padres y el 4.7% de las madres. En cuanto a lo laboral, el 14.28% de los padres y el 31.77% de las madres están desocupados.

4.2 Preparación, limpieza y transformación del Dataset

La preparación estuvo relacionada con categorización de variables (por ejemplo, si el estudiante cobró alguna beca), obtención de nuevas variables (determinar si el establecimiento secundario de procedencia es público o no), control de inconsistencias (por ejemplo, si un estudiante informó que egresó de una determinada escuela y por alguna

[18] Según el Régimen de correlatividades de ambos planes, para cursar AyED debe tener regular Análisis Matemático (Matemática para Informática es su correlativa) y Programación.

razón tuvo que recursar, en una próxima encuesta, el establecimiento secundario de procedencia debe coincidir). Es importante mencionar que algunas variables han sido previamente categorizadas por el SIU Guaraní (por ejemplo, nivel de estudios de los padres). En cuanto a limpieza de datos, se descartaron 15 registros (del total) debido a la gran presencia de valores ausentes. Esto se debe a problemas de validación en la encuesta provista por el Sistema.

4.3 Extracción de conocimiento e interpretación de resultados

4.3.1 Aplicación de Redes Neuronales

La utilización de las Redes Neuronales para realizar predicciones sobre los datos ha sido utilizada en diferentes campos de estudio. Por ejemplo, Knowler et al. [5] emplean un MultiLayer Perceptron (MLP) para la predicción de diabetes en pacientes mujeres. En este trabajo, proponemos una arquitectura de red neuronal multicapa [3] y un entrenamiento supervisado de ADAM para actualización de los pesos. El modelo está formado por un total de 6 capas completamente conectadas. La función de activación empleada es *relu* (rectified linear unit) en las 5 primeras capas y *sigmoide* en la capa de salida para garantizar que las salidas del modelo estén comprendidas en el rango [0, 1]. A continuación, se muestra la cantidad de neuronas por cada capa del modelo propuesto.

- **Primera**: 8000 neuronas.
- **Segunda**: 90 neuronas.
- **Tercera**: 62 neuronas.
- **Cuarta**: 32 neuronas.
- **Quinta**: 10 neuronas.
- **Sexta**: 3 neuronas. Es decir, una neurona por clase.

El MLP fue implementado en Python 2.7.3, utilizando la librería Keras [2] para el manejo de redes neuronales.

Para evaluar la performance del clasificador, calcularemos la precisión definida como las muestras clasificadas correctamente divididas por el número total de muestras del dataset. Esto es:

$$Precisión = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Donde TP corresponde al número de verdaderos positivos detectados, TN a verdaderos negativos, FP a falsos positivos y FN a falsos negativos respectivamente.

Finalmente aplicamos k-fold Cross validation con k=4, dividiendo el dataset en 4 grupos de 117 muestras cada uno. La Tabla 1 muestra la precisión promedio luego de 10 corridas por cada *fold*.

Tabla 9. *k*-fold Cross validation con $k = 4$. Cada iteración usa tres fold para el entrenamiento, la restante para validación. La segunda columna muestra la Precisión por cada fold. El promedio general muestra que la precisión final del clasificador es de ~72.62% para el conjunto de validación.

k-fold	Precisión
1	72.84%
2	72.64%
3	73.02%

4	71.98%
Promedio	**72.62%**

La precisión final del clasificador es de ~72%, es decir que de 100 casos acierta en 72, un número aceptable considerando la cantidad de información recolectada hasta el momento.

4.3.2 Aplicación de Arboles de Decisión

Los árboles de decisión son estructuras lógicas con amplia utilización en la toma de decisión, la predicción y la minería de datos. Son un conjunto de condiciones o reglas organizadas de manera jerárquica de manera tal que la decisión final se puede determinar siguiendo las condiciones desde la raíz hasta algunas de las hojas.

A los efectos de predecir (con otra técnica) la situación final de cursado, se propuso aplicar árboles de decisión. Para ello, se utilizó la librería *party* [13] incluida en R versión 3.2.3 ejecutada sobre entorno Linux.

Para llevar a cabo el experimento numérico, se incluyeron todas las variables de estudio (ver sección 4.1) de manera de obtener el mejor modelo posible, usando un 70% de los datos para entrenamiento y un 30% para pruebas.

En la Fig. 1 se muestran los resultados alcanzados por árboles de decisión. De ellos, se destacan dos situaciones de interés: los estudiantes que aprobaron el primer parcial y cuya edad no supere los 23 años, tienen una probabilidad superior al 80% de regularizar la asignatura; por otra parte, los estudiantes que no aprobaron el primer parcial y no aprobaron el examen final de Programación, tienen una probabilidad superior al 70% de abandonar el cursado. Finalmente, de lo obtenido por árboles de decisión, se aprecia que la situación de cursado del estudiante es independiente del entorno familiar como así también del establecimiento secundario de procedencia.

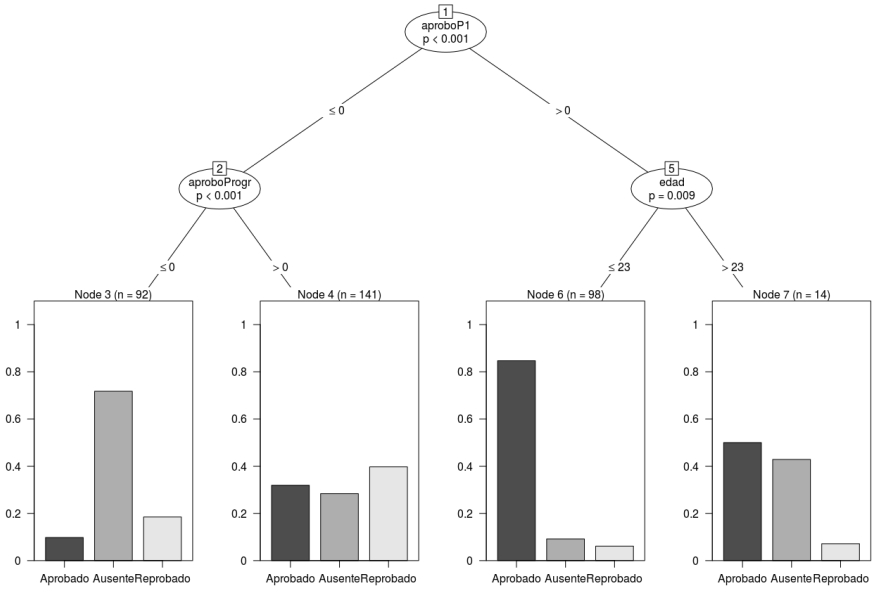


Fig. 1. Árbol de decisión obtenido sobre el dataset mediante R.

5 Conclusiones y trabajo futuro

El trabajo propuesto estuvo dedicado a conocer algunos factores que inciden en el rendimiento académico de los estudiantes que cursan la asignatura Algoritmos y Estructuras de Datos.

Los resultados obtenidos nos han permitido comenzar a planificar actividades que mejoren los porcentajes de regularidad y de abandono en la asignatura. Para el primer caso, se intensificarán las actividades de tutoría[19] desde el comienzo de clases de manera que el estudiante mejore sus conocimientos en Matemática Discreta y en Programación Orientada a Objetos. Respecto al abandono, se articulará con la Cátedra de Programación para organizar un taller de verano de manera que los estudiantes profundicen sus conocimientos en programación estructurada en C y así cursen AyED con más experiencia en programación y en resolución de problemas computacionales.

A futuro, se aplicarán Clustering Particional y Regresión Logística Multinomial para caracterizar mejor a nuestros estudiantes y por otra parte, validar los resultados alcanzados con Árboles de Decisión. Respecto a Redes Neuronales, se estudiarán otros modelos multicapas para la predicción, y se compararán arquitecturas publicadas en la bibliografía sobre temas similares. Consideramos que el estudio realizado y en específico los resultados alcanzados, pueden ser replicados en otras asignaturas de ambos planes de

[19] A partir de la acreditación de LAS, el Departamento de Informática accedió a fondos de la SPU para el mejoramiento de la carrera.

estudios y a su vez servir como referencia para la toma de decisión a nivel departamental y de Facultad.

Referencias

1. Chen, R., DesJardins, S. L. (2010). Investigating the impact of financial aid on student dropout risks: Racial and ethnic differences. The Journal of Higher Education, 81(2), 179-208.
2. Chollet, F. Keras. https://github.com/fchollet/keras. 2015.
3. Hagan, M., Demuth, H., Beale, M.: Neural Network Design. ISBN 0-534-94332-2 (1996).
4. Heredia, D., Amaya, Y., Barrientos, E. (2015). Student dropout predictive model using data mining techniques. IEEE Latin America Transactions, 13(9), 3127-3134.
5. Knowler, W., Pettitt, D., Saad, M., Bennett, P. Diabetes mellitus in the Pima indians: Incidence, risk factors and pathogenesis. Diabetes/Metabolism Reviews. 1990.
6. Kuna, H., García, R., Villatoro, F.: Identificación de causales de abandono de estudios universitarios. Uso de Procesos de explotación de información. Revista TE&ET, 5 39-44 (2009).
7. La Red Martínez, D., Giovannini, M., Pinto, N., Frisone, M., Báez, M.: Determinación de perfiles de rendimiento académico en la UTN-FrrRe. Anales del III CONAIISI. Buenos Aires (Argentina), 30--38 (2016).
8. Nghe, N., Janecek, P., Haddawy, P. (2007). A comparative analysis of techniques for predicting academic performance. In Frontiers in Education ConferenceGlobal Engineering: Knowledge Without Borders, Opportunities Without Passports. FIE 2007. 37th Annual IEEE (pp. T2G-7).
9. Romero, D., Rodríguez, S., Martínez, C.: Análisis cuantitativo y cualitativo de la deserción en la Facultad de Ciencias Exactas de la UNSa. Anales de la VI CLABES. Quito (Ecuador), 19-28 (2016).
10. Schumacher, P., Olinsky, A., Quinn, J., Smith, R. (2010). A comparison of logistic regression, neural networks, and classification trees predicting success of actuarial students. Journal of education for Business, 85(5), 258-263.
11. Vaira, S., Ricardi, P., Taborda, L., Arralde, Z., Manni, D.: Egreso y deserción universitaria: el caso de cohortes de carreras de la UNL. Perfil social, disponible en www.alfaguia.org/alfaguia/files/1320943930_5858.pdf (2010).
12. Xenos, M., Pierrakeas, C., & Pintelas, P. (2002). A survey on student dropout rates and dropout causes concerning the students in the Course of Informatics of the Hellenic Open University. Computers & Education, 39(4), 361-377.
13. Zhao, Y.: R and Data Mining: Example and Case Studies, Elsevier, 2013.

Un Simulador para la Comprensión de los Conceptos de la Macroeconomía

Francisco Ibáñez, Daniel Díaz, Sandra Oviedo, Nancy Alonso, Eduardo Ibáñez

LISI- Instituto de Informática – Dpto. de Informática FCEFyN – Universidad Nacional de San Juan
CUIM – Av. Ignacio de la Roza 590 (O), Rivadavia – J5402DCS San Juan
{fibanez,ddiaz,soviedo,nalonso,eibanez}@iinfo.unsj.edu.ar

Resumen. *Este trabajo describe el funcionamiento de un simulador de escenarios macroeconómicos desarrollado en Matalab. Con el objeto de describir el simulador se desarrolla una temática común en currícula de la enseñanza de la macroeconomía, la simulación de los efectos de una política fiscal. A lo largo del este artículo se hace referencia a una gran cantidad de videos, disponibles en la web, que complementan y profundizan los conceptos que aquí son presentados.*

Palabras claves. *Simuladores, Macroeconomía, Política Fiscal*

Introducción

En la currícula académica los simuladores han demostrado ser una herramienta poderosa [1]. En la actualidad, en la enseñanza de la economía sólo existen algunos simuladores tales como [2] del Banco Central Europeo y [3], estos se centran exclusivamente en política monetaria. También en [4] se describe un simulador de propósitos educativos en el cual ha sido diseñado específicamente para Bulgaria. En este trabajo se presenta un simulador implementado en Matalab, que implementa los modelos macroeconómicos de corto plazo basados en [5]. Se asume que en el corto plazo los salarios se mantienen constantes. El simulador usa tres mercados de referencia, el Mercado de Bienes y Servicios, el Mercado Monetario, y el Mercado Cambiario (moneda extranjera). Así, dados los valores de entrada para los tres mercados, el simulador muestra las gráficas correspondientes a los indicadores macroeconómicos, que representan los valores de las variables de salida. Por razones de espacio, en este artículo utilizamos sólo el Mercado de Bienes y Servicios, y dos indicadores macroeconómicos, el tipo de interés, y el Producto Bruto Interno (PBI). El simulador usa escenarios, cada par de valores de entrada y de salida constituye un escenario del simulador. Si los valores de entrada cambian, el simulador produce un escenario diferente, con una salida diferente. De este modo, es posible implementar una política fiscal, aumentando uno de los valores de entrada para el Mercado de Bienes y Servicios, como por ejemplo el Gasto Gubernamental, y el simulador muestra gráficamente el impacto de esta política en los indicadores macroeconómicos. En este caso, como se verá más adelante, aumentará tanto el tipo de interés como el producto bruto interno. Este artículo está organizado entorno a desarrollo de un ejemplo que permite simular los efectos de una política fiscal. A

continuación se explican los fundamentos del modelo circular de flujo de dinero para luego usar el simulador en la determinación de la curva IS. Por último, en el apartado el simulador en funcionamiento se analizan los efectos de una política fiscal.

Modelo Circular del flujo de dinero

La figura 3 representa una simplificación del Modelo Circular basado en [5]. Inicialmente se distinguen dos partes. A la izquierda están las Empresas (Firms) y a la derecha las Familias (Families). Parte de las Familias, están formadas por los trabajadores y por los empresarios. Las flechas verdes representan el flujo del dinero. La flecha grande de abajo representa el dinero que reciben las empresas por los Bienes y Servicios que les venden a las Familias. Los Bienes y Servicios van desde las Empresas a las Familias, pero no se incluye en el gráfico debido a sólo se muestra el flujo del dinero.

