



ISSN: 1646-9895

Revista Ibérica de Sistemas e Tecnologias de Informação
Revista Ibérica de Sistemas y Tecnologías de Información

M a r ç o 1 9 • M a r c h 1 9



©AISTI 2019 <http://www.aisti.eu>

Nº 31

Edição / Edición

Nº 31, 03/2019

Tiragem / Tirage: 1000

Preço por número / Precio por número: 17,5€

Subscrição anual / Suscripción anual: 30€ (2 números)

ISSN: 1646-9895

Depósito legal:

Indexação / Indexación

Academic Journals Database, CiteFactor, Dialnet, DOAJ, DOI, EBSCO, GALE, IndexCopernicus, Index of Information Systems Journals, ISI Web of Knowledge, Latindex, ProQuest, QUALIS, SciELO, SCImago, Scopus, SIS, Ulrich's.

Propriedade e Publicação / Propriedad y Publicación

AISTI – Associação Ibérica de Sistemas e Tecnologias de Informação

Rua Quinta do Roseiral 76, 4435-209 Rio Tinto, Portugal

E-mail: aistic@gmail.com

Web: <http://www.risti.xyz>

Director

Álvaro Rocha, Universidade de Coimbra

Coordenadores da Edição / Coordinadores de la Edición

Jezreel Mejia, Centro de Investigación en Matemática (CIMAT)

Mirna Muñoz, Centro de Investigación en Matemática (CIMAT)

Conselho Editorial / Consejo Editorial

Carlos Ferrás Sexto, Universidad de Santiago de Compostela

Gonçalo Paiva Dias, Universidade de Aveiro

Jose Antonio Calvo-Manzano Villalón, Universidad Politécnica de Madrid

Luís Paulo Reis, Universidade do Minho

Manuel Pérez Cota, Universidad de Vigo

Ramiro Gonçalves, Universidade de Trás-os-Montes e Alto Douro

Conselho Científico / Consejo Científico

Adolfo Lozano-Tello, Universidad de Extremadura, ES

Adrián Hiebra Pardo, Universidad de Santiago de Compostela, ES

Alberto Fernández, Universidad Rey Juan Carlos, ES

Alberto Bugarín, Universidad de Santiago de Compostela, ES

Alejandro Medina, Universidad Politécnica de Chiapas, MX

Alejandro Rodríguez González, Universidad Politécnica de Madrid, ES

Alejandro Peña, Escuela de Ingeniería de Antioquia, CO

Alexandre L'Erario, Universidade Tecnológica Federal do Paraná, BR

Ana Azevedo, Instituto Politécnico do Porto, PT

Ana Cristina Ramada Paiva, FEUP, Universidade do Porto, PT

Ana Isabel Veloso, Universidade de Aveiro, PT

Ana Maria Correia, ISEGI, Universidade Nova de Lisboa, PT

Ana Paula Afonso, Instituto Politécnico do Porto, PT

Anabela Mesquita, Instituto Politécnico do Porto, PT

Angelica Caro, Universidad del Bío-Bío, CL

Antoni Lluís Mesquida, UIB, España

António Coelho, FEUP, Universidade do Porto, PT

António Godinho, ISLA-Gaia, PT

Antonio Jesus Garcia Loureiro, Universidad de Santiago de Compostela, ES
António Pereira, Instituto Politécnico de Leiria, PT
Armando Mendes, Universidade dos Açores, PT
Arnaldo Martins, Universidade de Aveiro, PT
Arturo J. Méndez, Universidad de Vigo, ES
Baltasar García Pérez-Schofield, Universidad de Vigo, ES
Bell Manrique, Universidad de Medellín, Colombia
Benjamim Fonseca, Universidade de Trás-os-Montes e Alto Douro, PT
Bráulio Alturas, ISCTE - Instituto Universitário de Lisboa, PT
Brenda L. Flores-Rios, Universidad Autónoma de Baja California, MX
Brígida Mónica Faria, ESTSP, Instituto Politécnico do Porto, PT
Carlos Costa, Universidade de Aveiro, PT
Carlos Carreto, Instituto Politécnico da Guarda, PT
Carlos Morais, Instituto Politécnico de Bragança, PT
Carlos Mario Zapata, Universidad Nacional de Colombia, Colombia
Carlos Pon Soto, Universidad Católica del Norte, Chile
Carlos Vaz de Carvalho, Instituto Politécnico do Porto, PT
Ciro Martins, Universidade de Aveiro, PT
Daniel Castro Silva, Universidade de Coimbra, PT
Daniel Polónia, Universidade de Aveiro, PT
Daniel Riesco, Universidad Nacional de San Luis, AR
David Fonseca, Universitat Ramon Llull, ES
David Ramos Valcarcel, Universidad de Vigo, ES
Dora Simões, Universidade de Aveiro, PT
Eduardo Sánchez Vila, Universidad de Santiago de Compostela, ES
Emiliano Reynares, CIDISI - UTN FRSF - CONICET, AR
Enric Mor, Universitat Oberta de Catalunya, ES
Eusébio Ferreira da Costa, Escola Superior de Tecnologias de Fafe, PT
Fábio Magalhães, Universidade Nove de Julho, BR
Feliz Gouveia, Universidade Fernando Pessoa, PT
Fernando Bandeira, Universidade Fernando Pessoa, PT
Fernando Moreira, Universidade Portucalense, PT
Francisco Restivo, Universidade Católica Portuguesa, PT
Frederico Branco, Universidade de Trás-os-Montes e Alto Douro, PT

Gerardo Gonzalez Filgueira, Universidad da Coruña, ES
Germano Montejano, Universidad Nacional de San Luis, AR
Guilhermina Lobato Miranda, Universidade de Lisboa, PT
Hélder Zagalo, Universidade de Aveiro, PT
Hélia Guerra, Universidade dos Açores, PT
Henrique Gil, Instituto Politécnico de Castelo Branco, PT
Henrique Santos, Universidade do Minho, PT
Higino Ramos, Universidad de Salamanca, ES
Hugo Paredes, Universidade de Trás-os-Montes e Alto Douro, PT
Isabel Pedrosa, Instituto Politécnico de Coimbra, PT
Isaura Ribeiro, Universidade dos Açores, PT
Isidro Calvo, Universidad del País Vasco (UPV/EHU), ES
Ismael Etxeberria-Agiriano, Universidad del País Vasco (UPV/EHU), ES
Ivan Garcia, Universidad Tecnológica de la Mixteca, MX
Javier Garcia Tobio, CESGA-Centro de Supercomputacion de Galicia, ES
Jezreel Mejia, Centro de Investigación en Matemática (CIMAT), MX
João Pascual Faria, FEUP, Universidade do Porto, PT
João Paulo Costa, Universidade de Coimbra, PT
João Tavares, FEUP, Universidade do Porto, PT
Joaquim José Gonçalves, Instituto Politécnico do Cávado e do Ave, PT
Joaquim Madeira, Universidade de Aveiro, PT
Joaquim Reis, ISCTE - Instituto Universitário de Lisboa, PT
Jörg Thomaschewski, University of Applied Sciences OOW - Emden, DE
Jorge Dias Villegas, Universidad de la Frontera, CL
Jose Alfonso Aguilar, Universidad Autonoma de Sinaloa, MX
José Augusto Fabri, Universidade Tecnológica Federal do Paraná, BR
José Braga de Vasconcelos, Universidade Atlântica, PT
José Cascalho, Universidade dos Açores, PT
José Felipe Cocón Juárez, Universidad Autónoma del Carmen, MX
José Gallardo, Universidad Católica del Norte, Chile
Jose J. Pazos-Arias, Universidad de Vigo, ES
José Luís Silva, Instituto Universitário de Lisboa (ISCTE-IUL), PT
José Paulo Lousado, Instituto Politécnico de Viseu, PT
José Luis Pestrana Brincones, Universidad de Málaga

José Luís Reis, ISMAI - Instituto Superior da Maia, PT
Jose M Molina, Universidad Carlos III de Madrid, ES
Jose Maria Zavala Perez, Eticas Research & Consulting, ES
José Martins, Universidade de Trás-os-Montes e Alto Douro, PT
Jose Maria de Fuentes, Universidad Carlos III de Madrid, ES
Jose R. R. Viqueira, Universidade de Santiago de Compostela, ES
José Silvestre Silva, Academia Militar, PT
Josep M. Marco-Simó, Universitat Oberta de Catalunya, ES
Juan D'Amato, PLADEMA-UNCPBA-CONICET, AR
Juan M. Santos Gago, Universidad de Vigo, ES
Juan Manuel Fernández-Luna, Universidad de Granada, ES
Juan-Manuel Lopez-Zafra, Universidad Complutense de Madrid, ES
León Jaramillo, Tecnológico de Antioquia, Colombia
Leonardo Bermon, Universidad Nacional de Colombia, CO
Leila Weitzel, Universidade Federal Fluminense, BR
Lilia Muñoz, Universidad Tecnológica de Panamá, PA
Lorena Cardona, Universidad de Medellín, Colombia
Luis Alvarez Sabucedo, Universidad de Vigo, ES
Luís Bruno, Instituto Politécnico de Beja, PT
Luis de Campos, Universidad de Granada, ES
Luis Fernandez-Sanz, Universidad de Alcalá, ES
Luís Ferreira, Instituto Politécnico do Cávado e do Ave, PT
Luis Vilán-Crespo, Universidad de Vigo, ES
Luisa María Romero-Moreno, Universidad de Sevilla, ES
Luisa Miranda, Instituto Politécnico de Bragança, PT
Lus Sussy Bayona Ore, Universidad Nacional Mayor de San Marcos, PE
Magdalena Arcilla Cobián, Universidade Nacional de Educación a Distancia, ES
Manuel Jose Fernandez Iglesias, Universidad de Vigo, ES
Marco Painho, ISEGI, Universidade Nova de Lisboa, PT
Maria Hallo, Escuela Politécnica Nacional, EC
María J. Lado, Universidad de Vigo, ES
Maria João Castro, Instituto Politécnico do Porto, PT
Maria João Ferreira, Universidade Portucalense, PT
Maria João Gomes, Universidade do Minho, PT

Maria José Angélico, Instituto Politécnico do Porto, PT
Maria José Marcelino, Universidade de Coimbra, PT
Maria José Sousa, Universidade Europeia, PT
Marisol B. Correia, Universidade do Algarve, PT
Maristela Holanda, Universidade de Brasília, BR
Martín Llamas Nistal, Universidad de Vigo, ES
Matías García Rivera, Universidad de Vigo, ES
Mercedes Ruiz, Universidad de Cádiz, ES
Miguel A. Brito, Universidade do Minho, PT
Miguel Bugalho, Universidade Europeia, PT
Miguel Casquilho, IST, Universidade de Lisboa, PT
Miguel Hidalgo, INACOL, Mexico
Mirna Ariadna Muñoz Mata, Centro de Investigación en Matemáticas (CIMAT), MX
Nelson Rocha, Universidade de Aveiro, PT
Nuno Lau, Universidade de Aveiro, PT
Nuno Ribeiro, Universidade Fernando Pessoa, PT
Oscar Mealha, Universidade de Aveiro, PT
Paula Peres, Instituto Politécnico do Porto
Paula Prata, Universidade da Beira Interior, PT
Paulo Martins, Universidade de Trás-os-Montes e Alto Douro, PT
Paulo Pinto, FCT, Universidade Nova de Lisboa, PT
Pedro Abreu, Universidade de Coimbra, PT
Pedro Miguel Moreira, Instituto Politécnico de Viana do Castelo, PT
Pedro Nogueira Ramos, ISCTE - Instituto Universitário de Lisboa, PT
Pedro Sánchez Palma, Universidad Politécnica de Cartagena, ES
Pedro Sanz Angulo, Universidad de Valladolid, ES
Pilar Mareca Lopez, Universidad Politécnica de Madrid, ES
Raul Laureano, ISCTE - Instituto Universitário de Lisboa, PT
Reinaldo Bianchi, Centro Universitário da FEI, BR
Rita Santos, Universidade de Aveiro, PT
Rodolfo Miranda Barros, Universidade Estadual de Londrina, BR
Rogério Eduardo Garcia, Universidade Estadual Paulista, BR
Rubén González Crespo, Universidad Internacional de La Rioja, ES
Rui Cruz, IST, Universidade de Lisboa, PT

Rui José, Universidade do Minho, PT
Rui Pedro Marques, Universidade de Aveiro, PT
Santiago Gonzales Sánchez, Universidad Inca Garcilaso de la Vega, PE
Sergio Gálvez Rojas, Universidad de Málaga, ES
Sérgio Guerreiro, Universidade Lusófona de Humanidades e Tecnologias, PT
Silvia Fernandes, Universidade do Algarve, PT
Solange N Alves de Souza, Universidade de São Paulo, BR
Tomás San Feliu, Universidad Politécnica de Madrid, España
Tomás San Feliu Gilabert, Universidad Politécnica de Madrid, ES
Valéria Farinazzo Martins, Universidade Presbiteriana Mackenzie, BR
Victor Hugo Medina Garcia, Universidad Distrital Francisco José de Caldas, CO
Vitor Carvalho, Instituto Politécnico do Cávado e do Ave, PT
Vitor Santos, ISEGI, Universidade Nova de Lisboa, PT
Wagner Tanaka Botelho, Universidade Federal do ABC, BR
Xose A. Vila, Universidad de Vigo, ES

Tendencias en Tecnologías de Información y Comunicación

Trends in Information and Communication Technologies

Jezreel Mejia¹, Mirna Muñoz¹

{jmejia, mirna.munoz}@cimat.mx

1. Centro de Investigación en Matemáticas- Unidad zacatecas, Parque Quantum, Ciudad del Conocimiento, Avenida Lasec, andador Galileo Galilei, manzana 3 lote 7, C. P. 98160, Zacatecas, México

DOI: 10.4304/risti.31.0

1. Introducción

Los artículos presentados en este número fueron seleccionados de los artículos aceptados en el Séptimo Congreso Internacional de Mejora de Procesos Software (CIMPS) 2018, realizado en la Ciudad de Guadalajara, Jalisco, México del 17 al 19 de Octubre del 2018. Este congreso ha tenido una tasa de aceptación del 30% de los artículos enviados.

La evolución y el uso creciente de las Tecnologías de Información y Comunicación (TICs) genera nuevas áreas de interés, demandando una constante investigación. Por lo tanto, los artículos publicados en este número especial de la Revista Ibérica de Sistemas y Tecnologías de la Información (RISTI), aborda temas relacionados a “Tendencias en Tecnologías de Información y Comunicación”. Los artículos presentan versiones extendidas y mejoradas de las versiones originales que han sido presentados en el congreso CIMPS 2018.

2. Estructura

El primer artículo, propone un Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información en empresas navieras. Para ello, los autores han seleccionado a MAGERIT, OCTAVE y MEHARI como metodologías de análisis de riesgos considerando su uso en otras iniciativas de construcción de modelos de madurez. Para validar el

artículo presenta entrevistas a siete empresas navieras a fin de conocer su postura frente a la problemática de riesgos en este contexto.

En el segundo artículo, se presentan los resultados de la validación de una herramienta desarrollada como apoyo para llevar a cabo actividades relacionadas con el aseguramiento de la calidad. Los resultados obtenidos de los autores demuestran que la herramienta permite un entendimiento común entre los diferentes profesionales involucrados en el desarrollo de software referente a los objetivos de calidad establecidos, reforzando la comunicación y el análisis de resultados para toma de decisiones oportuna.

El tercer artículo, identifica un aspecto fundamental para gestionar y ejecutar procesos empresariales a través del modelado de procesos. Por lo tanto, este artículo presenta una revisión de 20 artículos para identificar los algoritmos mas utilizados para el modelado de procesos.

El cuarto artículo, presenta un estudio exploratorio en el que se analiza la influencia de la Teoría de Belbin en actividades de desarrollo de software como mecanismo para la mejora del mismo. La actividad considerada en el estudio exploratorio para los equipos, fue la medición de software mediante la técnica de puntos de función.

El quinto artículo, muestra los resultados de un caso de estudio para la medición de la preservación de la privacidad. Con ello, se desarrollan modelos de aprendizaje máquina para determinar el riesgo de predicción de atributos sensibles y como medio de verificación de la utilidad de los datos. Los hallazgos motivan la necesidad de adecuar la medición de la preservación de la privacidad a los requerimientos actuales y a medios de ataque sofisticados como el aprendizaje máquina.

El sexto artículo, presenta una aplicación que permite detectar de manera oportuna aplicaciones maliciosas presentes en Smartphones, particularmente aquellos con sistema operativo Android. Además, los autores presentan un estudio de caso que ha permitido validar la app a través de la detección de aplicaciones maliciosas instaladas en un dispositivo móvil.

El séptimo artículo, propone un prototipo totalmente funcional para el monitoreo del transporte público. En este prototipo, los autores han utilizado un conjunto de recursos y medios tecnológicos que se centran en las capacidades de una computadora de placa reducida Raspberry PI. El principal objetivo es proporcionar información en tiempo real a los usuarios que utilizan los autobuses como medio de transporte, y al mismo tiempo, proporcionar datos históricos de cada unidad para conocer la información relacionada con los horarios y retrasos de las rutas realizadas en los diferentes circuitos.

El octavo artículo, presenta un estudio sobre el desarrollo y la evolución de las herramientas de hardware que han permitido la interacción entre los usuarios y los entornos virtuales en 3D. Al mismo tiempo, los autores proponen una clasificación en eras para evolución de los dispositivos de realidad virtual.

Finalmente, en el noveno artículo, se introduce PulAm, una aplicación móvil basada en el sistema operativo móvil Android. Los autores diseñan esta aplicación como una herramienta de apoyo en el proceso de monitoreo de plagas de diferentes cultivos. Como

caso de estudio, se introduce el monitoreo de cultivos de sorgo, en el estado de Nayarit, México; específicamente para la campaña contra el “pulgón amarillo” del sorgo.

3. Agradecimientos

Agradecemos al Director de la Revista Álvaro Rocha por otorgar la oportunidad para dar a conocer los trabajos mas relevantes presentados en el congreso CIMPS 2018. Asimismo, expresamos nuestra gratitud a los autores de los trabajos publicados, lo cual hizo posible este número especial en RISTI. Damos la enhorabuena a todos aquellos autores cuyos artículos han sido publicados en este número especial de RISTI.

Índice / Index

EDITORIAL

Tendencias en Tecnologías de Información y Comunicación	ix
<i>Jezreel Mejía, Mirna Muñoz</i>	

ARTIGOS / ARTICULOS / ARTICLES

Modelo para Medir la Madurez del Análisis de Riesgo de los Activos de Información en el contexto de las Empresas Navieras.....	1
<i>Fresia Yanina Holguín García, Lohana Mariella Lema Moreta</i>	
Disminuyendo la Brecha del Aseguramiento de la Calidad entre Gestor y Miembros del Equipo Mediante el Uso de Herramientas de Software.....	18
<i>Mirna Muñoz</i>	
Algoritmos de Minería de Proceso para el Descubrimiento Automático de Procesos	33
<i>Marcos Hernán Rivas, Sussy Bayona-Oré</i>	
Influencia de la Teoría de Roles de Belbin en la Medición de Software: Un estudio exploratorio	50
<i>Raúl Aguilar Vera, Julio Díaz Mendoza, Juan Ucán Pech</i>	
Marco de medición de la privacidad	66
<i>Luis Gustavo Esquivel-Quirós, Elena Gabriela Barrantes, Fernando Esponda Darlington</i>	
Detectando aplicaciones maliciosas en Smartphone con sistema Android a través del uso de una aplicación	82
<i>Jezreel Mejía</i>	
Sistema inteligente para el monitoreo automatizado del transporte público en tiempo real.....	94
<i>Yadira Quiñonez, Carmen Lizarraga, Juan Peraza, Oscar Zatarain</i>	
El Desarrollo del Hardware para la Realidad Virtual.....	106
<i>Graciela Lara, Alexis Santana, Andrés Lira, Adriana Peña</i>	
Una aplicación móvil para el monitoreo de cultivos: caso de estudio campaña contra el pulgón amarillo del sorgo.....	118
<i>Alejandra Perez-Mena, José Alberto Fernández-Zepeda, Juan Pablo Rivera-Caicedo, Himer Avila-George</i>	

Modelo para Medir la Madurez del Análisis de Riesgo de los Activos de Información en el contexto de las Empresas Navieras

Fresia Yanina Holguín García¹, Lohana Mariella Lema Moreta²

fholguin@uees.edu.ec, lohanalema@uees.edu.ec

¹ Universidad Espíritu Santo, Samborondón, Ecuador.

² Universidad Espíritu Santo, Samborondón, Ecuador.

DOI: 10.17013/risti.31.1-17

Resumen: El avance tecnológico ha contribuido al incremento de eventos disruptivos de diferente naturaleza en las empresas que podrían producir pérdida de información; por tal motivo, es vital realizar análisis de riesgos adecuado. Considerando que las empresas navieras no están exentas de amenazas, ataques o vulnerabilidades, este artículo propone un Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información en éste contexto empresarial. Para cumplir esto, se ejecutaron tres actividades. En la primera, se analizó y seleccionó MAGERIT, OCTAVE y MEHARI como metodologías de análisis de riesgos considerando su uso en otras iniciativas de construcción de modelos de madurez. Luego, se realizaron entrevistas a siete empresas navieras a fin de conocer su postura frente a la problemática planteada. Finalmente, podemos determinar que la construcción de esta propuesta conlleva la ejecución de un proceso de análisis de riesgo formalizado y con técnicas proactivas acorde al contexto empresarial establecido.

Palabras-clave: modelo de madurez; análisis de riesgo; empresas navieras.

Model for Measuring the Maturity of the Risk Analysis of Information Assets in the context of Shipping Companies

Abstract: Technological progress has contributed to the increase of disruptive events of different nature in companies that could produce loss of information; for this reason, it is vital to carry out adequate risk analysis. Considering that shipping companies are not exempt from threats, attacks or vulnerabilities, this article proposes a Model to Measure the Maturity of Risk Analysis of Information Assets in this business context. To accomplish this, three activities were executed. In the first, MAGERIT, OCTAVE and MEHARI were analyzed and selected as risk analysis methodologies considering their use in other maturity modeling initiatives. Afterwards, seven shipping companies were interviewed in order to know their position regarding the proposed problem. Finally, we can determine that the construction of this proposal involves the execution of a formalized risk analysis process and proactive techniques according to the established business context.

Keywords: maturity model; risk analysis; shipping companies.

1. Introducción

Las Tecnologías de Información y Comunicación (TIC) se han convertido en un factor que impacta en la competitividad de todo tipo de organización, por ello es fundamental que los procesos del negocio estén fusionados de manera integral de lo contrario aspectos como la efectividad del nivel operativo, imagen corporativa o la rentabilidad estarían disminuidas (Gómez, Pérez, Donoso y Herrera, 2010). En este sentido, Areitio (2008) evidencia que al adquirir e implementar nuevas tecnologías se debe ser consciente de la coexistencia de una amplia gama de amenazas, vulnerabilidades y ataques informáticos; y cuyo objetivo principal es acceder a la información privada de las organizaciones. Según ESET (2015) una de cada cinco empresas sufrió ataques de explotación de vulnerabilidades en el año 2014, y por su parte ISACA (2015) manifiesta que los incidentes de seguridad se han incrementado en un 66% desde el año 2009.

En este contexto, Chu, Wei, & Chang (2013) aseveran que las organizaciones están obligadas a establecer protocolos que les permitan identificar las falencias que afectan sus actividades y procesos, evaluar los controles existentes para disminuir la posibilidad de que un riesgo potencial se cristalice en una pérdida cierta, y adoptar medidas para reducir o controlar el riesgo en aquellos sectores donde se observa que se está por encima de los límites permisibles.

Gómez, Pérez, Donoso y Herrera (2010) señalan que ante la necesidad latente de garantizar una seguridad efectiva se crearon estándares, normas, metodologías y guías para llevar a cabo análisis que prevengan, controlen y reduzcan los riesgos asociados a la violación o vulnerabilidad de la información. Además, según Jugdev & Thomas (2002) también se deben considerar los *Modelos de Madurez* orientados a la seguridad de la información como una alternativa para evaluar en toda su dimensión la situación organizacional en el manejo del riesgo respecto a una situación ideal, y considerando que estos se cimentan en el conocimiento de los procesos la toma de decisiones es más acertada.

Aunado a lo anterior, Becker, Niehaves, Poppelbus, & Simons (2010) establecen que un modelo de madurez en seguridad informática deberá evaluar desde los procesos incongruentes hasta los más óptimos de una empresa, lo cual permitirá identificar rápidamente las áreas donde la entidad debe enfocarse para mejorar y ello conllevará la implementación de buenas prácticas.

Por otra parte, según un estudio realizado por Llop et al. (2013) las empresas de transporte marítimo denominadas *Agencias Navieras* no están exentas de eventos conexos a la seguridad informática, además no cuentan con una herramienta específica para la identificación y categorización del riesgo de su ambiente tecnológico y considerando que sus actividades están relacionadas a la documentación, transporte terrestre de los contenedores, verificación de la carga, seguimiento de la carga desde el puerto origen al puerto destino, y otras operaciones portuarias; es preciso proponer un modelo que satisfaga este requerimiento no atendido.

Por lo antes expuesto el objetivo principal de este trabajo es diseñar un *Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información en el contexto de las Empresas Navieras*, de tal manera que permita autoevaluar la realidad de la

organización y establecer una ruta de mejoramiento con estrategias acordes al giro del negocio. Este modelo se cimentará en las mejores prácticas de las metodologías MAGERIT, OCTAVE y MEHARI; debido a que cuentan con una estructura definida, documentada y adaptable a todo tipo de empresa; sus fases de análisis de riesgo son homólogas, permiten distinguir los activos y sus riesgos de forma que sean un punto de referencia para tomar decisiones y hacer mejoras en los procesos internos organizacionales; y admiten después de su implementación la construcción de un plan de actividades que puede ser detallado, monitoreado y examinado de forma periódica.

Para alcanzar el objetivo planteado se construirá un modelo con tres elementos: *Niveles de Madurez, Categorías a Evaluar y Mapa de Control*; los cuales estarán sustentados en la revisión bibliográfica de las metodologías escogidas y de la profundización del modelo de madurez CMMI.

2. Marco Teórico

2.1. Análisis de Riesgo

Según Del Carpio (2006) el análisis de riesgo es un proceso de origen cuantitativo o cualitativo que permite evaluar los riesgos, y que además consiente una estimación de incertidumbre e impacto. Por su parte, Sikdar (2011) afirma que el análisis de riesgo es una etapa que comprende dos aspectos claves: determinar las amenazas a las cuales está expuesta una entidad y valorar su posible materialización. En cambio, Aguilera (2010) señala que un análisis de riesgo conlleva conocer todos los activos de información que componen un sistema para estipular su grado de vulnerabilidad y el impacto que un ataque causaría; lo cual será el sustento para seleccionar las medidas de protección más oportunas.

En este contexto, Curiman y Toth (2004) establecen que el análisis de riesgo no es un proceso aislado respecto a otras iniciativas que conllevan a un nivel óptimo de seguridad, al contrario, es una línea base que permitirá identificar con rapidez las áreas más vulnerables logrando también que la implantación de las salvaguardas sea acorde a las necesidades requeridas por el entorno.

Según Aguilera (2010) es conveniente conocer la diferencia entre el análisis y la gestión de riesgo; el análisis proporciona un bosquejo del sistema en términos de activos, amenazas y salvaguardas; es decir, que es la base para ejercer un control de todas las actividades; mientras que, la gestión se refiere a la estructuración de las acciones de seguridad para satisfacer las insuficiencias detectadas por el análisis.

Por otra parte, Echenique (2012) considera que dentro del análisis de riesgo es preciso distinguir seis elementos claves:

1. *Probabilidad*: estimación de la posibilidad que un recurso informático quede expuesto a un evento (Canal, 2004).
2. *Amenazas*: cualquier tipo de acción que puede producir un daño material o inmaterial en una empresa, y cuya naturaleza puede ser: física o lógica (Aguilera, 2010).

3. *Vulnerabilidades*: característica o circunstancia de debilidad de un recurso que posibilita la materialización de una amenaza (Canal, 2004).
4. *Activos*: recursos o elementos relacionados a un sistema de información y que tienen valor para la organización (Aguilera, 2010).
5. *Impacto*: es el efecto de una amenaza sobre un activo (Eterovic y Pagliari, 2011).
6. *Riesgo*: estimación del grado de exposición que una amenaza se materialice sobre uno o más activos causando daños a la organización (Gupta, & Xu, 2010).

Areitio (2008) señala que para llevar a cabo un análisis de riesgo robusto es recomendable utilizar los lineamientos establecidos por las metodologías vigentes, las cuales se caracterizan por contemplar entre sus fases aspectos como: realización de un inventario y valoración de los activos, identificación y estimación de las amenazas que puedan afectar a la seguridad de los activos, establecimiento y evaluación de las medidas de seguridad existentes, determinación y ponderación de las vulnerabilidades de los activos, y valoración del impacto que produciría un ataque.

En este ámbito, Alemán y Rodríguez (2014) agregan que las metodologías más sobresalientes en este dominio son: OCTAVE, MAGERIT, MEHARI, NIST SP 800:30, Coras, Cramm y Ebios; y aunque tienen características similares difieren esencialmente en la manera de estimar la probabilidad de ocurrencia de una amenaza y en la forma de determinar el impacto en la organización. Es preciso acotar que este artículo solo abarcará OCTAVE, MAGERIT Y MEHARI debido a que su proceso de análisis de riesgo es análogo, y porque las metodologías excluidas requieren un costo de licencia que las convierte en poco atractivas para su implementación.

2.2. OCTAVE

Alberts & Dorofee (2002) señalan que OCTAVE (Operationally Critical Threat, Asset and Vulnerability Evaluation) es una metodología desarrollada en el año 2001 por el CERT/CC, en la cual la tecnología es examinada en relación a las prácticas de seguridad y la toma de decisiones respecto a la protección de información, se basa en los riesgos de confidencialidad, integridad y disponibilidad a los que pueden estar sometidos los bienes con información crítica. En este mismo ámbito Januszkiewicz & Marek (2006) señalan que OCTAVE se diferencia de otras metodologías porque no se limita al análisis del riesgo tecnológico y aspectos tácticos, también abarca el riesgo de la organización y se enfoca en temas estratégicos relacionados con la práctica.

Según Alberts, Dorofee, Stevens & Woody (2003) OCTAVE es autodirigido, lo que significa que el personal de la organización asume el compromiso de establecer la estrategia de seguridad, por ende, es necesario la integración del departamento operativo y de TI así se equilibrarán los aspectos de: riesgos operativos, prácticas de seguridad y tecnología. Además, CERT (2013) indica que para la ejecución de OCTAVE se cuenta con tres fases claramente definidas:

- *Fase 1, Visión Organizativa*: En esta etapa se determinan los activos críticos, requerimientos de seguridad y amenazas sobre los activos críticos, prácticas de seguridad actuales y las vulnerabilidades organizativas (CERT, 2013).
- *Fase 2, Visión Tecnológica*: Se examinan los componentes claves y las vulnerabilidades tecnológicas vigentes (CERT, 2013).

- *Fase 3, Estrategia y Desarrollo del Plan:* Se crean medidas, estrategias y planes de mitigación de riesgo utilizando los resultados obtenidos de las fases anteriores (CERT, 2013).

Hasta la presente fecha esta metodología consta de los métodos: OCTAVE, OCTAVE – S, y OCTAVE ALLEGRO; los cuales se basan en los criterios del estándar con un enfoque en la práctica y evaluación de la seguridad fundamentada en la información de riesgo de la entidad objeto de análisis (Alberts, Dorofee, Stevens & Woody, 2003). A continuación, se describen las principales características de cada uno:

- *Método OCTAVE:* Fue desarrollado para organizaciones de treientos o más empleados, considerando la jerarquía de múltiples capas, así como la infraestructura que suelen tener este tipo de empresas. Durante su desarrollo se examinan los lineamientos organizacionales y tecnológicos, se crea una visión clara de la organización, y se definen sus necesidades de seguridad; por ello este método se enfoca en: identificar los elementos críticos y las amenazas de los activos, establecer las vulnerabilidades, y desarrollar una estrategia basada en las prácticas y planes de mitigación de riesgos (Alberts & Dorofee, 2002).
- *Método OCTAVE-S:* Fue concebido para las pequeñas organizaciones (con un personal de entre veinte a ochenta empleados), y a pesar de ser un método más simplificado produce el mismo tipo de resultados que OCTAVE. Para llevar a cabo su ejecución se requiere un equipo conformado por tres o hasta cinco personas, las cuales deben conocer a profundidad el sentido organizacional de la empresa pues se encargarán de recopilar información sobre los elementos significativos, los requisitos de seguridad, las amenazas y las estrategias de protección. Debido a que las entidades de este tipo generalmente externalizan servicios TI este método solo incluye una exploración limitada de la infraestructura informática (Januszkievicz & Marek, 2006).
- *Método OCTAVE ALLEGRO:* Es una variante simplificada del método de OCTAVE, sin embargo, debido a que su enfoque principal son los activos de la información, la identificación de otros importantes recursos se realiza en función de los activos de información a la que están conectados; lo cual elimina una posible confusión sobre el alcance de la evaluación (Caralli, Stevens, Young & Wilson, 2007).

2.3. MAGERIT

MAGERIT (Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información) es una metodología elaborada por el Consejo Superior de Administración Electrónica de España en respuesta a la percepción de que el gobierno dependía cada vez más de la tecnología de la información para conseguir sus objetivos de servicio, sin embargo, en la actualidad es de carácter público siendo utilizada por organizaciones de todo el mundo (Sylin, Hori & Sakurai, 2009).

Nagata, Amagasa & Kigawa (2009) señalan que MAGERIT tiene como objetivos: crear conciencia de la existencia de riesgos y la importancia de tratarlos a tiempo; proporcionar un método sistemático para analizar los riesgos garantizando la autenticidad, confidencialidad, integridad, disponibilidad y trazabilidad; diseñar estrategias para

mantener los riesgos controlados; y preparar a la empresa para los procesos de auditorías, certificaciones y acreditaciones. MAGERIT propone la realización de cinco pasos para efectuar el análisis de riesgos, los cuales se detallan en la Figura 1.