El Producto Bruto Interno (PBI), medido con el método del Ingreso (Income), se define como la suma de dinero que reciben las Familias. Este flujo de dinero está representado por la flecha grande de arriba que va de las Empresas a las Familias. Los trabajadores reciben dinero por el trabajo que realizan en las Empresas, y los empresarios reciben dinero en forma de ganancias. El PBI se representa mediante la variable *Y*. De este modo el dinero circula en el sentido de las agujas del reloj.

Un porcentaje del ingreso (digamos 20%) circula desde las familias a las empresas, y el resto (80%) sale de circulación, por ejemplo para ahorro. El porcentaje del ingreso que circula, denotado por la constante *c*, se denomina Propensión Marginal al Consumo.

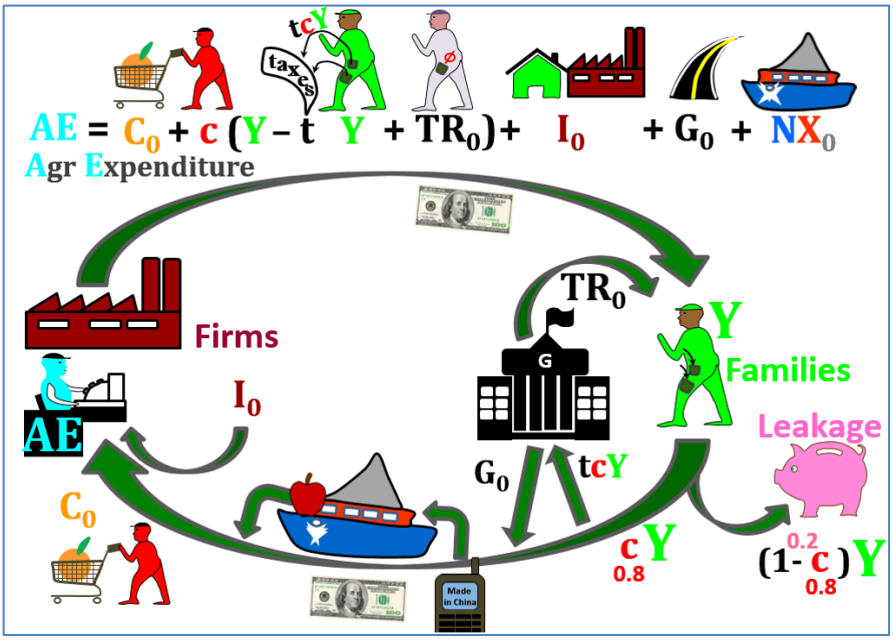


Figura 3. Modelo Circular del flujo de dinero basado en [10]

En la figura 3, el monto de dinero que circula es $c\,Y$, y está representado por la flecha que va desde las Familias hasta las Empresas, mientras que el resto del dinero que sale de circulación, igual a $(1-c)\,Y$, está representado por la flecha pequeña que va desde las familias hasta la alcancía (ahorros). La parte que sale de circulación se denomina fuga (Leakage). El dinero que circula a través de las empresas se denomina Gasto Agregado y se denota *AE* (Aggregate Expenditure). En el caso más simple en que la inversión es constante, el gasto agregado se define del siguiente modo:

$$AE = C_O + c\,Y - t\,c\,Y + c\,TR_O + I_O + G_O + NX_O$$ **Ecuación 1**

Donde C_O es el Consumo Autónomo, que constituye el gasto inicial en este flujo circular del dinero. El término algebraico $c\,Y$ representa la parte del ingreso *Y* que circula. La variable *t* representa impuestos (taxes). Por ejemplo, un valor de *t* igual a 0.3 representa un impuesto del 30% de la parte del ingreso que circula, es decir $c\,Y$. Tiene signo negativo porque sale de circulación. El gobierno inyecta dinero en el flujo circular de dos modos, a través de Transferencias (como por ejemplo subsidios), denotado por TR_O, y mediante Gasto Gubernamental, denotado por G_O. Notar que en la definición aparece $c\,TR_O$ porque solamente un porcentaje *c* de las transferencias que el gobierno otorga a las familias entra en circulación. Finalmente I_O denota Inversiones y NX_O las Exportaciones Netas (Net Export), que es la diferencia entre Exportaciones menos Importaciones. La ecuación 1 puede reordenarse del siguiente modo:

$$AE = C_O + c(Y - t\,Y + TR_O) + I_O + G_O + NX_O$$ **Ecuación 2**

La figura 3 ilustra la ecuación 2. Si las inversiones dependen del tipo de interés se obtiene la ecuación 3 que determina una expresión más general del Gasto Agregado.

$$AE = C_O + c(Y - t\,Y + TR_O) + I_O - b\,i + G_O + NX_O$$ **Ecuación 3**

Donde *i* denota el Tipo de Interés, y el coeficiente *b* denota la Sensibilidad de las Inversiones con respecto al Tipo de Interés. Por ejemplo, si b=10, un aumento del tipo de interés de 2 puntos porcentuales producirá una reducción de la inversión de 10*2 = 20. Intuitivamente, a mayor interés, menor inversión. Por esta razón, el término $b\,i$ debe tener signo negativo en ecuación 3.

Uso del simulador en la determinación de la curva IS

Asumamos los valores iniciales del panel de entradas para el mercado de Bienes y Servicios que se observa en la figura 4. Dado estos valores, el simulador debe determinar la salida, esto es, los valores de los indicadores macroeconómicos. Si el valor para i (Tipo de Interés) es conocido, la expresión 1 es constante, y por lo tanto la ordenada al origen queda determinada, como sugiere la figura 2. La pendiente es simplemente (c – c t).

$$C_O + c\,TR_O + I_O - b\,i + G_O + NX_O$$ **Expresión 1**

El mercado de Bienes y Servicios está en equilibrio cuando $AE = Y$. En otras palabras, si todo el dinero que ingresa a las empresas pasando por el cajero (*AE*) es igual al dinero que ingresa a los bolsillos de las familias (*Y*). Ver figura 3 y figura 5.

El punto importante a remarcar es que el grafico de la figura 5, sólo puede obtenerse si el valor del Tipo de Interés (i) es conocido (constante), que no es el caso inicial debido a que el Tipo de Interés es una variable Endógena, una variable que debe determinar el modelo. Para comprender cómo el simulador logra determinar el tipo de interés, utilizamos la curva IS [5], que se explica a continuación.

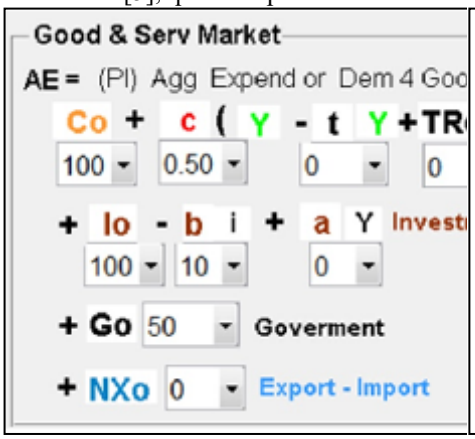


Figura 4 Panel del Merc. de B y Serv.

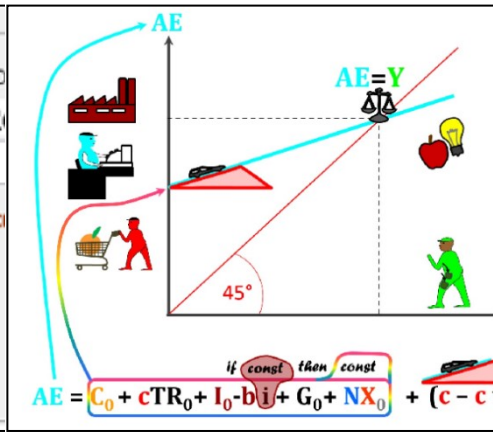


Figura 5. Equil en Mer. de B y Servicios

La ecuación 3 se puede reordenar dando origen a la ecuación 4 de la siguiente manera:

$$AE = C_O + c\,TR_O + I_O - b\,i + G_O + NX_O + (c - c\,t)\,Y \qquad \text{Ecuación 4}$$

La curva IS en la figura 6, representa todos los pares (*i, Y*), esto es, Tipos de Interés y PBI, tal que el mercado de Bienes y Servicios está en equilibrio, esto es, si $AE = Y$. Por lo tanto, reemplazando *AE* por *Y* en ecuación 4, se obtiene la ecuación 5.

$$C_O + c\,TR_O + I_O - b\,i + G_O + NX_O + (c - c\,t)\,Y = Y \qquad \text{Ecuación 5}$$

El punto clave para encontrar la curva IS, es abandonar la suposición inicial de que el Tipo de Interés (i) es constante, y considerarlo como una variable. Como sugiere la figura 6, la ecuación para la curva IS, ecuación 6, se obtiene despejando i de la ecuación 5 :

$$i = \frac{C_0 + c\,TR_0 + I_0 + G_0 + NX_0}{b} + \left(\frac{c\text{-}c\,t\text{-}1}{b}\right) Y \qquad \textbf{Ecuación 6}$$

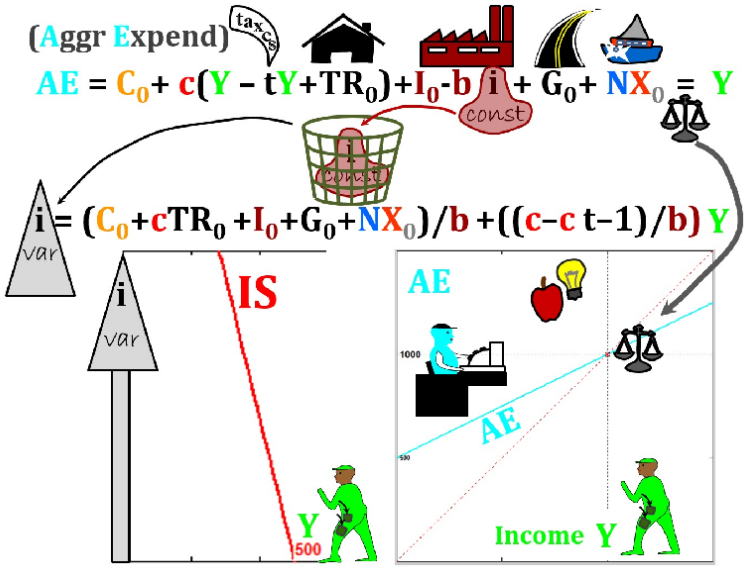


Figura 6. Curva IS

Reemplazando los valores del panel de entradas del mercado de Bienes y Servicios (ver figura 4) en la ecuación 6, se obtiene la representación gráfica de la curva IS de la figura 6. La obtención de la curva IS se puede ver en detalle en [6]. De un modo similar, el simulador determina la curva LM (recta azul en la figura 7), que representa todos los puntos en donde el Mercado Monetario está en equilibrio. El lector interesado puede encontrar la explicación detallada de la curva LM en [7]

A partir del gráfico de la figura 7, se determina el valor de i (Tipo de interés) y de Y (PBI) como la intersección de las curvas IS y LM. Debido a que la curva IS representa los pares (i, Y) donde el mercado de Bienes y Servicios está en equilibrio y la curva LM representa los pares (i, Y) en donde el Mercado Monetario está en equilibrio, el valor $i = 5$ y el valor $Y = 400$, son los valores para el Tipo de Interés y para el PBI en donde ambos mercados están en equilibrio.

En [8] se pueden ver los resultados de una primera versión del simulador. En [9] se presenta el uso de simulador para determinar el Tipo de Cambio, a partir de valores iniciales de variables exógenas. En el presente trabajo, se explican los fundamentos necesarios, para comprender los efectos de políticas macroeconómicas, mediante el uso del simulador.

La figura 8 muestra el efecto de la aplicación de una política fiscal expansiva. Si en el panel de la figura 4 se le asigna a G_O (Gasto Gubernamental) el valor 110, la nueva curva IS se desplaza paralelamente hacia la derecha como muestra la figura 8. Los valores exactos de la nueva curva IS, se obtienen cambiando el valor de G_O en la expresión (curva IS) (asignando el valor 110, en vez de 50).

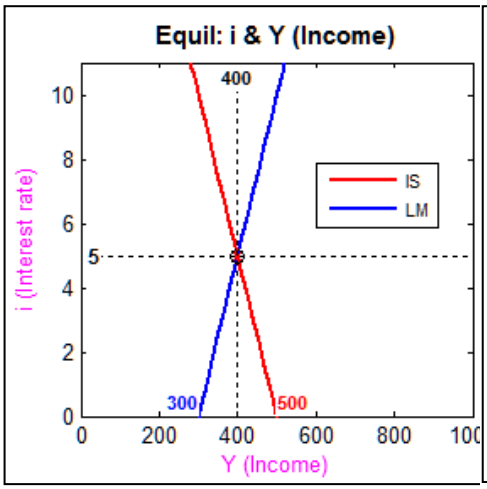


Figura 7 Curva LM

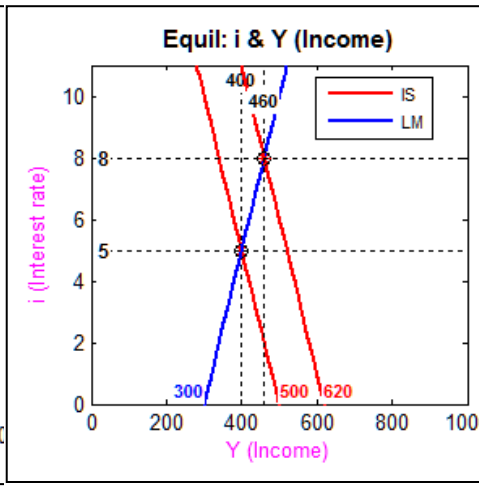


Figura 8. Efectos de una política fiscal expansiva

En [5] se explica gráfica y conceptualmente el aumento del Tipo de Interés (*i*) cuando se aplica política fiscal expansiva. En nuestro enfoque, se utiliza el simulador con dos propósitos. Por un lado, para determinar cuantitativamente el impacto de políticas macroeconómicas en función de los valores de las variables de entrada, cuyos valores se asignan en el panel de entrada, ver figura 4. Por otro lado, para ayudar a comprender la causalidad y la dinámica de los eventos, gráfica y analíticamente.

El simulador obtiene los diferentes escenarios macroeconómicos para las diferentes políticas (fiscal, monetaria, etc.) directamente, y también tiene la opción denominada "paso a paso", que muestra gráficamente toda la secuencia de eventos que conduce a la salida. En otras palabras, con la opción "paso a paso", el simulador exhibe todos los gráficos incluidos en este trabajo. Ver opción "paso a paso" en [10].

El simulador en funcionamiento

En la
figura 9, se muestra un ejemplo concreto del uso del simulador. Dados valores de entrada para los tres mercados, el simulador muestra las gráficas correspondientes a indicadores macroeconómicos, que representan los valores de la variable de salida. Cada par de valores de entrada y de salida representan un escenario del simulador.

En la figura 9, todos los recuadros de color verde representan ~~en~~ un escenario inicial, referenciado mediante escenario 1. Los valores de entrada para el escenario inicial (Escenario 1) están en el panel referenciado como "valores iniciales para escenario 1". Para estos valores de entrada, el simulador produce las gráficas referenciadas como "gráficas correspondientes al escenario 1". Entre estas gráficas, en la esquina superior izquierda se destaca con recuadro verde la gráfica referenciada con el título "Equil i & *Y*", que se explicará en detalle más adelante, y que puede observarse con mayor claridad en la

figura 7. El eje horizontal de esta grafica representa el PBI (variable *Y*), y el eje vertical representa el tipo de interés, (variable *i*). Para los valores iniciales referenciados como "valores iniciales para Escenario 1", el simulador produce, entre otras, la gráfica titulada "Equil i & Y". En la figura 7, la intersección de las rectas roja y azul, que se explican más adelante, corresponde a los valores $Y = 400,\ i = 5$. Esto significa que el simulador encuentra que el PBI (*Y*) será 400 y que el tipo de interés (*i*) será 5, para los valores iniciales correspondiente al escenario 1.

A los efectos de comprender más fácilmente la presentación, nos centraremos en una variable de entrada en particular que corresponde al gasto gubernamental representado por la variable G_O. El valor de esta variable para el escenario 1 es 50. Si los valores de entrada cambian el simulador producen un escenario diferente, con una salida diferente.

En la

figura 9, todos los recuadros de color violeta corresponden un segundo escenario (escenario 2) en el que los valores de entrada están referenciados como "valores iniciales para escenario 2", donde puede observarse que el gasto gubernamental, representado por la variable G_O, toma el valor 110. Los gráficos recuadrados con color violeta constituyen la salida que produce el simulador para estos valores de entrada. Para estos valores de entrada el simulador produce las gráficas recuadradas con color violeta. Entre estas gráficas, se destaca con recuadro violeta la gráfica referenciada con el título "Equil i & *Y*", que puede observarse con mayor claridad en la figura 8. Para los valores iniciales referenciados como "valores iniciales para escenario 2", el simulador produce un nueva grafica titulada "Equil i & Y", que se ve ampliada en la figura 8. La intersección de las rectas roja y azul, corresponde al punto $Y = 460,\ i = 8$. En otras palabras, si el gobierno adopta una Política fiscal expansiva aumentando el gasto público del valor 50 al valor 110, los indicadores macroeconómicos cambian, en particular el PBI aumenta del valor 400 al valor 460 y el Tipo de Interés aumenta el valor 5 al valor 8. De modo similar, el simulador permite observar otros indicadores macroeconómicos, como por ejemplo, el nivel de Precios, referenciado como "Aumento de P" en la

figura 9. En [11] se explica detalladamente el aumento del Nivel de Precios. En los otros gráficos de la

figura 9, están representados otros indicadores macroeconómicos, tales como nivel de inversión (explicación detallada en [7]), tipo de cambio (explicación detallada en [12]), etcétera.

Para el caso del mercado de Bienes y Servicios, el simulador permite cambiar otros valores del mercado de Bienes y Servicios (ver figura 4), como por ejemplo impuestos (t), consumo inicial (Co), etcétera. Además, el simulador permite cambiar valores de variables de entrada para los otros mercados, esto es, el mercado monetario y el mercado cambiario (ver [13] y [12]).

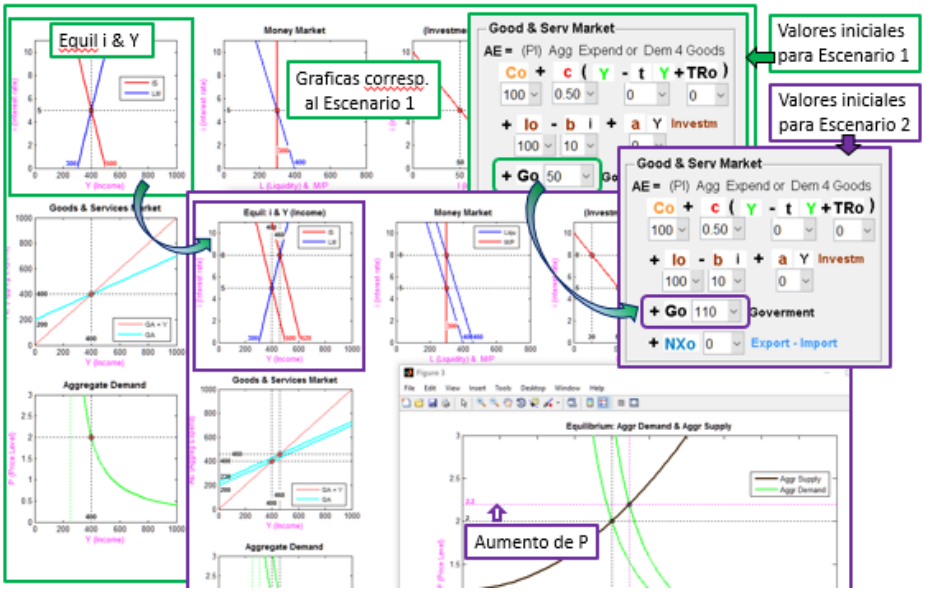


Figura 9 Incremento de Gasto Gubernamental en el Simulador

Conclusiones

Con el objeto de presentar un simulador macroeconómico se ha expuesto un modelo que permite simular los efectos de una política fiscal. Dicha temática es común en la currícula de la enseñanza de la macroeconomía. Primeramente se construyó paso a paso el modelo circular de flujo de dinero. Este modelo sirvió de base para presentar el desarrollo de la curva IS, posteriormente con la figura 2 se presentó la interface del simulador y se desarrolló sucintamente los efectos de una política fiscal expansiva. Por último, en el apartado 4 se ha intentado mostrar algunas particularidades del simulador en lo que se refiere al manejo de escenario. Es importante destacar que este artículo esta soportado por una gran cantidad de videos, disponibles en la web, que complementan y profundizan los conceptos que aquí son presentados y permiten observar al simulador en funcionamiento.

Referencias

1 Rutten, N., Van Joolingen, W.R., and Van der Veen, J.T.: 'The learning effects of computer simulations in science education', Computers & Education, 2012, 58, (1), pp. 136-153
2 €CONOMIA: 'European Central Bank €conomia game', Retrieved from https://www.ecb.europa.eu/ecb/educational/economia/html/index.en.html ; 2017
3 Presida-la-Fed: 'The Federal Reserve Bank of San Francisco. The chair the Fed game ', Retrieved from http://sffed-education.org/chairthefed/default.html. , 2017
4 Angelov, A.V., Aleksandar: 'A Simulator for Teaching Macroeconomics at Undergraduate Level', The economics network, https://economicsnetwork.ac.uk/showcase/angelov_simulator 2016
5 Blanchard Oliver, J.D.R.: 'Macroeconomics' (Pearson, 2013. 2013)
6 Ibáñez, F.: 'Lista de Reproducción: "The IS Curve". Video: " The IS Curve" (Parte 2 de 4) ', https://goo.gl/cQfGef, 2016a
7 Ibáñez, F.: 'Lista de Reproducción: "IS-LM Model". Video: "The Money Market in the Simulator" (Parte 3 de 3)', https://www.youtube.com/watch?v=CbJAayQNRJs&list=PLVBZ5QpEt37yheXumMScvZ8iU_kFMB8uL&index=3, 2016b
8 Ibáñez, F.: 'Propuesta de un proyecto interdisciplinario para construir un simulador macroeconómico', JATIC 2015, 2015
9 Ibáñez, F.S., Díaz Araya, D., Oviedo, S., Alonso, N., and Saffe, F.: 'Técnicas provenientes de las ciencias de la computación aplicadas a la simulación macroeconómica', in Editor (Ed.)^(Eds.): 'Book Técnicas provenientes de las ciencias de la computación aplicadas a la simulación macroeconómica' (2016c, edn.), pp.
10 Ibáñez, F.: 'Macroeconomics Short Run Overview', https://www.youtube.com/watch?v=ju8UMSpsnDc, 2016d
11 Ibáñez, F.: 'Lista de Reproducción: "Fiscal Policy and Policy Mix". Video: "Simulation of Expansionary Fiscal Policy When P increases " (Parte 2 de 4)', https://www.youtube.com/watch?v=WxqTx8DdaVw, 2016e
12 Ibáñez, F.: 'Lista de Reproducción: "Foreign Market". Video: "The Foreign Exchange Market " (Parte 1 de 2)', https://www.youtube.com/watch?v=6AMYVZFbkMY&index=1&list=PLVBZ5QpEt37x4ckX7RbZjnvtavKBScTWY 2016f
13 Ibáñez, F.: 'Lista de Reproducción: IS-LM Model. Video: "LM Curve Equations" (Parte 1 de 5),' https://www.youtube.com/watch?v=45885HyEbU4, 2016g

Maximización Del Throughput En Una Radio Cognitiva Implementando Los Protocolos de Enrutamiento DSR Y WCETT Sobre MACNG En CRAHNs En Un Estado de Alerta

D. León, G.A. Puerta, E. Aguirre y M. Quevedo

Resumen.El presente documento muestra un estudio comparativo de la métrica de enrutamiento para los protocolos: WCETT (Tiempo Promedio Acumulado de Transmisión Esperado) y DSR (Enrutamiento Dinámico desde el Origen), teniendo como técnica de acceso al medio MACNG (Control de Acceso al Medio de Nueva Generación), orientado a la utilización y maximización del throughput en redes de radio cognitiva sin infraestructura (CRAHNs) en estados de alerta. Los escenarios de trabajo se implementaron en el software de simulación NS2 (Network Simulator 2), donde se evaluó como medida de desempeño de los protocolos: El throughput promedio, los paquetes enviados y los paquetes recibidos, permitiendo concluir como el protocolo DSR en promedio debe recibir más paquetes que los que puede transmitir, para establecer un enlace de radio cognitiva, así como WCETT presenta un comportamiento uniforme en esta medida, permitiendo considerarlo como un protocolo más estable teniendo en cuenta que WCETT es un protocolo que realiza un trabajo más robusto en cuanto al manejo de los múltiple canales, el aprovechamiento de recursos y selección de rutas efectivas entre extremos con posibilidad de ser implementado en estado de alerta sin acceso a redes de infraestructura.

Keywords: Radio cognitiva, Estado de alerta, enrutamiento, control de acceso al medio, NS2, throughput, CRAHNs, WCETT, DSR, MACNG.

1. Introducción

En la actualidad la gran mayoría de redes están reguladas por políticas de espectro fijo las cuales son asignadas a distintas empresas y usuarios, donde las aplicaciones que se desarrollan son propietarias y definen estándares propios para el uso y apropiación de tecnologías, varios estudios han demostrado que el uso de gran parte del espectro esta subutilizado [13], por lo que el uso de las redes móviles y la reutilización del espectro licenciado actualmente son temáticas con amplia proyección en la investigación de las telecomunicaciones. La relevancia de este tipo de redes radica en su gran versatilidad [2], teniendo en cuenta las potenciales aplicaciones generadas por la masificación de equipos móviles y la portabilidad de éstos. El aporte que se ha generado está dado por una serie de protocolos de enrutamiento que han enfocado esfuerzos para mejorar el rendimiento de las redes inalámbricas, posibilitando el diseño de estrategias que permitan reutilizar y sincronizar [13], de manera oportunista sectores del espectro licenciado teniendo en cuenta el control de acceso al medio (MAC) [6 -15]. De manera paralela los recursos informáticos de simulación como NS2, permiten crear propuestas de escenarios dónde su configuración y los resultados son susceptibles de discusión académica.

En los estudios revisados se encontró el análisis por medio de simulación entre la comparación del tradicional DSR y el DSR extendido a través del simulador GloMoSim [1]. Así mismo en [7] se evalúa el rendimiento del protocolo de enrutamiento de fuentes dinámicas (RM-DSR) con diferentes tamaños de red. Aunque en algunos estudios se evidencio la evaluación del desempeño de los protocolos, no se encontró uno que haga la evaluación frente a su posible uso en estado de alerta donde se encuentre ausente la infraestructura de comunicaciones, pieza fundamental de este trabajo.

2. Protocolos

Las redes ad hoc de radio cognitiva (CRAHNs), contienen un conjunto de protocolos que han venido evolucionado, en este experimento en particular se observaron tres protocolos, debido a la naturaleza de las comunicaciones inalámbricas, el acceso al medio presenta el primer reto al determinar la forma de los parámetros funcionales para acceder al medio, así mismo como la dinámica del enrutamiento que permitan tener métricas adecuadas para la transmisión de información, en la figura 1 se describen estos tres elementos metodológicos del problema de investigación para determinar el comportamiento de protocolos.