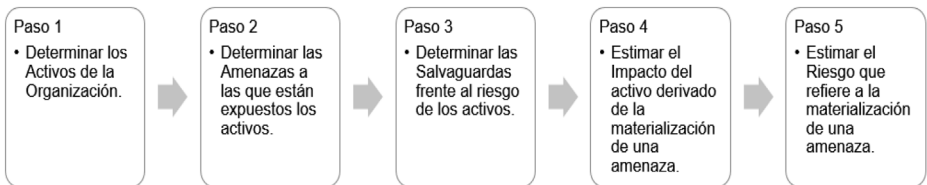


Figura 1 – Pasos para el Análisis de Riesgos de MAGERIT (Ministerio de Hacienda y Administración Pública del Gobierno de España, 2012).

2.4. MEHARI

MEHARI, también conocido como un método de análisis de riesgo armonizado, es una metodología desarrollada en el año 1995 por CLUSIF (**CLUB de la Sécurité de l'Information Français**) con la finalidad de que los responsables de la seguridad informática evalúen cuantitativamente o cualitativamente (según sea necesario) los principales factores de riesgos que puede percibir una organización según su contexto, y para ello se requiere que la entidad establezca previamente una política de seguridad y mantenimiento de riesgos a un nivel convenido, la misma que servirá de referencia para que el acople de los objetivos estratégicos existentes sea acorde a los nuevos métodos de funcionamiento de la empresa (Alemán y Rodríguez , 2014).

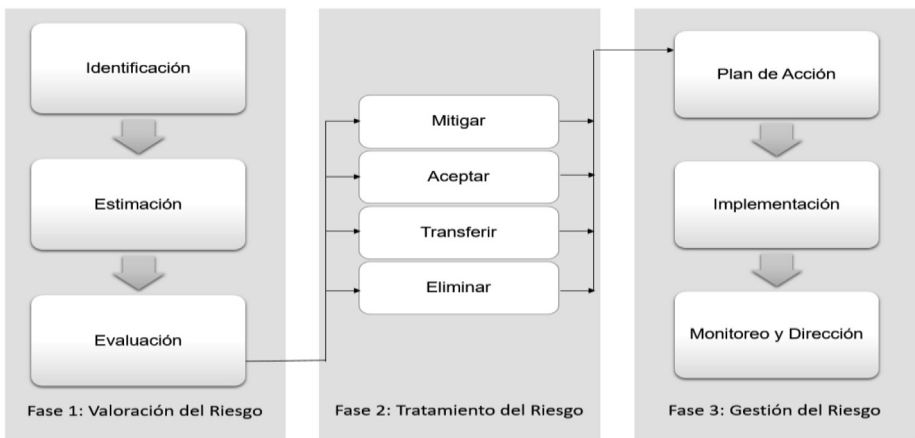


Figura 2 – Proceso de evaluación, tratamiento y gestión del Riesgo de MEHARI (CLUSIF, 2010).

MEHARI está estructurado por tres módulos: el primero tiene como finalidad el análisis de riesgos, el segundo está orientando a la evaluación de seguridad (con énfasis al análisis de vulnerabilidades) y el tercero permite el análisis de amenazas; cabe señalar que la ejecución de los mismos consentirá el diseño e implantación de los planes de acción que fomentaran la seguridad de la información (Aleman y Rodríguez, 2014).

El proceso de evaluación, tratamiento y gestión del riesgo de MEHARI, se exponen en la Figura 2.

2.5. Modelo de Madurez

Rosemann & De Bruin (2005) definen el término *madurez* como una medida para evaluar la capacidad de una organización respecto a una determinada disciplina. En cambio, Mettler (2009) afirma que la madurez es un proceso evolutivo en la demostración de una habilidad específica.

Partiendo de estas definiciones OPM3 (2003) establece que un modelo de madurez es un esquema con niveles jerárquicos, que en esencia permite a una organización comprender su situación actual y orientarla a la consecución de un nivel más elevado lo cual requiere la implementación de mejores prácticas o rutas de mejora. También, Saavedra, Dávila, Melendez & Pessoa (2017) evidencian que un modelo de madurez es un conjunto de elementos organizados en una estructura evolutiva con transiciones medibles entre niveles. Además, Carvalho, Rocha, & Abreu (2017) añaden que un modelo de madurez se sustenta en el principio de que las personas, las organizaciones, los procesos, etc., evolucionan hacia una mayor madurez por consecuencia su desarrollo abarca varias etapas diferentes.

En este mismo ámbito, Klimko (2001) señala que de forma general los modelos de madurez tienen como propiedades: un número limitado de niveles (generalmente de cuatro a seis), cada nivel consta de requisitos determinados que deben ser alcanzados, y los niveles de madurez están clasificados de forma secuencial siendo el último un nivel de perfección.

Por otra parte, considerando la importancia de evaluar la capacidad de una organización respecto a la posibilidad para cumplir con los objetivos de seguridad de la información se han originado varios modelos de madurez, sin embargo, es el CMMI (Capability Maturity Model Integrated) el utilizado con mayor preponderancia como marco de referencia para el desarrollo de nuevos modelos (Matrane, Talea & Okar, 2014).

Según Palomino, Dávila, Melendez & Pessoa (2017) el CMMI es un modelo que agrupa las mejores prácticas en actividades de desarrollo y mantenimiento. En este sentido, Crawford (2002) expone que el CMMI está orientado a la industria del software y contiene una cantidad de áreas definidas por proceso, las cuales cubren conceptos básicos que son indispensables para la mejora de los mismos. También Kerzner (2000) señala que este modelo clasifica a las organizaciones en cinco niveles de madurez basados en el grado de sofisticación de sus prácticas de ingeniería, los cuales son: inicial, repetible, definido, administrado y optimizado. En la Figura 3 se esquematizan estos niveles.

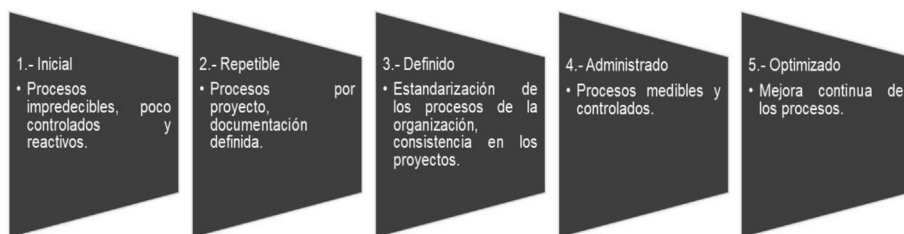


Figura 3 – Niveles de Madurez CMMI (Crawford, 2002).

3. Metodología

El enfoque de la investigación es de origen *Cualitativo* porque se utilizará la teoría como fundamento en la elaboración del Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información, en lugar de implementar una fundamentación empírica; mientras que el alcance es *Descriptivo*, debido a que se identificará la contribución del modelo en las empresas navieras a través de la profundización de información sobre un contexto en particular, de tal manera que se logre una descripción de sus características.

Por otra parte, el desarrollo de la investigación se llevará a cabo en tres fases:

1. Revisión de trabajos relacionados al modelo de madurez CMMI y las metodologías MAGERIT, OCTAVE y MEHARI.
2. Análisis de las Empresas Navieras en contexto a las Tecnologías de Información.
3. Construcción del Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información en el contexto de las Empresas Navieras basado en los resultados obtenidos del paso 1 y 2.

Para establecer las necesidades de las Agencias Navieras en relación al análisis de riesgo se utilizó como población las 17 entidades constituidas legalmente en el Ecuador (Autoridad Portuaria de Manta, 2016), de las cuales siete se encuentran en la ciudad de Manta, cinco en Guayaquil, tres en Esmeraldas, una en La Libertad y una en Galápagos. Considerando que el mayor número de empresas se encuentran radicadas en Manta y poseen una amplia trayectoria en el mercado se tomó como muestra los siete entidades de esta urbe. Los informantes claves fueron el personal del Departamentos de TI de la muestra obtenida, a quienes se les formuló como técnica de recolección de información la entrevista.

4. Análisis de Resultados

4.1. Revisión de trabajos relacionados al modelo de madurez CMMI y las metodologías MAGERIT, OCTAVE y MEHARI.

Mayer y Lemes (2008) crearon un *Modelo de Madurez para Evaluar los Procesos de Gestión de Riesgos en Seguridad de la Información* considerando como referencia el CMMI, debido a que una de sus principales fortalezas es la posibilidad de representar el modelo por etapas o de manera continua; siendo esta última opción la más idónea para establecer que cada actividad o buena práctica para la gestión de riesgo pueda alcanzar un nivel de madurez independientemente del nivel alcanzado por las demás actividades,

con lo que la empresa logra verificar en cuál de las actividades necesita enfocarse más. El modelo utiliza la Norma ISO/IEC 27005 y su diseño está fundamentado en la experiencia (empírico).

Por otra parte, Sotelo, Torres y Rivera (2012) desarrollaron un *Proceso Práctico de Análisis de Riesgos de Activos de Información* utilizando como eje de su temática la metodología MAGERIT porque los subprocesos planteados en el análisis de riesgos eran satisfactorios para el desarrollo de las acciones de seguridad que requiere la gestión del riesgo de los activos de información, pero los autores le agregaron un proceso de Análisis de Impacto del Negocio (BIA) para definir el impacto de la no disponibilidad de los servicios tecnológicos de la información. El proceso propuesto fue implementado en una entidad pública donde los resultados fueron satisfactorios.

En cambio, Amador (2014) definió una propuesta denominada *Gestión de Riesgo con base a la ISO 27005 adaptando OCTAVE-S*, en la cual manifiesta que este método provee un instrumento para identificar ágilmente las amenazas más relevantes en los activos de información proporcionando a su vez una estimación cualitativa de la probabilidad e impacto; lo cual ayuda a definir las estrategias de protección y controles para el tratamiento del riesgo. La propuesta se evaluó en un centro de estudios universitario privado con lo cual se logró una reducción significativa del riesgo luego de la implantación de los respectivos controles sugeridos en el tratamiento.

A través de los trabajos anteriormente expuestos queda evidenciado que el modelo CMMI puede ser utilizado en combinación con otros estándares, además que las metodologías MAGERIT y *OCTAVE-S* han generado resultados exitosos en el análisis y gestión de riesgos en instituciones públicas como privadas.

4.2. Análisis de las Empresas Navieras en contexto a las Tecnologías de Información

Para definir sus necesidades en relación al análisis de riesgo de los activos de información se utilizó como técnica de recolección de información la entrevista; la misma que se realizó a la muestra establecida (siete Agencias Navieras del país) y se escogió a siete empleados (un empleado de TI por entidad) que cumplieran actividades de infraestructura y seguridad. Es preciso señalar, que dos de las entidades escogidas pertenecen al grupo de grandes empresas, cuentan con una media de treinta y tres años en el mercado y su Área de TI la conforman seis personas; mientras que las otras cinco agencias son medianas empresas, tienen un promedio de diecisiete años de trayectoria y sus Departamentos de TI poseen una media de tres empleados. Las preguntas formuladas fueron:

- ¿Considera que existe una Metodología adecuada para realizar un Análisis de Riesgo en las Entidades Navieras?
- ¿Cuáles son los principales problemas que afronta el Departamento de TI durante el proceso de Análisis de Riesgos?
- ¿Considera usted que un Modelo de Madurez que determine el status actual en relación al Análisis de Riesgos de los Activos de Información, sería una herramienta útil en el Departamento de TI?

La información obtenida en las entrevistas ha permitido concluir que las Agencias Navieras tienden a expandirse a nuevas áreas geográficas, lo cual requiere maximizar

el uso de la tecnología para crear redes mundiales de puertos y así ofrecer niveles consistentes de servicios; pero esto genera un desconocimiento en la categorización del riesgo que pueden tolerar según la madurez de los procesos que se ejecutan en la actualidad, además que estos nuevos retos implican una capacitación permanente de todo el personal para poder evitar brechas en conocimientos. También, los cambios de jefes en el área de TI suelen ser un obstáculo en el diseño e implantación de estrategias unificadas y perdurables, lo cual se acrecienta con una política de seguridad ambigua.

Partiendo de las inferencias antes descritas y considerando que Applegate, McFarlan, & Austin (2002) señalan que en un negocio correctamente estructurado deben existir políticas, procesos e indicadores; que estén alineados a una adecuada identificación y tratamiento de riesgos; se han establecidos dos lineamientos para el desarrollo del modelo: estrategias claves el proceso de análisis de riesgo, y estrategias conexas al negocio en relación al riesgo tecnológico.

4.3. Construcción del Modelo para Medir la Madurez del Análisis de Riesgos de los Activos de Información en el contexto de las Empresas Navieras

La construcción del modelo prototipo se basa en los resultados obtenidos de las fases anteriores; y debe responder a las siguientes interrogantes: ¿Se adapta a los requerimientos de las Agencias Navieras?, ¿Las buenas prácticas sugeridas se enmarcan en las metodologías MAGERIT, OCTAVE y MEHARI? y ¿El modelo abarca las fases esenciales del análisis de riesgo?

Para definir el modelo se han desarrollado tres elementos: Niveles de Madurez, Categorías a Evaluar y Mejores Prácticas, y Mapa de Control.

4.3.1. Propuesta: Niveles de Madurez

El modelo de madurez propuesto consta de cinco niveles secuenciales que abarcan desde un nivel ad hoc hasta un nivel sofisticado; y para su elaboración se analizaron las directrices del CMMI en relación al grado de practicidad de los procesos que

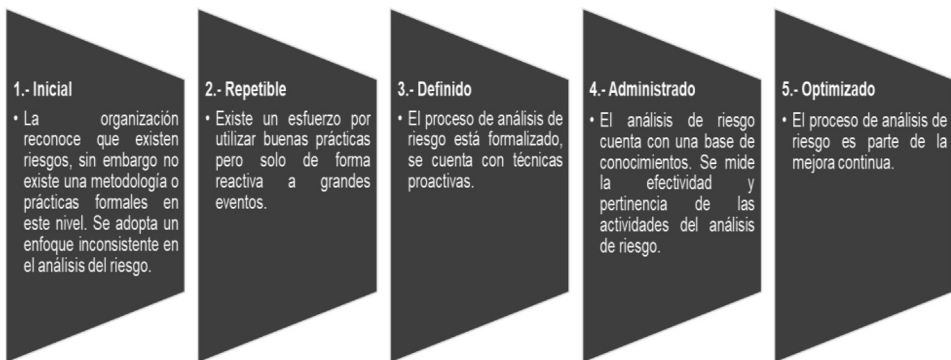


Figura 4 – Niveles de Madurez del Modelo Propuesto con base al CMMI (Elaboración propia, 2017).

conllevar el análisis de riesgos. En la Figura 4 se detallan los niveles de madurez con sus características.

4.3.2. Propuesta: Categorías a Evaluar y Mejores Prácticas

El modelo propuesto define once Categorías a Evaluar, de los cuales A, B, C y D corresponden a estrategias relacionadas al negocio y al riesgo tecnológico, y las categorías restantes comprenden los aspectos claves que se requieren en el proceso de análisis de riesgo. Además, por cada categoría se han establecido mejores prácticas en contexto a las metodologías MAGERIT, OCTAVE y MEHARI. En la Tabla 1 se evidencian los entes antes señalados, cabe detallar que en la tabla se encuentra marcado a que metodología corresponde la buena práctica.

4.3.3. Propuesta: Mapa de Control

El Mapa de Control propuesto expone las Categorías a Evaluar con las Mejores Prácticas que se sugieren estén implementadas según el nivel de madurez (se utilizaron los resultados obtenidos en la Figura 4 y Tabla 1). En la Tabla 2 se detalla el Mapa de Control.

5. Conclusiones

Un Modelo de Madurez es una medición estructurada del desempeño organizacional que permite identificar vacíos o debilidades para así establecer procesos de mejora continua. Dentro de estos modelos destaca el CMMI cuya principal fortaleza es la priorización de los esfuerzos iniciales como soporte para la transición evolutiva, por tal motivo fue escogido como referencia para delimitar los niveles y especificaciones del modelo propuesto.

Las metodologías MAGERIT, OCTAVE y MEHARI tienen características propias pero complementarias entre sí, lo que permite que al combinarlas se obtenga un análisis de riesgo más robusto; por ello para el modelo diseñado se escogieron las prácticas primordiales asociadas a los activos de información evitando las ambigüedades de estos estándares.

El Modelo planteado es el resultado de integrar los niveles de madurez con las mejores prácticas de las metodologías antes descritas; lo cual fue posible a través de la Propuesta de un Mapa de Control, el mismo que orienta su consecución paulatina. No obstante, las especificaciones de requerimientos pueden ser halladas en la documentación formal de las metodologías.

El modelo diseñado pretende establecer el grado de madurez en que se encuentra la organización según las prácticas de análisis de riesgo vigentes, lo que permitirá que a partir de ello se enfoquen en las debilidades o falencias que cada categoría evalúa, se tracen estrategias de mejora y se autoevalúe su cumplimiento; todo esto con la finalidad que en el futuro se alcance el nivel de perfección.

Por otra parte, el modelo propuesto tiene como limitante su validez teórica, por ello en una siguiente investigación, a través del método Delphi, se convocará a profesionales del sector para medir la eficiencia de las *Categorías a Evaluar* definidas, así como la pertinencia de las *Mejores Prácticas* según el nivel de madurez; lo cual podrá ser contrastado con los resultados obtenidos en este artículo.

Mejores Prácticas		MAGERIT	OCTAVE	MEHARI
Categorías a Evaluar				
A. Política de Riesgo	A.1.- Existe una política formal de riesgo aprobada por la autoridad responsable y que transcende a pesar de cambios de personal en la alta dirección.	X	X	
	A.2.- La política ha sido comunicada a toda la entidad.	X	X	
	A.3.- La política de riesgo es revisada y actualizada para incluir los cambios del entorno interno y externo.	X		
	A.4.- La política incluye una definición de la cantidad de riesgo (apetito de riesgo) que la entidad puede aceptar, y puede estar descrita en términos cuantitativos o cualitativos.	X		
B.- Responsabilidad	B.1.- La entidad posee un departamento de TI con un equipo de análisis de riesgo.	X		
	B.2.- Los roles del equipo de análisis de riesgo están claramente definidos, asignados y documentados. Por ejemplo: El personal verifica que las prácticas de respuestas al riesgo sean acordes a la política establecida, coordina los planes de acción así como también la consistencia y exactitud del proceso, aprueba el apetito de riesgo en la entidad.	X		
C.- Compromiso de la Alta Dirección	C.1.- La alta dirección es un apoyo activo para el equipo de análisis de riesgo, así por ejemplo: verifica que la política sea compatible con las necesidades de la empresa, evalúa la efectividad del proceso (revisión periódica de los informes del equipo de análisis riesgo) y está dispuesto a participar del mismo si es necesario.		X	
	C.2.- La alta dirección garantiza los recursos necesarios para llevar a cabo el análisis de riesgo, así como la formación y entrenamiento del personal.		X	
D.- Comunicación y Formación	D.1.- La empresa posee un personal altamente calificado en el departamento de TI, por ello se promueve su capacitación permanente.		X	
	D.2.- Existen programas de capacitación formalizada para establecer una cultura de riesgo y asegurar que el personal de la entidad entienda la política de riesgo.	X		
	D.3.- Se informa periódicamente y de forma inmediata a la alta dirección la consolidación de un riesgo.	X	X	
E.- Determinación y Valoración de los Activos de Información	E.1.- Los activos/recursos TI han sido clasificados según una perspectiva de: servicios, datos/información, aplicaciones, equipos informáticos, redes de comunicación, soporte de información, equipamiento auxiliar, instalaciones y personal.	X	X	X
	E.2.- Se cuenta con un perfil que describe características únicas, cualidades, dependencia (medida en que un activo superior se vea afectado por un incidente de seguridad en un activo inferior) y valor de los activos de información.		X	
Valorar los Activos Identificados	E.3.- Existe una escala de valoración definida, por ejemplo: de tipo logarítmica (la misma tiene como objetivo hacer una valoración cualitativa respondiendo a valoraciones subjetivas por parte del personal de la organización).	X	X	X

El Negocio y el Riesgo Tecnológico

Determinar las amenazas de los activos	F.- Identificación y estimación de Amenazas	F.1.- La empresa analiza exhaustivamente las amenazas externas e internas a los que pueden verse afectados los activos de información.	X	X	X
		F.2.- Por cada amenaza identificada se ha definido la dimensión de disponibilidad, integridad, y confidencialidad; que puede resultar afectada en el activo de información.	X	X	X
	Estimar el impacto de la amenaza	G.1.- Se ha establecido una escala de impacto considerando la gravedad de las consecuencias de la materialización de una amenaza; por ejemplo: reputación, pérdida de clientes, costos operativos, pérdida de ingresos, pérdidas financieras, horas de trabajo del personal, pérdidas significadas de vida y seguridad.		X	
Determinación del riesgo	H.- Evaluación del Riesgo	G.2.- Se ha ponderado el impacto acumulado tomando en cuenta la degradación causada por la amenaza, y el impacto repercutido teniendo en cuenta el valor propio del activo y las amenazas a los que está expuesto.		X	
		H.1.- Se ha definido el riesgo acumulado considerando la degradación causada y la frecuencia de la amenaza; y se ha definido el riesgo repercutido calculando el daño en los activos explícitamente valorados.	X		
		H.2.- Los riesgos son clasificados a partir de una escala que considera la probabilidad e impacto, con la finalidad de priorizar los más significativos.	X	X	X
Plan de Acción	I.- Respuesta a los Riesgos	I.1.- Se determina una respuesta por cada riesgo identificado (la cual es producto de una socialización y consenso) considerando su probabilidad e impacto.	X	X	X
		I.2.- Cada respuesta al riesgo está perfectamente desplegada, configurada, mantenida y recargará en una de estas categorías: <i>Mitigar</i> , contiene una o varias estrategias de mitigación, las cuales describen los controles a implantar y los recursos requeridos; <i>Acceptar</i> , no deben acarrear graves consecuencias a la organización; <i>Transferir</i> , deberán estar incluidos en el plan financiero de la empresa; <i>Eliminar</i> , constan de medidas estructurales para lograr su consecución.	X	X	X
		I.3.- Se ha definido el riesgo residual que permanecerá cuando una respuesta al riesgo se implemente.	X	X	X
Monitoreo y Dirección	J.- Actividades de Control	J.1.- Se definen indicadores de desempeño sobre la respuesta a los riesgos para determinar su validez.	X		X
		J.2.- Los efectos de las respuestas a los riesgos se miden frente al apetito de riesgo.		X	
		J.3.- Se evalúa el riesgo residual una vez aplicada la respuesta al riesgo.	X	X	
Fases del Análisis de Riesgo	K.- Mejora Continua del Análisis de Riesgo	K.1.- Exhaustiva recopilación de información para el seguimiento, revisión y aprendizaje del análisis de riesgo.	X		X
		K.2.- La presencia de nuevos riesgos se identifica sistemáticamente de manera oportuna y proactiva.	X	X	X

Tabla 1 – Propuesta de Categorías a Evaluar y Mejores Prácticas con base en las metodologías MAGERIT, OCTAVE y MEHARI (Elaboración propia, 2017).

Categorías a Evaluar	Nivel 1: Inicial	Nivel 2: Repetible	Nivel 3: Definido	Nivel 4: Administrado	Nivel 5: Optimizado
A. Política de Riesgo	No se esperan prácticas formales.		A1, A2	A3, A4	
B.- Responsabilidad		B1	B2		
C.- Compromiso de la Alta Dirección			C1	C2	
D.- Comunicación y Formación			D1	D2, D3	
E.- Determinación y Valoración de los Activos de Información		E1	E2, E3		
F.- Identificación y estimación de Amenazas			F1	F2	
G.- Estimación de Impacto			G1	G2	
H.- Evaluación del Riesgo			H1, H2		
I.- Respuesta a los Riesgos			I1, I2	I3	
J.- Actividades de Control				J1	J2, J3
K.- Mejora Continua del Análisis de Riesgo					K1, K2

Tabla 2 - Mapa de Control propuesto (Elaboración propia, 2017).

Referencias

- Aguilera, P. (2010). *Seguridad Informática*. Madrid, España: Editex.
- Alberts, C., & Dorofee, A. (2002). *Managing Information Security Risks: The OCTAVE Approach*. Boston, USA: Addison-Wesley,
- Alberts, C., Dorofee, A., Stevens, J., & Woody, C. (2003). *Introduction to the OCTAVE® Approach*. Recuperado de <https://www.itgovernance.co.uk/files/Octave.pdf>
- Alemán, H., & Rodríguez, C. (2014). Metodologías para el análisis de riesgo en los SGSI. *Revista Especializada en Ingeniería*, 9, 73-86. Recuperado de <http://oaji.net/articles/2017/5082-1501187567.pdf>
- Amador, S. (2014). *Gestión de Riesgo con base a la ISO 27005 adaptando OCTAVE-S*. (Tesis de Maestría). Universidad Internacional de la Rioja, Logroño, España.
- A.P.M. (2016). *Agencias Navieras*. Recuperado de <http://www.puertodemanta.gob.ec/clientes/agencias-navieras>
- Areitio, J. (2008). *Seguridad de la Información: Redes, Informática y Sistemas de Información*. Madrid, España: Paraninfo.

- Becker, J., Niehaves, B., Poppelbus, J., & Simons, A. (2010). Maturity Models in IS Research. *ECIS*, 42, 1-12. Recuperado de <https://webdocs.uni.li/public/04046167.PDF>
- Canal, V. (2006). *Seguridad de la información, Expectativas, Riesgos y Técnicas de Protección*. Ciudad de México, México: Editorial LIMUSA.
- Caralli, R., Stevens, J., Young, L., & Wilson, W. (2007). *Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process*. Recuperado de https://resources.sei.cmu.edu/asset_files/TechnicalReport/2007_005_001_14885.pdf
- Carvalho, J., Rocha, Á., & Abreu, A. (2017). HISMM - Hospital Information System Maturity Model: A Synthesis. En: Mejía J., Muñoz M., Rocha Á., San Feliu T., Peña A. (Eds). *Trends and Applications in Software Engineering, CIMPS 2016. Advances in Intelligent Systems and Computing*, vol 537. Cham: Springer.
- CERT. (2013). *OCTAVE*. Recuperado de <http://www.cert.org/resilience/products-services/octave/>
- Chu, Y., Wei, Y., & Chang, W. (2013). A risk recommendation approach for information security risk assessment. *IEICE*, 10, 1-3. Recuperado de <http://i-scover.ieice.org/proceedings/APNOMS/2013/pdf/P3-10-116166.pdf>
- CLUSIF. (2010). *MEHARI: Risk analysis and treatment Guide*. Recuperado de <http://meharipedia.x10host.com/wp/wp-content/uploads/2016/12/MEHARI-2010-Risk-Analysis-and-Treatment-Guide.pdf>
- Crawford, J. (2002). *Project management maturity model: Providing a proven path to project management excellence*. New York: Marcel Dekker.
- Curiman, F., & Toth, G. (2004). *Análisis de riesgos. Técnicas, metodologías y herramientas para el desarrollo de un Análisis de Riesgos*. Recuperado de <https://es.scribd.com/document/96314331/Analisis-de-Riesgos-Curiman-Toth>
- Del Carpio, J. (2006). *Análisis del riesgo en la administración de proyectos de tecnología de información*. Perú: Universidad Nacional Mayor de San Marcos.
- Echenique, J. (2012). *Auditoría Informática*. Ciudad de México, México: McGraw-Hill.
- ESET Security Report. (2015). *ESET Security Report, Latinoamérica 2015*. Ciudad de México: ESET.
- Eterovic, J., & Pagliari, G. (2011). Metodología de Análisis de Riesgos Informáticos. *Cyta*, 10(1), 1-5. Recuperado de <http://www.cyta.com.ar/ta1001/v10n1a3.htm>
- Gómez, R., Pérez, D., Donoso, Y., & Herrera, A. (2010). Metodología y Gobierno de la Gestión de Riesgos de Tecnologías de la Información. *Revista de Ingeniería Universidad de los Andes*, (31), 109-118. Recuperado de <http://www.scielo.org.co/pdf/ring/n31/n31a12.pdf>
- Gupta, S., & Xu, H. (2010). Examining the Relative Influence of Risk and Control on Intention to Adopt Risky Technologies. *Journal of technology management & innovation*, 5(4), 22-37. DOI: <https://dx.doi.org/10.4067/S0718-27242010000400003>

- ISACA. (2015). *State of Cybersecurity: Implications for 2015*. USA: Creative Commons
- Januszkiewicz, P., & Marek, P. (2006). *The OCTAVE methodology as a risk analysis tool for business resources*. Recuperado de <http://www.proceedings2006.imcsit.org/pliks/160.pdf>
- Jugdev, K., & Thomas, J. (2002). Project management maturity models: The silver bullets of competitive advantage. *Project Management Journal*, 33(4), 4–14. Recuperado de <https://dspace.ucalgary.ca/bitstream/1880/44250/1/2002%20PMJ%20PM%20maturity%20models.pdf>
- Kerzner, H. (2000). *Gestión de Proyectos: Un enfoque sistemático para la planificación, programación y control*. New York: John Wiley and Sons.
- Klimko, G. (2001). *Knowledge management and maturity models: Building common understanding*. In: Proceedings of the 2nd European Conference on Knowledge Management, Bled, Slovenia.
- Llop, M., Escamilla, M., Furió, S., Galdón, M., García, L., García, J., Lara, J., & Navarro, C. (2013). *Tendencias TIC en puertos*. Valencia, España: Fundación Valenciaport
- Matrane, O., Talea, M., & Okar, C. (2014). Towards A New Maturity Model for Information System. *International Journal of Computer Science Issues*, 12(3), 268-275. Recuperado de <https://www.ijcsi.org/papers/IJCSI-12-3-268-275.pdf>
- Mayer, J., & Lemes, L. (2008). *Modelo para Avaliar o Nível de Maturidade do Processo de Gestão de Riscos em Segurança da Informação*. Paper presented at: *VIII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*. São Leopoldo, Brasil.
- Mettler, T. (2009). *A design science research perspective on maturity models in Information Systems*. Recuperado de <https://www.alexandria.unisg.ch/214531/1/20090512%2520Maturity%2520Model%2520Design.pdf>
- Ministerio de Hacienda y Administración Pública del Gobierno de España. (2012). *MAGERIT – versión 3.0 Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información*. Recuperado de <https://www.ccn-cert.cni.es/documentos-publicos/1791-magerit-libro-ii-catalogo/file.html>
- Nagata, K., Amagasa, M., & Kigawa, Y. (2009). *Method to select Effective Risk Mitigation Controls Using Fuzzy Outranking*. Recuperado de <http://ieeexplore.ieee.org/document/5364934/>
- OPM3. (2003). *Organizational project management maturity model*. Recuperado de <http://faculty.kfupm.edu.sa/MGM/bubshait/project%20management/PDF/opm3KF.pdf>
- Palomino, M., Dávila, A., Melendez, K., & Pessoa, M. (2017). Agile Practices Adoption in CMMI Organizations: A Systematic Literature Review. En: Mejia J., Muñoz M., Rocha Á., San Feliu T., Peña A. (Eds). *Trends and Applications in Software Engineering. CIMPS 2016. Advances in Intelligent Systems and Computing*, vol 537. Cham: Springer.

- Rosemann, M., & De Bruin, T. (2005). *Towards a business process management maturity model*. Recuperado de https://eprints.qut.edu.au/25194/1/25194_rosemann_2006001488.pdf
- Saavedra, V., Dávila, A., Melendez, K., & Pessoa, M. (2017). Organizational Maturity Models Architectures: A Systematic Literature Review. En: Mejia J., Muñoz M., Rocha Á., San Feliu T., Peña A. (Eds). *Trends and Applications in Software Engineering. CIMPS 2016. Advances in Intelligent Systems and Computing*, vol 537. Cham: Springer.
- Sikdar, P. (2011). Alternate Approaches to Business Impact Analysis. *Information Security Journal: A Global Perspective*, 20, 128–134.
- Sotelo, M., Torres, J., & Rivera, J. (2012). Un proceso práctico de análisis de riesgos de activos de información. Paper presented at: *IV Congreso Internacional de Computación y Telecomunicaciones COMTEL*. Lima, Perú.
- Sylim, A., Hori, Y., & Sakurai, K. (2009). *Comparison of Risk Analysis Methods: Mehari, Magerit, NIST800-30 and Microsoft's Security Management Guide*. Recuperado de <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.6972&rep=rep1&type=pdf>

Disminuyendo la Brecha del Aseguramiento de la Calidad entre Gestor y Miembros del Equipo Mediante el Uso de Herramientas de Software

Mirna Muñoz

mirna.munoz@cimat.mx

Centro de Investigación en Matemáticas- Unidad Zacatecas, Parque Quantum, Ciudad el Conocimiento
Avenida Lassec ,Andador Galileo Galilei, Manzana, 3 Lote 7, 98160. Zacatecas, México.

DOI: [10.17013/risti.31.18-32](https://doi.org/10.17013/risti.31.18-32)

Resumen: El desarrollo del software se ha vuelto cada vez más complejo, involucrando la coordinación de diversos profesionales involucrados en su desarrollo en distintos niveles jerárquicos como: gerentes de calidad, líderes de proyecto, desarrolladores, arquitectos de software, analistas de software y testers. Esta interacción lleva consigo el surgimiento de retos y/o limitantes que deben ser gestionados durante el desarrollo del proyecto. Desafortunadamente, la interacción y entendimiento de los objetivos de calidad establecidos no siempre son los esperados afectando la obtención de resultados esperados. Este artículo presenta los resultados de la validación de una herramienta desarrollada como apoyo para llevar a cabo actividades relacionadas con el aseguramiento de la calidad. Los resultados demuestran que la herramienta permite un entendimiento común entre los diferentes profesionales involucrados en el desarrollo de software referente a los objetivos de calidad establecidos, reforzando la comunicación y el análisis de resultados para toma de decisiones oportuna.

Palabras-clave: aseguramiento de la calidad; desarrollo de software; herramientas de software; calidad en el software.

Reducing the Quality Assurance Gap between Middle Management and Operational through the Use of Software Tools

Abstract: The development of software has become more complex, involving the coordination of different professionals throughout its development, such as: quality manager, project leader, developers, software architects, software analyst, and testers. This interaction brings challenges and constraints that must be managed during the development of software. Unfortunately, the interaction and understanding of the quality goals established are not always the expected. This paper presents the results of validate a tool developed as support to carry out activities related to quality assurance. The results show that the tool allows a common understanding among different professionals involved in the software development regarding the quality goals established, reinforcing communication

and the analysis of the quality assurance activities results analysis for a timely decision making.

Keywords: quality assurance; software development; software tools; software quality.