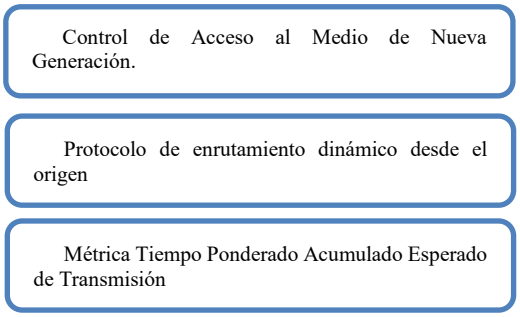


Figura 10. Tres elementos metodológicos del problema de investigación.

2.1 Control de Acceso al Medio de Nueva Generación.

El Control de Acceso al Medio de Nueva Generación (MACNG), se prueba bajo la funcionalidad de un radio que presenta el manejo de múltiples canales de acceso, vuelve operativo su funcionamiento en dos fases:

1. En la primera fase cada nodo envía paquetes de establecimiento con el canal de recepción, para calificar como canal de recepción hay que tener condición de canal preferido, el cual se adquiere al encontrarse disponible, en caso contrario el nodo comparte el canal con el nodo que se encuentre más lejos de él figura
2. En la segunda fase el nodo utiliza el canal seleccionado en la primera fase, para enviar y recibir datos y en la medida que el número de canales aumenta, adicional a esto el rendimiento se incrementa [3].

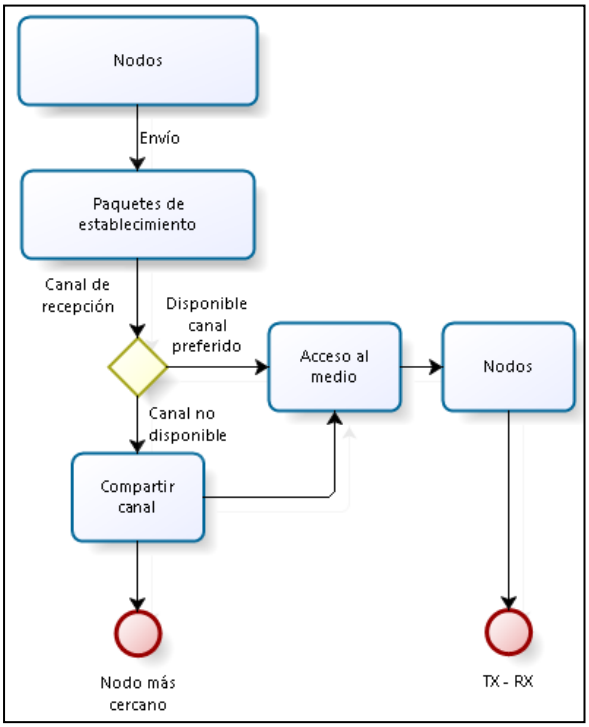


Figura 11. Establecimiento comunicaciones para control de acceso al medio

La movilidad de los nodos que conforman la red y la aparición y desaparición de los mismos modifican los posibles enlaces que se pueden establecer. Los protocolos clásicos de enrutamiento no están preparados para adaptarse a escenarios tan variantes.

En [10], se presenta la implementación de las características de acceso de MACNG para escenarios de CRAHNs desarrollado en la plataforma de simulación NS2. Para su respectivo estudio se implementa diversos protocolos de enrutamiento y acceso al medio, teniendo como características relevantes, el uso de múltiples canales y el análisis de diversos tipos de tráfico, en intervalos de tiempo definidos por el usuario, simulando la ausencia de infraestructura de telecomunicaciones clásica de un estado de alerta.

2.2 Protocolo de enrutamiento dinámico desde el origen.

En el protocolo Enrutamiento Dinámico desde el Origen (DSR), cuando se desea enviar un paquete de información desde el nodo origen a otro, el nodo origen decide

sobre la existencia o no de una ruta específica, luego de analizar los caminos y métricas posibles. La ruta elegida se incluye dentro de la cabecera del paquete de datos como un número de identificación único, por su parte los nodos intermedios hacen las veces de puente, reenviando el paquete al nodo siguiente. Esta característica limita el número máximo de saltos a realizar, pero disminuye el procesamiento requerido en los nodos intermedios [18]. DSR posee un mecanismo opcional que permite obtener información de enrutamiento observando el tráfico de datos. Este mecanismo se denomina *overhearing* y permite a los nodos conocer mejor la topología actual de la red y descubrir caminos más cortos que los utilizados en ese momento sin generar tráfico adicional de enrutamiento. Este mecanismo requiere procesamiento adicional en los nodos intermedios, pero ayuda al descubrimiento de mejores rutas y a reaccionar ante fallas en los enlaces [9].

Varias propuestas de aplicación del protocolo DSR han sido diseñadas para CRAHNs por ejemplo en [5] se soluciona el problema de asignación de canales teniendo en cuenta el parámetro de estimación SINR (Relación señal a ruido más interferencia) en una ruta elegida. De la información suministrada por DSR se hace uso de la variable de ruta (RREP) para estimar la SINR y la velocidad de datos máxima de transmisión de los nodos de la ruta elegida. De esta manera, el nodo fuente puede llevar a cabo la asignación de canal de una manera más eficiente.

Por su parte se presenta la arquitectura propuesta en [19], donde se implementa un receptor de banda ancha para RC. DSR se usa para contribuir en la detección eficiente del espectro y ofrece una mejoría en la exactitud de los datos recogidos, permitiendo optimizar el proceso de toma de decisión del canal con mejor rendimiento.

2.3 Métrica tiempo ponderado acumulado esperado de transmisión.

La métrica Tiempo Ponderado Acumulado Esperado de Transmisión (WCETT) permite determinar el camino o ruta que presenta las mejores condiciones para la transmisión de información entre un origen y un destino [14].

WCETT asigna valores a cada nodo mediante la variable temporal ETT (Tiempo Esperado de Transmisión), que es una relación entre el ancho de banda y la tasa de pérdida del canal entre el origen y destino. Esta variable cuyo comportamiento varía teniendo en cuenta los enlaces y las modificaciones que sufre el escenario[12].

La estimación de WCETT se presenta en la siguiente ecuación (1) [16]:

$$WCETT = (1 - \beta) * \Sigma\, ETT + \beta * \max x_j \quad (1)$$

Donde la sumatoria de los ETT representa el consumo total de recursos temporales para una transmisión entre extremos de la ruta, ß es un parámetro ajustable que se encuentra entre 0 y 1 [17]. Dependiendo de la cantidad de saltos que están al servicio de la transmisión requerida para enviar la información.

El trabajo implementado en [8], presenta una comparación de las métricas WCETT y AODV en términos de rendimiento y retardo entre extremos de una ruta. El algoritmo MISD (Interferencia Mínima Acumulada y Canal de Retardo Conmutado) tiene como función realizar un trabajo cooperativo entre capas (capa de red, MAC y física) donde se optimiza el retardo y el número de saltos.

3. Desarrollo del estudio comparativo

Para desarrollar la propuesta comparativa de trabajo, se propuso un escenario específico con 10 nodos distribuidos aleatoriamente, ubicados de manera estática durante un periodo de 500 segundos. Simulando un posible escenario de estado de alerta donde cada uno de los nodos pueden ser personas.

Se utilizó el programa de simulación NS2, el cual permitió emular las características de acceso a la capa MAC implementada con MACNG y la evaluación del rendimiento en el establecimiento y enrutamiento de tráfico en los protocolos DSR Y WCETT. No se tuvieron en cuenta otros simuladores, debido a la facilidad de NS2, para la implementación de nuevos protocolos y el análisis de las trazas en cada una de las capas.

El escenario de simulación se presenta de forma general en la Tabla 1:

Parámetro	Descripción
Tipo de canal	WirelessChannel
Tipo de propagación	TwoRayGround
Tipo de interfaz	Física /Wireless
Tipo de antena	OmniAntenna
Tipo de enrutamiento	DSR y WCETT
Tipo de encolamiento	DropTail - Prioridad de encolamiento
Tamaño de la cola	50 paquetes
Tipo de acceso al medio	Mac/Macng

Tabla 1. Parámetros del escenario de simulación.

La distribución geográfica de los nodos se hace de forma aleatoria donde es posible observar de manera animada la actividad de cada nodo a lo largo de los 500 segundos que tarda la simulación, como se presenta en la Figura 3.

Figura 12. Escenario de simulación radiando con 10 nodos fijos con posición inicial aleatoria.

4. Discusión de resultados

En el archivo de traza generado en NS2 se evidencio que se simula el establecimiento del protocolo en un escenario de radio cognitiva, más no el comportamiento del protocolo de enrutamiento frente a algún tipo de tráfico específico, este análisis permitió evaluar de manera comparativa el comportamiento que presenta el throughput promedio de cada protocolo en el establecimiento, los resultados obtenidos se muestran en la Figura 4.

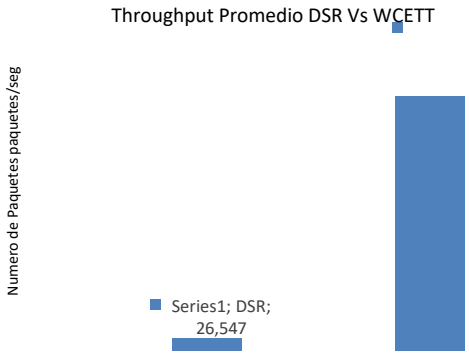


Figura 13. Comparativo throughput promedio de red

Por otra parte, la Figura 5 muestra el comportamiento promedio de las variables: paquetes recibidos y paquetes enviados por unidad de tiempo, en los protocolos DSR y WCETT, para el establecimiento de un enlace de CRAHNs.

Cuantificar y comparar dichas variables posibilita el análisis de la estabilidad de los protocolos respecto de los paquetes por unidad de tiempo.

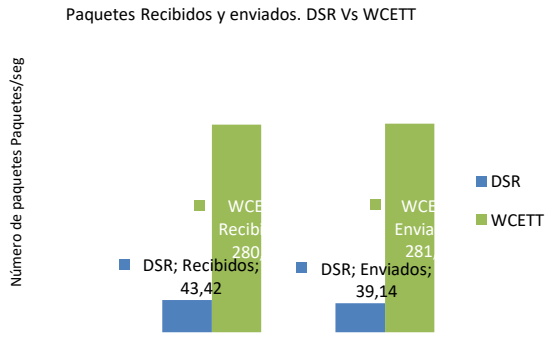


Figura 14. Comparativo paquetes recibidos y enviados.

5. Conclusiones y trabajos futuros

Al comparar los protocolos DSR y WCETT se observa como el protocolo WCETT presenta un mayor valor en la magnitud en la maximización del throughput promedio de todos los nodos, lo cual podría permitir a un usuario que se encuentra en una situación o estado d alerta enviar de una manera efectiva información relevante de su posición y estado. Hay que tener en cuenta que, entre los protocolos en cuestión, WCETT es un protocolo que requiere mayor cantidad de procesamiento en el cálculo de rutas y métricas óptimas, ya que es un protocolo de enrutamiento desarrollado para redes radio cognitiva (CRAHNs). Adicionalmente debido a su naturaleza es un protocolo multicanal por lo cual debe realizar este proceso de manera redundante por cada canal.

Por su parte DSR es un protocolo diseñado para redes básicas inalámbricas evaluado en un entorno de radio cognitiva, como una posible situación de alerta o alerta, mostrando una maximización del throughput baja en magnitud con respecto al otro protocolo evaluado, ya que el protocolo solamente soporta un canal. Cabe destacar que las aplicaciones de múltiple radio ofrecen un mejor throughout de desempeño del enrutamiento.

14. El estudio de los protocolos DSR y WCETT en términos de la cantidad de paquetes recibidos y enviados, permite concluir como DSR en promedio debe recibir más paquetes, que los que puede transmitir para establecer un enlace de radio cognitiva, siendo poco efectivo para el establecimiento de enlaces sin infraestructura en un estado de alerta o alerta. Por su parte WCETT presenta un comportamiento uniforme en esta medida, permitiendo considerarlo como un protocolo más estable teniendo en cuenta que WCETT es un protocolo que realiza un trabajo más robusto en cuanto al manejo de múltiples canales, aprovechamiento de recursos y selección de rutas efectivas entre extremos, de esta forma postulándose como un candidato para llevar paquetes de información, como geolocalización de personas en situación o estado de alerta o de emergencia.
15.
16. La pertinencia de este tipo de trabajos se sustenta en el aprovechamiento de las herramientas informáticas, que posibilitan medir y cuantificar variables respecto del desempeño de los protocolos en escenarios controlados. De tal forma, que le ofrecen al investigador ilimitadas posibilidades de trabajo que van desde el estudio comparativo de protocolos, hasta el diseño y posterior evaluación de los mismos, la posible aplicación a escenarios de distinto tipo de operación y relación.

Como trabajo futuro se propone el desarrollo de una plantilla que permita evaluar de manera sistematizada diferentes tipos de variables como: interferencia, jitter, rutas eficientes por canal y por nodos, entre otros. Permitiendo al investigador realizar propuestas de trabajo que evalúe de manera eficiente el desempeño de los protocolos

en estados de alerta o emergencia sin ningún tipo de infraestructura de telecomunicaciones.

6. Referencias

[1] Ahmad, S., Awan, I., Waqqas, A., & Ahmad, B. (2008). Performance Analysis of DSR & Extended DSR Protocols. In *2008 Second Asia International Conference on Modelling & Simulation (AMS)* (pp. 191–196). IEEE. http://doi.org/10.1109/AMS.2008.72

[2] Arias, Y., & Cumanda, P. (2010). Estudio de los canales con desvanecimiento sobre redes fijas y móviles en sistemas de radio comunicación. Retrieved from http://bibdigital.epn.edu.ec/handle/15000/1455

[3] Bian, K., Park, J.-M., & Gao, B. (2014). *Cognitive Radio Networks*. Cham: Springer International Publishing. http://doi.org/10.1007/978-3-319-07329-3

[4] Charles E. Perkins. (2008). *Ad Hoc Networking*. Retrieved from http://dl.acm.org/citation.cfm?id=1481270

[5] Dai, Y., & Wu, J. (2011). Efficient Channel Assignment under Dynamic Source Routing in Cognitive Radio Networks. In *2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems* (pp. 550–559). IEEE. http://doi.org/10.1109/MASS.2011.58

[6] Haykin, S. (2005). Cognitive radio: brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*, *23*(2), 201–220. http://doi.org/10.1109/JSAC.2004.839380

[7] Husieen, N. A., Hassan, S., Ghazali, O., & Siregar, L. (2013). The Robustness of RM-DSR Multipath Routing Protocol with Different Network Size in MANET. *International Journal of Mobile Computing and Multimedia Communications*, *5*(2), 46–57. http://doi.org/10.4018/jmcmc.2013040104

[8] Jiao Wang, & Yuqing Huang. (2010). A cross-layer design of channel assignment and routing in Cognitive Radio Networks. In *2010 3rd International Conference on Computer Science and Information Technology* (pp. 542–547). IEEE. http://doi.org/10.1109/ICCSIT.2010.5564800

[9] Johnson, David B. Maltz, Dave. Broch, J. (2001). DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. *Ad Hoc Networking*.

[10] Ljialian. (2014). MACNG for escanaries CR. Retrieved from http://stuweb.ee.mtu.edu/~ljialian/

[11] Luis Fernando Pedraza, Felipe Forero, I. P. P. (2014). Evaluación de ocupación del espectro radioeléctrico en Bogotá-Colombia. *Ingeniería Y Ciencia*.

[12] Ma, L., & Denko, M. K. (2007). A Routing Metric for Load-Balancing in Wireless Mesh Networks. In *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)* (pp. 409–414). IEEE. http://doi.org/10.1109/AINAW.2007.50

[13] Puerta, G., Aguirre, E. and Alzate, M. (2010). Effects of Topology and Mobility in Bio-Inspired Synchronization of Mobile Ad Hoc Networks. *EEE Latincom*. http://doi.org/10.1109/LATINCOM.2010.5640976

[14] Renu, B., lal, M. H., & Pranavi, T. (2013). Routing Protocols in Mobile Ad-Hoc Network: A Review (pp. 52–60). Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-642-37949-9_5

[15] Wang, H., Qin, H., & Zhu, L. (2008). A Survey on MAC Protocols for Opportunistic Spectrum Access in Cognitive Radio Networks. In *2008 International Conference on Computer Science and Software Engineering* (pp. 214–218). IEEE. http://doi.org/10.1109/CSSE.2008.1546

[16] Xiao, Y. (2008). *Cognitive Radio Networks.*Taylor & Francis.
[17] Yau, A. K.-L., Komisarczuk, P., & Teal, P. D. (2008). On Multi-Channel MAC Protocols in Cognitive Radio Networks. In *2008 Australasian Telecommunication Networks and Applications Conference* (pp. 300–305). IEEE. http://doi.org/10.1109/ATNAC.2008.4783340
[18] Yinfei Pan. (2008). Design Routing Protocol Performance Comparison in NS2: AODV comparing to DSR as Example. Retrieved March 2, 2017, from https://www.researchgate.net/publication/241888201_Design_Routing_Protocol_Performance_Comparison_in_NS2_AODV_comparing_to_DSR_as_Example
[19] Zamat, H., & Natarajan, B. (2009). Practical architecture of a broadband sensing receiver for use in cognitive radio. *Physical Communication*, *2*(1–2), 87–102. http://doi.org/10.1016/j.phycom.2009.02.005.

CICCSI 2017
Mendoza 13 al 17 de noviembre de 2017

Buenas Prácticas en la Implementación del Gobierno de las Tecnologías de la Información en universidades

Alex Armando Torres Bermúdez[1]
Antonio Fernández Martínez[2]

[1]Corporación Universitaria Comfacauca (Colombia)
atorres@unicomfacauca.edu.co
[2]Universidad de Almería (España)
afm@ual.es

Resumen. Un buen sistema de Gobierno de TI en una universidad concebirá a las TI no solo como elementos tácticos, sino que hará parte de la planificación global de la universidad generando una nueva cultura donde se podrá extraer de las TI el máximo valor posible para la universidad. Este artículo presenta un proyecto de arranque en la implementación del gobierno de las TI en universidades de Colombia tomando como caso piloto la Corporación Universitaria Comfacauca donde se socializará los resultados obtenidos en este proceso. En primera instancia se describe el proyecto al lado de una metodología propuesta aplicada a la universidad, se presentan las evidencias de buenas prácticas relacionadas con el Gobierno de las TI, seguidamente se identifica la madurez inicial y objetivos de mejora del Gobierno de las TI y finaliza presentando un Plan de mejora del Gobierno de las TI.

Palabras Clave: Madurez, TI, COBIT, Gobierno de TI, GTI4U, PAGTI

1 Introducción

La gestión de las Tecnologías de la Información (TI) en las Instituciones de Educación Superior se ha centrado hasta ahora en lograr una administración eficiente de los recursos tecnológicos como soporte fundamental del resto de servicios universitarios.

Sin embargo, no convendría concebir las TI sólo como elementos tácticos de las universidades, no deberían gestionarse verticalmente o planificarse de manera aislada, sino que tendrían que formar parte de la planificación global de la universidad, pues tienen un carácter estratégico y horizontal. Sólo de esta manera se alcanzará la máxima eficiencia y se podrá extraer de las TI el máximo valor posible para la universidad.

Actualmente, los sistemas de gobierno de las TI se encuentran implantados con éxito en otros sectores (banca, seguros, industria, etc.) alcanzando una madurez de 2,33 sobre 5 en la escala propuesta por el IT Governance Institute (ITGI, 2011). También se están incorporando al gobierno de las TI universidades de todo el mundo, y según el estudio realizado por Yanosky y Borreson (2008) ya alcanzan una madurez de 2,30 sobre 5, lo

que significa que las universidades se encuentran todavía en una situación incipiente y en proceso de maduración pero cerca de la media global.

En consecuencia el establecimiento de un buen sistema de gobierno (gobernanza) de las TI significa, entre otras cosas, que las universidades lleven a cabo una planificación estratégica e integral de las tecnologías de la información de manera alineada con los objetivos globales de la organización. Para ello, las principales responsabilidades relacionadas con el gobierno corporativo de las TI deben recaer y ser apoyadas directamente por la más alta dirección universitaria (rector, gerente y vicerrectores).

1.1 Proyectos de colaboración interinstitucional

Conscientes de la pertinencia de implementar un buen gobierno de TI en las Instituciones de Educación Superior en Colombia, la Corporación Universitaria Comfacauca ubicada en el departamento del Cauca a través de su grupo TIC-UNICOMFACAUCA ha venido trabajando desde su línea de investigación "Gobierno y Gestión de TI", el modelo de gestión y gobierno de TI descrito en el apartado anterior. Este proyecto permitió establecer un convenio de colaboración con la Universidad de Almería (España) donde investigadores expertos de ambas instituciones iniciaron una experiencia piloto de gobernanza de TI en la Corporación Universitaria Comfacauca UNICOMFACAUCA donde el principal objetivo es llevar a cabo una evaluación de la situación actual de madurez del gobierno de las TI en esta universidad.

1.2 Propuesta Metodológica de Implementación del Gobierno de las TI

En junio de 2009 la Comisión Sectorial TIC de la Conferencia de Rectores de las Universidades Españolas (CRUE) decide lanzar un Proyecto de Arranque para la implantación del modelo GTI4U (Gobierno de las TI para Universidades) en el Sistema Universitario Español (SUE).

El Modelo GTI4U ha sido elaborado por Fernández (2011) en colaboración con el Grupo de Análisis, Planificación y Gobierno de las TI de la Sectorial TIC de la CRUE, en base a la norma internacional ISO/IEC 38500 (2008) "Corporate Governance of IT". Actualmente el PAGTI ha sido implantado ya en 10 universidades españolas.

En este sentido como resultado del convenio para proyectos de colaboración interinstitucional descrito en el apartado 1.2 se implementa una propuesta metodológica de implementación del gobierno de las TI para las universidades mediante el modelo GTI4U. La validación de la propuesta metodológica se realizará en el dominio o sector de la educación superior en Colombia; se presenta un caso de estudio en la Corporación Universitaria Comfacauca.

La propuesta metodológica describe cómo llevar a cabo un proyecto piloto del gobierno de las TI mediante GTI4U en la Corporación Universitaria Comfacauca donde

el principal objetivo fue llevar a cabo una evaluación de la situación actual de madurez del gobierno de las TI en la Corporación. La implementación de dicha propuesta dio inicio a un Proyecto Piloto del gobierno de las TI en Unicomfacauca, el cual consta de una serie de etapas que darán que permitirá identificar y socializar las evidencias de buenas prácticas relacionadas con el Gobierno de las TI, también se presentar la madurez inicial y objetivos de mejora del gobierno de las TI y finalmente mostrar un plan de mejora del gobierno de las TI y unas acciones de mejora, debidamente priorizadas y planificadas para la Corporación Universitaria Comfacauca permitiendo de esta manera llevar a cabo la implantación de un buen gobierno de TI, con el fin de llevar a cabo una planificación estratégica e integral de las tecnologías de la información de manera alineada con los objetivos globales de la Corporación.

Esta producción como memoria de resultados de investigación no intenta usurpar la independencia de la que disfruta cada universidad para seleccionar el modelo de gobierno de las TI que desee, sino poner a disposición de las Instituciones de Educación Superior en Colombia, una serie de herramientas que le faciliten dicha implantación si así lo estimaran conveniente.

Sea cual sea el camino elegido, los autores desean que las Instituciones de Educación Superior en Colombia dispongan de sistemas de gobierno de las TI maduros que les permitan extraer el máximo valor a sus TI, al mismo tiempo que las sitúe con cierta ventaja competitiva en relación a las universidades de su entorno.

2 Proyecto de Arranque del Gobierno de las TI

La propuesta metodológica de implementación del Gobierno de las Tecnologías de la Información se realizará en la Corporación Universitaria de Comfacauca como caso piloto para las universidades en Colombia.

Inicialmente este proyecto tiene como objetivo principal llevar a cabo una evaluación de la situación actual de madurez del gobierno de las TI en la Corporación para luego completar el proceso de implantación de su sistema de gobierno de las TI.

Los resultados de este proyecto disponen de una herramienta de referencia a la hora de implantar sus sistemas de gobierno de las TI. Pero además, cada universidad participante en el Proyecto de Arranque del Gobierno de las TI (PAGTI) obtendrá los siguientes beneficios:

- El Equipo de Gobierno de la Universidad conoce mejor la situación integral de las TI en su universidad.
- El Equipo de Gobierno puede potenciar el valor que proporcionan las TI a su universidad aplicando mejores políticas, procedimientos y cambios organizativos.

- Ahora son capaces de definir fácilmente objetivos TI a medio plazo.
- Disponen de un catálogo de acciones de mejora cuya ejecución va a satisfacer dichos objetivos de TI.
- A partir de ahora, podrán compararse con otras universidades y situar su nivel de madurez de gobierno TI en relación a la ISO 38500.
- Todo ello va a transmitir una imagen más moderna e innovadora de la universidad.

2.1 Etapas del Proyecto de Arranque de gobierno de las TI

Para la realización del Proyecto de Arranque del Gobierno de las TI en Unicomfacauca se definió una propuesta metodológica de implementación del Gobierno de las Tecnologías de la Información mediante el modelo GTI4U (Figura 1) como una experiencia piloto que sirva de referente para las Instituciones de Educación Superior en Colombia.

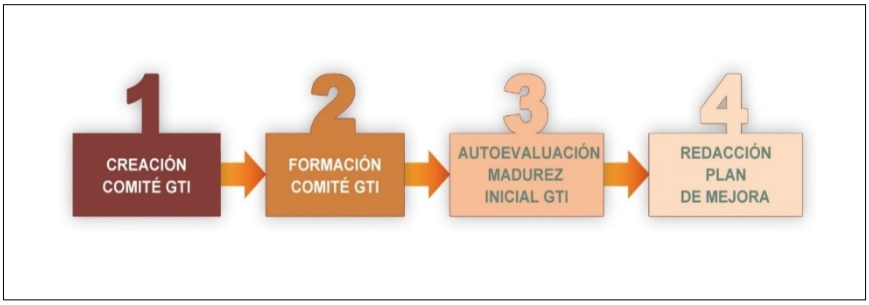


Fig.1. Etapas del Proyecto de Arranque de Gobierno de las TI
Antonio Fernández a partir de Van Grembergen y De Haes (2008)

A continuación se describe cada una de las siguientes etapas que se realizaron en la Corporación Universitaria Comfacauca UNICOMFACAUCA:

a. Creación del Comité de Gobierno de las TI (CGTI): responsable de llevar a cabo el Proyecto de Arranque de Gobierno de las TI (PAGTI) en Unicomfacauca.

Para la ISO 38500 (2008) el Gobierno de las TI es responsabilidad de la más alta dirección de una organización. En el caso de las universidades, recomendamos que esta responsabilidad recaiga en el Equipo de Gobierno, en especial en el rector que debe ser el principal promotor y líder de esta iniciativa.
Se recomienda que el Comité GTI esté compuesto por:

- El Vicerrector (o cargo equivalente) responsable directo de la planificación estratégica de la universidad.
- El Gerente o Vicerrector responsable de la planificación económica.
- Todos los vicerrectores que tengan responsabilidad estratégica sobre las TI.

- Director del Área de TI.
- Otros directivos universitarios que tengan responsabilidad estratégica sobre las TI
- Varios responsables de los principales Servicios Universitarios: Biblioteca, Personal, Asuntos Económicos, etc.