1. Introducción

Hoy en día asegurar la calidad del software se ha convertido es una necesidad debido a la incursión del software en diferentes sectores industriales como minería, automotriz, aeroespacial, agroindustria, turismo, etc., donde se está convirtiendo núcleo para la realización de sus actividades cotidianas (Muñoz, Mejía & Ibarra, 2018). Además, 4 de 5 de las empresas más capitalizadas en el mundo bursátil son empresas de desarrollo de software (Rebolledo, 2017).

El desarrollo de software para estos distintos sectores industriales hace que su desarrollo se vuelva más complejo, involucrando la interacción entre profesionales de diversos niveles jerárquicos, dando como resultado el surgimiento tanto de retos y limitantes para lograr la calidad esperada (Suali, Fauzi, Sobri, & Nasir, 2017).

En este contexto es importante distinguir el concepto de calidad del de aseguramiento de calidad (Muñoz, Mejía & Ibarra, 2018): (1) calidad es el nivel en el que el producto de software cumple con los requisitos; (2) aseguramiento de la calidad son aquellas actividades que se realizan para generar evidencia que permite evaluar que los procesos y productos de software cumplen con los estándares aplicables, adecuados al tipo de producto de software que se desea obtener.

Por lo tanto, para que pueda ser lograda la calidad esperada en el producto de software, se deben ejecutar actividades que permitan evaluar, controlar y tomar decisiones oportunas de acuerdo a los resultados. Esta situación resalta, por un lado, la importancia de las actividades para el aseguramiento de la calidad, y por otro lado, la necesidad de mejorar la coordinación de estas actividades para garantizar que la obtención de productos con la calidad requerida por el cliente, convirtiéndose en actividades cada vez más comunes y vitales en el desarrollo de proyectos de software (Suali, Fauzi, Sobri, & Nasir, 2017).

Esta necesidad se hace crítica en micro empresas, en las que una persona desarrolla más de un rol que conlleva la realización de múltiples actividades, y la mayoría de las veces no se cuenta con el conocimiento de expertos para la implementación de este tipo de actividades (Muñoz, M., Gasca G. & Valtierra, C., 2014).

Como una solución a esta situación, este artículo presenta una herramienta cuyo objetivo es dar soporte a la implementación de las actividades para el aseguramiento de la calidad, con el fin de guiar y facilitar la implementación de estas actividades en las empresas. Esta herramienta pretende mejorar tanto la comunicación, como el seguimiento y control en actividades relacionadas con la gestión de la calidad en proyectos de desarrollo de software, involucrando a los gestores de proyectos y los miembros del equipo.

Después de la introducción, este artículo está estructurado como sigue: en la sección 2 se presentan los trabajos relacionados; en la sección 3 se presenta la herramienta desarrollada; en la sección 4 se muestra el estudio de caso realizado; en la sección 5 se analizan los resultados y en la sección 6 se presentan las conclusiones.

2. Trabajos relacionados

En (Heinemann, Hummel, & Steidl, 2014), se propone una herramienta para realizar análisis de calidad por medio de pruebas dinámicas de integración de código utilizando un control de versiones, mediante la ejecución automática de una serie de pruebas por cada commit ejecutado.

En (Jain & Gupta, 2011), se propone una herramienta para generar conjuntos de pruebas que satisfagan criterios de salida particulares, a partir del conjunto de pruebas se identifican debilidades y fortalezas de los conjuntos de pruebas.

En (Lin, He, Zhang, & Song, 2015), se propone un marco de trabajo enfocado en la creación de un entorno que contenga un conjunto de herramientas por medio de un plan en el que se detalla las métricas y requisitos relevantes.

En (Mathrani, 2014), se propone un marco de trabajo enfocado al análisis dinámico de sistemas distribuidos, en donde los equipos de desarrollo realizan pruebas dinámicas al código para identificar y para resolver errores de código, facilitando la trazabilidad entre piezas de software y cambios realizados.

En (Öztürk, CİL, & Zengin, 2015), se propone un marco de trabajo para facilitar el proceso de selección de herramientas para la ejecución de pruebas de código mediante una ontología de errores y herramientas ad-hoc para resolver determinados errores, proporcionado un conjunto base de herramientas para análisis dinámico de código.

En (Zaytsev & Morrison, 2013), se propone un marco de trabajo para la integración continua del código mediante el control de versiones y un software para pruebas de integración automática de código.

Al analizar los trabajos relacionados, se observa, que la mayoría de las herramientas están enfocadas en la ejecución de pruebas dinámicas para resolver problemas de código, así como en la integración continua de los componentes de software. Sin embargo, controlar la calidad a partir del código potencializa el costo de corregir los errores detectados en esta etapa. Además, para que éstas herramientas se utilicen de forma adecuada, debe haber un previo entendimiento de qué se tiene artefactos se tienen que controlar a lo largo del desarrollo de software, así como los criterios y métricas que se van a utilizar. Este es un problema muy común que se identifica en las organizaciones y que es el enfoque de la herramienta propuesta en este artículo.

3. Herramienta para la Gestión del Aseguramiento de la Calidad

3.1. Antecedentes

Para el desarrollo de esta herramienta, se realizó en primer lugar una revisión sistemática (Muñoz, Mejía & Ibarra, 2018) mediante la cual se obtuvieron los siguientes hallazgos:

- a. *Identificación de las principales actividades llevadas a cabo para el aseguramiento de la calidad:* se identificaron 3 prácticas principales llevadas a cabo para el aseguramiento de la calidad de software: (1) identificar elementos relevantes para la calidad; (2) identificar y asignar métricas que tengan mayor impacto en la calidad de la organización; y (3) identificar y asignar criterios para

la aceptación del producto. Estas actividades son establecidas por el Estándar IEEE 730 (McGrath et al. 2014), que es un estándar para el aseguramiento de la calidad, el cual establece los requisitos para iniciar, planificar, controlar y ejecutar el aseguramiento de la calidad dentro de un proyecto de desarrollo de software, y son mantenidas en modelos y estándares de calidad como CMMI-Dev v1.3, ISO 15504, MoProsoft, PMBOK.

Como parte del análisis a los resultados se identifico la falta de actividades importantes para realizar de manera correcta el aseguramiento de la calidad: (1) verificar la configuración del aseguramiento de la calidad y (2) realizar seguimiento y control de la calidad.

- b. *Identificación de un conjunto de herramientas para la gestión de la calidad:* se identificó que además de herramientas, han sido desarrollados modelos, meta-modelos y metodologías. La mayoría de estas herramientas se enfocan en la fase de desarrollo enfocado en el análisis estático y dinámico de código, la búsqueda, selección y ejecución de pruebas automáticas de software.

De los resultados mostrados anteriormente, se concluyó que las herramientas existentes no cubren todas las actividades principales para garantizar el aseguramiento de la calidad, debido a que éstas únicamente se enfocan en la fase de código y unas pocas en la verificación de los requisitos. Esta situación refleja el estado actual de la implementación de las actividades de aseguramiento de la calidad en las organizaciones, donde no existe un entendimiento común entre los gestores o líderes de proyecto, quienes realizan los planes para el aseguramiento de la calidad, y el equipo de trabajo quienes son los encargados de ir llevando a cabo las actividades relacionadas con el aseguramiento de calidad a través de las distintas etapas del desarrollo del software.

Por lo tanto, resalta la necesidad del desarrollo de una herramienta que sirva como plataforma de comunicación para la interacción de diferentes roles, en la que se tenga identificado en todo momento: (1) él o los artefactos de software se deben analizar; (2) los roles intervienen en la revisión y (3) las métricas establecidas, de tal manera que, se logre un entendimiento común entre los diferentes roles, de cómo sus actividades y productos de trabajo, suman para lograr la calidad del producto y para la toma de decisiones oportuna ante alguna desviación de los objetivos de calidad establecidos desde la planificación del proyecto.

3.2. Desarrollo de la herramienta

Tomando en cuenta los hallazgos de la revisión de literatura se identifica la necesidad de una herramienta que sirva como plataforma de comunicación y control para que las actividades de aseguramiento de calidad puedan ser llevadas a cabo durante el desarrollo de un proyecto de manera oportuna.

La herramienta desarrollada se estructura utilizando elementos del proceso, como a continuación se lista:

1. *Fases:* las fase representa cada una de las etapas del aseguramiento de la calidad, cada fase contiene una fecha límite, un conjunto de actividades y uno o más productos de trabajo que resultan de la ejecución de las actividades.

2. **Actividad:** esta formada por un conjunto de tareas a realizar para la generación de una o más salidas, y que servirán de apoyo para dar por concluida la fase.
3. **Tarea:** acción específica que el equipo de trabajo debe realizar para producir las salidas y/o productos de trabajo requeridas por la actividad.
4. **Productos de trabajo:** elemento generado a través del ciclo de vida del desarrollo del proyecto, este puede ser: documentos, diagramas, código ejecutable, script, etc.
5. **Salidas:** elemento temporal que ayuda en el aseguramiento de la calidad de un producto o bien, proporciona información del estado actual de la calidad del producto de trabajo.
6. **Roles:** Define un conjunto de responsabilidades y habilidades específicas relacionadas con el aseguramiento de la calidad que deben ser llevadas a cabo dentro del proyecto de desarrollo de software por un miembro del equipo.

La Tabla 1 muestra la definición de los elementos de la herramienta propuesta, es importante resaltar que en la fase de inicio son configuradas las fases que apliquen al proyecto de acuerdo al ciclo de desarrollo de software utilizado.

Fase	Descripción	Actividades	Producto de Trabajo	Roles
<i>Inicio</i>	Fase diseñada para que el líder de proyecto configure la herramienta, seleccione las fases, productos, criterios y métricas considerados para el control de la calidad.	1) Configurar la herramienta respecto al plan de calidad, plan de Proyecto 2) Asignar métricas de calidad y criterios de aceptación a los productos de trabajo agregados 3) Verificar la aceptación de la configuración de la herramienta	- Configuración de las fases de desarrollo - Configuración de productos relevantes - Criterios de aceptación asignados a los productos relevantes - Métricas de calidad asignadas a los productos relevantes - Configuración aprobada	Líder de Proyecto Responsable de calidad
<i>Análisis</i>	Fase diseñada para que el analista realice las verificaciones correspondientes y el gerente de calidad evalúe los resultados de medición obtenidos	1) Evaluar la aceptabilidad del documento de requisitos 2) Analizar los resultados de medición al documento de requisitos	- Documento de requisitos verificado - Documento de requisitos validado - No conformidades respecto a los resultados de medición	Analista Responsable de calidad
<i>Evaluación de Riesgos</i>	Fase diseñada para que el líder de proyecto verifique el cumplimiento de los criterios de aceptación y el gerente de calidad evalúe los resultados de medición.	1) Evaluar la aceptabilidad del documento de riesgos 2) Analizar los resultados de medición al documento de riesgos	- Documento de gestión de riesgos verificado - No conformidades respecto a los resultados de medición	Líder de Proyecto Responsable de calidad

Fase	Descripción	Actividades	Producto de Trabajo	Roles
<i>Diseño</i>	Fase diseñada para que el arquitecto de software o el responsable del diseño, evalúe el cumplimiento de los criterios de aceptación registrados para el diseño del sistema.	1) Evaluar la conformidad del diseño 2) Evaluar la aceptabilidad del diseño del sistema 3) Analizar los resultados de medición al diseño del sistema	- Diagrama del sistema verificado - Diseño del sistema verificado - Diseño del sistema validado - No conformidades respecto a resultados esperados	Arquitecto de software/ responsable de diseño Responsable de calidad
<i>Desarrollo</i>	Fase diseñada para que el desarrollador verifique el cumplimiento de los criterios de aceptación establecidos para el código fuente y el gerente de calidad evalúe los resultados de medición obtenidos.	1) Evaluar conformidad del código fuente 2) Evaluar aceptabilidad del código fuente 3) Analizar los resultados del código fuente	- Diagramas del sistema verificados contra requisitos - Diagramas del sistema verificados contra el diseño - Código fuente verificado - No conformidades respecto a los resultados esperados	Desarrollador Responsable de calidad
<i>Pruebas</i>	Fase diseñada para que el téster verifique el cumplimiento de los criterios de aceptación, de igual manera que el gerente de calidad evalúe los resultados de medición obtenidos.	1) Evaluar la conformidad del plan de pruebas y/o el procedimiento de pruebas 2) Evaluar la aceptabilidad del plan de pruebas y los resultados de pruebas 3) Análisis de los resultados de medición del plan de pruebas y los resultados de la pruebas	- Plan de pruebas o procedimientos de pruebas verificados contra los requisitos - Plan de pruebas y resultados de pruebas verificados - Resultados de pruebas validados - No conformidades respecto a los resultados esperados	Téster Responsable de calidad
<i>Liberación</i>	Fase diseñada para que el líder de proyecto verifique el cumplimiento de los criterios de aceptación y el gerente de calidad evalúe los resultados de medición obtenidos del procedimiento de entrega	1) Evaluar aceptabilidad del procedimiento de entrega 2) Análisis de los resultados de medición del procedimiento de entrega	- Procedimiento de entrega verificado - procedimiento de entrega validado - No conformidades respecto a los resultados esperados	Líder de Proyecto Responsable de calidad
<i>Cierre</i>	Fase diseñada para que el líder de proyecto confirme el cumplimiento de los criterios de aceptación y de igual manera el gerente de calidad evalúe los resultados de medición obtenidos.	1) Evaluar aceptabilidad del procedimiento de entrega 2) Mediciones de calidad al procedimiento de entrega	- Documento de cierre de proyecto verificado	Líder de proyecto

Tabla 1 – Definición de las fases de la herramienta

Como se observa en la Tabla cada fase ha sido diseñada para disminuir la brecha de entendimiento en cuanto al aseguramiento de calidad de las personas involucradas en el desarrollo de un proyecto, como a continuación se describe:

- Mejorar el común entendimiento referente a la calidad de producto debido a que la herramienta está diseñada para todos miembros de un equipo de desarrollo de software tengan un común entendimiento referente a la visualización de la calidad por parte del responsable de calidad y el líder del proyecto.
- Permite la configuración de las fases en las que se va llevar a cabo el aseguramiento de la calidad con base en el ciclo de desarrollo de software utilizado.
- Mejora el conocimiento de cómo se llevará a cabo el aseguramiento de la calidad a lo largo de proyecto, ya que permite la selección de artefactos de proyecto por fase que debe ser medidos y los criterios que deben cubrir.
- Facilita a los miembros del equipo a realizar actividades de aseguramiento de calidad dependiendo de su rol y fase del ciclo de desarrollo de software.
- En cada fase se implementan las 3 actividades principales para el aseguramiento de la calidad. Además, en la primera fase se implementa la actividad adicional de verificar la configuración, la cual asegura que todos están de acuerdo en las actividades de aseguramiento de calidad a llevarse a cabo en el proyecto. Finalmente, con la ejecución de la herramienta se está cumpliendo con la segunda actividad adicional referente a realizar seguimiento y control de la calidad.

3.3. Ejemplo de interfaz de la herramienta

El diseño completo de la herramienta propuesta, así como algunas de sus pantallas, se encuentra publicado en (Ibarra & Muñoz, 2018). En esta sección se incluye a manera de ejemplo un conjunto de pantallas significativas para mostrar los elementos de la herramienta.

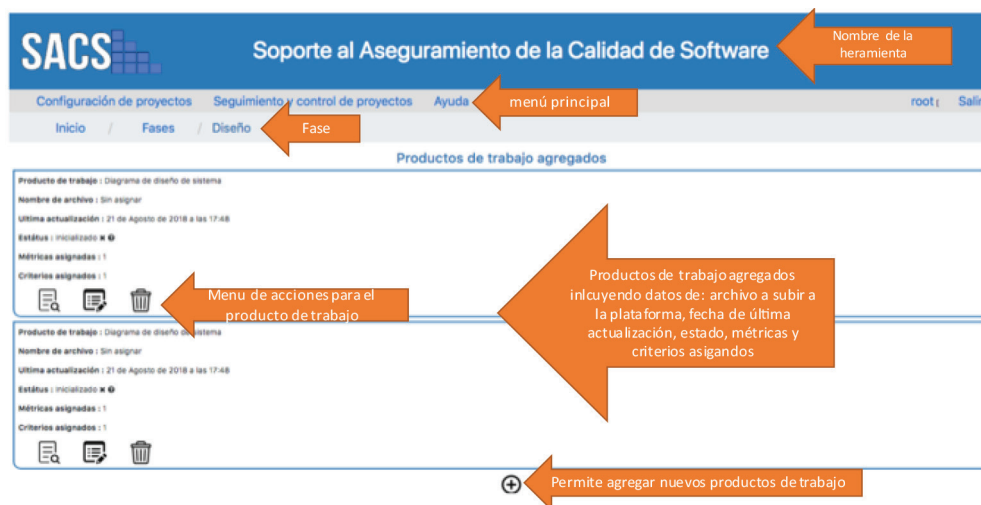


Figura 1 – Pantalla de la herramienta para agregar productos de trabajo de la fase de Diseño

La Figura 1, muestra la pantalla de la fase de diseño, en la que se tienen agregados dos productos de trabajo a ser revisados como parte del aseguramiento de la calidad.

La Figura 2 muestra la pantalla de los criterios y métricas asignadas para el producto de trabajo de la fase de diseño: Diagrama de diseño del sistema.

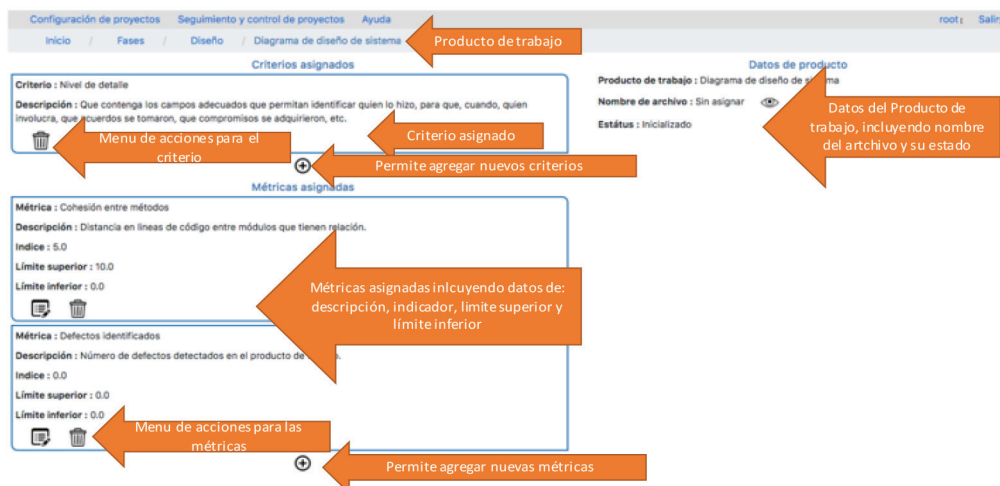


Figura 2 – Pantalla de gestión de criterios y métricas para un producto de trabajo de la fase de Diseño

La Figura 3 muestra la pantalla para la revisión de los resultados de medición.

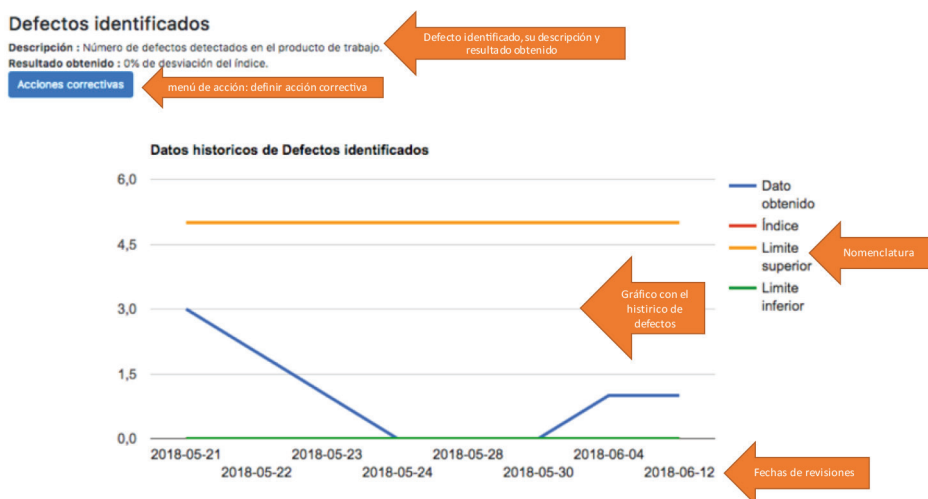


Figura 3 – Pantalla en la que se revisan los resultados de medición [adaptado de (Ibarra & Muñoz, 2018)]

La Figura 4 muestra la pantalla para establecimiento de acciones correctivas según sea necesario.

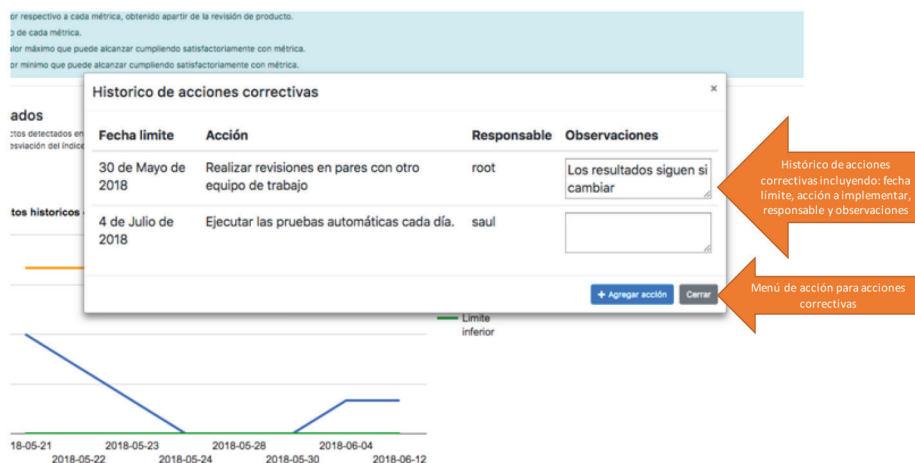


Figura 4 – Pantalla en la que se establecen acciones correctivas con base en resultados de medición [adaptado de (Ibarra & Muñoz, 2018)]

4. Estudio de Caso

Para validar la herramienta se realizó un estudio de caso en el que participaron 4 empresas, de las cuales 2 son centros de desarrollo de software de universidades, un centro de investigación y una microempresa de desarrollo de software.

El objetivo del estudio fue evaluar el nivel en el que las prácticas integradas como parte de herramienta eran percibidas el conjunto de empresas de desarrollo de software para apoyarlas en la realización de las actividades de aseguramiento de software que éstas llevan a cabo.

Las empresas de desarrollo de software participantes se describen a continuación:

1. **Empresa 1:** Es un Centro de Desarrollo de Software dentro de una universidad, que se dedica a desarrollar software para clientes internos y externos a la institución, certificada en el estándar ISO/IEC 29110. De este centro de desarrollo de software se encuestaron 4 personas con los siguientes roles: desarrollador, téster, diseñador y gestor.
2. **Empresa 2:** Es un Centro de Desarrollo de Software dentro de una universidad, que se dedica a desarrollar software para clientes internos y externos a la institución, certificado en el estándar ISO/IEC 29110. De este centro de desarrollo de software se encuestó a 1 persona con el rol de gestor.
3. **Empresa 3:** Es una microempresa de desarrollo de software embebido, certificada en el estándar ISO/IEC 29110. De esta empresa se encuestó a 1 persona con el rol de desarrollador, téster, diseñador y analista.

4. Empresa 4: Laboratorio de un centro de investigación en el que se integran equipos de profesionales para trabajar en proyectos específicos. De este centro de laboratorio se encuestaron 4 personas con los siguientes roles: desarrolladores, analistas, téster, arquitectos y gerentes de calidad.

Es importante resaltar que de las 10 personas participantes en el estudio de caso, no todas tienen la misma experiencia, 1 persona tiene menos de un año de experiencia; 5 personas tienen de uno a tres años de experiencia y 4 personas tienen más de cinco años de experiencia en la realización de actividades relacionadas con la calidad y el aseguramiento de la calidad.

Las preguntas de investigación definida:

1. ¿La herramienta facilita la gestión de la calidad en el desarrollo de software?.
2. ¿Las prácticas contenidas en la herramienta permiten implementar el aseguramiento de la calidad del producto?.
3. ¿Se mejora la identificación de defectos en los productos de software obtenidos?.

Para la recogida de datos se diseñó un cuestionario en google forms, dividido en dos secciones; la primera sección enfocada en la percepción de los elementos de aseguramiento de calidad y la segunda sección en la percepción de la herramienta. El cuestionario completo se lista a continuación:

- a. Sección enfocada en los elementos de aseguramiento de la calidad incluidos como parte de la herramienta:
 1. ¿Considera que las fases disponibles son las adecuadas para la cobertura total de la metodología de desarrollo que realiza en su trabajo?
 2. ¿Considera que los tipos de producto de trabajo disponibles son suficientes para llevar a cabo en aseguramiento de calidad básico?
 3. ¿Considera que las actividades de aseguramiento de calidad que realiza se pudieron ajustar a la configuración de la herramienta?
 4. ¿Considera que con las actividades realizadas se obtendrá una mejora en la prevención de defectos?
- b. Sección enfocada en la herramienta:
 - Facilidad de entendimiento
 1. ¿Se reduce el esfuerzo para realizar las acciones en pocos pasos?
 2. ¿El estado de la calidad se transmite en forma clara y concisa?
 3. ¿La herramienta permite fácilmente al usuario identificar qué acciones debe llevar a cabo?
 - Facilidad de uso
 1. ¿Los usuarios realizan sus tareas en el menor tiempo posible?
 2. ¿Las tareas están diseñadas para realizarse de la forma más rápida e intuitiva posible?
 - Satisfacción en uso
 1. ¿Los resultados que obtendrías tras la interacción con la aplicación web son los deseados?
 2. ¿La interfaz es amigable al usuario?

El proceso de recogida de datos consistió en 5 pasos listados a continuación:

1. Realizar una reunión con cada una de las empresas participantes, durante la cuál se muestra la herramienta, describiendo de manera breve su funcionamiento.
2. Una vez finalizada la presentación, se proporciona acceso a la herramienta.
3. Se permite a los participantes interactuar con la herramienta.
4. Se proporciona el link del cuestionario y se invita a los participantes a responder la encuesta.
5. Se analizan los resultados.

Los resultados obtenidos de la ejecución del estudio de caso se analizan en la siguiente sección.

5. Análisis de resultados.

En esta sección se mostrarán los resultados mediante la respuesta de cada una de las preguntas definidas en el estudio de caso.

Pregunta 1. ¿La herramienta facilita la gestión de la calidad en el desarrollo de software?

Para responder la pregunta 1 se analizaron las respuestas de la segunda sección del cuestionario enfocada en la herramienta. La Figura 4 muestra un resumen de los datos.

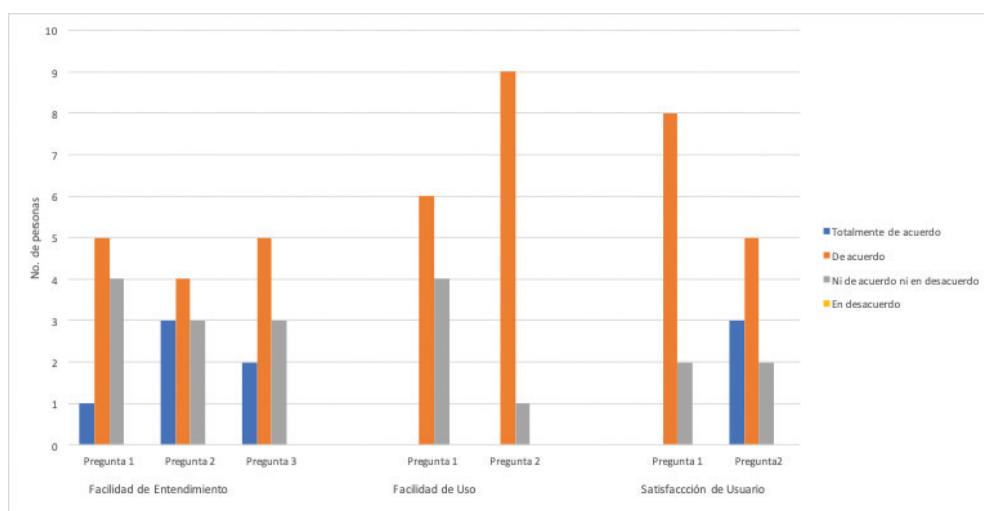


Figura 5 – Gráfico de respuestas de preguntas relacionadas con la percepción de facilidad del aseguramiento de la calidad mediante el uso de la herramienta

- a. Bloque de preguntas relacionadas con la facilidad de entendimiento:
 - o En cuanto a la pregunta 1 relacionada con la reducción del esfuerzo para realizar acciones en pocos pasos el 50% de los encuestados está de acuerdo y 10% de los encuestados está totalmente de acuerdo, mientras que 40% de los encuestados tiene una opinión neutral, por lo que se considera que

la herramienta puede apoyar en la reducción de esfuerzo para la realización de las actividades, aunque es importante identificar los puntos de mejora de la misma.

- En cuanto a la pregunta 2 relacionada con la transmisión del estado de la calidad 30% de los encuestados están totalmente de acuerdo; 40% de los encuestados está de acuerdo; y 30% de los encuestados tienen una opinión neutra, por lo que se considera que la herramienta apoya en la comunicación y transmisión del estado de la calidad del producto en sus diferentes fases.
 - En cuanto a la pregunta 3 relacionada con facilitar la identificación de las acciones que se tienen que llevar a cabo por usuario (rol), 10% de los encuestados está totalmente de acuerdo, 50% de los encuestados están de acuerdo y 30% de los encuestados tiene una opinión neutra, por lo que se considera que la herramienta es percibida de apoyo para que los miembros de un equipo tengan conocimiento de sus actividades referente a la calidad a través de las distintas fases del desarrollo de software.
- b. Bloque de preguntas relacionadas con la facilidad de uso:
- En cuanto a la pregunta 1 relacionada con la correcta realización de las tareas en menor tiempo posible, 60% de los encuestados están de acuerdo y 40% de los encuestados tienen una opinión neutra, por lo que se considera que la herramienta es percibida como fácil de usar por la mayoría de los encuestados. Aunque deben detectarse oportunidades de mejora de los 40% encuestados que tienen opinión neutra
 - En cuanto a la pregunta 2 relacionada con el diseño de la herramienta para realizar las tareas de forma más rápida e intuitiva, 90% de los encuestados están de acuerdo y sólo el 10% tiene una opinión neutra, por lo que se considera que las tareas están diseñadas para realizarse de una forma más rápida e intuitiva.
- c. Bloque de preguntas relacionadas con la satisfacción:
- En cuanto a la pregunta 1 relacionada con los resultados a la percepción de si se obtendrían los resultados deseados con la aplicación web, 80% de los encuestados están de acuerdo y 20% de los encuestados tienen una opinión neutra, por lo que se considera que con la implementación de la herramienta se obtendrá una mejora en la gestión de las actividades de calidad de los productos de trabajo.
 - En cuanto a la pregunta 2 relacionada con los resultados de si la interfaz es percibida como amigable, 30% de los encuestados están totalmente de acuerdo; 50% de los encuestados están de acuerdo y el 20% de los encuestados tienen una opinión neutra, por lo que se considera que la interfaz de la herramienta es amigable.

Pregunta 2. ¿Las prácticas contenidas en la herramienta permiten implementar el aseguramiento de la calidad del producto?.

Para responder la pregunta 2 se analizaron las respuestas de las preguntas 1, 2 y 3 del cuestionario. La Figura 5 muestra un resumen de los datos.

En cuanto a la pregunta 1 del cuestionario, relacionada con la percepción de si las fases incluidas en la herramienta permiten cubrir las fases de la metodología usada en la

empresa, como se observa en la figura, el 80% están de acuerdo y el 20% totalmente de acuerdo, por lo que se considera que la herramienta cubre las fases de desarrollo de software y pueden ser adecuadas a su forma de trabajo.

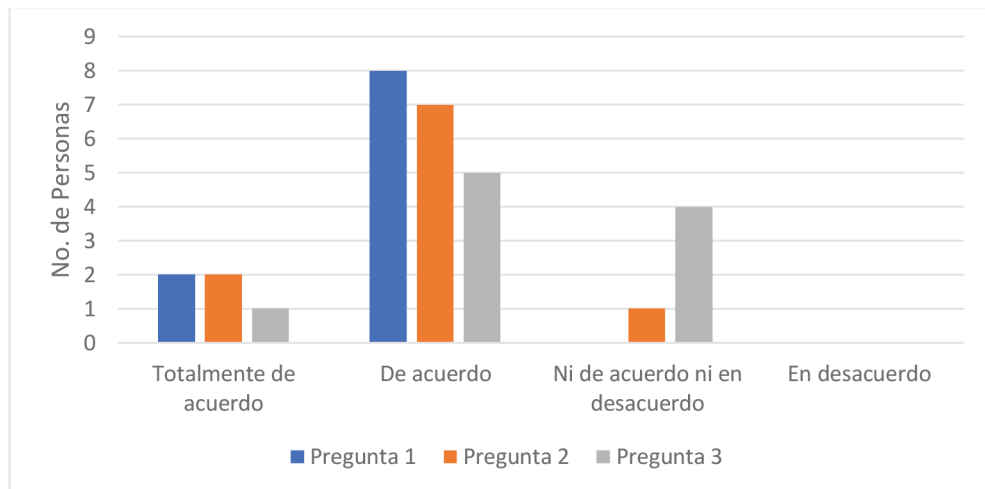


Figura 6 – Gráfico de respuestas de preguntas relacionadas con la percepción de facilidad en el aseguramiento de la calidad

En cuanto a la pregunta 2 del cuestionario, relacionada con si los productos de trabajo considerados en la herramienta son suficientes para una gestión básica del aseguramiento de la calidad, como se observa en la figura el 20% está totalmente de acuerdo y el 70% está de acuerdo, y únicamente el 10% tiene una percepción neutra, por lo que se considera que la herramienta contiene los productos de trabajo que permiten llevar a cabo una gestión básica del aseguramiento de la calidad.

Finalmente, en cuanto a la pregunta 3, relacionada con si las actividades que normalmente llevan a cabo para el aseguramiento de la calidad pueden ser configuradas utilizando la herramienta, como se observa en la figura aquí hay un cambio de percepción ya que el 10% están totalmente de acuerdo y el 50% está de acuerdo. Sin embargo el 40% tiene una percepción neutra. Al ser la mayoría la que considera que pueden ser configuradas sus actividades, se considera que la herramienta apoya en la implementación de actividades para el aseguramiento de la calidad.

Pregunta 3. ¿Se mejora la identificación de defectos en los productos de software obtenidos?.

Para responder la pregunta 3 se analizó la respuesta 4 de la primera sección del cuestionario, relacionada con la percepción de una mejora en la prevención de los defectos, el 40% de los encuestados está totalmente de acuerdo; 40% de los encuestados está de acuerdo y el 20% tiene una opinión neutra, por lo que se concluye que la herramienta apoyará en la reducción de los defectos.

Es importante resaltar que como limitante del caso de estudio únicamente se tuvo una sesión, sin embargo ninguna de las empresas utilizaron la herramienta en un desarrollo de software completo, aún cuando tuvieron acceso a la misma, únicamente exploraron su contenido, y pudieron revisar el ejemplo cargado en la misma.

6. Conclusiones.

El aseguramiento de la calidad es una actividad esencial en los proyectos de desarrollo, desafortunadamente aún cuando se establezcan planes de mejora, éstos no son transmitidos de manera adecuada al equipo de trabajo, lo que resulta en una falta de entendimiento en actividades a realizar y en cómo se va gestionando la calidad a través del ciclo de desarrollo de software. Esta situación es más crítica en PYMES y MiPYMES donde cada miembro del equipo ejecuta más de un rol.

Como una solución a esta situación, en este artículo se presenta una propuesta de herramienta software cuyo objetivo es brindar soporte a los equipos de desarrollo para la gestión de las actividades relacionadas con el aseguramiento de la calidad.

El estudio de caso realizado, en el que mostró la herramienta a un conjunto de 4 empresas, las cuales tuvieron acceso a la misma y pudieron explorarla y revisar el ejemplo precargado de la misma, demuestra que la herramienta es percibida como útil para soportar éstas actividades relacionada con el aseguramiento de la calidad, teniendo como principales características: (1) implementa las 3 actividades principales de aseguramiento de calidad; (2) es fácilmente configurable de acuerdo a las fases del ciclo de desarrollo de software utilizado en la organización; (3) facilita a los integrantes de un equipo el conocer los productos de trabajo a revisar y las fechas en las que debe realizarse; (4) habilita una plataforma de comunicación para obtener un entendimiento común entre el gestor o líder un proyecto y el equipo de trabajo relacionado con las actividades que deben llevarse a cabo para el aseguramiento de la calidad y (5) permite el análisis de los resultados y; el establecimiento y seguimiento de acciones correctivas.

Es importante mencionar que el estudio de caso fue realizado con una muestra pequeña de empresas, y con restricciones como la falta de uso de la herramienta durante todo un proyecto de desarrollo de software, y que se establece como trabajo futuro.

Finalmente, con base en los resultados obtenidos se concluye que con la realización de este tipo de herramientas puede ser disminuida la brecha entre gestores de proyecto y equipo de trabajo hacía una visión más clara de la gestión de la calidad, mediante la implementación de prácticas para el aseguramiento de la calidad.

Agradecimientos

Al alumno Saúl Alonso Ibarra Luévano quien ha desarrollado la herramienta presentada en este artículo como parte de su tesis para obtención del grado de Maestro en Ingeniería de Software ofrecida en el Centro de Investigación en Matemáticas.

Referencias

- Heinemann, L., Hummel, B., & Steidl, D. (2014). Teamscale: software quality control in realtime. In: *Companion Proceedings of the 36th International Conference on Software Engineering - ICSE 2014*, (pp.592–595). New York, New York, USA: ACM Press. doi:10.1145/2591062.2591068.
- Ibarra, S., & Muñoz, M. (2018). Support tool for software quality assurance in software development. In: *Proceedings of the 7th, International Conference On Software Process Improvement (CIMPS)*, (pp.13-19). IEEE. doi: 10.1109/CIMPS.2018.8625617.
- Jain, R. S., & Gupta, A. (2011). Refinement of the Test Bed Using Various Prioritization Techniques for Assuring Software-Quality (pp. 657–661). Berlin: Springer. doi:10.1007/978-3-642-25734-6_113.
- Lin, L., He, J., Zhang, Y., & Song, F. (2015). Quality Assurance through Rigorous Software Specification and Testing: A Case Study. *Procedia Computer Science*, 62, 257–265. doi: 10.1016/j.procs.2015.08.448.
- Mathrani, A. (2014). Quality assurance strategy for distributed software development using managed test lab model. In: 2014 IEEE International Technology Management Conference (pp. 1–4). Doi: 10.1109/ITMC.2014.6918609.
- Muñoz, M., Mejía, J., & Ibarra, S. (2013). Herramientas y prácticas para el aseguramiento de la calidad del software: Una revisión sistemática. IEEEExplore.
- Muñoz, M., Gasca, G., & Valtierra, C. (2014). Caracterizando las necesidades de las pymes para implementar mejoras de procesos software: una comparativa entre la teoría y la realidad. *Revista Ibérica de Sistemas y Tecnologías de Información*, E1, 1-15.
- Öztürk, M. M., CİL, İ., & Zengin, A. (2015). Development of a Multi-Agent Framework for Software Quality. *ACM SIGSOFT Software Engineering Notes*, 40(1), 1–10. Doi: 10.1145/2693208.2693234.
- Rebolledo, R., (2017). Las 20 mejores empresas del mundo en el 2017. *El Economista*, [Online]. Available: <https://www.eleconomista.com.mx/empresas/Las-20-mejores-empresas-del-mundo-en-el-2017-20170628-0113.html>.
- Suali, A. J., Fauzi, S. M., Sobri, W. A. W. M., & Nasir, M. H. N. M. (2017). Developers' coordination issues and its impact on software quality: A systematic review. In: 2017 3rd International Conference on Science in Information Technology (ICSITech), (pp. 659–663).
- Zaytsev, Y. V., & Morrison, A. (2013). Increasing quality and managing complexity in Neuroinformatics Software Development with Continuous Integration. *Frontiers in Neuroinformatics*, 6, 31. Doi: 10.3389/fninf.2012.00031.

Algoritmos de Minería de Proceso para el Descubrimiento Automático de Procesos

Marcos Hernán Rivas, Sussy Bayona-Oré

mrivasp@unmsm.edu.pe, sbayona@hotmail.com

Universidad Nacional Mayor de San Marcos, Av. Germán Amézaga s/n, Lima, Perú.

DOI: 10.17013/risti.31.33-49

Resumen: Un aspecto fundamental para gestionar y ejecutar procesos empresariales es el modelado de procesos. Para establecer las diferencias que existen entre los modelos preestablecidos y los modelos que han sido ejecutados se hace necesario revisar las huellas y los registros de eventos. Los registros de eventos son utilizados por la minería de procesos para descubrir el proceso real, mediante la extracción de conocimiento. Para conocer que algoritmos han sido desarrollados para el descubrimiento automático de procesos de negocios, se realizó una revisión de literatura de los artículos publicados en el periodo 2004-2017. Como resultado de la revisión 20 artículos primarios fueron identificados y analizados. Un total de 20 diferentes algoritmos fueron identificados. En el desarrollo de los algoritmos se han utilizado diferentes enfoques con predominio del enfoque general de algoritmo. Los algoritmos identificados en su mayoría utilizan redes Petri como técnica de modelamiento de procesos.

Palabras-clave: redes Petri; descubrimiento automático; minería de procesos; modelo de procesos; registro de eventos.

Process Mining Algorithms for Automated Process Discovery

Abstract: A fundamental aspect to manage and execute business processes is the process modeling. In order to establish the differences that exist between the pre-established models and the models that have been executed, the traces and records of events must be reviewed. Process Mining uses event logs to discover the real processes, through the extraction of knowledge. To know which algorithms have been developed for the automatic discovery of business processes, a literature review of articles published in the period 2004-2017 was carried out. As a result of the review, 20 primary articles were identified and analyzed. A total of 20 algorithms were identified using different approaches with predominance of the general algorithm approach. The algorithms identified mostly use Petri networks as a process modeling technique.

Keywords: Petri networks; automatic discovery; process mining; process model; event log.

1. Introducción

Una organización que desea supervivencia en el mundo competitivo empresarial del momento requiere efectividad, eficacia y eficiencia en sus procesos de negocios. Los procesos que no han sido bien diseñados tienen alta probabilidad de tener como resultado prolongados tiempos de respuesta, niveles de servicio por debajo de lo que ha sido establecido, utilización de recursos de forma desproporcionada, costos operativos en constante crecimiento y clientes descontentos con las entregas (Hermida, Rodríguez, Padrón, Domas, & López, 2016). Es por ello la relevancia de monitorear los procesos previos a su implementación, con la finalidad de descubrir las posibles fallas que podría haber en el diseño, fallas que se pueden presentar durante su ejecución e identificar posibles oportunidades de mejora de los procesos (van Der Aalst, 2015).

Para analizar y mejorar procesos se pueden utilizar como alternativas novedosas la Simulación y la Minería de Procesos, ambas soportadas por las Tecnologías de la Información y Comunicaciones (TICs) (Hermida et al, 2016). Agrawal, Gunopulos & Leymann (1998) fueron los primeros en presentar los conceptos de la minería de procesos. La minería de procesos puede ser visto desde dos puntos de vista: (1) el algorítmico, que ubica a la minería de procesos entre la minería de datos y la inteligencia computacional y (2) por la operatividad de los negocios que la ubica entre la modelación y el análisis de procesos (van der Aalst, 2016). El propósito es descubrir, monitorear y mejorar los procesos en ejecución (i.e., no los procesos supuestos), mediante la extracción de conocimiento de los registros de eventos dejados por los sistemas de información (van der Aalst et al., 2011).

Hoy para el análisis de procesos se usan técnicas que requieren mucho tiempo y los resultados que se esperan obtener dependen de la experiencia y las habilidades de los que realizan el estudio, con el riesgo de que se cometan errores. Sin embargo, los métodos empleados por la minería de procesos tienen menores costos en tiempo y recursos (Hermida et al, 2016). Para el descubrimiento del modelo de procesos se utilizan algoritmos que usan registros de eventos (representado como una tabla que contiene todos los eventos relacionadas con las actividades empresariales ejecutadas).

El modelo resultante puede ser una Red de Petri, un Modelo en Notación para la Gestión de Procesos de Negocio (BPMN, por sus siglas en inglés Business Process Model and Notation), una Cadena de Procesos Impulsadas por Eventos, (EPC, por sus siglas en inglés Event-driven Process Chain) o un diagrama en Lenguaje unificado de modelado (UML, por sus siglas en inglés, Unified Modeling Language) (Pourmasoumi & Bagheri, 2016).

Para determinar los algoritmos más utilizados en el descubrimiento automático de procesos se realizó una revisión sistemática de la literatura.

Este artículo está estructurado en 6 secciones incluido la Introducción. En la Sección 2 se describe los conceptos de minería de procesos, los tipos de minería, registros de eventos y el descubrimiento automático de procesos. En la Sección 3 se presenta la metodología utilizada para realizar la revisión de literatura. La Sección 4 muestra los resultados de la revisión de literatura. En la Sección 5 se discute los resultados. Finalmente, en la Sección 6 se presenta las conclusiones.

2. Marco Teórico

2.1. Minería de Procesos

Las organizaciones cada día requieren ser más competitivas y desean saber sobre la operatividad real de sus procesos, esta necesidad incentiva el desarrollo de nuevas técnicas de minería de procesos. La minería de procesos es un campo que utiliza elementos de la minería de datos y el modelamiento de procesos para realizar tareas de descubrimiento (Norambuena & Zepeda, 2017). Para satisfacer las necesidades de los especialistas en el campo de la minería de procesos, diversos algoritmos basados en algoritmos de otros campos de la informática han sido desarrollados (Tiwari, Turner & Majeed, 2008). La minería de procesos se inicia con Cook & Wolf (1998), quienes descubren modelos de procesos de software a partir de registros de eventos. Posteriormente Cook, Du, Liu & Wolf (2004) trasladaron sus resultados obtenidos en el campo de procesos de software a los sistemas de flujo de trabajo descubriendo modelos de comportamiento concurrentes. La minería de procesos puede ser utilizado para explorar los procesos históricos de la organización con el propósito de implementar mejoras (Méjia & Muñoz). Las técnicas de minería de procesos son utilizadas para descubrir, monitorear y mejorar los procesos reales en una organización. Estas técnicas facilitan la extracción de conocimiento de los registros de eventos dejados por los sistemas de información y como resultado brinda un modelo que describe los procesos. Para aplicar minería de procesos se requiere (a) contar con los registros de eventos de los procesos ejecutados, (b) identificar qué eventos pertenecen a la misma instancia del proceso y (c) que exista un orden de ejecución para los eventos en la misma instancia del proceso (van der Aalst, Weijters & Maruster, 2004).

2.2. Tipos de Minería de Procesos.

Dustdar, Hoffmann, & van der Aalst (2005) establecieron tres perspectivas en la minería de procesos de negocios: (a) perspectiva de tener un control de flujo, (b) perspectiva del desarrollo organizacional y (c) perspectiva de seguimiento del caso. El control de flujo es una perspectiva que se interesa por el orden de ejecución de las actividades, y el objetivo es poder mapear todas las rutas posibles que se pueden ejecutar en el proceso. La perspectiva organizativa tiene como objetivo construir la red social de la organización clasificando a las personas por su interacción teniendo en cuenta funciones y unidades organizativas a la que pertenecen. La perspectiva del caso se interesa por los elementos que pueden caracterizar el caso como la ruta de ejecución en el proceso, los recursos que se utilizan o por los valores que toman sus elementos de datos. La perspectiva de control de flujo puede extenderse con eventos que tienen marcas de tiempo. Van der Aalst & Vanthienen (2011) adiciona una perspectiva del tiempo que se ocupa de la sincronización y de la frecuencia de ejecución de las actividades dentro del proceso. Adicionalmente a las perspectivas descritas (Dustdar, Hoffmann, & van der Aalst, 2005), en (van der Aalst & Vanthienen, 2011), describen tres tipos básicos de minería de procesos: (a) *Descubrimiento* de procesos, cuando la entrada es un registro de eventos y la salida es un modelo de proceso, (b) *Conformidad*, cuando el modelo de proceso inicialmente diseñado y sus reglas se compara con los resultados obtenidos de un registro de eventos del mismo proceso, y (c) *Mejora*, cuando se pretende mejorar el modelo de proceso existente utilizando el conocimiento que se extrae del registro de eventos.

2.3.Registro de Eventos (RE)

La minería de Procesos utiliza un enfoque de registro de eventos para almacenar la información sobre la ejecución real del proceso de negocio, para su construcción extrae los datos almacenados por los sistemas de información que se crea durante el procesamiento de transacciones. Los registros de eventos representan la base de datos de entrada para ejecutar algoritmos de minería de procesos (van der Aalst et al., 2003). Un registro de eventos es representado como una tabla que contiene todos los eventos relacionadas con las actividades empresariales ejecutadas. Un caso tiene asignados eventos. Una sola ejecución de un proceso empresarial se denomina instancia (o caso) del proceso y se reflejan en el registro de eventos como un conjunto de eventos que se asignan al mismo caso. La secuencia de eventos registrados en un caso se llama traza (vanden Broucke & De Weerd, 2017). El modelo que describe la ejecución de una única instancia de proceso se denomina modelo de instancia de proceso. Casos y eventos se caracterizan por clasificadores y atributos. Los clasificadores aseguran la distinción de casos y eventos al asignar nombres únicos a cada caso y evento. Los atributos almacenan información adicional que se puede utilizar con fines de análisis.

Un modelo de la estructura de un registro de eventos se presenta en la Tabla 1. El registro de eventos de un proceso particular (Tabla 1), tiene 4 elementos de datos: Case ID, que implica una instancia de proceso, Activity, que es la tarea que se está realizando, Resource, se refiere al actor responsable de realizar la actividad (persona, grupo, departamento, etc.), timestamp indica el momento en que se registró un evento en el sistema. Un

Case id	Event id	Properties				
		Timestamp	Activity	Resource	Cost	...
1	35654423	30-12-2010:11.02	Register request	Pete	50	...
	35654424	31-12-2010:10.06	Examine thoroughly	Sue	400	...
	35654425	05-01-2011:15.12	Check ticket	Mike	100	...
	35654426	06-01-2011:11.18	Decide	Sara	200	...
	35654427	07-01-2011:14.24	Reject request	Pete	200	...
2	35654483	30-12-2010:11.32	Register request	Mike	50	...
	35654485	30-12-2010:12.12	Check ticket	Mike	100	...
	35654487	30-12-2010:14.16	Examine casually	Pete	400	...
	35654488	05-01-2011:11.22	Decide	Sara	200	...
	35654489	08-01-2011:12.05	Pay compensation	Ellen	200	...
3	35654521	30-12-2010:14.32	Register request	Pete	50	...
	35654522	30-12-2010:15.06	Examine casually	Mike	400	...
	35654524	30-12-2010:16.34	Check ticket	Ellen	100	...
	35654525	06-01-2011:09.18	Decide	Sara	200	...
	35654526	06-01-2011:12.18	Reinitiate request	Sara	200	...
	35654527	06-01-2011:13.06	Examine thoroughly	Sean	400	...
	35654530	08-01-2011:11.43	Check ticket	Pete	100	...
	35654531	09-01-2011:09.55	Decide	Sara	200	...
4	35654533	15-01-2011:10.45	Pay compensation	Ellen	200	...
	35654641	06-01-2011:15.02	Register request	Pete	50	...
	35654643	07-01-2011:12.06	Check ticket	Mike	100	...
	35654644	08-01-2011:14.43	Examine thoroughly	Sean	400	...
	35654645	09-01-2011:12.02	Decide	Sara	200	...
...	...	...	...	...	...	...

Tabla 1 – Estructura de un registro de eventos (van der Aalst, 2011)

proceso consiste en casos, un caso consiste en eventos tales que cada evento se relaciona precisamente con un caso, se ordenan los eventos dentro de un caso, los eventos pueden tener atributos (tiempo, actividad, costos y recursos) (van der Aalst, 2011).

2.4. Descubrimiento Automático de Procesos

El descubrimiento automático de procesos tiene como objetivo la construcción de un modelo de procesos ejecutando un algoritmo que tiene como entrada un registro de eventos y como resultado una representación del comportamiento del proceso. El modelo de proceso que se ha construido debería tener la capacidad de reproducir los casos del registro de eventos utilizado y prohibir el comportamiento de lo que no existe en los registros (van der Aalst et al., 2010). Teniendo en cuenta el requisito que cualquier evento debe estar vinculado a un caso (instancia de proceso) y a una actividad, y suponiendo que sólo esta información está disponible, un evento es descrito por un par (c, a) donde c se refiere al caso y a se refiere a la actividad. Por lo tanto, suponemos que cada caso se ejecuta independientemente de otros casos, es decir, el enrutamiento de un caso no depende del enrutamiento de otros casos (aunque pueden competir por los mismos recursos). Como resultado, podemos centrarnos en el ordenamiento de actividades dentro de casos individuales. Por lo tanto, un solo caso σ puede representarse como una secuencia de actividades, es decir, un trazo $\sigma \in A$, donde A es el conjunto de actividades. En consecuencia, un registro puede ser visto como una colección de trazos L , es decir, $(L \subseteq A)$. En la Figura 1, tomado de (van der Aalst et al., 2010), se muestra un ejemplo de registro y el correspondiente modelo de proceso descubierto utilizando el algoritmo α (van der Aalst, Weijters, & Maruster, 2004). Se visualiza que la red de Petri generada es capaz de reproducir el registro, es decir, hay un buen ajuste entre el registro y el modelo de proceso descubierto.

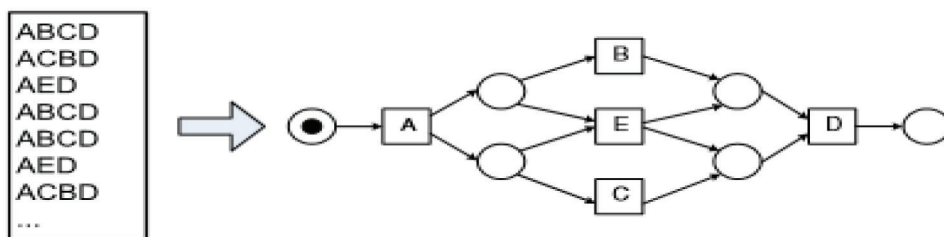


Figura 1 – Log representado por secuencias de actividades y el modelo de proceso que se descubre utilizando el algoritmo α (van der Aalst et al., 2010)

2.5. Enfoque en Desarrollo de Algoritmos de Descubrimiento.

Considerando los trabajos de (Tiwari, Turner & Majeed, 2008; Turner, Tiwari, Olaiya, & Xu, 2012; De Weerd, De Backer, Vanthienen, & Baesens, 2012; Pourmasoumi & Bagheri, 2016), existen diferentes enfoques en las técnicas de descubrimiento automático de procesos tales como: Enfoque algorítmico general como el algoritmo α , (b) Algoritmo Genético que utiliza un enfoque evolutivo, (c) Enfoque Markoviano que son algoritmos que examinan el comportamiento pasado y futuro para definir el estado actual (Cook

& Wolf, 1998), (d) Redes Neuronales caracterizados por su capacidad de “aprender” (Cook & Wolf, 1998), (e) Técnicas de agrupamiento que divide un grupo de soluciones en subgrupos homogéneos (Schimm, 2004), (f) Minería de Procesos basado en regiones que pueden ser de dos tipos, basados en el estado (van Dongen, Busi, Pinna & van der Aalst, 2007) y basados en el lenguaje (Bergenthum, Desel, Lorenz & Mauser, 2007), (g) Enfoque Heurístico mediante la representación de redes causales (Weijters & Ribeiro, 2011), (h) Enfoque de Minería declarativa que introduce eventos negativos generados artificialmente que muestran que no es posible una ruta particular a través de un proceso (Goedertier, Martens, Vanthienen & Baesens, 2009) o (i) Minería borrosa que es recomendado para procesos menos estructurados (Günther & van der Aalst, 2007).

3. Metodología

Para realizar la revisión de la literatura se ha utilizado la técnica de revisión sistemática sobre la base de las directrices originales propuestas por (Kitchenham, 2004). Una revisión sistemática de la literatura permite identificar, evaluar e interpretar la investigación disponible y relevante para dar respuesta a la pregunta de investigación en particular. Los estudios individuales que contribuyen a una revisión sistemática se denominan estudios primarios y la revisión sistemática es un estudio secundario (Kitchenham, 2004). Las etapas en el método de revisión sistemática de la literatura son: planificar la revisión sistemática, conducir la revisión sistemática y presentar los resultados. A continuación, se describen cada una de estas etapas.

3.1. Planificar la Revisión Sistemática

En esta etapa se elaboró el protocolo a ser usado para conducir la revisión sistemática. El protocolo incluye (i) la pregunta de investigación, (ii) la estrategia de búsqueda, (iii) los criterios de inclusión y exclusión, (iv) los criterios de calidad y (v) el método de extracción de datos. El objetivo de la investigación es determinar el estado actual de los métodos / técnicas / Algoritmos para realizar descubrimiento automático de proceso de negocios.

La pregunta principal de investigación definida en este estudio es:

P1: ¿Qué algoritmos de minería de procesos existen para realizar descubrimiento automático de procesos de negocios?”.

Para crear la cadena de búsqueda se utilizaron las siguientes palabras clave: Process mining, data mining, machine learning, Workflow mining, event log, BPMN, Petri nets, Discovery, Techniques, Algorithms. En la construcción de la cadena de búsqueda se ha utilizado los operadores booleanos AND y OR.

(“Process mining” OR “data mining” OR “machine learning” OR “Workflow mining”) AND (“event log”) AND (“BPMN” OR “Petri nets”) AND (“Discovery”) AND (“Techniques” OR “Algorithms”)

Las bases de datos seleccionadas para la búsqueda de los artículos fueron ACM, IEEE Xplore, ScienceDirect, Scopus, Springer y Google Scholar.

3.2. Conducir la Revisión Sistemática

Según el protocolo establecido en la etapa de planificación, se procedió con la búsqueda de artículos en las bases de datos electrónicas seleccionadas, haciendo uso de la cadena de búsqueda (artículos publicados desde el año 2004 a mayo de 2017). Al aplicar la cadena de búsqueda en cada una de las bases de datos seleccionadas, se identificaron un total de 295 artículos. Después de aplicar los criterios de inclusión, exclusión (véase Tabla 2) y los criterios de calidad se seleccionaron 40 artículos potenciales.

Criterios de Inclusión	Criterios de Exclusión
(a) La investigación propone alguna técnica / algoritmo / Método de minería de procesos para el descubrimiento automático de procesos de negocios, esto es realiza la construcción de un modelo de procesos como una representación abstracta de los registros de eventos	(a) El estudio está centrado en otro objetivo de la minería de proceso diferente al de descubrimiento de procesos de negocios
(b) El modelo de procesos representado desde los registros de eventos utiliza BPMN o Redes de Petri	(b) El estudio sólo muestra una aplicación utilizando software sin evaluar el algoritmo
(c) El enfoque presentado es aplicación a casos de estudio o evaluación experimental.	

Tabla 2 – Criterios de Inclusión / Exclusión

Finalmente se hizo una revisión detallada de los algoritmos en los artículos quedando definido 20 artículos primarios a estudiar en el presente trabajo (véase Tabla 3). Los criterios de calidad para evaluar la calidad de los estudios fueron:

- Están los objetivos de la investigación claramente especificados?
- ¿Se diseñó el estudio para lograr estos objetivos?
- ¿Se describen claramente los métodos / técnicas/ algoritmos utilizados?
- ¿Se presenta de manera clara el registro de eventos utilizado en el estudio?
- ¿Los hallazgos del estudio están plenamente justificados?

En el proceso se detectó que varios artículos carecían de detalles suficientes sobre las limitaciones de algoritmo, muchos estudios se centran sólo en superar algunas deficiencias del algoritmo α y no evalúan limitaciones de otros problemas que fueron definidos en el descubrimiento automático de procesos. En la Figura 2 se presenta el diagrama del proceso para conducir la revisión.

En los 20 estudios primarios seleccionados se exponen algoritmos para el descubrimiento automático de procesos de negocios bajo diferentes enfoques. Se diseñó un formato de extracción de datos para registrar los siguientes datos: Algoritmo, Autor(s), Año, Característica funcional del Algoritmo, Enfoque utilizado, Algoritmo detallado (si/no), Limitaciones, Fortaleza, Técnica de modelamiento, Caso de estudio utilizado. Los resultados de la revisión sistemática se presentan con mayor detalle en la Sección 4.

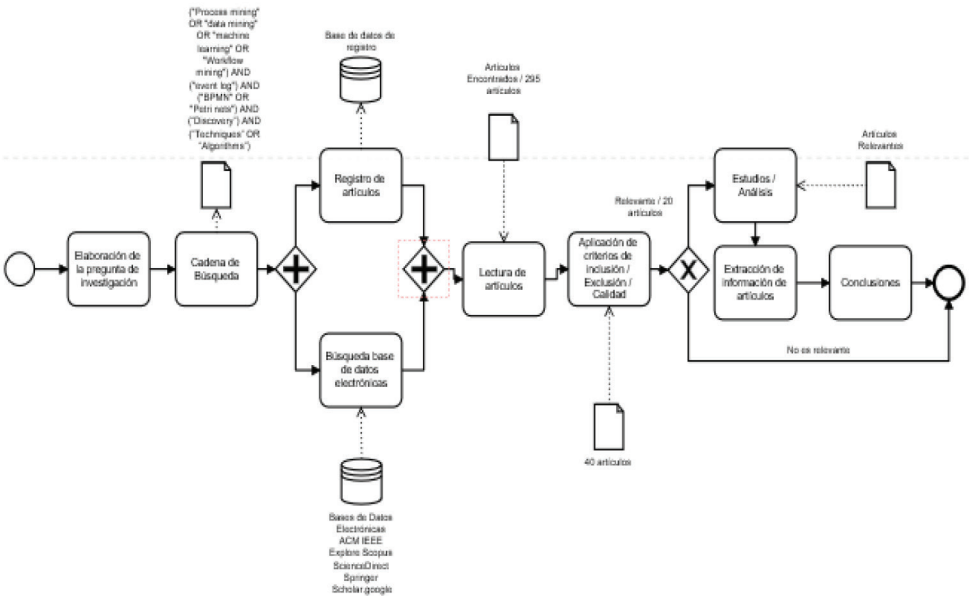


Figura 2 – Proceso para conducir la revisión

Bases de Datos	Estudios			
	Cadena Búsqueda	Inclusion / Exclusion	Calidad	Algoritmos
ACM	30	7	7	6
IEEE Xplore	13	4	4	1
Scopus	59	15	9	1
ScienceDirect	123	15	12	5
Springer	70	5	2	1
Scholar.google		7	6	6
TOTAL	295	53	40	20

Tabla 3 – Resumen de resultados de búsqueda

4. Resultados

Según los resultados de la revisión, el desarrollo de los algoritmos ha evolucionado a lo largo del tiempo (véase Figura 3).

En total se identificaron 20 algoritmos desarrollados en los últimos años (véase Tabla 4). En la Tabla 4, se muestra el enfoque utilizado para cada algoritmo identificado en

el estudio, que se representa con la siguiente nomenclatura: EAG: Enfoque algorítmico general, AE: Aproximación evolutiva y EH: Enfoque heurístico. La columna TMU representa la Técnica de modelado.

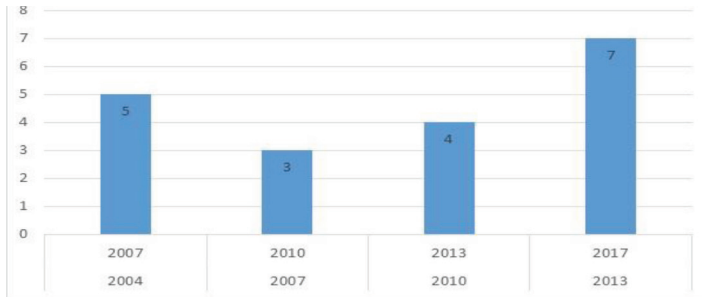


Figura 3 – Frecuencia de artículos por intervalos de años

De los resultados, se observa que la mayoría de los algoritmos fueron contruidos tomando como base los resultados obtenidos en el algoritmo α . En varios estudios los autores se enfocan en solucionar los casos de estructuras de procesos donde el algoritmo α no da como resultado respuesta satisfactoria, y otros lo utilizan en la etapa de experimentación para evaluar las ventajas del nuevo algoritmo desarrollado. Se han utilizado diferentes enfoques de desarrollo y diferentes técnicas de modelado para el flujo de resultado. Todos los algoritmos funcionan para diferentes condiciones de entradas o suposiciones. En la Tabla 4 se presenta información relevante para obtener buenos resultados en el uso de los algoritmos.

ID	Algoritmos	TMU	Enfoque Utilizado	Información Importante
A1	α Algorithm	Petri net	EAG	Asegura buen resultado si no hay ruido y además se requiere que el registro de flujo de trabajo sea completo es decir todos los casos deben estar completos de inicio a fin (van der Aalst, Weijters & Maruster, 2004).
A2	$\alpha+$ Algorithm	Petri net	EAG	Presupone información perfecta: (i) el registro debe estar completo y (ii) no hay ruido en el registro (es decir, todo lo que está registrado en el registro es correcto) (Alves de Medeiros, Van Dongen, van der Aalst & Weijters, 2004).
A3	Genetic Algorithm	Petri net	AE	El principal valor agregado de usar un algoritmo genético es la capacidad de lidiar con el ruido y cuando el registro no está completo (van der Aalst, Alves de Medeiros, Weijters, 2005).
A4	The Heuristics Miner algorithm	Causal Net	EH	Puede tratar con ruido y pocos tipos de casos, es decir, puede trabajar con un registro de eventos con miles de trazas, pero con poca frecuencia de comportamiento y algo de ruido (Weijters, van der Aalst & Alves De Medeiros, 2006).

ID	Algoritmos	TMU	Enfoque Utilizado	Información Importante
A5	$\alpha++$ Algorithm	Petri net	EAG	Se asume información perfecta: (i) el registro debe estar completo y (ii) el registro está libre de ruidos (es decir, cada evento registrado en el registro es correcto y representativo del modelo que debe descubrirse) (Wen, van der Aalst, Wang & Sun, 2007).
A6	β -algorithm	Petri net	EAG	Suponiendo un registro completo, se puede demostrar que el algoritmo β puede descubrir correctamente cualquier red WorkFlow (Wen, Wang, van der Aalst, Huang & Sun, 2009).
A7	<i>Region-based workflow mining</i>	Petri net	Enfoque basado en regiones	El clásico algoritmo α es incapaz de redescubrir varias redes, mientras que el enfoque basado en regiones puede explotarlas correctamente, pero es demasiado complejo (Busi & Pinna, 2009).
A8	<i>Mining algorithm $\alpha\#$</i>	Petri net	EAG	La capacidad de extracción del algoritmo está limitada a las redes DIWF (es una subclase grande de redes WF) y se supone que el registro de eventos dado está completo (Wen, Wang, van der Aalst, Huang & Sun, 2010).
A9	<i>Mining Algorithm χ- (Algorithm Based on Event Type)</i>	Petri net	EAG	Propone una estructura novedosa de registros de eventos para descubrir las dependencias causales. Luego propone un nuevo algoritmo para redescubrir el modelo de proceso con este tipo de registros de eventos (Wang, Ge, Hu, & Luo, 2011).
A10	<i>Mining Algorithm λ</i>	Petri net	EAG	No se centra en los problemas de ruido (el registro de eventos es correcto) y se supone que el registro de eventos contiene información suficiente (el registro de eventos está completo) (Wang, Ge, Hu, Luo, & Huang, 2012).
A11	<i>Guided Local Search</i>	BPMN	Búsqueda local guiada	Con un mismo registro de eventos compara los resultados obtenidos en el descubrimiento del proceso con los resultados obtenidos con el algoritmo Heuristic Miner, obteniendo mejores resultados (Chinches & Salomie, 2013).
A12	<i>Ant Colony Optimization Algorithm</i>	BPMN	Optimización de la colonia de hormigas	Compara el proceso descubierto por ACO BP Miner con el proceso de descubrimiento realizado con Genetic Miner, ACO BP Miner encontró significativamente menos errores que Genetic Miner (Chinches & Salomie, 2013b).
A13	<i>Algorithm: Evolution Strategy</i>	BPMN	AE	La estrategia de evolución encuentra de manera eficiente los modelos de proceso que mejor representan un registro de eventos (Molka, Redlich, Drobek, Zeng, & Gilani, 2015).
A14	<i>The Fusion Miner algorithm</i>	Petri net	Fusión de algoritmos	Los resultados muestran que este enfoque produce modelos más precisos y completos, especialmente en entornos con múltiples capas de flexibilidad (De Smedt, De Weerd, & Vanthienen, 2015).