El Comité GTI debe estar formado por un mínimo de 6 y un máximo de 12 miembros. Para facilitar la gestión del Comité y facilitar la coordinación con los investigadores se recomendó nombrar a un presidente y a un secretario del Comité GTI. El presidente debe ser nombrado de entre los miembros del Comité GTI y no sólo se ocupará de velar por el cumplimiento de la planificación y el éxito del Proyecto de Arranque de Gobierno de las TI (PAGTI) sino que además será el líder de dicho proyecto y su principal valedor en el Equipo de Gobierno.

Aunque este comité se crea para llevar a cabo un proceso determinado y finito, la evaluación de la madurez de gobierno de las TI, una vez ya finalizado este proyecto, se dio continuidad convirtiéndolo en un Comité de Estrategia de las TI.

Este nuevo comité es el responsable de diseñar la estrategia relacionada con las TI de UNICOMFACAUCA. La experiencia y la alta responsabilidad de los miembros de este comité contribuye a llevar a cabo su principal función: alinear la planificación de las TI con las necesidades estratégicas de la organización.

b. Formación del CGTI. La segunda fase consistió en "educar" (según Van Grembergen y De Haes, 2008) a los directivos universitarios. Se supone que actualmente las universidades desconocen la importancia de realizar un buen gobierno (gobernanza) de sus TI. Y aunque, en ocasiones, algunos de los directivos universitarios conozcan dicha importancia, se puede suponer que no se ha extendido suficientemente esta cultura organizacional relacionada con las TI. Por ello, se planteó este proceso formativo inicial que sirva para recalcar la importancia de adoptar modelos de gobierno de las TI como para concienciar a los directivos de que asuman su responsabilidad en relación a la gobernanza de las TI. En resumen, el objetivo de esta formación fue que cada directivo conozca cuál es su responsabilidad en relación a la gobernanza de las TI.

c. **Autoevaluación de la Madurez inicial del GTI**, que tiene por objetivo determinar cuál es la madurez inicial del gobierno de las TI en UNICOMFACAUCA. Para descubrir esta situación de partida el Comité GTI se va someter a una serie de encuestas y dinámicas de consenso. Las encuestas se realizaron en base a plantillas disponibles en la aplicación web KTI que servirá de soporte a este proceso de autoevaluación. La aplicación web kTI (Figura 2) proporciona todo el soporte necesario para que cualquier universidad lleve a cabo la tercera fase o fase de autoevaluación del Gobierno de las TI.

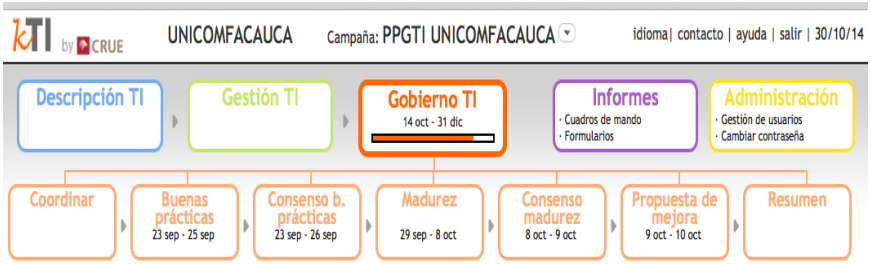


Fig. 2. Menú principal de Gobierno TI en kTI

Una vez constituido el Comité GTI y capacitado en Gobierno de las TI, comienza el proceso de Autoevaluación. Este proceso se compone a su vez de dos grandes fases, que se llevan a cabo en orden cronológico: primero se recogen los valores de los indicadores que evidencian si se están aplicando las mejores prácticas relacionadas con el gobierno de las TI y después se establece el nivel de madurez de gobierno de las TI en relación al modelo de madurez propuesto por GTI4U (Figura 3):

Fig. 3. Fases del proceso de Autoevaluación de la madurez del gobierno de las TI

Antonio Fernández

d. **Redacción de un Plan de Mejora del GTI**. En el último paso del Proyecto de Arranque del Gobierno de TI (PAGTI) el CGTI reflexionó sobre cuál es la situación actual del gobierno de las TI y propuso el estado de madurez que se desea alcanzar a un año. Para establecer la madurez objetivo, se tuvo en cuenta los

resultados del proceso de autoevaluación, pero también los objetivos estratégicos definidos por UNICOMFACAUCA.

Posteriormente, se consensuaron un conjunto de acciones de mejora, o buenas prácticas de gobierno de las TI, que se han incluido en el Plan de Mejora del Gobierno de las TI. Los resultados obtenidos fueron entregados a la rectora de la Corporación Universitaria Comfacauca una vez finalizado el PAGTI.

A partir de este momento son los miembros del Comité GTI los que trabajan para: conseguir el apoyo a este plan por parte del resto de los directivos universitarios, crear las estructuras necesarias y establecer las responsabilidades relacionadas con las TI en UNICOMFACAUCA, comunicar el plan y generar cultura entorno al buen gobierno de las TI, llevar a cabo el seguimiento del plan y asegurarse de su pleno cumplimiento y llevar a cabo autoevaluaciones periódicas de la madurez de su GTI con la posterior revisión del plan.

El Proyecto de Arranque de Gobierno de TI (PAGTI) para Unicomfacauca tuvo una duración estimada de 16 semanas, que se distribuyó entre las diferentes etapas como indica la Figura 1. El cronograma con las acciones llevadas a cabo aparece reflejado en la Figura 4.

Esta proyección tuvo una carga de trabajo para cada miembro del CGTI de 40 horas distribuidas a lo largo de este periodo. Esta planificación incluía 6 reuniones presenciales de unas 4 horas de duración, mientras que el resto de la carga fue dedica a trabajo autónomo durante el que se visionan elementos polimedia, se leen artículos y se rellenan encuestas.

Fig. 4. Cronograma del PAGTI en la UNICOMFACAUCA

R1: Presentación del Proyecto de Arranque al equipo de Gobierno y Constitución del Comité de Gobierno de las TI. (2 horas).

R2: Taller de formación sobre Gobierno de las TI (4 horas).

R3: Consenso de los indicadores de Evidencia de Gobierno de las TI (4 horas).

R4: Consenso del nivel de madurez para cada principio de gobierno (4 horas).

R5: Reflexión sobre situación actual y deseable, selección de acciones de mejoras (4 horas).

R6: Presentación resultados Plan de Mejora de Gobierno de las TI a la rectora (2 horas).

3 Evidencias de buenas prácticas relacionadas con el Gobierno de las TI

Después del proceso de formación, el siguiente objetivo fue el de establecer la situación inicial del gobierno de las TI en la Corporación Universitaria Comfacauca - UNICOMFACAUCA.

El primer paso consistió en descubrir qué buenas prácticas relacionadas con el gobierno de las TI se encontraban ya presentes en UNICOMFACAUCA. Para ello se llevó a cabo un taller de trabajo y una encuesta, realizada mediante la aplicación kTI (kti.crue.org), que incluyó los siguientes pasos:

- Los miembros del CGTI llenaron de manera individual los valores de los indicadores de evidencia de buenas prácticas de gobierno de las TI recogidos en la encuesta. Estos valores fueron introducidos de manera online y asíncrona por el propio CGTI, durante la sesión presencial (R3*).
- Después, con la ayuda de kTI, los miembros del CGTI consensuaron, también de manera presencial (R3*), un valor único para cada indicador de evidencia de buenas prácticas.

Las cuestiones recogidas en esta encuesta se han diseñado a partir de las mejores prácticas que se han encontrado recogidas en los principales marcos de referencia, estándares y publicaciones profesionales internacionales.

Se realizó interpretación de dichos resultados clasificados por cada uno de los principios que propone la norma ISO 38500, que también han sido incorporados al modelo GTI4U: *Responsabilidad, Estrategia, Adquisición, Desempeño, Cumplimiento y Comportamiento Humano*.

- **Responsabilidad**

Este principio de la norma ISO 38500 pretende *"que cada individuo o grupo de personas de la organización comprendan y acepten sus responsabilidades relacionadas con la demanda y prestación de servicios de TI. Quienes tengan la responsabilidad sobre las acciones también tienen la autoridad para llevarlas a cabo"* (ISO 38500).

- **Estrategia**

Este principio pretende establecer que *"a la hora de diseñar la estrategia actual y futura de la organización hay que tener en cuenta el potencial de las TI. Los planes estratégicos de las TI deben recoger y satisfacer las necesidades estratégicas de negocio de la organización"* (ISO 38500).

- **Adquisición**

Este principio establece que *"las adquisiciones de TI deben realizarse después de un análisis adecuado, en base a criterios válidos e incluirá decisiones claras y transparentes. Debe existir un equilibrio apropiado entre beneficios, oportunidades, coste y riesgos, tanto a corto como a largo plazo."* (ISO 38500).

- **Desempeño**

Este principio establece que *"las TI son la herramienta más adecuada para dar soporte a los procesos de negocio, ofreciendo servicios con el nivel y la calidad requerida para satisfacer los objetivos actuales y futuros de la organización"* (ISO 38500). Fundamentalmente, las organizaciones necesitan de sus TI para funcionar bien en cualquier momento.

Las TI serán "adecuadas" si consiguen dar soporte a los procesos universitarios en la medida en que estos las necesiten, ajustándose a un valor, coste y riesgo equilibrado.

- **Cumplimiento**

Este principio establece que *"las TI deben cumplir con toda la legislación y normativas publicadas que le afecte, y las organizaciones también deben tener claramente definidas sus propias políticas y procedimientos internos y apoyar su implantación y cumplimiento"* (ISO 38500). El incumplimiento de la legislación vigente es un gran riesgo que no puede justificar la dirección de la universidad argumentando desconocimiento de la misma o delegándola sin supervisión a otros niveles de la organización.

- **Comportamiento Humano**

Este principio pretende establecer *"la importancia que tiene la interacción de las personas con el resto de elementos de un sistema, con la intención de alcanzar el buen funcionamiento y un alto rendimiento del mismo. El comportamiento de las personas incluye su cultura, sus necesidades y sus aspiraciones, tanto a nivel individual como en grupo"* (ISO 38500).

3.2 Análisis aplicación de buenas prácticas

Tras el largo proceso de evaluación realizado a UNICOMFACAUCA y al resto de universidades participantes en el Proyecto de Arranque, se pueden extraer las primeras conclusiones relacionadas sobre cuáles son las buenas prácticas de gobierno de las TI presentes en el conjunto de las ocho universidades que han participado en el Proyecto de Arranque. En la Figura 5 se aprecia, en relación a la media del P. de Arranque, que los principios *Responsabilidad* y *Estrategia* satisfacen 1 de cada 3 prácticas, los principios *Adquisición y Desempeño* satisfacen 1 de cada 4 y los principios *Cumplimiento y Comportamiento Humano* llega a 1 de cada 5 buenas prácticas implantadas. Estos resultados ponen de manifiesto que las universidades participantes en el Proyecto de Arranque se encuentran en una situación incipiente en cuanto a la implantación de las mejores prácticas relacionadas con el gobierno de las TI. Lo cual no quiere decir que desempeñen mal sus responsabilidades o desarrollen una

inadecuada política relativa a las TI, pero sí que resulta aconsejable formalizar su gobierno de las TI e incorporar las mejores prácticas de referencia.

De manera particular, UNICOMFACAUCA presenta un nivel inferior a la media del Proyecto de arranque en los principios de Responsabilidad (17% de buenas prácticas), Estrategia (6%), Cumplimiento (5%) y Comportamiento Humano (7%). Mientras tanto los principios de Desempeño (25%) se mantiene en valores cercanos a la media del Proyecto de Arranque. Sin embargo, el principio de Adquisición (47%) se encuentra por encima de la media.

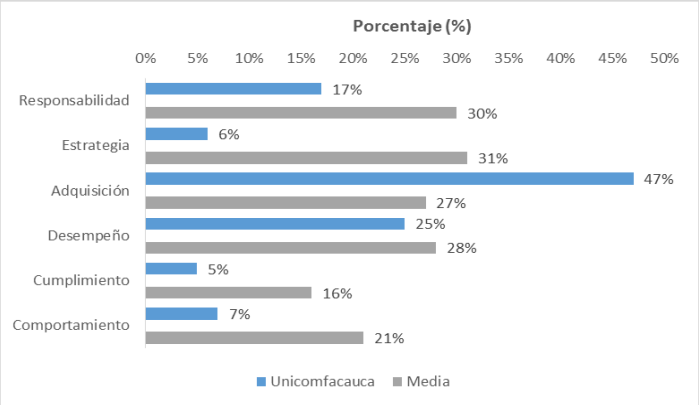


Fig. 5. Buenas prácticas de gobierno de las TI presentes en el SUE

4. Madurez inicial y objetivos de mejora del Gobierno de las TI

Una vez establecidas las evidencias de buenas prácticas de gobierno de las TI, los miembros del Comité GTI procedieron a determinar el nivel de madurez actual del gobierno de las TI en UNICOMFACAUCA y el valor objetivo a alcanzar a medio plazo.

Para alcanzar este objetivo se llevaron a cabo las siguientes acciones:

1. Cada miembro del CGTI respondió de manera individual a una serie de cuestiones que establecían de manera automática cuál es el nivel de madurez de gobierno de las TI en relación al modelo de referencia propuesto por GTI4U. Estas cuestiones fueron rellenadas de manera presencial (R4*).
2. En la misma sesión presencial, los miembros del CGTI consensuaron un valor único para cada cuestión de madurez.
3. A partir de las respuestas obtenidas y de manera automática, gracias a una lógica diseñada por los investigadores, se propuso un valor de madurez para cada principio de gobierno de las TI.

4. En la siguiente reunión presencial (R5), los miembros del CGTI analizaron los valores propuestos y establecieron finalmente el valor actual de madurez para cada principio de gobierno de las TI.
5. En la misma reunión, los miembros del CGTI analizaron la situación inicial de madurez, los objetivos institucionales de la universidad a medio plazo y propusieron el nivel de madurez deseable, como objetivo de mejora a medio plazo.