ID	Algoritmos	TMU	Enfoque Utilizado	Información Importante
A15	<i>Technique BPMN Miner</i>	BPMN	EH	Se requiere que los eventos en el registro incluyan atributos de datos. Además, el conjunto de atributos debe incluir una clave principal para cada subprocesso a descubrir, así como una clave externa para cada Proceso - subprocesso (Conforti, Dumas, García-Bañuelos & La Rosa, 2016).
A16	<i>Algorithm (τ) with token carried data</i>	Petri net	Utilización de Tokens	En el manejo de lugares paralelos con entradas múltiples, sus resultados son mejores que los algoritmos $\alpha +$, $\alpha ++$, β y λ (Li, C., Ge, J., Huang, L., Hu, H., Wu, B., Yang, H., et al., 2016).
A17	<i>Discovery algorithm of synchronization-core-based process models</i>	BPMN	Árbol de decisión	Se requiere que los registros del evento estén completos. Además, el enfoque propuesto no considera el ruido, los modelos de procesos múltiples y las tareas duplicadas (Lu, Zeng, & Duan, 2016).
A18	<i>Algorithm SLAD (Splitting Labels After Discovery)</i>	Petri net	EAG	Es un algoritmo para abordar tareas duplicadas en un modelo ya descubierto (Vázquez-Barreiros, Mucientes, & Lama, 2016).
A19	<i>Fodina</i>	Petri net	EH	Mejora y expande el algoritmo Heuristics Miner (vanden Broucke & De Weerd, 2017).
A20	Mining $\lambda +$	Petri net	EAG	Si el modelo contiene la subred simétrica que no está bien estructurada, podría no descubrir el modelo (Li, Ge, Huang, Hu, Wu, Hu et al., 2017).

Tabla 4 – Algoritmos utilizados en el descubrimiento automático de procesos

Los algoritmos han sido desarrollados utilizando diferentes enfoques tales como enfoque general, enfoque evolutivo, enfoque heurístico, enfoque basado en regiones entre otros (véase Tabla 5).

Enfoque Utilizado	Estudios	Descripción
<i>Enfoque Algorítmico General</i>	[A1] [A2] [A5] [A6] [A8] [A9] [A10] [A18] [A20]	Es un enfoque de descubrimiento que produce una red de Petri para el comportamiento del proceso registrado en los registros de eventos (van der Aalst, Weijters, & Maruster, 2004).
<i>Enfoque evolutivo</i>	[A3] [A13]	Utiliza un enfoque evolutivo (Alves de Medeiros, Weijters & van der Aalst, 2004a).
<i>Enfoque heurístico</i>	[A4] [A15] [A19]	Utiliza la representación de redes causales, en el que los nodos son actividades y los arcos son dependencias causales (Weijters & Ribeiro, 2011).
<i>Enfoque basado en regiones</i>	[A7]	Utiliza la Teoría de Regiones en la minería de procesos para modelar los conjuntos de estados que caracterizan en una red Petri.

Enfoque Utilizado	Estudios	Descripción
Otros enfoques	[A11] [A12] [A14] [A16] [A17]	Redes Neuronales como modelos de la mente humana en su capacidad de “aprender” e identificar patrones en los datos (Cook & Wolf, 1998). Técnicas de agrupamiento que divide un grupo de soluciones en subgrupos homogéneos. Enfoque de Minería declarativa que introduce eventos negativos generados artificialmente que muestran que no es posible una ruta particular a través de un proceso. Minería borrosa que es recomendado para procesos menos estructurados, permite simplificar la visualización a través de cluster.

Tabla 5 – Algoritmos y enfoques utilizados en su desarrollo

5. Discusión

5.1. Algoritmos desarrollados

De la revisión de los estudios primarios fueron identificados 20 algoritmos desarrollados utilizando diferentes enfoques tales como enfoque general, enfoque evolutivo, enfoque heurístico, enfoque basado en regiones entre otros. Un registro completo de eventos es un requisito para obtener buenos resultados, es decir, con casos que comienzan y terminan. Los algoritmos en su mayoría están enfocados en superar los problemas de manejo de bucle 1 y 2 que el algoritmo α no puede resolver. Una tendencia es utilizar BPMN que se ha convertido en el estándar para modelar los procesos, pero la mayoría de los algoritmos estudiados utiliza redes Petri.

5.2. Sobre el manejo de los algoritmos para registro de eventos con presencia de ruidos

El resultado de los algoritmos depende de un buen registro de eventos, la mayoría de ellos requieren registros de eventos sin ruido, por ejemplo, en los algoritmos que utilizan el enfoque evolutivo, a un mayor porcentaje de ruido en el registro de eventos hay menos probabilidad de que el algoritmo entregue como resultado una red de proceso correcta. El estudio muestra que no es posible descubrir arbitrariamente flujo de trabajo, los algoritmos funcionan para ciertos tipos de flujos de trabajo y en los nuevos desarrollos se observan que se enfocan en resolver los problemas encontrados básicamente por el algoritmo α . Todavía está pendiente resolver el manejo del registro de eventos con ruido. Los datos de los eventos suelen ser dispersos en diferentes fuentes de datos y, a menudo, se necesita un esfuerzo para recopilar los datos relevantes y filtrarlos para evitar el ruido. En (van der Aalst, Schonenberg & Song, 2011) se realiza una descripción general del flujo de trabajo para obtener un buen registro de eventos para la minería de procesos de fuentes de datos heterogéneas.

5.3. Sobre la técnica de modelamiento utilizada por el algoritmo

Los algoritmos para representar el proceso descubierto utilizan técnicas de modelamiento de procesos. Se ha observado que la técnica más utilizada es la de redes Petri ([A1] [A2] [A3] [A5] [A6] [A7] [A8] [A9] [A10] [A14] [A16] [A18] [A19] [A20]) y lo utilizan para la generación del modelo final como paso intermedio una relación de orden. Es decir, según la revisión, en el 70% de los estudios se usa la técnica de redes Petri. En segundo lugar, se utiliza la técnica de los árboles de procesos BPMN que hoy es el estándar en modelamiento de procesos ([A11] [A12] [A13] [A15] [A17]). La tendencia es que los algoritmos descubran el proceso en el lenguaje BPMN, estándar adoptado por la industria de proceso. En (Kalenkova, van der Aalst, Lomazova & Rubin, 2016) se proponen algunos algoritmos que permiten la conversión de formalismos conocidos como Redes de Petri, Redes Causales y Árboles de Proceso a BPMN que es el estándar que se está imponiendo para modelar procesos de negocios. De los artículos revisados el 25% usa el lenguaje BPMN. Finalmente, los algoritmos con enfoque heurístico se utilizan para representar el proceso descubierto Causal Net, el algoritmo SLAD (Splitting Labels After Discovery) usa Causal Net como paso intermedio para mejorar y obtener una Red Petri (Alves de Medeiros, Weijters & van der Aalst, 2004a). Solo en el 5% de los estudios seleccionados se utiliza la técnica Causal Net.

6. Conclusiones

La minería de procesos utiliza los registros de eventos para descubrir los procesos que se están enjutando en el mundo reales del negocio, mediante la extracción de conocimiento. Los resultados de la revisión de los estudios seleccionados (20) muestran que la mayoría de los algoritmos utilizan un enfoque general. Los resultados son aceptables cuando los registros de datos son completos y limpios. El algoritmo α es la base para el desarrollo de las nuevas propuestas que tratan de resolver problemas que en los inicios no pudieron ser resueltos. Para validar los algoritmos se utilizan casos cuyos resultados son comparados con los resultados obtenidos de otros algoritmos desarrollados previamente. Los algoritmos utilizan técnicas de modelamiento para descubrir los procesos y como resultado se cuenta con un flujo de trabajo mapeado en un lenguaje de modelado. Una tendencia es utilizar BPMN que se ha convertido en el estándar para modelar los procesos, pero la mayoría de los algoritmos estudiados utiliza redes Petri. A pesar de los avances mostrados, la presencia de ruido en los registros de eventos es un problema que no ha sido solucionado para descubrir los procesos de manera satisfactoria.

Referencias

- Agrawal, R., Gunopulos, D., & Leymann, F. (1998). Mining process models from workflow logs. Paper presented at the *International Conference on Extending Database Technology*, (pp. 467-483).
- Alves de Medeiros, A. A., Van Dongen, B., Van der Aalst, W., & Weijters, A. (2004). *Process Mining: Extending the α -Algorithm to Mine Short Loops*. Eindhoven: Technische Universiteit Eindhoven.

- Alves de Medeiros, A.K., Weijters, A.J.M.M., & van der Aalst, W.M.P. (2004). *Using genetic algorithms to mine process models: representation, operators and results*. In: Beta Working Paper Series, WP 124. Eindhoven: Eindhoven University of Technology.
- Bergenthum, R., Desel, J., Lorenz, R., & Mauser, S. (2007). Process Mining Based on Regions of Languages. In: G. Alonso, P. Dadam, & M. Rosemann, (Eds). International Conference on Business Process Management (BPM 2007), volume 4714 of Lecture Notes in Computer Science, (pp 375–383). Berlin: Springer.
- Bratosin, C., Sidorova, N., & van der Aalst, W. (2010). Distributed genetic process mining. Paper presented at the *Evolutionary Computation (CEC), 2010 IEEE Congress on*, (pp. 1-8).
- Busi, N., & Pinna, G. M. (2009). Process discovery and petri nets. *Mathematical Structures in Computer Science*, 19(06), 1091–1124.
- Chinces, D., & Salomie, I. (2013). Business process mining using guided local search. In: *Proceedings of the Parallel and Distributed Computing (ISPDC), 2013 IEEE 12th International Symposium on*, (pp. 177-181).
- Chinces, D., & Salomie, I. (2013). Process discovery using ant colony optimization. In: *Proceedings of the Control Systems and Computer Science (CSCS), 2013 19th International Conference on*, (pp. 448-454).
- Conforti, R., Dumas, M., García-Bañuelos, L., & La Rosa, M. (2016). BPMN miner: Automated discovery of BPMN process models with hierarchical structure. *Information Systems*, 56, 284–303.
- Cook, J. E., & Wolf, A. L. (1998). Discovering models of software processes from event-based data. *ACM Transactions on Software Engineering and Methodology*, 7(3), 215–249.
- Cook, J. E., Du, Z., Liu, C., & Wolf, A. L. (2004). Discovering models of behavior for concurrent workflows. *Computers in Industry*, 53(3), 297–319.
- De Smedt, J., De Weerd, J., & Vanthienen, J. (2015). Fusion miner: Process discovery for mixed-paradigm models. *Decision Support Systems*, 77, 123–136.
- De Weerd, J., De Backer, M., Vanthienen, J., & Baesens, B. (2012). A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. *Information Systems*, 37(7), 654–676.
- Dustdar, S., Hoffmann, T., & Van der Aalst, W. (2005). Mining of ad-hoc business processes with TeamLog. *Data & Knowledge Engineering*, 55(2), 129–158.
- Goedertier, S., Martens, D., Vanthienen, J., & Baesens, B. (2009). Robust process discovery with artificial negative events. *Journal of Machine Learning Research*, 10, 1305–1340.

- Günther, C., & Van der Aalst, W. (2007). Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics. En: Alonso, G., Dadam, P., & Rosemann, M. (Eds.), *Business Process Management* (Vol. 4714, 2007, p. 328-343). Berlín: Springer Berlin Heidelberg.
- Hermida, A. G., Rodríguez, M. S., Padrón, M. V., Domas, L. R., & López, R. R. (2016). Alternativa para realizar análisis de rendimiento de procesos automatizados empleando técnicas de minería de procesos y simulación sobre redes de petri. *Investigación Operacional*, 37(1), 93–103.
- Kalenkova, A. A., van der Aalst, W.M.P., Lomazova, I. A., & Rubin, V. A. (2016). Process mining using BPMN: Relating event logs and process models. In: *Proceedings of the ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems*, (pp. 123-123).
- Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele University*, 33(2004), 1–26.
- Li, C., Ge, J., Huang, L., Hu, H., Wu, B., Hu, H., et al. (2017). Software cybernetics in BPM: Modeling software behavior as feedback for evolution by a novel discovery method based on augmented event logs. *Journal of Systems and Software*, 124, 260–273.
- Li, C., Ge, J., Huang, L., Hu, H., Wu, B., Yang, H., et al. (2016). Process mining with token carried data. *Information Sciences*, 328, 558–576.
- Lu, F., Zeng, Q., & Duan, H. (2016). Synchronization-core-based discovery of processes with decomposable cyclic dependencies. *ACM Transactions on Knowledge Discovery from Data*, 10(3), 31.
- Mejía, J., & Muñoz, M. (2017). Tendencias en Tecnologías de Información y Comunicación. RISTI - Revista Ibérica de Sistemas y Tecnología de información, 21, 51-66. Doi: 10.17013/risti.21.51-66.
- Molka, T., Redlich, D., Drobek, M., Zeng, X., & Gilani, W. (2015). Diversity guided evolutionary mining of hierarchical process models. In: *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation*, (pp. 1247-1254).
- Norambuena, B., & Zepeda, V. (2017). Minería de procesos de software: una revisión de experiencias de aplicación. RISTI - Revista Ibérica de Sistemas y Tecnología de información, 21, 51-66. Doi: 10.17013/risti.21.51-66.
- Pourmasoumi, A., & Bagheri, E. (2016). Business process mining. *Encyclopedia with Semantic Computing*, 1(1), 1–4.
- Schimm, G. (2004). Mining exact models of concurrent workflows. *Computers in Industry*, 53, 265-81.
- Tiwari, A., Turner, C. J., & Majeed, B. (2008). A review of business process mining: State-of-the-art and future trends. *Business Process Management Journal*, 14(1), 5–22.

- Turner, C. J., Tiwari, A., Olaiya, R., & Xu, Y. (2012). Process mining: From theory to practice. *Business Process Management Journal*, 18(3), 493–512.
- Van Der Aalst, W., Rubin, V., Verbeek, H. M., van Dongen, B. F., Kindler, E., & Günther, C. W. (2010). Process mining: A two-step approach to balance between underfitting and overfitting. *Software and Systems Modeling*, 9(1), 87–111.
- Van der Aalst, W., & Vanthienen, J. (2011). IEEE task force on process mining. *Lecture Notes in Business Information Processing*, 99, 169–194.
- Van Der Aalst, W., Adriansyah, A., & Van Dongen, B. (2011). Causal nets: A modeling language tailored towards process discovery. In: *Proceedings of International Conference on Concurrency Theory*, (pp. 28-42).
- Van der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9), 1128–1142.
- Van der Aalst, W., & Weijters, A. (2004). Process mining: A research agenda. *Computers in Industry*, 53(3), 231–244.
- Van der Aalst, W., Alves De Medeiros, A. A., & Weijters, A. (2005). Genetic process mining. In: *Proceedings of International Conference on Application and Theory of Petri Nets*, (pp. 48-69).
- Van der Aalst, W., Schonenberg, M. H., & Song, M. (2011). Time prediction based on process mining. *Information Systems*, 36(2), 450–475.
- Van der Aalst, W., van Dongen, B. F., Herbst, J., Maruster, L., Schimm, G., & Weijters, A. J. (2003). Workflow mining: A survey of issues and approaches. *Data & Knowledge Engineering*, 47(2), 237–267.
- Van der Aalst, W. (2011). Getting the data. In: *Process mining* (pp. 95-123). Berlin: Springer.
- Van Der Aalst, W. (2015). Business process simulation survival guide. In: *Handbook on business process management 1* (pp. 337-370). Berlin: Springer.
- Van der Aalst, W. (2016). *Process mining: Data science in action*. Berlin: Springer.
- van Dongen, B.F., Busi, N., Pinna, G.M., & van der Aalst, W. (2007). An Iterative Algorithm for Applying the Theory of Regions in Process Mining. In: W. Reisig, K. van Hee, & K. Wolf, (eds). *Proceedings of the Workshop on Formal Approaches to Business Processes and Web Services (FABPWS'07)*, (pp.36– 55). Siedlce: Publishing House of University of Podlasie.
- Van den Broucke, S. K., & De Weerd, J. (2017). Fodina: A robust and flexible heuristic process discovery technique. *Decision Support Systems*.
- Vázquez-Barreiros, B., Mucientes, M., & Lama, M. (2016). Enhancing discovered processes with duplicate tasks. *Information Sciences*, 373, 369–387.

- Wang, D., Ge, J., Hu, H., & Luo, B. (2011). A new process mining algorithm based on event type. In: *Proceedings of Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on*, (pp. 1144-1151).
- Wang, D., Ge, J., Hu, H., Luo, B., & Huang, L. (2012). Discovering process models from event multiset. *Expert Systems with Applications*, 39(15), 11970–11978.
- Weijters, A., & Ribeiro, J. (2011). Flexible heuristics miner (FHM). In: *Proceedings of Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on*, (pp. 310-317).
- Weijters, A., van Der Aalst, W., & Alves De Medeiros, A. A. (2006). Process mining with the heuristics miner-algorithm. *Tech.Rep.WP*, 166, 1–34.
- Wen, L., van der Aalst, W., Wang, J., & Sun, J. (2007). Mining process models with non-free-choice constructs. *Data Mining and Knowledge Discovery*, 15(2), 145-180.
- Wen, L., Wang, J., van der Aalst, W., Huang, B., & Sun, J. (2009). A novel approach for process mining based on event types. *Journal of Intelligent Information Systems*, 32(2), 163–190.
- Wen, L., Wang, J., van der Aalst, W., Huang, B., & Sun, J. (2010). Mining process models with prime invisible tasks. *Data & Knowledge Engineering*, 69(10), 999–1021.

Influencia de la Teoría de Roles de Belbin en la Medición de Software: Un estudio exploratorio

Raúl Aguilar Vera¹, Julio Díaz Mendoza¹, Juan Ucán Pech¹

avera@correo.uady.mx, julio.diaz@correo.uady.mx, juan.ucan@correo.uady.mx

¹ Facultad de Matemáticas, Universidad Autónoma de Yucatán, C.P. 97000, Mérida, Yucatán, México.

DOI: 10.4304/risti.31.50-65

Resumen: Se presenta un estudio exploratorio en el que se analiza la influencia de la Teoría de Belbin en actividades de desarrollo de software como mecanismo para la mejora del mismo. El estudio comprende un primer experimento controlado con estudiantes y una réplica similar-interna. La actividad considerada para los equipos, fue la medición de software mediante la técnica de puntos de función. Los resultados obtenidos en el primer experimento no fueron concluyentes debido a que no fueron confirmados en la réplica, no obstante, las lecciones aprendidas con el estudio, permiten identificar y proponer varios experimentos controlados como trabajos futuros.

Palabras-clave: Experimentación en Ingeniería de Software; Integración de Equipos; Medición de Software; Roles de Belbin.

Influence of Belbin's Role Theory in Software Measurement: An Exploratory Study

Abstract: An exploratory study is presented in which the influence of the Belbin Theory on software development activities as a mechanism to improve it is analyzed. The study includes a first controlled experiment with students and a replication. The activity considered for the teams, was the measurement of software using the function point technique. The results obtained in the first experiment were not conclusive because they were not confirmed in the experimental replica, however, the lessons learned with the study allow to identify and propose several controlled experiments as future works.

Keywords: Belbin Roles; Experimentation in Software Engineering; Software Measurement; Teams Integration.

1. Introducción

El desarrollo de software, desde el momento mismo de su concepción como disciplina ingenieril, ha mantenido una dinámica de mejora continua en cuanto a sus métodos, técnicas, herramientas y prácticas. Medio siglo después de la denominada “Crisis del Software” la Ingeniería de Software (IS) cuenta con un cuerpo de conocimientos

aceptado en lo general, por profesionistas e investigadores de la disciplina (Bourque & Fairley, 2014), sin embargo, la dinámica de mejora en cuanto a calidad, tanto en sus procesos, como en los artefactos generados a través de los mismos, sigue siendo fuente de investigación para dicha disciplina ingenieril (p.e. Miramontes et al, 2016; Norambuena y Vega, 2017).

El estudio de los enfoques vinculados con el desarrollo, operación y mantenimiento del software, ha sido analizado con base en la relación de diversas variables, no obstante, el aspecto social intrínseco de la disciplina (Juristo & Moreno, 2001) ha llevado a considerar al factor humano, como un aspecto de singular interés para su investigación. En un estudio reciente, Morales y Vega (2018) proponen un catálogo de factores humanos críticos para el éxito de propuestas de mejora al proceso software; uno de los cuatro factores se relaciona con el rol que desempeña un Ingeniero de Software al interior del equipo de trabajo; Humphrey (2000) afirma que el proceso de formación y creación de un equipo de desarrollo de software no se produce por accidente, si no que el equipo debe establecer relaciones de trabajo, ponerse de acuerdo sobre los objetivos y determinar las funciones de los miembros del grupo. En la última década, incluso elementos de gamificación han sido utilizados como estrategia para integrar equipos de trabajo (Hernández et al, 2016).

El presente estudio, tiene como propósito explorar si el uso de la teoría de roles de Belbin (1993) para la formación de equipos, tiene alguna influencia en una de las tareas vinculadas con los procesos de gestión de software, particularmente, en la tarea de medición. El documento se encuentra organizado de la siguiente manera: la sección 2 presenta trabajos relacionados con el estudio, la sección 3 describe brevemente la técnica de medición de software conocida como puntos de función. La sección 4 presenta un primer experimento realizado. La sección 5 describe una réplica experimental. Finalmente, las conclusiones se presentan en la sección 7.

2. Trabajos relacionados con es estudio

Son varios los investigadores (Belbin, 1981; Briggs-Myers & McCaulley, 1985; Margerison & McCan, 1985; Mumma, 2005) que afirman haber identificado roles que definen el comportamiento de los individuos al interior de un equipo de trabajo, dichos roles son conocidos bajo el nombre de roles de equipo, y aunque no existe el reconocimiento de la existencia de una relación funcional con algún proceso o tarea particular, su ausencia o presencia, al parecer tiene una influencia significativa en los logros del equipo (Aritzeta, Swailes & Senior, 2005).

Entre los estudios existentes sobre roles de equipo, el trabajo de Belbin (1981, 1993) es probablemente el más recurrido entre consultores e investigadores, su popularidad radica en que además de ofrecer una categorización de roles (Impulsor:IM, Implementador:IP, Finalizador:F, Cerebro:C, Monitor-Evaluador:ME, Especialista:E, Coordinador:CO, Investigador de Recursos:IR y Cohesionador:CH), presenta recomendaciones para la integración de equipos con base en roles compatibles, así como mecanismos para la identificación de los roles que un individuo puede asumir en un equipo de trabajo, en función de su comportamiento. Johansen (2003) en su disertación sobre la gestión del recurso humano, validó el inventario de autopercepción propuesto por Belbin para la

identificación del rol natural o roles que un individuo puede asumir en un equipo de trabajo, y entre sus conclusiones, indicó que dicho inventario es definitivamente una herramienta que vale la pena utilizar para evaluar la composición y posible desempeño del equipo.

En cuanto a las investigaciones sobre el uso de la teoría de Belbin en el contexto de la Ingeniería de Software, se encuentra el reporte de Henry y Stevens (1999) cuyo esfuerzo de investigación se centró en el análisis de la utilidad de los roles de Belbin para mejorar la efectividad de los equipos, en particular, la influencia del liderazgo en el grupo; dichos autores comentan que los roles de Belbin se pueden utilizar en la formación de nuevos equipos, así como en la evaluación de los equipos existentes, y luego del desarrollo de un experimento controlado con estudiantes, concluyen que un solo líder mejora el rendimiento de un equipo, en comparación con equipos que tienen múltiples líderes o ningún líder. Considerando también a la efectividad como el aspecto a mejorar, Pollock (2009) exploró la influencia de contar con una diversidad de roles y personalidades, como estrategia para mejorar la efectividad del equipo, y concluye, que aunque no encontró evidencia de que dicha diversidad tiene influencia en el equipo, pudo observar que la presencia de ciertos roles como son el caso del Impulsor, Coordinador o Finalizador, pueden incrementar la efectividad.

Otros estudios se han centrado en analizar la relación de los roles de Belbin con tareas vinculadas con el proceso software. Estrada y Peña (2013) reportan un experimento controlado con estudiantes en ciertas actividades vinculadas con las fases de requisitos, diseño y codificación; las autoras concluyen que algunos roles presentan mayor aportación en ciertas actividades, particularmente señalan al rol Implementador en la tarea de codificación. Así mismo, en un estudio previo en el contexto de la fase de codificación, comparamos la calidad de la legibilidad del código generado por equipos integrados con roles compatibles, de acuerdo con la Teoría de Belbin, en contraste con equipos formados por integrantes asignados aleatoriamente, los resultados obtenidos mediante un experimento controlado con estudiantes, nos permitieron concluir que los equipos que fueron integrados utilizando como referencia la Teoría de Belbin, presentan resultados significativamente mejores que los equipos integrados mediante un criterio de aleatoriedad (Aguileta, Ucán y Aguilar, 2017).

3. Metodología

Aunque el cuerpo de conocimientos de la IS comenzó a integrarse a finales de la década de los sesenta, fue hasta la década de los ochenta cuando la comunidad académica comenzó a adoptar y utilizar enfoques de investigación para estudiar de manera rigurosa los diferentes aspectos y problemas involucrado con el proceso de desarrollo de software (Glass, Vessey & Ramesh, 2002).

Una de las metodologías más utilizadas en el campo de la IS, es la experimentación, en particular, la experimentación en entornos controlados; el énfasis en aplicar la experimentación en IS se remonta a la década de los años ochenta (Basili, Selby & Hutchens, 1986) y tiene como objetivo identificar las causas por las cuales se producen ciertos resultados, su aplicación nos ayuda a identificar y comprender las posibles relaciones entre factores y variables dependientes, ambos elementos involucrados en los

procesos de desarrollo de software. Entre los elementos característicos de los estudios exploratorios encontrados en la literatura, se destaca el uso de grupos de estudiantes como sujetos experimentales; Genero, Cruz-Lemus y Piattini (2014) indican al respecto, que dicha muestra académica permite al investigador obtener evidencia preliminar para confirmar o refutar hipótesis que pueden ser contrastadas posteriormente en contextos industriales. Por otro lado, un elemento clave en el paradigma experimental, es el desarrollo de réplicas, su propósito radica en verificar o refutar los hallazgos previamente observados (Gómez, Juristo & Vegas, 2014).

El presente estudio tiene como propósito explorar, mediante la ejecución de un experimento controlado con estudiantes, y una réplica experimental similar-interna, la influencia del uso de la Teoría de Belbin en la integración de equipos de desarrollo de software con integrantes que presentan roles compatibles, en el producto de una actividad, así como en el tiempo invertido en dicha actividad; en el caso del estudio se seleccionó una actividad vinculada con las tareas de gestión, particularmente, la medición del tamaño lógico o funcional de un sistema de software propuesto.

3.1. Factor y Tratamientos

El factor considerado para nuestro estudio, es la estrategia utilizada para la integración de los equipos de trabajo, particularmente, con dos tratamientos:

Equipos Compatibles (EC): Equipos de trabajo de tres integrantes, conformados con roles compatibles —identificados con el inventario de autopercepción— de acuerdo con la Teoría de Belbin.

Equipos Tradicionales (ET): Equipos de trabajo de tres integrantes, seleccionados aleatoriamente para su conformación.

3.2. Hipótesis y Variables Dependientes

De acuerdo con el propósito de nuestro estudio, el objetivo planteado para el experimento así como para su réplica, consiste en comparar métricas —del producto y del proceso— obtenidas mediante equipos integrados con roles compatibles —con base en la Teoría de Belbin— con las obtenidas por equipos integrados en forma aleatoria.

El primer par de hipótesis estadísticas utiliza como variable dependiente, una métrica del producto, es decir, el número de puntos de función sin ajustar obtenidos (PFSA) por los equipos al medir el sistema (Garmus & Herron, 1996).

- H01: La media de los PFSA obtenidos con la Técnica de Puntos de Función por los ET es igual a la media de los PFSA obtenidos por los EC.
- H11: La media de los PFSA obtenidos con la Técnica de Puntos de Función por los ET difiere de la media de los PFSA obtenidos por los EC.

En cuanto a la variable vinculada con el proceso, utilizaremos el tiempo en minutos (TM) utilizado para concluir la actividad, las hipótesis estadísticas planteadas son las siguientes:

- H02: El tiempo medio registrado por los ET en la actividad de medición es igual al reportado por los EC.

- H12: El tiempo medio registrado por los ET en la actividad de medición difiere del reportado por los EC.

3.3. Parámetros

Aspectos como la complejidad del problema a resolver, la presión del tiempo disponible para concluir la actividad, así como la experiencia de los participantes, se consideran parámetros que no afectan o sesgan los resultados del estudio, debido a que son parámetros que se mantuvieron homogéneos para todos los equipos de desarrollo.

En el caso de la experiencia de los sujetos, al ser equipos integrados por estudiantes que se encuentran en su proceso de formación y que antes del experimento, aún no habían estudiado la técnica, podemos considerar que son homogéneos —la técnica fue analizada con todos los estudiantes en una sesión de trabajo. Por su parte, en lo referente a la complejidad del sistema por estimar, se redactó un documento de Especificación de Requisitos para un Software Educativo, con base en el estándar IEEE Std. 830 1998, dicho documento fue el mismo que se entregó a todos los equipos en la tarea. Finalmente, en cuanto al tiempo disponible para la realización de la actividad, se realizó una estimación del tiempo requerido para la actividad, y considerando un estimado de 120 minutos, se dispuso de dos sesiones presenciales (180 minutos) para la actividad de medición.

3.4. Diseño Experimental

De acuerdo con las características del estudio, el diseño más apropiado para el experimento es el diseño factorial con una fuente de variación (ver tabla 1), la variable dependiente corresponde, en el primer par de hipótesis, a la métrica obtenida por los equipos en la tarea (PFSA) y en la segunda, al tiempo utilizado por los equipos para su finalización (Tiempo en segundos). En ambos experimentos, los tratamientos experimentales se corresponden con: (1) equipos integrados con base en la teoría del roles de Belbin (Equipos Belbin) y (2) equipos integrados de manera forma aleatoria (Equipos Tradicionales).

	Tratamientos	Equipos Integrados con Estudiantes
Factor	<i>Equipos Belbin</i>	<i>Mediciones sobre: PFSA/Tiempo</i>
	<i>Equipos Tradicionales</i>	<i>Mediciones sobre: PFSA/Tiempo</i>

Tabla 1 – Diseño factorial con una fuente de variación

3.5. Análisis estadístico

El análisis de los datos que serán recolectados en ambos experimentos, contendrá una sección descriptiva a partir de la generación de gráficos de caja y bigotes, y en el caso del análisis inferencial, se utilizará el modelo estadístico del análisis de varianza simple. Los gráficos de caja y bigotes permiten generar una presentación visual en la que se pueden observar diversos datos de los conjuntos de mediciones por analizar: valor mínimo y máximo, mediana, cuartil inferior y superior, así como valores atípicos; así mismo, permiten observar la forma y dispersión de los conjuntos de datos, pero sobre todo, permiten evaluar empíricamente el comportamiento de los conjuntos de mediciones por comparar.

Para realizar el análisis inferencial, elegimos el análisis de varianza de una vía, debido a que nos permite realizar pruebas de hipótesis para determinar si hay o no diferencias significativas entre las medias y varianzas, de los valores recogidos en la variable dependiente, para los diferentes niveles del factor (tratamientos). El análisis de la varianza (Gutierrez y de la Vara, 2012) permite construir con los datos, un modelo estadístico que describe el impacto de un solo factor categórico sobre una variable dependiente, dicho modelo lineal asociado es el siguiente:

$$Y_{ij} = \mu + \beta_i + \epsilon_{ij};$$

donde Y_{ij} es la ij -ésima medición, μ es un parámetro común para todos los tratamientos, denominado media global, β_i es un parámetro que mide el efecto del i -ésimo tratamiento, y ϵ_{ij} es el error sistemático atribuible a la medición Y_{ij} .

4. Primer Experimento

4.1. Participantes/Sujetos Experimentales

La muestra por conveniencia utilizada para el experimento estuvo conformada por 27 estudiantes de carrera de Ingeniero de Software de la Universidad Autónoma de Yucatán, los participantes cursaban la asignatura “Diseño de Experimentos en Ingeniería de Software” ubicada en el quinto semestre de la carrera. Con los 27 participantes, se formaron 9 equipos de trabajo —sujetos experimentales. La tabla 2 ilustra la asignación de los equipos a los tratamientos.

Tratamientos	Equipos
EB	I, II, III, IV, V
ET	VI, VII, VIII, IX

Tabla 2 – Asignación de sujetos experimentales por tratamiento

La integración de los Equipos Compatibles, con base en los roles identificados mediante el inventario de autopercepción, se ilustra en la figura 1.

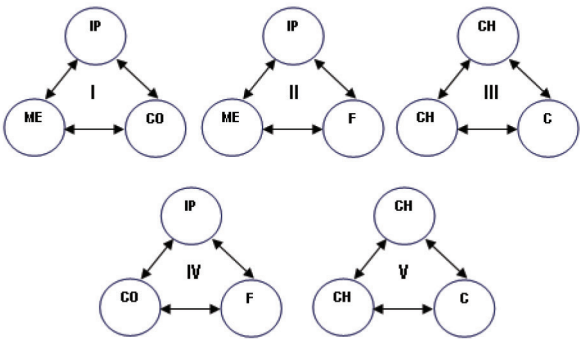


Figura 1 – Equipos integrados con la Teoría de Belbin (Equipos Compatibles)

4.2. Ejecución del Experimento

El experimento se llevó a cabo en tres sesiones de trabajo, en la primera, se administró el inventario de autopercepción, a partir del cual se identificaron los roles primarios de los participantes; posteriormente, en una segunda sesión que duró dos horas, los participantes recibieron capacitación sobre la técnica del análisis de puntos de función; finalmente, la sesión experimental fue realizada.

Al inicio de la sesión experimental, se integraron los nueve equipos, de acuerdo con lo indicado en la tabla 2, se les proporcionó el documento de Especificación de Requisitos de un caso, así como una hoja de informe con el esquema de la tabla 1; posteriormente, se describió la actividad y se les pidió que identificaran en la hoja de informe: el número de equipo, hora de inicio y de finalización de la actividad, así como las métricas correspondientes a la técnica. La sesión transcurrió sin problemas.

4.3. Análisis y Resultados del Experimento

En esta sección se presenta tanto el análisis descriptivo, como el análisis inferencial con base en las métricas seleccionadas para el experimento.

Primeramente se analizan los gráficos de Caja y Bigotes generados para las dos variables dependiente consideradas, a efecto de tratar de identificar intuitivamente el comportamiento de los datos (ver figura 2).

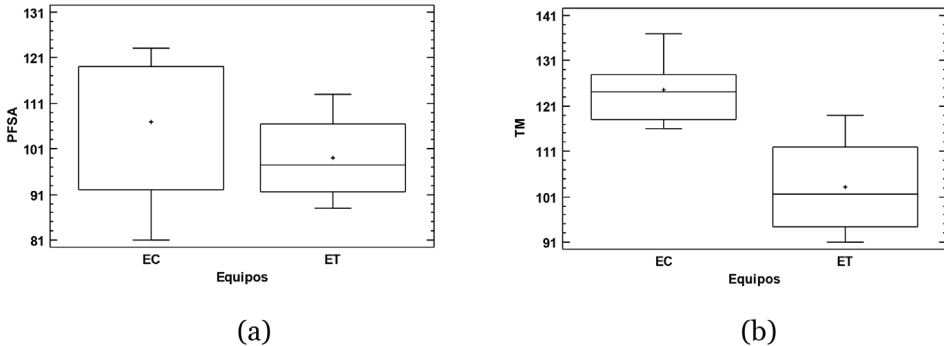


Figura 2 – Gráficos de Caja y Bigotes para las variables PFSA (a) y TM (b)

Como se observa en el gráfico de la figura 2(a), en cuanto a la variable PFSA, los EC presentan mayor dispersión y un sesgo negativo respecto de su mediana, por su parte los ET tienen un comportamiento más simétricos; si observamos ambas gráficas al mismo tiempo, vemos que el área de la caja de los ET se encuentra incluida en el área de los EC, lo cual nos da indicios de que no haber diferencias en cuanto a la variable PFSA. Por su parte, la figura 2(b) ilustra el comportamiento de los datos vinculados con la variable TM, observamos que los ET presentan mayor dispersión que los EC, sin embargo, dado que las áreas de ambas cajas se encuentran desfasadas, podemos suponer que el factor tiene cierta influencia.

Con el objeto de evaluar estadísticamente las diferencias observadas en el análisis descriptivo previo, se plantearon las siguientes hipótesis estadísticas.

$$H_{0_{PFSA}}: \mu_{EC} = \mu_{ET} \quad ; \quad H_{1_{PFSA}}: \mu_{EC} <> \mu_{ET} \quad (I)$$

$$H_{0_{Time}}: \mu_{EC} = \mu_{ET} \quad ; \quad H_{1_{Time}}: \mu_{EC} <> \mu_{ET} \quad (II)$$

El resultado de evaluar con ANOVA ambas hipótesis, se ilustra en la tabla 3.

Métrica	F	p-valor
PFSA	0.53	0.4894
TM	9.96	0.0160 *

*p-valor < 0.05

Tabla 3 – ANOVA para las variable PFSA y TM

La tabla ANOVA descompone la varianza de la variable en estudio en dos componentes: un componente intergrupo y un componente en grupos. Dado que el valor p de la prueba F —en la variable PFSA— es mayor que 0.05 (PFSA: 0.4894) la hipótesis de nulidad se acepta, es decir, se afirma que no existe diferencias entre las medias de los dos tratamientos.

Por su parte, el valor P de la prueba F —en la variable TM— es menor que 0.05 (TM: 0.016), entonces podemos rechazar la hipótesis de nulidad y podemos afirmar que existe una diferencia estadísticamente significativa entre las medias de la variable TM de los dos tratamientos.

El modelo ANOVA tiene asociado tres supuestos que es necesario validar antes de utilizar la información que nos ofrece; los supuestos del modelo son: (1) Los errores experimentales de sus datos se distribuyen normalmente, (2) No existe diferencia entre la varianza de los tratamientos (Homocedasticidad) y (3) existe independencia entre las muestras.

Para validar el primer supuesto, usaremos el gráfico de probabilidad normal de los residuos. Es una técnica gráfica para evaluar si un conjunto de datos se distribuye de acuerdo con la distribución normal. Como se puede ver en la gráfica que se ilustra en la figura 3, los puntos, en ambas gráficas, no muestran desviaciones de la diagonal, por lo que es posible asumir que los residuos tienen una distribución normal en ambos casos.

En el caso de la homocedasticidad, generamos un gráfico de residuales vs. predichos y observamos si es posible detectar que el tamaño de los residuales aumenta o disminuye sistemáticamente a medida que aumenta los valores predichos. Como podemos ver en los dos gráficos de la figura 4, no se observa patrón particular alguno, con el cual podamos aceptar que se cumple la hipótesis de varianza constante de los residuos.

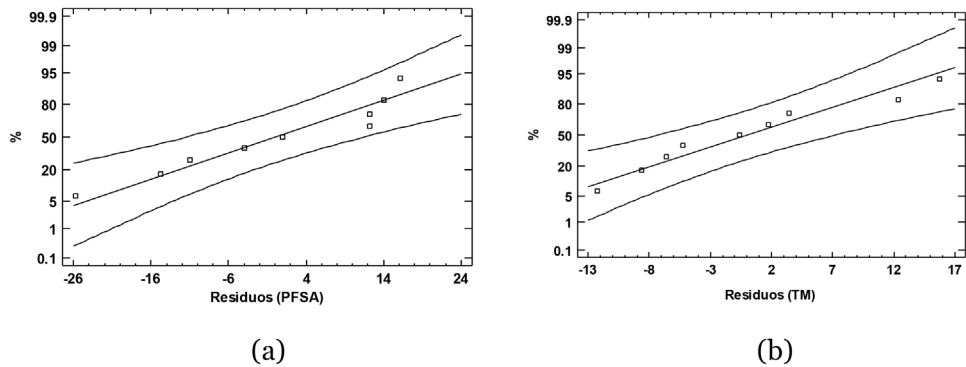


Figura 3 – Gráficos de Probabilidad Normal de Residuos: PFSA (a) y TM (b)

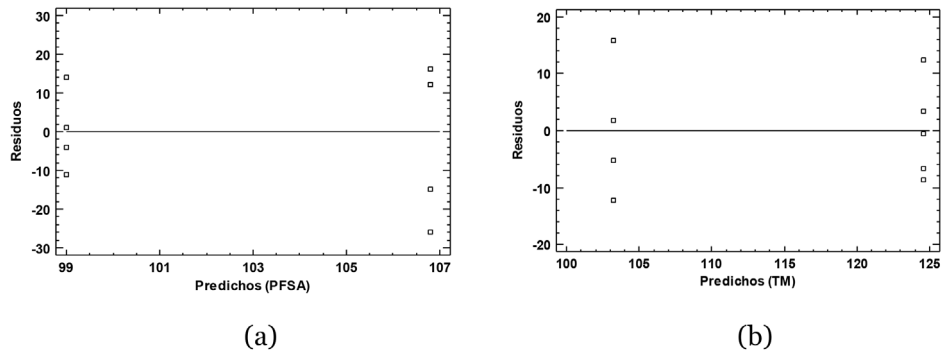


Figura 4 – Gráficos de Residuos vs. Predichos: PFSA (a) y TM

Finalmente, para validar el supuesto de independencia de los datos, generamos un gráfico de residuales vs. secuencia de los datos; en este caso, observamos si es posible detectar

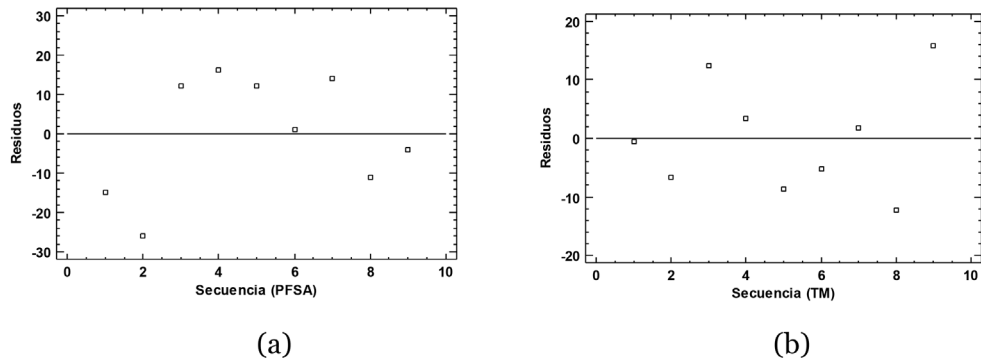


Figura 5 – Gráficos de Residuos vs. Secuencia: PFSA (a) y TM

alguna tendencia a tener rachas con residuos positivos y negativos; en el caso de nuestro análisis, podemos ver en la figura 5 que en ambos casos no se identifica una tendencia, por lo que es posible suponer que los datos provienen de poblaciones independientes.

Con el propósito de profundizar en el análisis, utilizamos el ANOVA para evaluar pruebas de hipótesis relacionadas con cada uno de los cinco componentes de la métrica PFSA (Ficheros Lógicos Internos: ILF, Ficheros de interface externos: EIF, Entradas Externas: EI, Salidas Externas: EO y Consultas Externas EQ); la tabla 4 resume los valores obtenidos para dicho análisis inferencial.

Métrica	F	p-valor
ILF	0.09	0.7730
EIF	0.01	0.9200
EI	0.01	0.9069
EO	10.94	0.0130 *
EQ	0.08	0.7916

*p-valor < 0.05

Tabla 4 – ANOVA para las cinco métricas que componen los PFSA

Como se observa en la tabla 4, únicamente uno de los cinco componentes de la métrica de PFSA, presentó diferencias estadísticamente significativas; el componente vinculado con el elemento de transacciones “salidas externas”.

Resulta importante mencionar, que aunque no se incluye en el artículo, la validación de los tres supuestos del modelo fue realizada por los autores, y en los cinco casos antes citados, el modelo resultaba válido.

5. Réplica Experimental

5.1. Participantes/Sujetos Experimentales

La muestra por conveniencia disponible para la réplica experimental estuvo conformada por 18 estudiantes de la carrera de Ingeniero de Software de la Universidad Autónoma de Yucatán, los participantes cursaban por segunda ocasión —en un curso de verano— la asignatura “Fundamentos de Ingeniería de Software” ubicada en el primer semestre de la carrera. Con los 18 participantes —de primer curso— se integraron 6 equipos de trabajo que fungieron como sujetos experimentales. La tabla 5 ilustra la asignación de sujetos a los tratamientos.

Tratamientos	Equipos
EB	I, II, III
ET	IV, V, VI

Tabla 5 – Asignación de sujetos experimentales por tratamiento

La integración de los Equipos Compatibles, con base en los roles identificados mediante el inventario de autopercepción, se ilustra en la figura 6.

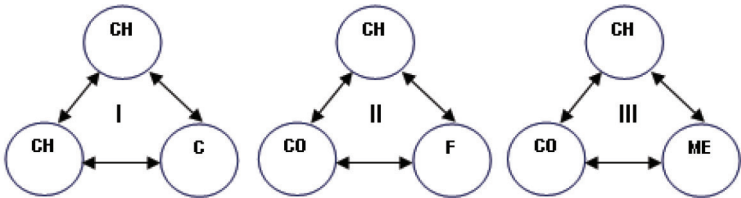


Figura 6 – Equipos integrados con la Teoría de Belbin (Equipos Compatibles)

5.2. Ejecución del Experimento

La ejecución de la réplica experimental se desarrolló en tres sesiones, en forma similar al primer experimento, es decir, para la sesión de capacitación, se utilizó el mismo material de instrucción, incluso fue el mismo instructor quién impartió dicho taller; en cuanto al problema por resolver, se utilizó el mismo Documento de Especificación de Requisitos, y en relación con el tiempo disponible para la actividad, también se dispuso de 180 minutos. Un parámetro que se consideró homogéneo respecto del primer experimento, fue el de la experiencia de los participantes, ya que en el caso de la réplica, los alumnos se encontraban en su primer año de la carrera.

5.3. Análisis y Resultados del Experimento

En esta sección se presenta tanto el análisis descriptivo, como el análisis inferencial con base en las variables seleccionadas para el experimento. Los gráficos de Caja y Bigotes generados para las variables dependientes consideradas se presentan en la figura 7. Como se observa en el gráfico de la figura 7(a), en cuanto a la variable PFSA, los ET presentan mayor dispersión y un sesgo negativo respecto de su mediana, por su parte los EC tienen poca dispersión pero también presentan un sesgo negativo respecto de su mediana.; si observamos ambas gráficas al mismo tiempo, vemos que el área de la caja de los EC se encuentra totalmente desfasada respecto de los ET, lo cual nos da indicios de diferencias con a variable PFSA.

Por su parte, la figura 7(b) ilustra el comportamiento de los datos vinculados con la variable TM, observamos que los EC presentan mayor dispersión que los ET, y ambos mantienen un sesgo positivo, también se puede observar que el área de la caja de los ET se encuentra incluida en el área de los EC, lo cual nos da indicios de que no haber diferencias en cuanto a la variable TM.

Con el objeto de evaluar estadísticamente las diferencias observadas en el análisis descriptivo previo, se plantearon las mismas hipótesis estadísticas que en el primer experimento.

$$H_{0_{PFSA}}: \mu_{EC} = \mu_{ET} \quad ; \quad H_{1_{PFSA}}: \mu_{EC} < \mu_{ET} \quad (I)$$

$$H_{0_{Time}}: \mu_{EC} = \mu_{ET} \quad ; \quad H_{1_{Time}}: \mu_{EC} <> \mu_{ET}$$

(II)

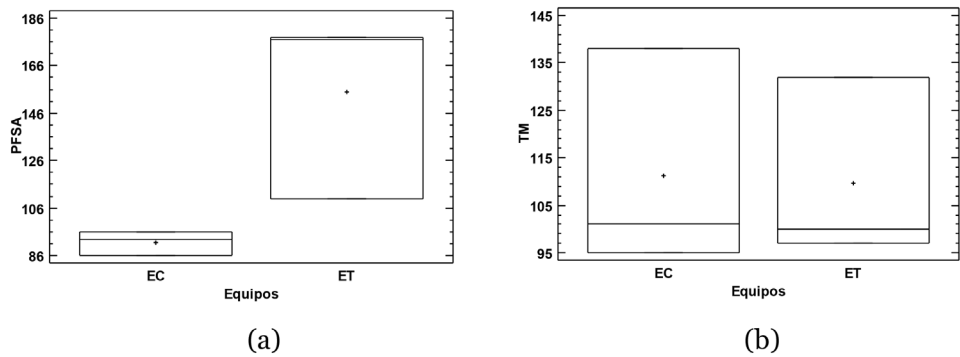


Figura 7 – Gráficos de Caja y Bigotes para las variables PFSA (a) y TM (b)

El resultado de evaluar con ANOVA ambas hipótesis, se ilustra en la tabla 6.

Métrica	F	p-valor
PFSA	7.79	0.0494 *
TM	1.01	0.9287

*p-valor < 0.05

Tabla 6 – ANOVA para las variable PFSA y TM

En virtud de que el valor P de la prueba F —en la variable PFSA- es menor que 0.05 (PFSA: 0.0494), entonces podemos rechazar la hipótesis de nulidad y afirmar que existe una diferencia estadísticamente significativa entre las medias de la variable PFSA de los tratamientos. Por su parte el valor p de la prueba F —en la variable TM— es mayor que

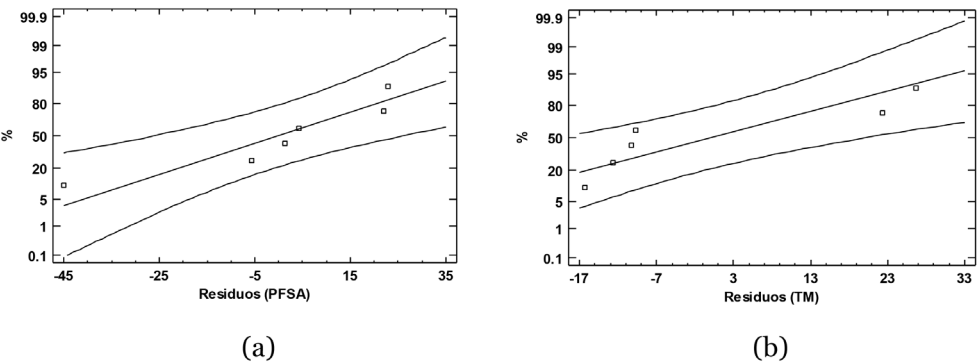


Figura 8 – Gráficos de Probabilidad Normal de Residuos: PFSA (a) y TM (b)

0.05 (TM: 0.9287) la hipótesis de nulidad se acepta, es decir, se afirma que no existe diferencias entre las medias de los dos tratamientos.

Para validar el primer supuesto del modelo ANOVA, generaremos el gráfico de probabilidad normal de los residuos para ambas variables. Como se puede ver en la gráfica de la figura 8, los puntos, en ambas gráficas, no muestran desviaciones graves de la diagonal, por lo que es posible asumir que los residuos tienen una distribución normal en ambos casos.

Para analizar la homogeneidad de varianzas, observamos el gráfico de residuales vs. predichos; y como podemos observar en ambas variables, no se tiene un comportamiento de aumento o disminución de valores predichos, por lo que consideramos homogeneidad de varianzas.

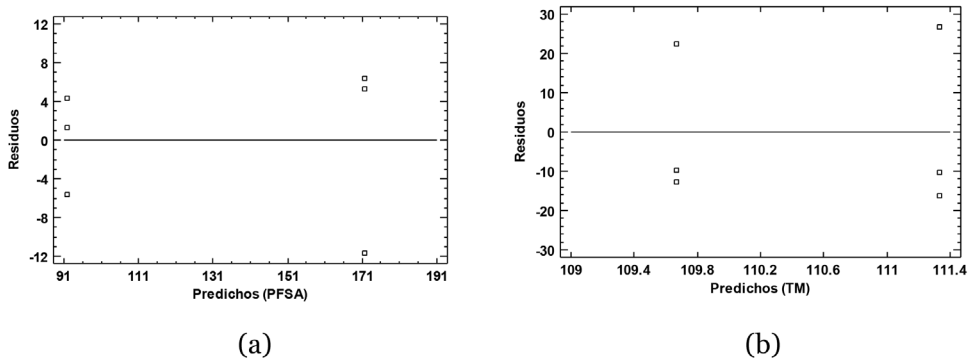


Figura 9 – Gráficos de Residuos vs. Predichos: PFSA (a) y TM

Finalmente, para validar el supuesto de independencia de los datos, generamos un gráfico de residuales vs. secuencia de los datos, y como podemos ver en la figura 10, en ambos casos no se identifica tendencia alguna en los datos, por lo que podemos suponer que los datos provienen de poblaciones independientes.

De manera similar a lo reportado en el primer experimento, se realizaron los cinco análisis de varianza —para cada componente de la métrica PFSA— con sus respectivas pruebas de validez al modelo, la cuales, por las limitaciones de espacio no serán incluidas en el artículo; la tabla 7 resume los valores obtenidos para el ANOVA.

Métrica	F	p-valor
ILF	13.31	0.0218 *
EIF	0.84	0.4104
EI	1.50	0.2879
EO	0.13	0.7406
EQ	0.23	0.6542

*p-valor < 0.05

Tabla 7 – ANOVA para las cinco métricas que componen los PFSA

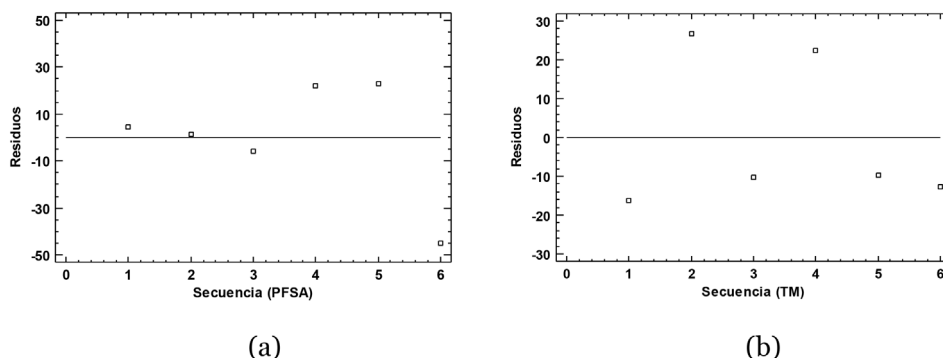


Figura 10 – Gráficos de Residuos vs. Secuencia: PFSA (a) y TM

Como se observa en la tabla 7, aunque de manera global la variable PFSA presenta diferencias significativas entre ambos tratamientos, en cuanto a sus componentes, solamente uno de los cinco presentó diferencias estadísticamente significativas, particularmente, el componente vinculado con el elemento de datos “ficheros lógicos internos”.

6. Conclusiones y Trabajos Futuros

En este estudio presentamos un primer experimento controlado con el propósito de explorar si el uso de la teoría de roles de Belbin puede ser de utilidad para la integración de equipos que pudiesen evidenciar mayor calidad, tanto en el proceso de desarrollo, como en los productos generados. En particular, se seleccionó la actividad de medición de software en el contexto de un entorno académico. Con el propósito de conformar o refutar los hallazgos del primer experimento, se realizó una réplica similar-interna. En el primer experimento se encontró con que la variable PFSA no presentó diferencias entre tratamientos, sin embargo, la variable Tiempo en minutos para concluir la actividad, presentó diferencias significativas. Con el objetivo de profundizar en los componentes de la variable PFSA, se analizaron posibles diferencias entre sus cinco elementos, presentando diferencias solo uno de éstos. Posteriormente, con la réplica experimental, los resultados no fueron confirmados, en este caso, la variable PFSA presentó diferencias significativas. Por su parte, lo obtenido en la variable Tiempo para concluir la tarea, en la réplica no se identificaron diferencias entre tratamientos. Siendo un estudio exploratorio, y luego de un razonamiento abductivo, podemos comentar que posiblemente la madurez de los participantes en el estudio tuvo alguna influencia en los resultados, si bien, en ambos experimentos se mantuvo un mismo nivel de experiencia en la técnica, la madurez en procesos de software en el primer experimento, es mayor que la de los sujetos de la réplica.

Con lo obtenido en el estudio, los autores proponen continuar con el desarrollo de experimentos controlados con estudiantes; una variante es analizar el factor madurez académica, para identificar si éste tiene alguna influencia en las variables respuesta. En cuando a la incorporación de una nueva variable dependiente, se tiene el interés

de incorporar una métrica vinculada con la colaboración de los integrantes del equipo, durante la realización de una tarea. Por otro lado, considerando los estudios previos, y siendo la actividad de medición de software, una actividad que pudiese realizarse de manera individual, nos resulta interesante investigar si alguno de los roles de Belbin presenta diferencias significativas en cuanto a la calidad de la medición de software.

Referencias

- Aguileta, A. Ucán, J., & Aguilar, R. (2017). Explorando la influencia de los roles de Belbin en la calidad del código generado por estudiantes en un curso de ingeniería de software. *Revista Educación en Ingeniería*, 12(23), 93–100.
- Aritzeta, A., Swailes, S., & Senior, B. (2005). *Team roles: Psychometric evidence, construct validity and team building*. Hull, UK: University Hull.
- Basili, V., Selby, R., & Hutchens, D. (1986). Experimentation in Software Engineering. *IEEE Transactions on Software Engineering*, 12(7), 733–743.
- Belbin, M. (1981). *Management teams: Why they succeed or fail*. New York: John Wiley & Sons.
- Belbin, M. (1993). *Team roles at Work*. Oxford, USA: Elsevier Butterworth Heinemann.
- Bourque, P., & Fairley, R. (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK V3.0)*. IEEE Computer Society.
- Briggs-Myers, I., & McCaulley, M. (1985). *Manual, A Guide to the Development and Use of the Myers-Briggs Type Indicator*. Palo Alto, USA: Consulting Psychologists Press.
- Estrada, E., & Peña, A. (2013). Influencia de los roles de equipo en las actividades del desarrollador de software. *Revista Electrónica de Computación Informática, Biomédica y Electrónica*, 2(1), 1–19.
- Garmus, A. & Herron, D. (1996). *Measuring the Software Process: a practical guide to functional measurements*. New Jersey, USA: Prentice Hall.
- Genero, M., Cruz-Lemus, J., & Piattini, M. (2014). *Métodos de investigación en ingeniería de software*. Madrid, España: Ra-Ma.
- Glass, R., Vessey, I., & Ramesh, V. (2002). Research in Software Engineering: An analysis of the literature. *Information and Software Technology*, 44(8), 491–506.
- Gómez, O., Juristo, N., & Vegas, S. (2014). Understanding replication of experiments in software engineering: A Classification. *Information and Software Technology*, 56(8), 1033–1048. doi: 10.1016/j.infsof.2014.04.004.
- Gutiérrez, H., & De la Vara, R. (2012). *Análisis y Diseño de Experimentos*, (3ª Ed). Ciudad de México: McGraw Hill.
- Henry, S., & Stevens, K. (1999). Using Belbin's leadership role to improve team effectiveness: An empirical investigation. *The Journal of Systems and Software*, 44, 241–250.

- Hernández, L., Muñoz, M., Mejía, J., Peña, A., & Rangel, N. (2017). Una Revisión Sistemática de la Literatura Enfocada en el uso de Gamificación en Equipos de Trabajo en Ingeniería de Software. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (21), 33-50. doi: 10.17013/risti.21.33-50.
- Johansen, T. (2003). *Predicting a Team's Behaviour by Using Belbin's Team Role Self Perception Inventory*. (Dissertation at Department of Management & Organisation), University of Stirling, UK.
- Juristo, N., & Moreno, A. (2001). *Basics of Software Engineering Experimentation*. Boston, USA: Kluwer Academic Publishers.
- Margerison, C.J., & McCann, D.J. (1985). Team Management Profiles: Their use in Managerial Development. *Journal of Management Development*, 4(2), 34-37. doi: 10.1108/eb051580.
- Miramontes, J., Muñoz, M., Calvo-Manzano, J., & Corona, B. (2016). Establecimiento del estado del arte sobre el aligeramiento de procesos de software. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (17), 16-25. doi: 10.17013/risti.17.16-25.
- Morales, N., & Vega, V. (2018). Factores Humanos y la Mejora de Procesos de Software: Propuesta inicial de un catálogo que guíe su gestión. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (29), 30-42. doi: 10.17013/risti.29.
- Mumma, F.S. (2005). *Team-work & team-roles: what makes your team tick? facilitator guide*. King of Prusia, PA: HRDQ Press.
- Norambuena, B., & Vega, V. (2017). Minería de procesos de software: una revisión de experiencias de aplicación. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Información*, (21), 51-66. doi: 10.17013/risti.21.51-66.
- Senior, B. (1997). Team roles & Team performance: Is there 'really' a link?. *Journal of Occupational and Organizational Psychology*, (70), 85-94.

Marco de medición de la privacidad

Luis Gustavo Esquivel-Quirós¹, Elena Gabriela Barrantes², Fernando Esponda Darlington³.

luis.esquivel@ucr.ac.cr, gabriela.barrantes@ecci.ucr.ac.cr, fernando.esponda@itam.mx

^{1,2} Universidad de Costa Rica, Escuela de Ciencias de la Computación e Informática, Sede “Rodrigo Facio Brenes” Montes de Oca, 11501-2060, San José, Costa Rica.

³ Ciencias en Computación, Instituto Tecnológico Autónomo de México, Río Hondo #1 Colonia Progreso Tizapán Álvaro Obregón, 01080, Ciudad de México, México.

DOI: 10.17013/risti.31.66–81

Resumen: El aumento de sanciones por violaciones de la privacidad motiva la definición de una metodología de evaluación de la utilidad de la información y de la preservación de la privacidad de datos a publicar. Al desarrollar un caso de estudio se provee un marco de trabajo para la medición de la preservación de la privacidad. Se exponen problemas en la medición de la utilidad de los datos y se relacionan con la preservación de la privacidad en datos a publicar. Se desarrollan modelos de aprendizaje máquina para determinar el riesgo de predicción de atributos sensibles y como medio de verificación de la utilidad de los datos. Los hallazgos motivan la necesidad de adecuar la medición de la preservación de la privacidad a los requerimientos actuales y a medios de ataque sofisticados como el aprendizaje máquina.

Palabras-clave: aprendizaje máquina; preservación de la privacidad; publicación de datos; medición de la privacidad.

Privacy measurement framework

Abstract: The grown penalties for privacy violations motivate the definition of a methodology for evaluating the usefulness of information and the privacy-preserving data publishing. We developing a case study and we provided a framework for measuring the privacy-preserving. Problems are exposed in the measurement of the usefulness of the data and relate to privacy-preserving data publishing. Machine learning models are developed to determine the risk of predicting sensitive attributes and as a means of verifying the usefulness of the data. The findings motivate the need to adapt the privacy measures to current requirements and sophisticated attacks as the machine learning.

Keywords: machine learning; privacy-preserving; data publishing; privacy measurement.

1. Introducción

El uso exponencial de Internet en el mundo ha modificado la forma de compilar, intercambiar y manipular datos. Las velocidades de procesamiento, los volúmenes de información y las relevancias de los contenidos han cambiado rápidamente hasta niveles cada vez menos imaginables. Esto se ve asociado al interés en crear servicios personalizados basados en la información disponible. Diferentes organizaciones tienen datos sobre las personas, estos datos son un elemento vital y se estima que hasta el 80% de todos los datos almacenados en las organizaciones, pueden clasificarse como grandes datos (big data) (Khan et al., 2014).

Los datos que almacenan las organizaciones por lo general provienen de múltiples fuentes (dispositivos, personas, organizaciones, entre otros) y estas fuentes se convierten en los productores de los datos. Estos datos pueden ser publicados o compartidos con otras organizaciones o individuos, de esta forma los receptores de estas publicaciones se convierten en los consumidores de la información. Una publicación de datos en la mayoría de los casos contiene información de múltiples fuentes y muchas organizaciones utilizan análisis de datos ("*data mining*") para extraer conocimiento relevante (Castillo-Rojas, Medina-Quispe, & Vega-Damke, 2017; Norambuena & Zepeda, 2017). Desde una perspectiva individual esto plantea preguntas acerca de cuánto conocimiento puede ser recabado sobre la vida de una persona, por ejemplo, sobre su situación actual o su paradero (Ohm, 2010). Una vez realizada la publicación de datos, las consecuencias para el dueño o productor de los datos pueden ser muy variadas, exponiendo al individuo a discriminación o escarnio público, entre otros (Martínez et al., 2017).

Las leyes y estándares apenas se mantienen al día con el potencial de invasión a la privacidad. La integración económica y social resultante del funcionamiento de los mercados ha llevado a un aumento sustancial de los flujos transfronterizos de datos y esto converge en el surgimiento de nuevas regulaciones, como por ejemplo, el Reglamento General de Protección de Datos de la Unión Europea (GDPR, por sus siglas en inglés) (El Parlamento Europeo y el Consejo de la Unión Europea, 2016). En consecuencia a estas regulaciones, la Preservación de la Privacidad en Datos a Publicar (PPDP, por sus siglas en inglés) se ha convertido en un área de interés para los investigadores y profesionales.

La PPDP supone que quienes intentan descubrir información confidencial sobre las personas, se pueden encontrar entre los destinatarios de los datos. Por lo tanto, el objetivo de las técnicas de PPDP es modificar los datos haciéndolos menos específicos, de modo que la privacidad de los dueños de los datos esté protegida y a la vez se mantenga la utilidad de los datos tratados. La preservación de la privacidad de los datos requiere del estudio del equilibrio entre el respeto a los deseos o preferencias de privacidad de múltiples dueños de datos, ante una posible inferencia autorizada o no, y la posibilidad de la reidentificación de cada dueño o el enlace de información sensible sobre el conjunto de datos publicado.

Regulaciones como la GDPR definen que las tareas deben estar acotadas, por ejemplo al solicitar el consentimiento informado de todos los posibles escenarios de uso para los datos recopilados. Es importante destacar que esto resulta sumamente difícil, por ejemplo, en iniciativas de datos abiertos, es casi imposible identificar todos los destinatarios y los posibles usos que les den a los datos (Conradie & Choenni, 2012).

Por lo tanto, cualquier publicador de datos necesita aplicar mecanismos de preservación de la privacidad (Ayala-Rivera, McDonagh, Cerqueus, & Murphy, 2014).

La aplicación de formas para medir el potencial abuso y pérdida de la información, mediante la experimentación sobre conjuntos de datos reales, permite brindar garantías científicas sobre la preservación de la privacidad. En este artículo, se estudia el trabajo de la profesora Latanya Sweeney, de la Universidad Carnegie Mellon, y su modelo de preservación de la privacidad conocido como k-anonimato (Sweeney, 2002a). Se aplica de forma práctica una variante multidimensional conocida como algoritmo de Mondrian (LeFevre, DeWitt, & Ramakrishnan, 2006a, 2006b) y se analiza una metodología de evaluación de la utilidad de la información y de la preservación de la privacidad de la información resultante.

La sección 2 describe formalmente la preservación de la privacidad en datos a publicar, así como algunos algoritmos y métricas alrededor de la preservación de la privacidad de los datos. En la sección 3 se describe trabajo relacionado y se enfatizan las contribuciones respecto a iniciativas similares. La evaluación experimental se describe en la sección 4, que incluye la metodología utilizada y el análisis de resultados. Las conclusiones se presentan en la sección 5.

2. Preservación de la privacidad de datos a publicar

La preservación de la privacidad de datos a publicar o PPDP requiere una definición clara de la preservación de la privacidad. En 1977, Dalenius proporcionó una definición muy estricta, donde especifica, que el acceso a los datos publicados, no debe permitir que un atacante aprenda algo adicional sobre cualquier víctima objetivo, en comparación con la información que obtendría si no contara con el acceso a la base de datos publicadora, incluso con la presencia de conocimiento previo obtenido de otras fuentes (Dalenius, 1977).

Con base en esta definición, la principal motivación alrededor de la creación de modelos de preservación de la privacidad de datos a publicar es la de proveer garantías científicas. Donde estas permitan asegurar una utilidad práctica de uso sobre el conjunto de datos publicado y un grado de dificultad al intentar realizar una identificación de los dueños de los datos a partir de la información provista. La forma más básica e intuitiva de proveer garantías científicas de preservación de la privacidad es la desidentificación. La cual consiste en eliminar los datos que permiten identificar o relacionar directamente al dueño original de los datos, como por ejemplo eliminar el número de identificación (cédula en el caso de Costa Rica), número de teléfono o el nombre completo. Al proceso mediante el cual se identifican los datos desidentificados es conocido como reidentificación (Bayardo & Agrawal, 2005).