4.1. Modelo de madurez propuesto por GTI4U

Las universidades, al igual que cualquier otra organización, necesitan implantar sistemas de gobierno de sus TI si desean mejorar su rendimiento y efectividad. Para ello, el primer paso es conseguir la implicación de sus altos directivos, que deben comprender cuales son los principios de un adecuado gobierno de las TI. Este objetivo se puede alcanzar utilizando la norma ISO 38500 (2008). La norma incluye un modelo propio de gobierno de las TI y una guía de sugerencias y buenas prácticas muy útiles. Por ello, se ha diseñado y validado un marco de referencia de Gobierno de las TI para Universidades (GTI4U).

Este marco se basa y respeta por completo al modelo de gobierno TI propuesto por la norma ISO 38500. Pero a la vez, proporciona una serie de herramientas para que sea fácilmente implementado en un entorno universitario. El objetivo último sería que la universidad que implemente el modelo GTI4U también consiga, en un futuro, certificarse fácilmente con la norma ISO 38500.

El modelo GTI4U está compuesto por tres niveles: El primer nivel incluye todos los elementos de la norma ISO 38500; modelo de gobierno TI, principios, buenas prácticas y diccionario de términos. El segundo está compuesto por un Modelo de Madurez (MM) para cada principio, que se utilizará para establecer en qué nivel de madurez de gobierno de las TI se encuentra cada universidad. El tercero incluye a los indicadores que van servir para medir hasta qué punto se satisfacen los criterios presentados en la norma. En la figura 6, se muestra los elementos del modelo GTI4U.

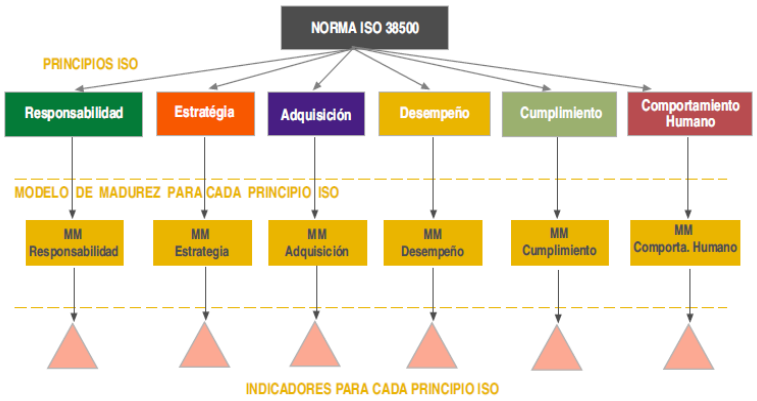


Fig.6. Modelo GTI4U

En todo este proceso se ha utilizado como referencia el modelo de madurez propuesto por GTI4U. Dicho modelo de madurez incluye 6 posibles niveles:

- Inexistente (0), la universidad no conoce el principio y no es consciente de necesitarlo.
- Inicial (1), el principio está establecido pero los procesos son desorganizados y *ad hoc*.
- Repetible (2), el principio está inmaduro, los procesos siguen un patrón regular.
- Definido (3), el principio comienza a madurar, los procesos se documentan y comunican.
- Medible (4), principio bastante maduro, los procesos se monitorizan y se miden.
- Óptimo (5), principio a nivel óptimo, procesos basados en las mejores prácticas.

4.2. Análisis de la Madurez inicial y objetivos de mejora del gobierno de las TI en UNICOMFACAUCA

El nivel de madurez "Actual" y "Objetivo" de cada principio de gobierno de las TI de UNICOMFACAUCA se ha consolidado los respectivos resultados. Se puede comprobar cómo UNICOMFACAUCA se plantea como objetivo alcanzar el nivel 1 en el principio de *Desempeño y Cumplimiento*; para el nivel 2 el resto de principios.

En la Figura 6 se presenta el cuadro de mando que incluye el nivel actual y el objetivo de madurez de gobierno de las TI para cada principio y la relación con la media del proyecto piloto. También aparece el porcentaje de buenas prácticas ya implementadas

y el porcentaje de mejora previsto a corto plazo (próximo año). Por tanto, este cuadro resume todo el trabajo realizado durante el PAGTI en UNICOMFACAUCA.

También se muestra cómo para alcanzar dichos niveles de madurez se necesita incrementar el porcentaje de buenas prácticas de los principios de Responsabilidad, Adquisición y Comportamiento Humano hasta llegar a 4 de cada 10. Para el principio de Desempeño serán necesarias 3 de cada 10 buenas prácticas mientras que para el principio de Estrategia será necesario llegar a 4 de cada 10. Mientras en el principio de Cumplimiento será necesario incrementar el porcentaje hasta 2 de cada 10 buenas prácticas, lo que significaría que la UNICOMFACAUCA superaría la media del Proyecto de Arranque el año próximo.

El CGTI ha decidido concentrar los esfuerzos en todos los principios y poner la puesta en marcha de mejoras en cada uno de ellos, con un amplio margen de mejora por delante.

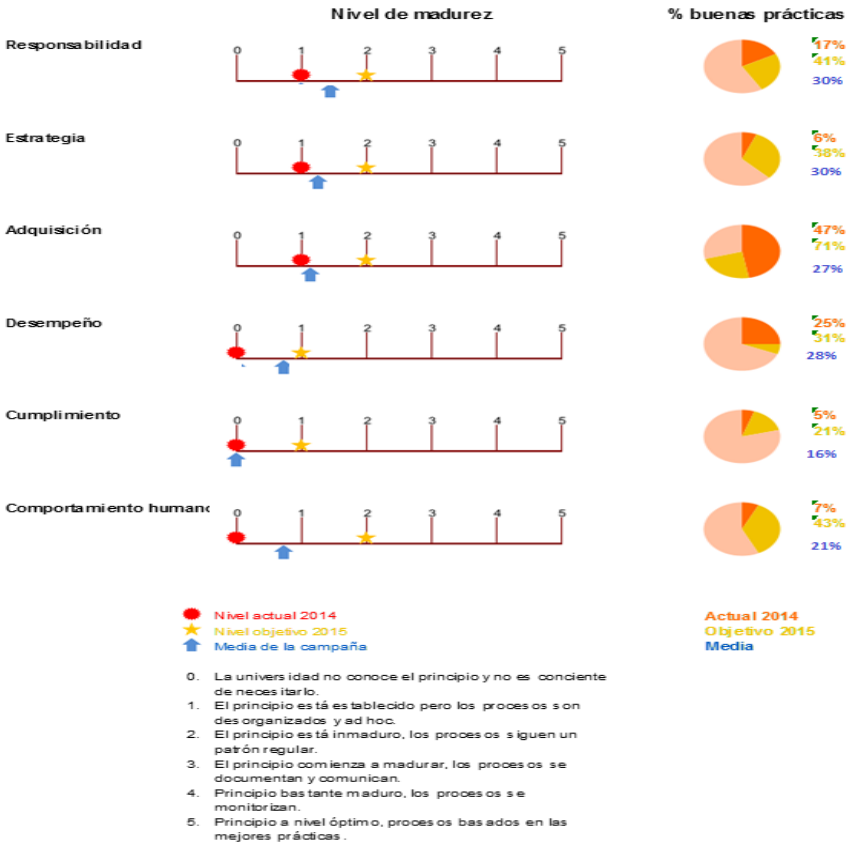


Fig. 6. Mejores prácticas y madurez de gobierno de las TI actual y objetivo de Unicomfacauca

5. Plan de mejora del Gobierno de las TI

5.1. Acciones de mejora, debidamente priorizadas y planificadas.

Una vez seleccionados los objetivos de mejora de la madurez de gobierno TI por parte de los miembros de CGTI de UNICOMFACAUCA, se pasó a establecer cuáles serían las acciones de mejora que se deberían poner en marcha para conseguir satisfacerlos (este proceso se llevó a cabo durante la R5). El modelo GTI4U ha definido una relación entre cada uno de los ítems de las tablas del modelo de madurez y las acciones, por tanto, determinar cuáles son dichas acciones es una operación inmediata. Con los resultados obtenidos a lo largo del Proyecto de Arranque el Comité de GTI (con la ayuda de los investigadores) se redactó un Plan de Mejora que incluye esencialmente un listado de acciones de mejora, debidamente priorizadas y planificadas.

Se presentó una relación de un conjunto de acciones de mejora propuestas, donde se priorizan y clasifican en iniciativas diferentes. Esto quiere decir que las acciones que aparecen en una misma fila se pueden ejecutar independientemente de que se ejecuten o no las acciones de otra fila. Es importante establecer un orden dentro de cada fila, de manera que se ejecuten primero las acciones que se encuentran más a la izquierda.

Se recomienda que el Equipo de Gobierno de UNICOMFACAUCA ponga en marcha las siguientes acciones de mejora durante el próximo año:

- **El Equipo de Gobierno debe asumir el gobierno de las TI** (R1) y establecer un modelo de toma de decisiones que incluya cuales son los elementos de TI sobre los que debe decidir el Equipo de Gobierno y cuales puede delegar (R8).
- El Equipo de Gobierno debe **liderar el diseño de la estrategia de las TI** (R2), encomendando al CIO la elaboración de una estrategia específica de las TI (R3) que esté incluida, y por tanto alineada, en la estrategia global de la universidad (D2).
- El Equipo de Gobierno debe **crear una estructura que brinde soporte al sistema de gobierno de las TI**, para ello debe comenzar por:
    - Crear un **Comité de Estrategia de las TI,** de carácter consultivo, del que formarían parte el CIO, otros miembros del Equipo de Gobierno y otros responsables TI, que tendría como responsabilidad proponer el diseño de la estrategia, las políticas y el gobierno de las TI (R5).
    - Crear un **Comité de Dirección de las TI** AR6), del que formarían parte el CIO, otros responsables de TI y otros expertos, e invitará a participar a los responsables y usuarios de los principales proyectos TI (R7). Su principal función sería supervisar la implementación de la estrategia TI y de los proyectos propuestos para llevarla a cabo.

El Equipo de Gobierno debe ocuparse de que estos comités se reúnan periódicamente y funcionen adecuadamente.

- El Equipo de Gobierno debe diseñar y **publicar una política que oriente sobre los diferentes tipos de adquisiciones** (A1) y disponer de un **procedimiento para las adquisiciones de TI** que incluyan el análisis de diferentes ofertas en base a los objetivos estratégicos, y no solo en base a criterios técnicos o económicos (A2). También revisar el **rendimiento de los servicios TI externalizados** y determina su continuidad (A10).
- El Equipo de Gobierno debe **crear una Cartera de Proyectos TI** (A3) que se actualice anualmente. Sería deseable que esta iniciativa se complementara con las siguientes acciones:
    - Establecer una **plantilla para la redacción de los proyectos TI** que incluya toda la información relevante (objetivos, beneficios, pasos a seguir, criterios de rendimiento y riesgos asociados) que necesita el EG para establecer el orden de ejecución de los mismos (A6)
    - Calcular adecuadamente **el coste de un proyecto TI,** teniendo en cuenta los costes de inversión y mantenimiento de las TI, pero también el coste de los recursos humanos, su formación y en general el coste de los cambios organizativos que provocará dicho proyecto (A5)
    - Incluir la plantilla para la redacción de los proyectos TI los **criterios a evaluar regularmente para decidir sobre la continuidad** o el momento de la interrupción del servicio o la retirada de un equipamiento TI (A8)
    - Identificar en el **análisis de riesgos estratégico** de los proyectos los factores relacionados con la **resistencia al cambio** de las personas o grupos afectados (H3) y establecer los mecanismos que potencien **el máximo nivel de compromiso** de los implicados mediante el diseño de acciones formativas de los implicados (A7)
    - Publicar periódicamente los **objetivos iniciales de los proyectos TI** (A4 y R4) y establecer un procedimiento para medir el nivel de **éxito alcanzado** al finalizar cada proyecto (A9).

- El Equipo de Gobierno debe **realizar el seguimiento del rendimiento de los Servicios TI** para mantenerlos dentro de los niveles deseables por sus usuarios (D1). A ello puede contribuir:
    - Analizar periódicamente cuales son los **requerimientos de los usuarios** y diseñar un procedimiento para analizar la **satisfacción de los diferentes grupos de interés** en relación a los servicios universitarios basados en TI en explotación.
    - Saber **cuántos desarrollos TI no se encuentran aún integrados** y sin embargo deberían estarlo (E3) y diseñar un **programa a largo plazo** que tenga por objetivo llevar a cabo todos los desarrollos TI que la universidad necesita para cubrir las necesidades de los usuarios.

- El Equipo de Gobierno debe **definir y publicar un catálogo con todo tipo de políticas relacionadas con las TI** para orientar al resto de los universitarios sobre cómo desarrollar las TI en el campus (E1). Se sugiere establecer al menos las siguientes políticas:
    - Política que oriente sobre los diferentes tipos de **adquisiciones** (A1) y sobre el rendimiento esperado de los procesos universitarios (D1)
- El Equipo de Gobierno debe ocuparse de que se **cumplan las leyes externas y normas internas** establecidas, para ello se recomiendan las siguientes acciones:
    - Asignar formalmente **la responsabilidad de conocer la legislación** relacionada con las TI a una persona o grupo de ellas (C1) y revisar periódicamente que cumple con esta responsabilidad (C5).
    - Elaborar y mantener actualizado **un catálogo** de referencia **que contenga las normas y leyes externas relacionadas con las TI** que afectan a la universidad (C2)
- El Equipo de Gobierno debe **promover una gestión de las TI basada en estándares** (por ejemplo ITIL, ISO 27000, ISO 9000, etc.) Para ello, las primeras acciones a llevar a cabo son:
    - Asignar formalmente **la responsabilidad de conocer los estándares** relacionados con las TI a una persona o grupo de ellas (C3) y revisar periódicamente que cumple con esta responsabilidad.
    - Elaborar y mantener actualizado **un catálogo de estándares** relacionadas con las TI aplicables o ya aplicados a la universidad (C4)
- Por último, el Equipo de Gobierno debe **procurar la implicación de las personas** en las iniciativas de TI. Para ello debe:
    - **Identificar los diferentes grupos de interés** y documentar formalmente cómo van a participar cada uno en las nuevas iniciativas de TI (H1). Realizar diferentes agrupamientos en los grupos de interés para darles un trato diferenciado a lo hora de implicarlos en los procesos de cambio soportados por las TI (H2)
    - **Evitar sobrecargar de trabajo** a las personas, realizando un adecuado análisis de los recursos humanos y de sus cargas de trabajo.
    - **Analizar** los factores relacionados con la **resistencia al cambio** de las personas o grupos afectados y la falta de compromiso de los implicados (H3).
    - Incluir en la planificación de los proyectos TI acciones destinadas a **paliar** el riesgo relacionado con la **falta de compromiso de los participantes** (H4)

Este Plan de Mejora será fue presentado al Equipo de Gobierno durante la R6, quedando a su disposición para ser analizado, aprobado y puesto en marcha.