La desidentificación dentro de la PPDP se refiere al proceso mediante el cual un administrador de datos tiene una tabla T, donde cada columna se puede categorizar como identificador directo, cuasi-identificador, sensible o no sensible. Cada columna solo puede pertenecer a una de estas categorías. Las columnas definidas como identificadores directos permiten identificar explícitamente a los dueños de los datos; las columnas el cuasi-identificadoras contienen valores que pueden identificar a los dueños de los datos,

por medio de la vinculación a información externa que permita reidentificar los dueños originales. Si un atributo es cuasi-identificador su característica más importante es que tan disponibles estén datos externos con una variable que corresponda al potencial valor de la columna cuasi-identificadora. Las columnas sensibles contienen información susceptible a crear algún perjuicio o discriminación específica hacia el dueño de los datos, como por ejemplo la religión, una enfermedad, el salario o el estado de discapacidad, entre otras. Las columnas no sensibles son todas aquellas que no se inscriben en las tres categorías anteriores (Fung, Wang, Fu, & Yu, 2010).

Dado esto, la PPDP toma la tabla T (ID : identificadores, CID : cuasi-identificadores, S : Columnas sensibles, NS : Columnas no sensibles) y la lleva a un estado T' (CID' , S' , NS). Donde CID' y S' es una versión anónima del CID y S originales. Esto al aplicar operaciones de preservación de la privacidad (anonimización) a las columnas en CID y S de la tabla original T . Uno de los modelos más conocidos para preservar la privacidad de datos es el k -anonimato, el cual fue propuesto por primera vez por Samarati y Sweeney en (Samarati & Sweeney, 1998) y extendido por Sweeney en (Sweeney, 2002a). El aporte del trabajo de Sweeney es definir una propiedad de un grupo de datos tal que si es cumplida, dichos datos se pueden publicar con una reducción significativa en la posibilidad de realizar una reidentificación a partir de ellos.

2.1. K-anonimato

El k -anonimato es un marco de desarrollo que trabaja sobre un conjunto de columnas cuasi-identificadoras con un objetivo de privacidad definido por un parámetro k (Pierangela Samarati, 2001). El k -anonimato se puede ver como una propiedad de una tabla T . Se dice que una tabla T es k -anónima si, para cualquier combinación de valores de los campos cuasi-identificadores de la tabla, hay al menos k tuplas con la misma combinación (Sweeney, 2002b).

Las tablas no son k -anónimas de forma natural. Por lo tanto, se debe aplicar un proceso para forzar la propiedad sobre ellas. La forma más común de lograr esto es mediante el uso de la generalización y/o la supresión. Para la generalización, los valores se agrupan en clases de equivalencia de acuerdo con algún principio organizativo que depende de las particularidades de la información.

En la Fig. 1 se puede observar una estructura árbol que representa una distribución de clases de equivalencia disponibles para números de 0 a 99. La distribución de clases de equivalencia para una columna i , dado su papel en el proceso de generalización de los datos, se conoce como jerarquía de generalización. En la jerarquía de generalización, cada nivel proporciona un conjunto de clases de equivalencia. Estas clases permiten agrupar más o menos registros para obtener el valor k solicitado y cada clase de equivalencia no comparte elementos en común con otra clase.

Las operaciones de k -anonimato ocultan información detallada para que los múltiples registros se vuelvan indistinguibles con respecto a los valores en el CID . En consecuencia, si el dueño de los datos está vinculado a un registro a través de un valor en el CID , el dueño de los datos también está vinculado a todos los demás registros que tienen el mismo valor para el CID , lo que hace al enlace ambiguo con los demás dueños de los datos. De esta forma, podemos entender que el problema del k -anonimato es producir

una T anónima (T^*) y que la misma satisfaga un requisito de privacidad determinado por el nivel del valor del k en el modelo de privacidad, de forma que retenga la mayor utilidad de datos posible.

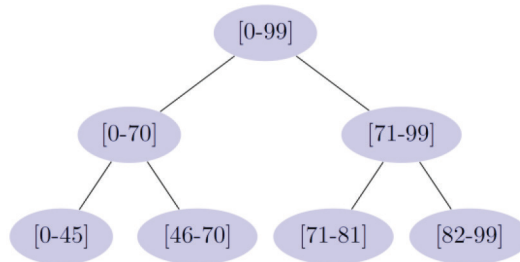


Figura 1 – Jerarquía de generalización con las clases de equivalencia para números de 0 a 99

En este trabajo utilizaremos una versión multidimensional del k -anonimato conocida como Mondrian Básico, propuesta por Kristen LeFevre en (LeFevre et al., 2006a) y modificada en (LeFevre et al., 2006b).

2.2. K-anonimato multidimensional – Mondrian Básico

Mondrian es un algoritmo de preservación de la privacidad de datos voraz descendente que permite la generalización de atributos CID de forma multidimensional. La versión inicial del algoritmo está diseñada para trabajar solo con atributos numéricos (LeFevre et al., 2006a). En un trabajo posterior LeFevre propuso una modificación a su algoritmo para que aceptara atributos categóricos, además de numéricos (LeFevre et al., 2006b).

Para satisfacer el requerimiento del k -anonimato sobre la tabla T , se debe validar que para cada valor distinto en una columna CID , este valor se encuentra k veces. En muchos casos esto no se cumple en la tabla original y mucho menos si se cuenta con varias columnas CID . Por este motivo la visión multidimensional de los datos ayuda a satisfacer el requerimiento de forma más rápida. Permitiendo que si tenemos como CID a varias columnas, y si por cada tupla única al seleccionar estas columnas sobre T se producen al menos k tuplas iguales, se dice que cumple el principio de k -anonimato en el grado del k reportado. Es decir, el tamaño de cada clase de equivalencia en T con respecto a las columnas es al menos de tamaño k o en otras palabras el tamaño de la clase de equivalencia se determina contando el número de tuplas que son agrupadas mediante ella (Fung et al., 2010).

La utilidad de datos de Mondrian se puede ver afectada por la distribución de los datos y el mecanismo utilizado para la creación de particiones (Ayala-Rivera et al., 2014). Por este motivo es importante contar con alguna métrica que verifique la utilidad de los datos y su preservación de la privacidad. En general se crea una pérdida de información cada vez que un valor en una columna CID se generaliza a una clase de equivalencia mayor en la jerarquía de generalización, esto debido a que cada clase de equivalencia es más genérica que la anterior. El objetivo de un buen algoritmo de PPDP es encontrar

una transformación de los datos originales, de forma que satisfaga un requerimiento de privacidad al tiempo que minimice la pérdida de información y maximice la utilidad de los datos resultantes. Por lo tanto, una medida es necesaria para indicar la calidad y la preservación de la privacidad de los datos.

2.3. Métricas de privacidad

En la literatura se presentan varias métricas que evalúan la calidad de los datos utilizando medidas simples basadas en el tamaño de la clase de equivalencia o el número total de generalizaciones (Aggarwal et al., 2005; Bayardo & Agrawal, 2005; LeFevre et al., 2006a; Pierangela Samarati, 2001; Sweeney, 2002a). Dado el efecto del uso de jerarquías de generalización, en este trabajo es de interés la métrica de *Penalización de Certeza Normalizada* (NCP, por sus siglas en inglés) (Xu et al., 2006). Esta métrica intenta capturar la incertidumbre causada por la generalización de cada clase de equivalencia en el espacio de columnas *CID*. Para un valor continuo u ordinal en una columna *CID*, el NCP en una clase de equivalencia *G* se define como la división de los rangos de valores de la columna *CID* en los que se agrupa el valor del campo en la clase de equivalencia *G*, entre el dominio completo de la columna *CID*. En el caso de los valores de columnas categóricas, donde no exista un orden total o alguna función de distancia, el NCP se define con respecto a la jerarquía de generalización de la columna *CID*. Podemos expresar el NCP como cero cuando el número de hojas (es decir, el número de valores agrupados de la columna) en el subárbol que contiene el valor del campo es igual a uno y en caso contrario el NCP es el resultado de la división del número de hojas entre el número total de valores distintos en la columna categórica.

El valor NCP de la clase de equivalencia sobre todas las columnas *CID* es el resultado de la sumatoria de *i* igual a 1 hasta el número de columnas continuas, ordinales o categóricas del NCP de la clase de equivalencia. NCP mide la pérdida de información para una única clase de equivalencia y caracteriza la pérdida de información de una partición completa al sumar el NCP de todas las tuplas en cada clase de equivalencia. Para este trabajo adoptamos la formulación normalizada de la versión agregada de NCP, llamada Penalización de Certeza Global (Global Certainty Penalty - GCP) (Ghinita, Karras, Kalnis, & Mamoulis, 2009). Esta mide la pérdida de información de toda la tabla anonimizada, tomando *P* como el conjunto de todas las clases de equivalencia en la tabla anonimizada y se define como:

$$GCP(P) = \frac{\sum_{G \in P} |G| * NCP(G)}{d * N}$$

Figura 2 – Ecuación GCP

En la Fig. 2 tenemos que *d* representa todas las columnas *CID* y *N* el número de registros en la tabla original *T*, *|G|* es la cardinalidad de la clase de equivalencia *G*. La ventaja de esta ecuación es la capacidad de medir la pérdida de información entre tablas con cardinalidad y dimensionalidad variable. El rango de valores de GCP está entre 0 y 1, donde 0 significa que no existe pérdida de información y 1 corresponde a la pérdida

total de información. Por facilidad de uso, en este trabajo el GCP se calcula dividiendo el valor de GCP original con el número de valores en el conjunto de datos para pasarlo a porcentaje.

La importancia de una métrica de privacidad está vinculada de manera innata a los cálculos que se pueden realizar sobre los datos y el objetivo de la PPDP es producir conjuntos de datos que tengan una “buena” utilidad para una gran variedad de trabajos.

El que esa variedad de trabajos sea desconocida es una premisa esencial para muchos investigadores, pero regulaciones como la GDPR (El Parlamento Europeo y el Consejo de la Unión Europea, 2016) se traen abajo esta premisa al solicitar informarle al usuario los procesos que se realizarán con sus datos. Al conocer de antemano los trabajos a realizar, el publicador de los datos simplemente puede ejecutar los trabajos en los datos originales y publicar solo los resultados o publicar una versión anonimizada que no permita realizar los trabajos que el dueño de los datos no autorizó. Otros investigadores han utilizado el aprendizaje máquina para medir la utilidad de los datos (Chen, LeFevre, & Ramakrishnan, 2008; LeFevre et al., 2006b). En este documento se propone utilizar el aprendizaje máquina no solo para medir la utilidad, sino también la preservación de la privacidad.

2.4. Aprendizaje máquina

El aprendizaje máquina permite en muchos casos extraer información útil, interesante y previamente desconocida de grandes conjuntos de datos. El éxito siempre se basa en la disponibilidad de datos de alta calidad y el intercambio efectivo de estos. Se ha logrado un impulso en la publicación de datos tanto por beneficio mutuo, como por regulaciones que obligan a la publicación de ciertos datos (Conradie & Choenni, 2012). La publicación de datos es omnipresente en muchos dominios. Por ejemplo, en el 2006, el proveedor de Internet AOL lanzó un conjunto de datos que contenían 3 meses de búsquedas de 650 000 usuarios. Los nombres fueron enmascarados con identificadores aleatorios y aun así, en cuestión de días, un reportero del New York Times identificó a Thelma Arnold, una viuda de 62 años como el usuario 4417749 (Barbaro & Zeller, 2006). Situaciones como esta han impulsado a los investigadores en PPDP a utilizar algoritmos de aprendizaje de clasificación y regresión en sus investigaciones (Chen et al., 2008; LeFevre et al., 2006b; Machanavajjhala et al., 2007).

En estos modelos de aprendizaje máquina, los atributos normalmente se caracterizan en al menos uno de los siguientes tipos (Han, Kamber, & Pei, 2012):

- *Atributo objetivo o de interés*: es el atributo nominal cuyo valor busca predecir con precisión el modelo de clasificación construido. En el caso de la regresión, es el atributo numérico cuyo valor tiene como propósito predecir el modelo de regresión construido.
- *Atributos de predicción*: son el conjunto de atributos (discretos o continuos, según el algoritmo) que se utilizan como entradas para construir el modelo que intenta predecir el atributo objetivo.

La utilidad requiere potenciar el uso de la información disponible y por ello se debe tomar en cuenta que los atributos no sensibles se publican si son importantes para la tarea de

minería de datos. El atributo objetivo en la minería es definido como no sensible en el proceso de preservación de la privacidad para que conserve su semántica. Por lo general cuando se considera un algoritmo de clasificación o de regresión junto con la PPDP, cada atributo tiene solo dos caracterizaciones (atributo de predicción y atributo objetivo). En el resto de este documento, se asumirá que el conjunto de atributos de predicción es un conjunto de cuasi-identificadores (*CID*). Bajo esta suposición, puede parecer contradictorio mantener la categoría de atributo sensible, pero para efectos de este trabajo se utiliza como atributo objetivo cuándo se evalúa la preservación de la privacidad.

El transformar un atributo sensible en un atributo objetivo permite medir la precisión del modelo de aprendizaje máquina creado para predecirlo y por lo tanto medir la preservación de la privacidad en el conjunto de datos en estudio. De esta manera en unos casos se toma como atributo objetivo el atributo sensible y en otros como atributo objetivo un atributo de interés hipotético. La evaluación del uso de modelos de aprendizaje máquina como métrica de preservación de la privacidad se llevara a cabo en la siguiente sección.

3. Trabajo relacionado

La selección de un algoritmo apropiado para proteger la privacidad cuando se difunden datos es una preocupación general para la PPDP. Como resultado, la comparación de múltiples algoritmos de preservación de la privacidad desde la perspectiva de la utilidad de los datos y la efectividad en la preservación de la privacidad, representa un importante trabajo de investigación. Algunos autores discuten que la eficacia de la preservación de la privacidad se evalúa mejor por la utilidad que proporciona a las aplicaciones de destino y que las métricas pueden comportarse de manera diferente con diferentes algoritmos de preservación de la privacidad (Nergiz & Clifton, 2007). Además de esto, se presenta un marco de evaluación en (Bertino, Fovino, & Provenza, 2005) para estimar y comparar diferentes tipos de algoritmos de preservación de la privacidad específicamente en la minería de datos. Aun cuando estos estudios están estrechamente relacionados con la PPDP, las tareas de minería de datos que se consideran están estrechamente relacionadas con las soluciones propuestas.

La PPDP publica los datos modificados mediante algoritmos de preservación de la privacidad a múltiples destinatarios que pueden usar los datos de muchas maneras diferentes. Por lo tanto, no sería adecuado evaluar los métodos de preservación de la privacidad utilizando estudios comparativos que solo tengan en cuenta métricas con fines específicos (es decir, dependientes de la aplicación). Esto se debe a que las métricas que tienen en cuenta un escenario de uso particular solo pueden capturar la utilidad de los datos protegidos según los requisitos para ese escenario. En cambio, en (Ayala-Rivera et al., 2014) se indica que un conjunto de métricas que pueden aplicarse a la mayoría de los escenarios de publicación proporciona un mejor enfoque para realizar una comparación sistemática. Aunque este mismo trabajo hace énfasis en las variaciones del rendimiento de los algoritmos, su aporte es importante, pero escaso de discusión sobre la incompetencia de algunas métricas en los distintos contextos.

El objetivo de las métricas de preservación de la privacidad es medir el grado y la cantidad de protección que ofrecen las tecnologías de preservación de la privacidad.

De esta forma, las métricas de preservación de la privacidad contribuyen a mejorar la privacidad de los dueños de los datos. La diversidad y la complejidad de las métricas de preservación de la privacidad en la literatura hacen que una elección informada de métricas sea desafiante. Como resultado, en lugar de utilizar métricas existentes, se proponen nuevas métricas con frecuencia, y los estudios de privacidad a menudo son incomparables (Wagner & Eckhoff, 2018).

En (Fung et al., 2010) se afirma que para abordar el objetivo de clasificación, la distorsión debe medirse por error de clasificación en casos futuros y sugieren que el conocimiento de clasificación útil es capturado por diferentes combinaciones de atributos. La generalización y la supresión pueden destruir algunas de estas “estructuras de clasificación” útiles, pero pueden surgir otras estructuras útiles para ayudar. En algunos casos, la generalización y la supresión pueden incluso mejorar la precisión de la clasificación, al eliminar ruido en los datos. También se afirma que es esencial evaluar experimentalmente el impacto de la preservación de la privacidad mediante la construcción de un clasificador a partir de los datos protegidos y ver cómo funciona en los casos de prueba. Algunos trabajos (Iyengar, 2002; LeFevre et al., 2006b) han llevado a cabo o analizan tales experimentos, aunque en la práctica general se utiliza alguna métrica específica (Ghinita et al., 2009; Gong, Luo, Yang, Ni, & Li, 2017).

Este enfoque no es abordado al analizar métricas y modelos de privacidad, por ejemplo en (Wagner & Eckhoff, 2018) analizan una selección de más de ochenta métricas de privacidad e introducen caracterizaciones basadas en los aspectos de la privacidad que miden, sus requerimientos de entrada y el tipo de datos que protegen. Además, presentan un método sobre cómo elegir las métricas de privacidad basadas en nueve preguntas que ayudan a identificar las métricas de privacidad adecuadas según un escenario dado. Pero rescatan que se necesita trabajo adicional sobre métricas de privacidad. Esto fundamenta la necesidad de una metodología de evaluación de la preservación de la privacidad como la presentada en este artículo.

4. Evaluación experimental

La evaluación experimental tiene como objetivo proporcionar una metodología de evaluación de la utilidad y la preservación de la privacidad. Se describen los pasos de un protocolo experimental para evaluar un algoritmo de preservación de la privacidad con respecto a un conjunto de datos y se desarrolla un caso de estudio que permite comparar los resultados de varios modelos de aprendizaje máquina creados a partir de publicaciones construidas con diferentes niveles de privacidad y distintas configuraciones de cuasi-identificadores.

4.1. Metodología

Una forma de evaluar la calidad de la preservación de la privacidad es creando un modelo de predicción utilizando los datos anonimizados y luego evaluar el modelo según la precisión en la predicción tanto del atributo sensible, como del atributo de interés para el estudio. Para lograr este propósito se presenta la siguiente metodología de evaluación de la utilidad y la preservación de la privacidad:

1. Preprocesado de los datos.
2. Categorización de los datos en atributos identificadores, cuasi-identificadores, sensibles y no sensibles.
3. Definición de niveles de preservación de la privacidad e identificación de los cuasi-identificadores a utilizar.
4. Anonimización de los datos por medio del algoritmo de preservación de la privacidad (Mondrian Básico).
5. Construcción de conjunto de datos que comprende los datos anonimizados, los datos no sensibles, el atributo de interés y el atributo sensible.
6. Preprocesado y recodificación del conjunto de datos generado para poder ser utilizado en el algoritmo de aprendizaje máquina.
7. Categorización de los atributos como atributos de predicción y atributos objetivo.
8. Segmentación del conjunto en datos de entrenamiento y de prueba.
9. Generación del modelo de predicción mediante el algoritmo de aprendizaje máquina seleccionado.
10. Evaluación de la precisión de la predicción del modelo generado.

Esta metodología propuesta para la evaluación de la utilidad de los datos y la preservación de la privacidad de los datos se utiliza en un caso de estudio práctico para demostrar su usabilidad. El caso de estudio desarrollado mantiene el atributo de interés y el atributo sensible en los datos a publicar durante las fases de preservación de la privacidad y durante la creación del modelo de aprendizaje máquina.

Se utiliza el algoritmo Mondrian Básico implementado en el lenguaje Python como algoritmo de preservación de la privacidad, así como implementaciones de los algoritmos de aprendizaje máquina proporcionados por el paquete de software “scikit-learn” (Pedregosa et al., 2011). Específicamente árboles de decisión, bosques aleatorios y regresión logística. El conjunto de datos de prueba es el AdultDatabase (Dheeru & Karra Taniskidou, 2017). Este conjunto contiene 32 561 filas con 15 atributos, una vez finalizado el preprocesado quedaron 30 162 filas. Se definió como atributo sensible (no cuasi-identificador): “race”, como atributo de interés (no cuasi-identificador): “income” y se define como atributo no sensible: “education” (para asegurar un grado de utilidad constante, esto aunque es equivalente a el atributo “education-num”). Los demás atributos se utilizan como posibles cuasi-identificadores.

En los experimentos, se varió el tamaño del conjunto de atributos cuasi-identificadores en un rango de 1, 6 y 12. Cuando se modifica el número de atributos cuasi-identificadores, el atributo que deja de ser cuasi-identificador automáticamente pasa a ser no sensible para mantener un mismo número de entradas en los algoritmos de aprendizaje máquina. Las jerarquías de generalización para los atributos categóricos están disponibles en desarrollos de código abierto de GitHub. También, se definió como nivel de privacidad k el rango de valores: 2, 5, 10, 25, 50 y 100.

4.2. Análisis de resultados

El caso de estudio sobre la metodología propuesta incluye evaluar la utilidad de los datos y la preservación de la privacidad de los datos a publicar, para esto se crearon modelos de predicción utilizando los algoritmos de aprendizaje máquina seleccionados.

Se distinguen dos tipos de modelos creados, los que realizan predicción de la variable de interés para comprobar la utilidad de los datos y los que realizan predicción de la variable sensible para comprobar la preservación de la privacidad. Como se observa en la Fig. 3, los árboles de decisión tienen una pérdida en su precisión en la utilidad de los datos (predicción de variable de interés) al aumentar el nivel de privacidad k con doce cuasi-identificadores. Este comportamiento resulta normal si se considera que los datos son generalizados a partir de alguna de las clases de equivalencia. Lo que no resulta consecuente es que la pérdida de precisión no continua aumentando. Podemos observar que en general la precisión de la predicción del árbol de decisión no sufre variaciones conforme se cambia el nivel de privacidad.

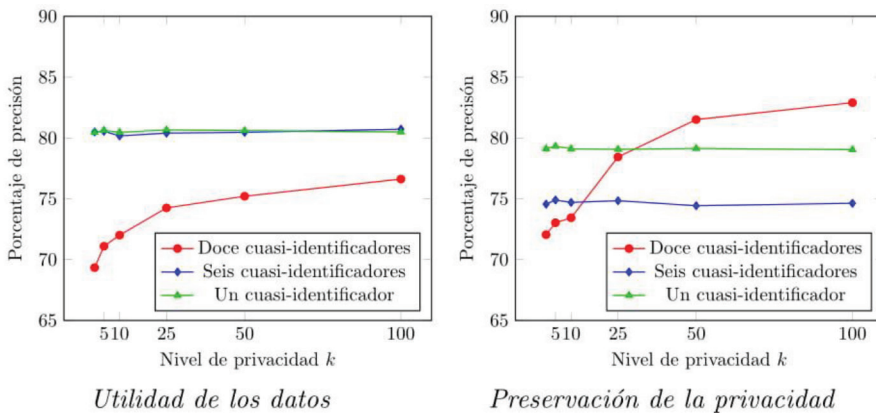


Figura 3 – Evaluación de modelos creados con árboles de decisión

También se puede observar en la Fig. 3 que los modelos que miden la preservación de la privacidad por medio de árboles de decisión mantienen una precisión superior al 70%. Si bien la medición de la precisión demuestra cambios respecto a los modelos que miden la utilidad, se puede observar que ambos obtienen comportamientos similares y con resultados superiores al 70%.

El comportamiento de la precisión en la predicción de los modelos creados con el algoritmo de bosques aleatorios se presenta en la Fig. 4. Se observa que la variación de la precisión para la medición de la utilidad de los datos es similar a la de los modelos creados con el algoritmo de árboles de decisión.

Los modelos creados con el algoritmo de regresión logística comparten similitud con los modelos evaluados anteriormente. Se observa en la Fig. 5 que todos los modelos creados tienen una precisión superior al 75%. Este resultado es consecuente con los arrojados por los modelos evaluados anteriormente y permiten teorizar que se mantiene una utilidad buena de los datos. Al mismo tiempo se puede teorizar que la preservación de la privacidad es muy baja. Esto al destacar que ni el número de cuasi-identificadores utilizado, ni el nivel de preservación de la privacidad parece influir en el resultado de la predicción para el dato sensible.

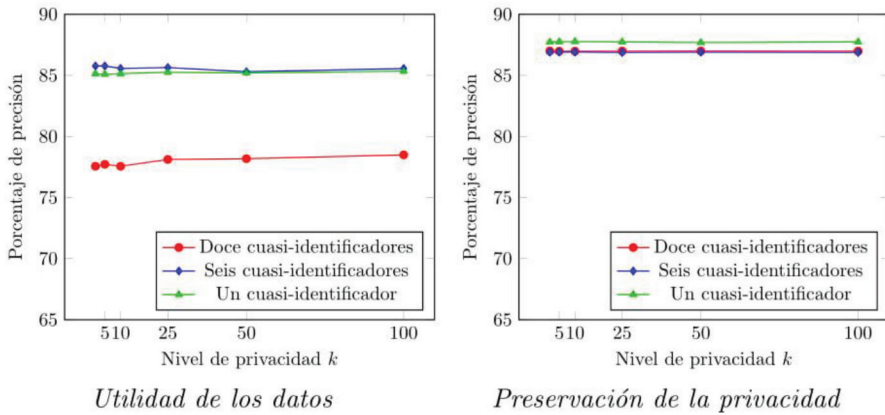


Figura 4 – Evaluación de modelos creados con bosques aleatorios

Destaca que la preservación de la privacidad no cambia significativamente al variar el nivel de privacidad k o el número de cuasi-identificadores. Al tomar como métrica de privacidad los resultados obtenidos por los algoritmos de aprendizaje máquina, podemos indicar que la precisión en la predicción sobre el atributo sensible es muy alta. Dados los resultados obtenidos se puede indicar que existe una preservación de la privacidad muy débil.

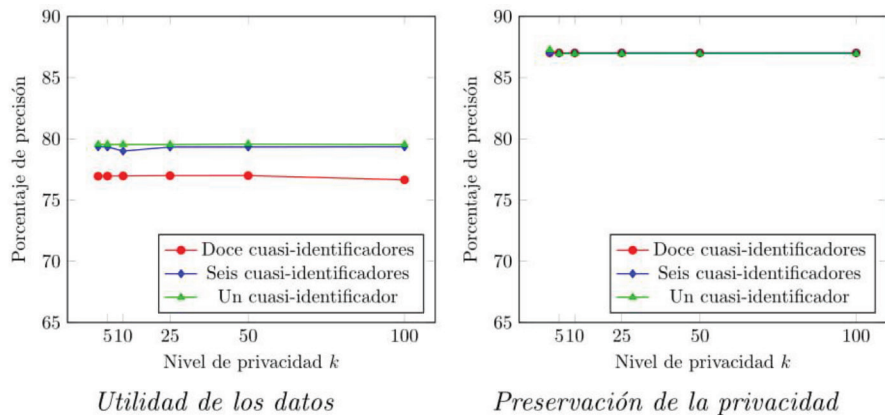


Figura 5 – Evaluación de modelos creados con regresión logística

En particular, como se explicó en la sección 2, es importante contar con una métrica de preservación de la privacidad. Para contrastar los resultados obtenidos por los modelos de aprendizaje máquina se evaluó la métrica GCP. Los resultados de la métrica GCP en la Fig. 6 indican que la pérdida de información puede llegar a 40% y este resultado no es consecuente con los resultados de los modelos creados con los algoritmos de aprendizaje máquina.

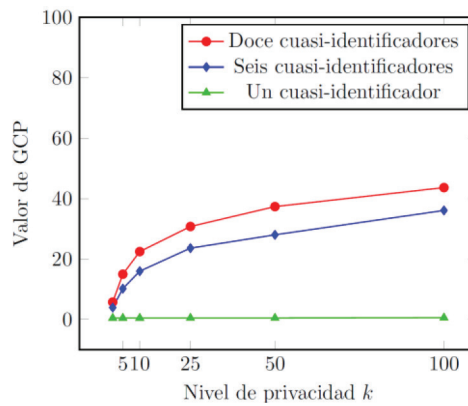


Figura 6 – Relación entre el nivel de privacidad k , el número de cuasi-identificadores y el valor obtenido por la métrica GCP

La métrica GCP reporta un incremento en la pérdida de información al modificar el nivel de privacidad k , situación que no se ve reflejada en la precisión de la predicción de los algoritmos de aprendizaje máquina. Es probable que la reducción de los dominios de los atributos influyera en la conservación de la precisión en las predicciones. Otro aspecto es que la métrica GCP no toma en cuenta la diversidad de valores en el atributo sensible y eso disminuye la calidad de la medición.

Dados estos resultados se puede afirmar que los modelos creados mediante los algoritmos de aprendizaje máquina mantienen el mismo porcentaje de éxito en la predicción aun cuando los datos han sido procesados por un modelo de preservación de la privacidad como Mondrian Básico.

5. Conclusiones

La utilidad de los datos es un aspecto fundamental para motivar el uso de técnicas de preservación de la privacidad, pero al mismo tiempo es posible que sean la llave para respetar los deseos o derechos de los dueños de los datos. Los algoritmos de preservación de la privacidad en datos a publicar son ampliamente estudiados en la búsqueda de proteger la privacidad pero resulta complicado indicar que datos corren más riesgo de reidentificación. Este artículo proporciona indicios sobre la existencia de dificultades para establecer buenos parámetros en algoritmos de privacidad como Mondrian Básico. El estudio experimental genera evidencia sobre la poca efectividad de la métrica GCP respecto a la utilidad de los datos anonimizados en modelos de aprendizaje máquina que efectúen predicciones. Además, la poca efectividad del GCP está presente tanto en la predicción del atributo de interés (midiendo la utilidad), como sobre el atributo sensible (midiendo la preservación de la privacidad). Los algoritmos de aprendizaje máquina se plantean como un instrumento para asegurar la privacidad y se espera que la metodología sea base para un aseguramiento de la privacidad más riguroso.

Agradecimientos

Un agradecimiento especial a la Profesora Ileana Castillo Arias de la UCR por todas sus recomendaciones y revisiones. Este trabajo fue apoyado por el Programa de Posgrado en Computación e Informática (PCI), la Escuela de Ciencias de la Computación e Informática (ECCI), el Centro de Investigaciones en Tecnología de la Información y la Comunicación (CITIC)), y el Sistema de Estudios de Posgrado (SEP) todos en la Universidad de Costa Rica (UCR). Así como también por el Ministerio de Ciencia, Tecnología y Telecomunicaciones (MICITT), y por el Consejo Nacional para Investigaciones Científicas y Tecnológicas (CONICIT) del Gobierno de Costa Rica.

Referencias

- Aggarwal, G., Feder, T., Kenthapadi, K., Motwani, R., Panigrahy, R., Thomas, D., & Zhu, A. (2005). Anonymizing tables. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3363, 246–258. Doi: 10.1007/978-3-540-30570-5_17.
- Ayala-Rivera, V., McDonagh, P., Cerqueus, T., & Murphy, L. (2014). A Systematic Comparison and Evaluation of k -Anonymization Algorithms for Practitioners. *Transactions on Data Privacy*, 7(3), 337–370.
- Barbaro, M., & Zeller, T. (2006). A Face Is Exposed for AOL Searcher No. 4417749. *New York Times*, (4417749), 1–3. Doi: 4417749.
- Bayardo, R. J., & Agrawal, R. (2005). Data privacy through optimal k-anonymization. In: *Proceedings - International Conference on Data Engineering* (pp. 217–228). Doi: 10.1109/ICDE.2005.42.
- Bertino, E., Fovino, I. N., & Provenza, L. P. (2005). A Framework for Evaluating Privacy Preserving Data Mining Algorithms. *Data Mining and Knowledge Discovery*, 11(2), 121–154. Doi: 10.1007/s10618-005-0006-6.
- Brickell, J., & Shmatikov, V. (2008). The cost of privacy: destruction of data-mining utility in anonymized data publishing. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 70–78. Doi: 10.1145/1401890.1401904.
- Castillo-Rojas, W., Medina-Quispe, F., & Vega-Damke, J. (2017). Esquema de visualización para modelos de clústeres en minería de datos. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, (21), 67–84. Doi: 10.17013/risti.21.67-84.
- Chen, B.-C., LeFevre, K., & Ramakrishnan, R. (2008). Adversarial-knowledge dimensions in data privacy. *The VLDB Journal*, 18(2), 429–467. Doi: 10.1007/s00778-008-0118-x.
- Conradie, P., & Choenni, S. (2012). Exploring process barriers to release public sector information in local government. In: *Proceedings of the 6th International Conference on Theory and Practice of Electronic Governance - ICEGOV '12* (p. 5). New York, USA: ACM Press. Doi: 10.1145/2463728.2463731.

- Dalenius, T. (1977). Towards a methodology for statistical disclosure control. *Statistik Tidskrift*, 15, 429–444. Doi: 10.1145/320613.320616.
- Dheeru, D., & Karra Taniskidou, E. (2017). UCI Machine Learning Repository. Retrieved from: <http://archive.ics.uci.edu/ml>.
- El Parlamento Europeo y el Consejo de la Unión Europea. (2016). Reglamento (UE) 2016/679 del parlamento europeo y del consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la D. Diario Oficial de La Unión Europea, 2014(119), 1–88.
- Fung, B., Wang, K., Fu, A., & Yu, P. (2010). Introduction to Privacy-Preserving Data Publishing (Vol. 17). CRC Press. Doi: 10.1201/9781420091502.
- Ghinita, G., Karras, P., Kalnis, P., & Mamoulis, N. (2009). A framework for efficient data anonymization under privacy and accuracy constraints. *ACM Transactions on Database Systems*, 34(2), 1–47. Doi: 10.1145/1538909.1538911
- Gong, Q., Luo, J., Yang, M., Ni, W., & Li, X. B. (2017). Anonymizing 1:M microdata with high utility. *Knowledge-Based Systems*, 115, 15–26. Doi: 10.1016/j.knosys.2016.10.012.
- Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques*. San Francisco, CA: Morgan Kaufmann Publishers. Doi: 10.1016/B978-0-12-381479-1.00001-0.
- Iyengar, V. S. (2002). Transforming data to satisfy privacy constraints. In: *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD'02* (p. 279). New York, USA: ACM Press. Doi: 10.1145/775047.775089.
- Khan, N., Yaqoob, I., Hashem, I. A. T., Inayat, Z., Mahmoud Ali, W. K., Alam, M., ... Gani, A. (2014). Big Data: Survey, Technologies, Opportunities, and Challenges. *The Scientific World Journal*, 2014, 1–18. Doi: 10.1155/2014/712826.
- LeFevre, K., DeWitt, D. J., & Ramakrishnan, R. (2006a). Mondrian multidimensional K-anonymity. In: *Proceedings - International Conference on Data Engineering* (Vol. 2006, p. 25). IEEE.
- LeFevre, K., DeWitt, D. J., & Ramakrishnan, R. (2006b). Workload-aware anonymization. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06* (p. 277). New York, USA: ACM Press. Doi: 10.1145/1150402.1150435.
- Machanavajjhala, A., Kifer, D., Gehrke, J., & Venkitasubramaniam, M. (2007). L-diversity. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 3–es. Doi: 10.1145/1217299.1217302.
- Martinez, F. R. C., Candelaria, A. D. H., Lozano, M. A. R., Zúñiga, A. R. R., Peláez, R. M., & Michel, J. R. P. (2017). Después de presionar el botón enviar, se pierde el control sobre la información personal y la privacidad: Un caso de estudio en México. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, (21), 115–128. Doi: 10.17013/risti.21.115-128.