6. Conclusiones

El Proyecto de Arranque del gobierno de las TI en el SUE ha permitido que las universidades participantes conozcan las principales ventajas que aporta un sistema de gobierno de las TI a su organización, su nivel de madurez actual en relación a la ISO 38500 y cuáles son las mejores prácticas a llevar a cabo para mejorarlo. Entendemos que el cambio más importante llevado a cabo es comprender la importancia del gobierno de las TI y ser conscientes de que la responsabilidad de tomar las decisiones de Gobierno TI corresponde al Equipo de Gobierno de la Universidad incorporando al CIO de forma activa en esta toma de decisiones; esto permitirá que las TI se alineen con la estrategia global de la Universidad y aumenten el valor de los procesos universitarios.

La segunda gran aportación del Proyecto de Arranque tiene que ver con el modelo GTI4U, que se ha aplicado en diez universidades, siendo validado con satisfacción por los responsables de las TI de las mismas. Este modelo ha sido mejorado con las sugerencias recibidas durante el proceso y ahora se encuentra disponible una nueva versión que es más rica, versátil y sólida que la anterior. Además del modelo, se ha validado el proceso global de implantación del sistema de gobierno de las TI, al menos su fase de arranque, ya que la validación definitiva no va a llegar hasta que no se revise dicha implantación dentro de un año, cuando se hayan ejecutado las acciones de mejora sugeridas.

La primera implementación del PAGTI en Latinoamérica se ha llevado a cabo en la UNICOMFACAUCA. El primer resultado a destacar es que su manera de gobernar no es muy diferente a la de las universidades españolas y por tanto el modelo GTI4U es válido también para esta universidad.

Durante este proceso se ha puesto de manifiesto que la UNICOMFACAUCA es una organización de carácter ofensivo y autosuficiente en relación al gobierno de las TI y al mismo tiempo preocupada por implementar mejoras, lo cual ya se podía deducir por el simple hecho de solicitar su participación en este proyecto piloto. De hecho, hay que valorar muy positivamente la plena disponibilidad e interés mostrado por los miembros del CGTI durante todo su desarrollo.

Los resultados muestran que su madurez inicial y las buenas prácticas relacionadas con el gobierno de las TI no están aún extendidas y por ello su madurez se encuentra en los primeros niveles de la escala (no pasa del nivel 1 en ningún principio). Sin embargo, también muestran el deseo de mejorar de manera inmediata y permanente, lo que se demuestra por la puesta en marcha de una Cartera de Proyectos TI para el presente año.

UNICOMFACAUCA se plantea como objetivo alcanzar el nivel 2 en todos los principios y el nivel 1 en *Desempeño y Cumplimiento*, lo cual significaría dar un salto de madurez que la situaría en todos ellos por encima de la media de las 10 universidades españolas participantes. Para alcanzar dichos niveles de madurez se necesita

incrementar el porcentaje de buenas prácticas de los principios de Responsabilidad, Adquisición y Comportamiento Humano hasta llegar a 4 de cada 10. Para el principio de Desempeño serán necesarias 3 de cada 10 buenas prácticas mientras que para el principio de Estrategia será necesario llegar a 4 de cada 10. Mientras en el principio de Cumplimiento será necesario incrementar el porcentaje hasta 2 de cada 10 buenas prácticas, lo que significaría que la UNICOMFACAUCA superaría la media del P. de Arranque durante el próximo año.

El que la madurez de gobierno de las TI se encuentren en los primeros niveles de la escala no significan que UNICOMFACAUCA desempeñen mal sus responsabilidades o desarrollen una inadecuada política relativa a las TI, pero sí que resulta aconsejable formalizar su gobierno de las TI e incorporar las mejores prácticas de referencia, sustentar la acción de gobierno en unos procesos bien definidos y transparentes, soportados en la documentación adecuada, etc. Pero, el verdadero potencial del gobierno de UNICOMFACAUCA no puede establecerse ahora, sino que se descubrirá en los meses venideros durante los cuales se procurará ejecutar las acciones de mejora para conseguir una mayor madurez en su gobierno de las TI. Si el actual gobierno de las TI es suficientemente sólido entonces las acciones de mejora serán más fáciles de aplicar y se alcanzarán los objetivos establecidos inmediatamente.

El plan de mejora comienza por aplicar una serie de medidas que afectan a la organización, al gobierno y a la toma de decisiones. En este sentido, consideramos que es fundamental la implicación del Equipo de Gobierno de UNICOMFACAUCA en la labor de definir y aprobar las políticas y normas internas aplicables a las TI, de evaluar y apoyar su puesta en práctica y de conocer y monitorizar los resultados obtenidos. Las TI trascenderán así su carácter técnico y se convertirán en una herramienta fundamental de apoyo a la dirección estratégica global de la universidad.

Se desea que este Proyecto de Arranque no concluya ahora sino que UNICOMFACAUCA emprenda las acciones de mejora planificadas y que tras un breve periodo (entre 1 y 2 años), vuelvan a autoevaluar su madurez de gobierno de las TI para establecer el grado de crecimiento del mismo. Esto será bueno para la propia universidad y para el conjunto de universidades latinoamericanas, que verán en las universidades participantes y en el proceso seguido un referente para mejorar. El objetivo es que este ejemplo sea seguido próximamente por muchas otras universidades latinoamericanas que apuesten por el gobierno de las TI.

Referencias

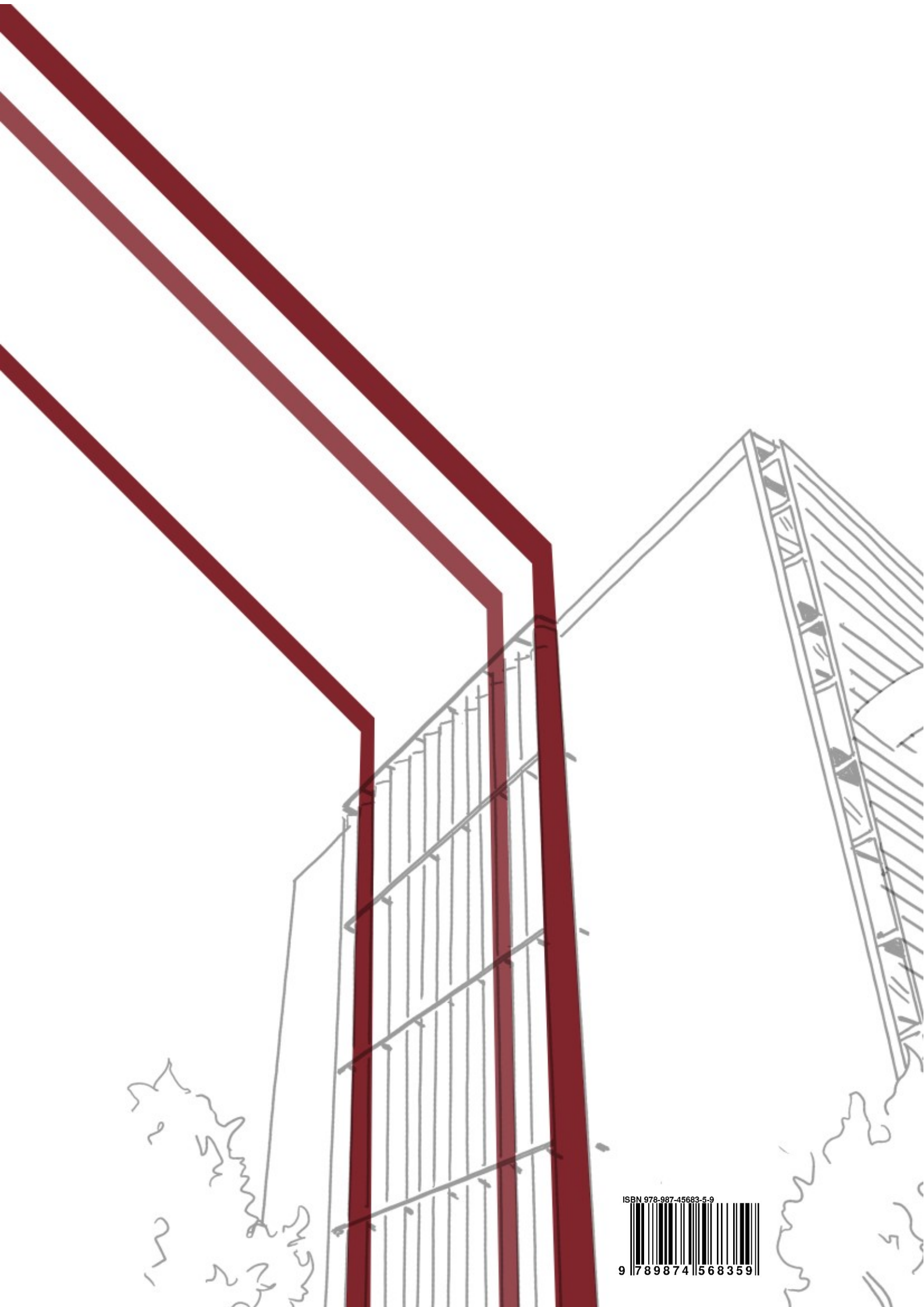
1. Calder-Moir (2008). *Calder-Moir IT Governance Framework*. it Governance. www.itgovernance.co.uk/calder_moir.aspx
2. COBIT (2005). *CoBIT, 4th Edition*. IT Governance Institute (ITGI). www.itgi.org

3. Council, C. L. (2006). Implementing COBIT in Higher Education: Practices that work best. *Information Systems Control Journal*. ISACA. www.isaca.org
4. CRUE (2010). *La Universidad española en cifras 2010*. CRUE http://www.crue.org/Publicaciones/UEC.html
5. CRUE (2011). *Gobierno de las TI para Universidades*. CRUE http://www.crue.org/Publicaciones/GobiernoTI.html
6. Fernández, A. (2009). Análisis, Planificación y Gobierno de las TI en las universidades. Tesis doctoral. Universidad de Almería.
7. Fernández, A. y Llorens, F. (2011). *Gobierno de las TI para Universidades*. CRUE. http://www.crue.org/Publicaciones/GobiernoTI.html
8. Fernández, A. (2011). Capítulo 10: Modelo de Gobierno de las TI para Universidades (GTI4U). *Gobierno de las TI para Universidades*. CRUE.
9. Fernández, A y otros (2011). Una experiencia de gobierno de las TI en el SUE. En Uceda y otros (2011). *UNIVERSITIC 2011: Descripción, Gestión y Gobierno de las TI en el SUE.* CRUE, Madrid. http://www.crue.org/Publicaciones/universitic.html
10. Golden, C., Holland, N., Luker, M. y Yanosky, R. (2007). *A Report on the EDUCAUSE Information Technology Governance Summit*. September 10-11. EDUCAUSE. http://net.educause.edu/ir/library/pdf/CSD5228.pdf
11. ISO 38500 (2008). *ISO/IEC 38500:2008 Corporate Governance of Information Technology*. http://www.iso.org/iso/pressrelease.htm?refid=Ref1135
12. ITGI (2011). *IT Governance Global Status Report*. IT Governance Institute. www.itgi.org
13. JISC (2007). *A Framework for Information Systems Management and Governance*. Joint Information Systems Committee (JISC). www.ismg.ac.uk/Portals/18/Governance%20Framework.pdf
14. Llorens, F. y Fernández, A. (2008). Encuesta de Satisfacción de UNIVERSITIC y COITIC. Informe interno de la Comisión Sectorial TIC de la CRUE. 2008.
15. Martín y Fernández (2011). Capítulo 12: ¿Cómo implantar el gobierno de las TI en una universidad?. *Gobierno de las TI para Universidades*. CRUE.
16. Nolan y McFarlan (2005). Information Technology and the Board of Directors. Harvard Business Review. October
17. Petrorius, J. (2006). A Structured Methodology for Developing IT Strategy. *Proceedings of the Conference on Information Technology in Tertiary Education*. Pretoria.
18. Píriz, S., Jímenez, T., Fernández, A., Llorens, F., Fernández, S., Rodeiro, D., Ruzo, E. y Canay, R. (2013). *UNIVERSITIC 2013: Descripción, Gestión y Gobierno de las TI en el SUE.* CRUE, Madrid.
19. PLS RAMBOLL Management (2004) Studies in the Context of the E-learning Initiative: Virtual Models of European Universities (Lot 1). Draft Final Report to the UE25 Commission, DG Education & Culture. http://www.elearningeuropa.info/extras/pdf/virtual_models.pdf
20. Ridley, M. (2006). Information Technology (IT) Governance. A position paper.

http://www.isc.uoguelph.ca/documents/061006ITGovernance-PositionPaper-September2006.pdf
21. Sierra, L. (2012). ¿Cómo implantar el Gobierno de las Tecnologías de Información en Instituciones de Educación Superior?

(página dejada intencionalmente en blanco)

(página dejada intencionalmente en blanco)



ISBN 978-987-45683-5-9