- Nergiz, M. E., & Clifton, C. (2007). Thoughts on k-anonymization. *Data & Knowledge Engineering*, 63(3), 622–645. Doi: 10.1016/J.DATAK.2007.03.009
- Norambuena, B. K., & Zepeda, V. V. (2017). Minería de procesos de software: Una revisión de experiencias de aplicación. *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, 21(21), 51–66. Doi: 10.17013/risti.21.51-66
- Ohm, P. (2010). Broken Promises of Privacy: Responding to the Surprising Failure of Anonymization. *UCLA Law Review*, 57, 1701. Retrieved from: <http://www.uclalawreview.org/?p=1353>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12, 2825–2830. Retrieved from: <http://dl.acm.org/citation.cfm?id=1953048.2078195>.
- Samarati, P., (2001). Protecting respondents' identities in micro- data release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6), 1010–1027.
- Samarati, P., & Sweeney, L. (1998). Protecting Privacy when Disclosing Information: k-Anonymity and its Enforcement Through Generalization and Suppresion. In: *Proceedings of the IEEE Symposium on Research in Security and Privacy*, (pp. 384–393). Doi: 10.1145/1150402.1150499.
- Sweeney, L. (2002a). Achieving k-anonymity Privacy Protection using Generalization and Suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05), 571–588. Doi: 10.1142/S021848850200165X.
- Sweeney, L. (2002b). k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5), 557--570. Doi: 10.1142/S0218488502001648.
- Wagner, I., & Eckhoff, D. (2018). Technical Privacy Metrics: a Systematic Survey. *ACM Computing Surveys*, 51(3), 1–38. Doi: 10.1145/3168389.
- Xu, J., Wang, W., Pei, J., Wang, X., Shi, B., & Fu, A. W.-C. (2006). Utility-based anonymization using local recoding. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06* (Vol. 18, p. 785). *JMLR.org*. Doi: 10.1145/1150402.1150504.

Detectando aplicaciones maliciosas en Smartphone con sistema Android a través del uso de una aplicación

Jezreel Mejía

jmejia@cimat.mx

Centro de Investigación en Matemáticas, Unidad Zacatecas, Parque Quantum, Ciudad el Conocimiento
Avenida Lassec, Andador Galileo Galilei, Manzana, 3 Lote 7, 98160. Zacatecas, México.

DOI: 10.17013/risti.31.82-93

Resumen: En los últimos años, la consolidación de Android, como el sistema operativo más utilizado en Smartphone lo convierte en blanco del mayor número de amenazas creciente; tendencia que se ha mantenido en los últimos años. De acuerdo al informe de 2016, Tendencias en Seguridad publicado por ESET, el número de vulnerabilidades móviles incrementa cada año desde 2013 y, a lo largo de 2015 la tasa de incremento dentro del ecosistema Android que ponen en riesgo la privacidad de su información, ha promediado de 200 nuevas muestras de malware por mes. Por lo tanto, en este artículo se presenta una aplicación (app) para la detección oportuna a potenciales aplicaciones maliciosas presentes en Smartphone, particularmente aquellos con sistema operativo Android, presentando un estudio de caso que ha permitido validar la app a través de la detección de aplicaciones maliciosas instaladas en un dispositivo móvil.

Palabras-clave: Smartphone, Android, Ciberseguridad, Hooking.

Detecting malicious applications on Smartphones with Android operating system through of an app

Abstract: In recent years, the Android operating system is the most used in Smartphones, which makes it the target of the greatest number of threats; trend that has remained in recent years. According to the 2016 report, Trends in Security published by ESET, the number of mobile vulnerabilities increases every year since 2013 and, throughout 2015, the rate of increase within the Android ecosystem that put the privacy of your information at risk, has averaged 200 new malware samples per month. Therefore, this article presents an application (app) for the timely detection of potential malicious applications present in smartphones, particularly those with Android operating system, presenting a case study that has allowed validating the app through the detection of malicious applications installed on a mobile device.

Keywords: Smartphone, Android, Cybersecurity, Hooking.

1. Introducción

El desarrollo, crecimiento y evolución de las Tecnologías de Información y Comunicación (TICs) ha creado nuevas formas para compartir, transferir y distribuir información por medios digitales (Cisco, 2017), (Sherman, 2019), (Dai, Miao, 2005); (Jin, et al., 2005), en este contexto, el panorama de la seguridad se ha vuelto más complejo al aumentar el número de amenazas y tipos de amenazas. En la última década, el uso de las tecnologías de la información en el ámbito personal, académico y empresarial ha aumentado de 30% a 70% (INEGI, 2014), lo que ha propiciado nuevas oportunidades de desarrollo tecnológico.

De acuerdo, al informe de 2016, ICT Realidad y cifras (Sanou, 2016) muestra que los Smartphone se han convertido en el medio masivo de comunicación más utilizado debido a su portabilidad y relativo bajo costo. Asimismo, al informe de 2016, Visión de la Ciberseguridad publicado por Norton (Symantec Corp., 2016), existe registro de alrededor de 594 millones de afectaciones por vulnerabilidades y amenazas informáticas que ponen en riesgo la privacidad de la información de estos dispositivos, debido a que su exposición en Internet es cada vez más frecuente.

El reporte de Gartner (Woods, 2016), muestra que Android se ha consolidado definitivamente como el sistema más utilizado en el segmento de usuarios de Smartphones con un 86,2% de las ventas en el segundo cuartil (Q2) del año 2016, seguido por iOS de Apple Inc., como la segunda plataforma preferida con el 12,9%, estadística que se ha mantenido desde Q1 de 2013. Además, de acuerdo al informe de 2016, Tendencias en Seguridad publicado por (ESET, 2016), el número de vulnerabilidades móviles ha aumentado cada año desde 2013 y, a lo largo de 2015 la tasa de creación de amenazas dentro del ecosistema Android creció con un promedio de 200 nuevas muestras de malware por mes, debido a la exposición de estos dispositivos en Internet.

De acuerdo a los informes mencionados, implican que la seguridad de los Smartphone es un asunto cada vez más importante. Sin embargo, la complejidad que conlleva analizar estos volúmenes de información generados, requiere de técnicas y herramientas sofisticadas para mejorar su seguridad (Ponemon, 2014).

Por lo tanto, para poder analizar la información generada por los Smartphones, es necesario contar con herramientas de análisis automático de datos (Moço, Lobato, 2014), (Maloof, 2006), (Spreitzenbarth, 2015), para examinar aplicaciones que representen brechas de seguridad; particularmente las de dispositivos con sistema operativo Android (ESET, 2014), (Martínez, 2015).

El objetivo de este artículo es presentar una aplicación (app) para la detección oportuna a potenciales aplicaciones maliciosas presentes en Smartphones, particularmente aquellos con sistema operativo Android. Para el desarrollo, de esta herramienta se ha utilizado técnicas de hooking y se ha realizado un estudio de caso para su validación.

El artículo está estructurado de la siguiente manera: la sección 2 presenta los conceptos generales relacionada a la seguridad; la sección 3 presenta la Herramienta para identificar aplicaciones potencialmente maliciosas; la sección 4 presenta un estudio de caso que se ha utilizado para validar la herramienta; finalmente, en la sección 5 se establecen las conclusiones.

2. Conceptos generales

En esa sección, se describirán los conceptos principales: Ciberseguridad, Seguridad, Seguridad de la Información, Hooking y amenazas principales en Smartphone con sistema operativo Android, con el objetivo de establecer la contextualización de terminología para el desarrollo de este artículo.

La ciberseguridad conforme a la norma ISO 27001, define Activo de Información como “el conocimiento o datos de valor para una organización”, mientras que un Sistema de Información incluye aplicaciones, servicios, activos de tecnologías de información u otros componentes que permiten el manejo de la misma.

La Seguridad de la Información, incluye métodos, procesos o técnicas para la protección de la información en formato digital de una organización o persona, incluyendo redes e infraestructura tecnológica. La principal diferencia entre Ciberseguridad y Seguridad de la Información (SI), es que esta la SI se encarga de las medidas adecuadas de protección de la información, de acuerdo a su importancia y criticidad, independientemente de su formato, medio de almacenamiento (físico o digital) y de transmisión.

Por otro lado, el término Hooking (Tinajero, 2017) abarca un conjunto de técnicas utilizadas para alterar o aumentar el comportamiento de un sistema operativo, sus aplicaciones o sus componentes, interceptando mediante un código llamado Hook, llamadas de función, mensajes o eventos. En el ámbito de la Ciberseguridad, Hooking ayuda a detectar comportamientos irregulares para así prevenir, por ejemplo, la ejecución de aplicaciones maliciosas.

Con respecto a las amenazas más relevantes para la plataforma Android, es el software malicioso desarrollado específicamente para la plataforma (Mendoza-López, 2015). Su cantidad, complejidad y diversidad crece de forma constante y exponencial, de tal forma que, nuevas familias de códigos maliciosos y sus variantes se desarrollan con distintos objetivos, entre los que se encuentran, por ejemplo: (a) troyanos que suscriben al usuario a servicios de mensajería SMS premium sin su consentimiento; (b) botnets que buscan convertir en zombi al dispositivo móvil mientras roban su información; (a) adware para el envío de publicidad no deseada y; (d) ransomware que cifra la información y solicita un pago como rescate para que el usuario pueda recuperar sus datos. (Mendoza-López, 2015).

De acuerdo con el reporte de 2016, Amenazas de Seguridad en Internet, publicado por Norton (Symantec Corp., 2016a), tan sólo en el año de 2015, se identificaron 528 nuevas vulnerabilidades móviles, 18 nuevas familias de malware para Android y 3,944 nuevas variantes de malware para el ecosistema Android.

3. Herramienta para identificar aplicaciones potencialmente maliciosas

3.1 Antecedentes de la Herramienta

Para establecer las bases de la herramienta propuesta se llevo a cabo una revisión sistemática (Mejía, et al., 2017). A partir del análisis de los resultados del proceso de Revisión Sistemática, y de las diferentes áreas de interés orientadas en la detección de aplicaciones maliciosas y de vulnerabilidades en Smartphones, se identifican los siguientes resultados.

Entre las técnicas utilizadas se encuentran: 1) Hooking, que se basa en la inyección de código, para la detección y tratamiento de aplicaciones maliciosas (Tinajero-Manjarez et al., 2017); 2) Máquinas de Soporte Vectorial (Support Vector Machines, por sus siglas en inglés SVM), utilizadas para clasificar aplicaciones desconocidas como maliciosas o benignas y en la detección de actividad de los usuarios, mencionadas en (Kaghyan & Sarukhanyan, 2013) y (Spreitzenbarth et al., 2015); 3) Redes Neuronales para localización en interiores y exteriores, aplicado a tráfico y demografía, descritas en (Su et al., 2014), (Huang et al., 2014), (Paul & George, 2015), (Mugagga & Winberg, 2015), (Mugagga & Winberg, 2015), (Carlos E. Galván-Tejada et al., 2015), y (Dou et al., 2015); 4) K-Means y Local Outlier Factor (por sus siglas en inglés LOF) para detección de anomalías, utilizado en dominios como: medicina, detección de intrusos y fraude, abordadas en (Karim et al., 2014) y (Pasillas D. & Ratté, 2016); 5) Árboles de Decisión, Redes Bayesianas y Regresión, en cómputo forense y para la localización y detección de actividad utilizado en la detección de acoso escolar o bullying, analizadas en (Moço & Lobato-Correira, 2014) y (Garcia-Ceja et al., 2014), y; 6) Big Data y Autómatas Celulares para localización en interiores/exteriores y demografía, cubierta en (Tosi et al., 2014), (Asri et al., 2015), y análisis de comportamiento humano en entornos sociales (Human Social Behavior, por sus siglas en inglés HSB), estudiado en (More & Lingam, 2015) y (Dou et al., 2015).

De igual forma, se observó que los sensores de Smartphones son utilizados, principalmente, para: 1) la detección de la actividad del usuario, por medio del giroscopio y el acelerómetro, como se menciona en (Paul & George, 2015), (Moço & Lobato-Correira, 2014), (Carlos E. Galván-Tejada et al., 2015) y (Garcia-Ceja et al., 2015); 2) la localización en interiores y exteriores con ayuda de, por ejemplo el campo magnético de la tierra, utilizando sensores de orientación, micrófono e iluminación, estudiada en (Carlos E. Galván-Tejada et al., 2015) autenticación, cubierta en (Haque et al., 2013).

Por otro lado, los tipos de dato más utilizados para el análisis de comportamiento de un Smartphone son: 1) datos crudos, utilizados en (Tuttle et al., 2010), (Su et al., 2014), (Tosi et al., 2014), (Pasillas D. & Ratté, 2016) y (Garcia-Ceja et al., 2014), y; 2) logs de información, como se muestra en (Asri et al., 2015), (Mestry et al., 2015), (More & Lingam, 2015), (Mugagga & Winberg, 2015), (Moço & Lobato-Correira, 2014); mientras que, las principales vulnerabilidades detectadas son: 1) la infección por malware, analizada en (Karim et al., 2014); 2) el acceso o disposición indebida de la información y comportamiento sospechoso, cubiertos en (Pasillas D. & Ratté, 2016) y (Haque et al., 2013), y; 3) la aplicación de código malicioso, estudiado en (Chen et al., 2015).

Con respecto a herramientas se encontraron las siguientes: 1) ARES – Adaptable Reverse Engineering System, presentada en (Tuttle et al., 2010), una herramienta para Sistemas Móviles Forenses que asiste en el análisis de los datos recuperados en las investigaciones de actividad de Smartphones, por ejemplo: Logs de llamadas, email, mensajes de texto, eventos de calendarios, entre otros; 2) Log Analysis, mostrada en (Catanese & Fiumara, 2010), otra herramienta de Cómputo Forense que, utilizando técnicas de clustering, representa de manera gráfica el resultado de un análisis de la información, haciendo uso del formato GraphML (basado en lenguaje de marcas XML); 3) Demographic-Vis, cubierta en (Dou et al., 2015), el cual analiza información demográfica, en base al contenido generado por el usuario a través de una encuesta

para determinar características sociales, económicas y de comportamiento, por medio de técnicas como SVM y Redes Bayesianas, y; 4) Xposed framework (Barolli & Enokido, 2018), herramienta enfocada en la detección de aplicaciones maliciosas.

3.2. Desarrollo de la herramienta

De acuerdo a los resultados de la revisión sistemática la herramienta propuesta debe otorgar funcionalidades con respecto a:

- Detectar oportunamente aplicaciones potencialmente maliciosas mediante la ponderación de los permisos concedidos estática y dinámicamente a una aplicación, mediante la suma de los pesos asignados a cada uno de los permisos clasificados como peligrosos.
- Prevenir la disposición indebida de información sensible de usuario.
- Notificar al usuario de cualquier riesgo potencial mediante semaforización: (1) aplicación no maliciosa – negra; (2) aplicación sospechosa – amarilla; y (3) aplicación potencialmente maliciosa – roja.

Para el desarrollo de esta herramienta se llevo a cabo la siguiente tecnología:

- Android Studio, API 18+ como IDE de desarrollo.
- Xposed framework API (XposedBridge) 5.4 (jar), marco de trabajo para desarrollo de los módulos que cambian el comportamiento del sistema Android y sus respectivas aplicaciones.
- Smartphone con sistema operativo Android v4.0 API 18+.
- KingRootV5.0.5.397172 (apk), software para obtener privilegios de superusuario.
- Xposed Installer 2.6.1 (apk), gestor de los módulos desarrollados con el Xposed framework.

La aplicación propuesta, está conformado por una interfaz de usuario, habituales para una app de Android, con objetos View y ViewGroup para desplegar los elementos gráficos, tales como: aplicaciones, servicios, permisos, etc. La Figura 1, muestra un ejemplo de la vista principal de la aplicación.

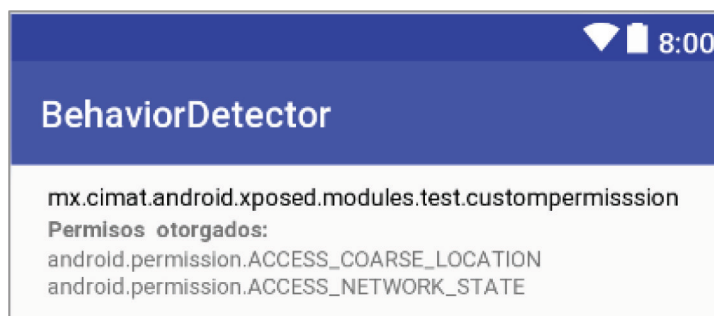


Figura 1 – Vista principal de la aplicación “BehaviorDetector”

La arquitectura de la aplicación propuesta, la cual es llamada BehaviorDetector, y los elementos que la conforman está dividida en 3 paquetes principales (Figura 2):

- **base**, representa las clases, interfaces y entidades que implementa la funcionalidad principal mediante el uso del framework de Xposed, p.e: la clase hook, que es la encargada de realizar las llamadas a la API de Xposed
- **hooks**, donde se definen las clases que implementan tanto la funcionalidad estática como dinámica de la aplicación propuesta; y
- **clase monitor**, encargada de administrar los llamados a la funcionalidad estática y dinámica de la aplicación

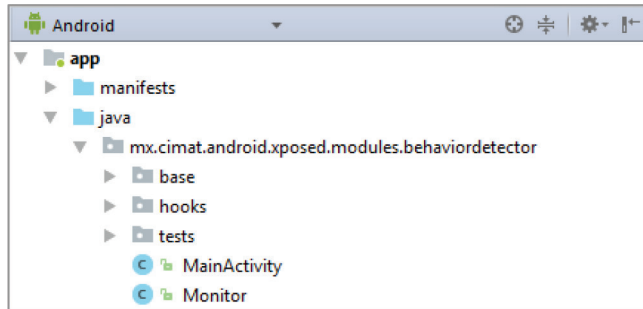


Figura 2 – Paquetes principales de la aplicación “BehaviorDetector”

Cuando se ejecuta instala la apk, se debe habilitar la aplicación con el gestor de módulos del Xposed Installer. Una vez iniciada la aplicación, el programa captura todos los paquetes que se cargan al iniciar el sistema operativo y, mediante la ponderación de los permisos estáticos que tienen definidos en sus manifiestos, determina si es una aplicación o proceso potencialmente peligroso o que ponga en riesgo la integridad del dispositivo, listándolo en la pantalla principal de la aplicación con su correspondiente semáforo.

4. Estudio de Caso

Para validar la herramienta, se realizó el análisis de un grupo de aplicaciones que, de manera nativa, forman parte del sistema operativo Android, versión 4.0 o superior, así como una aplicación personalizada que dinámicamente cambia los permisos solicitados, tanto de manera estática como en tiempo ejecución. El conjunto de permisos sobre los que se evaluó el riesgo potencial de cada aplicación, pertenece al que Android tiene clasificado como riesgosos, es decir, aquellos que podrían afectar la privacidad del usuario o el funcionamiento normal del dispositivo. El sistema operativo solicita al usuario que otorgue explícitamente esos permisos:

- En tiempo de ejecución (para Android 6.0 API 23 o posteriores). El usuario puede revocar los permisos en cualquier momento.
- Al instalar o actualizar la app (para Android 5.1 API 22 o anteriores). La única manera de revocar los permisos, es desinstalando la app.
- También, pueden aplicar permisos en los escenarios siguientes:
 - Cuando una llamada ingresa al sistema, para evitar que la app ejecute determinadas funciones.

- Cuando comienza una actividad, para evitar que las apps inicien actividades de otras apps.
- Cuando se envían y reciben transmisiones, para controlar los receptores y transmisores.
- Cuando se inicia o vincula un servicio.

Los permisos riesgosos abarcan áreas en las cuales una app requiere datos o recursos que incluyen información privada del usuario, o bien que podrían afectar los datos almacenados del usuario o el funcionamiento de otras apps. Por ejemplo, la capacidad de leer los contactos del usuario, se considera es un permiso riesgoso.

Todos los permisos riesgosos del sistema Android pertenecen a grupos de permisos. Si el dispositivo tiene Android 6.0 (nivel de API 23) instalado, el siguiente comportamiento del sistema tiene lugar cuando una app solicita un permiso riesgoso:

- Si una app solicita un permiso riesgoso incluido en su manifiesto y no tiene permisos actualmente en el grupo de permisos, el sistema muestra un cuadro de diálogo al usuario en el que se describe el grupo de permisos al cual la app desea acceder. En el cuadro de diálogo no se describe el permiso específico dentro de ese grupo. Por ejemplo, si una app solicita el permiso `READ_CONTACTS`, en el cuadro de diálogo del sistema se indica únicamente que la app necesita acceso a los contactos del dispositivo. Si el usuario brinda la aprobación, el sistema otorga a la app solamente el permiso que solicitó
- Si una app solicita un permiso riesgoso incluido en su manifiesto y ya tiene otro permiso riesgoso en el mismo grupo de permisos, el sistema lo otorga de inmediato sin interacción con el usuario. Por ejemplo, si a una app ya se le otorgó el permiso `READ_CONTACTS` y luego esta solicita el permiso `WRITE_CONTACTS`, el sistema lo otorga de inmediato.

Cualquier permiso puede pertenecer a un grupo de permisos. Sin embargo, el grupo de un permiso solo afecta la experiencia del usuario si es riesgoso. Se puede ignorar el grupo de permisos para los permisos normales. Por lo que se realizaron pruebas en Aplicaciones personalizadas y nativas.

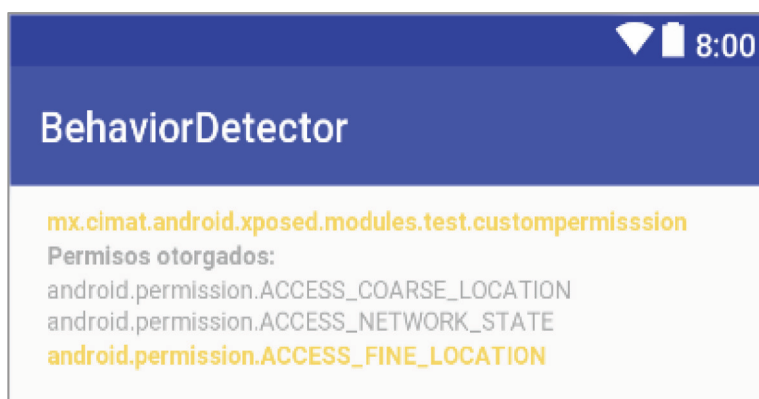


Figura 3 – Detección de una aplicación “sospechosa”.

Aplicación personalizada: Se realizaron varias pruebas durante el desarrollo del módulo con el objetivo de probar la efectividad de la detección de aplicaciones potencialmente maliciosas, por la solicitud de permisos riesgosos en tiempo de ejecución. Se probó usando una aplicación que usa permisos similares a los utilizados por aplicaciones maliciosas. El resultado obtenido, muestra que, conforme se van incrementando los permisos riesgosos, el semáforo para esa aplicación cambia, dependiendo de los permisos solicitados. En las Figura 3 y Figura 4, se muestra como el módulo desarrollado por la aplicación propuesta, alerta al usuario la ejecución de aplicaciones potencialmente maliciosas.

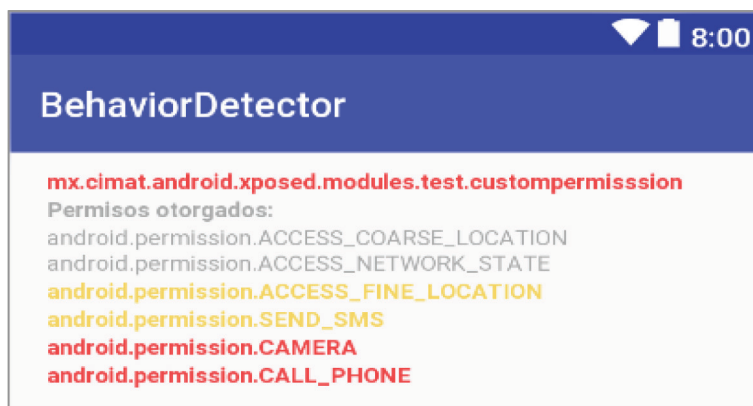


Figura 4 – Detección de una aplicación “potencialmente maliciosa”.

Los permisos detectados, normalmente serían utilizados en Android para realizar cualquier tipo de trabajo malicioso. Este conjunto de permisos representa un subconjunto

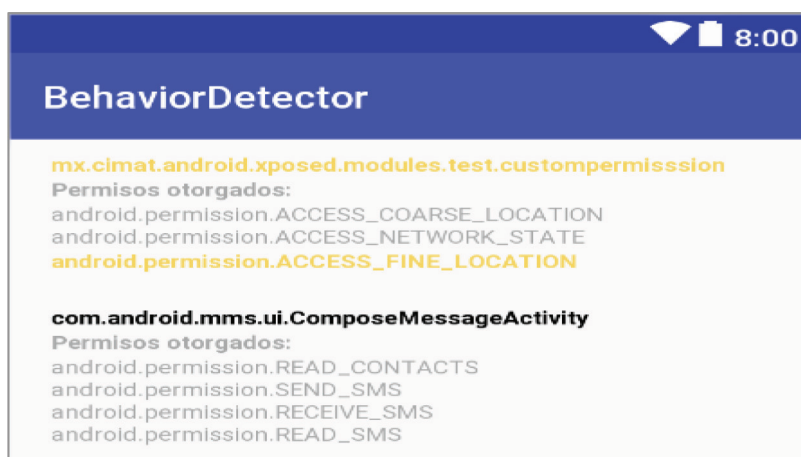


Figura 5 – Detección de una aplicación nativa “no maliciosa”.

de los permisos riesgosos utilizados para acceder a información privada y un medio de comunicación con el mundo exterior.

Aplicaciones nativas: También se probó este enfoque de detección con aplicaciones nativas que se cargan al inicio del sistema operativo. La Figura 5, muestra un ejemplo del análisis realizado durante el arranque del sistema operativo, donde se encuentra una aplicación no maliciosa.

Al observar los resultados obtenidos al utilizar la herramienta propuesta se pudo observar que: (1) se detectan aplicaciones con permisos potencialmente riesgosos declarados estáticamente; (2) se detectan las aplicaciones que solicitan permisos riesgosos de forma dinámica o en tiempo de ejecución; y (3) se notifica al usuario sobre las aplicaciones o servicios potencialmente riesgosos que estén instalados o en ejecución.

6. Conclusiones.

El creciente uso de las Tecnologías de Información ha revolucionado a tal grado de evolucionar en nuevos dispositivos como son los Smartphone, los cuales se encuentran conectados a internet, lo que ha permitido nuevas formas de transmitir información. Sin embargo, esto mismo ha generado que la seguridad en estos dispositivos sea más complejo, ya que se han vuelto el medio masivo de comunicación portátil más utilizado. En este contexto, Android como el sistema operativo más utilizado actualmente en Smartphone lo ha convertido en blanco del mayor número de amenazas creciente. En este contexto, analizar la información generada por estos dispositivos, es necesario contar con herramientas de análisis automático de datos. Por lo tanto, la herramienta desarrollada ha permitido detectar de manera oportuna aplicaciones personalizadas o nativas que son maliciosas. Al observar los resultados obtenidos se pudo observar que el uso de la herramienta es viable para ayudar a usuarios con un nivel de habilidad y conocimiento mayor al básico del sistema operativo Android.

Agradecimientos

Al alumno Ricardo E. Melchor Velásquez quien ha desarrollado la herramienta presentada en este artículo como parte de su tesis para obtención del grado de Maestro en Ingeniería de Software ofrecida en el Centro de Investigación en Matemáticas.

Referencias

- Asri, H., Mousannif, H., Al Moatassime, H., & Noel, T. (2015). Big data in healthcare: Challenges and opportunities. In: 2015 International Conference on Cloud Technologies and Applications (CloudTech) (pp. 1–7). IEEE. Doi: 10.1109/CloudTech.2015.7337020.
- Barolli, L., & Enokido, T. (2018). Innovative Mobile and Internet Services in Ubiquitous Computing. In: 11th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2017) (Vol. 612, pp. 952–958). Torino, Italia: Springer. Doi: 10.1007/978-3-319-61542-4.

- Catanese, S. A., & Fiumara, G. (2010). A visual tool for forensic analysis of mobile phone traffic. In: MiFor '10 Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence (pp. 71–76). New York, NY. USA: ACM. Doi: 10.1145/1877972.1877992.
- Cisco. (2017) ¿Qué es un firewall? Retrieved from: http://www.cisco.com/c/es_mx/products/security/firewalls/what-is-afirewall.html.
- Chen, Y., Lee, H., Jeng, A. B., & Wei, T. (2015). DroidCIA: A Novel Detection Method of Code Injection Attacks on HTML5-Based Mobile Apps. In: 2015 IEEE Trustcom/BigDataSE/ISPA (pp. 1014–1021). IEEE. Doi: 10.1109/Trustcom.2015.477.
- Dai J., Miao H. (2005). D_DIPS: An Intrusion Prevention System for Database Security. In: Qing S., Mao W., López J., Wang G. (eds). Information and Communications Security. ICICS 2005. Lecture Notes in Computer Science, (vol 3783). Berlin, Heidelberg: Springer.
- Dou, W., Cho, I., ElTayeb, O., Choo, J., Wang, X., & Ribarsky, W. (2015). DemographicVis: Analyzing demographic information based on user generated content. In: 2015 IEEE Conference on Visual Analytics Science and Technology (VAST) (pp. 57–64). IEEE. Doi: 10.1109/VAST.2015.7347631.
- ESET LLC. (2014). Guía de seguridad para smartphones: ¿Cómo configurar tu Android de la forma más segura?. Buenos Aires, Argentina. Retrieved from: http://www.welivesecurity.com/wp-content/uploads/2014/06/guia_seguridad_android_eset.pdf.
- ESET LLC. (2016). Trends 2016 (in) Security Everywhere. Retrieved from: <http://www.welivesecurity.com/wp-content/uploads/2016/01/eset-trends-2016-insecurity-everywhere.pdf>.
- Galván-Tejada, C. E., García-Vázquez, J. P., Galván-Tejada, J. I., Delgado-Contreras, J. R., & Brena, R. F. (2015). Infrastructure-Less Indoor Localization Using the Microphone, Magnetometer and Light Sensor of a Smartphone. Sensors (Basel), 15(8), 20355–20372. Doi: 10.3390/s150820355.
- García-Ceja, E., Brena, R., & Galván-Tejada, C. E. (2014). Contextualized hand gesture recognition with smartphones. In: 2014 Mexican Conference on Pattern Recognition (MCPR) (pp. 122–131). Springer. Doi: 10.1007/978-3-319-07491-7_13.
- Haque, M. M., Zawoad, S., & Hasan, R. (2013). Secure techniques and methods for authenticating visually impaired mobile phone users. In: 2013 IEEE International Conference on Technologies for Homeland Security (HST) (pp. 735–740). IEEE. Doi: 10.1109/THS.2013.6699095.
- Huang, C., Ying, J. J., Tseng, V. S., & Zhou, Z. (2014). Location semantics prediction for living analytics by mining smartphone data. In: 2014 International Conference on Data Science and Advanced Analytics (DSAA) (pp. 527–533). IEEE. Doi: 10.1109/DSAA.2014.7058122.

- INEGI. (2014). Estadísticas sobre la disponibilidad y uso de tecnología de información y comunicaciones en los hogares, 2013. Aguascalientes, Ags. México. Retrieved from: <http://www3.inegi.org.mx/sistemas/biblioteca/ficha.aspx?upc=702825062378>.
- Jin H., Yang Z., Sun J., Tu X., & Han Z. (2005) CIPS: Coordinated Intrusion Prevention System. In: Kim C. (eds). Information Networking: Convergence in Broadband and Mobile Networking. ICOIN 2005. Lecture Notes in Computer Science, (vol 3391). Springer, Berlin, Heidelberg: springer.
- Kaghyan, S., & Sarukhanyan, H. (2013). Accelerometer and GPS sensor combination based system for human activity recognition. In: Ninth International Conference on Computer Science and Information Technologies (CSIT) Revised Selected Papers (pp. 1–9). IEEE. Doi: 10.1109/CSITechnol.2013.6710352.
- Karim, A., Salleh, R. B., Shiraz, M., Shah, S. A. A., Awan, I., & Anuar, N. B. (2014). Botnet detection techniques: review, future trends, and issues. Journal of Zhejiang University SCIENCE C, 15(11), 943–983. Doi: 10.1631/jzus.C1300242.
- Maloof, M. A. (2006). Machine Learning and Data Mining for computer security: Methods and applications. Advanced Information and Knowledge Processing. Washington, DC. USA: Springer.
- Martínez Retenaga, A. (2015). Situación del Malware para Android. Madrid, España. Retrieved from: <https://www.certs.es/guias-y-estudios/estudios/android-malware-situation>.
- Mendoza-López, M. Á. (2015). Herramientas de detección - Investigación, desarrollo y acción. Coordinación de Seguridad de La Información UNAM-CERT, (23), 34. Retrieved from: http://revista.seguridad.unam.mx/sites/revista.seguridad.unam.mx/files/revistas/pdf/23_RevistaSeguridad-HerramientasDeDeteccion.pdf.
- Mejía-Miranda, J., Melchor-Velásquez, R.E., & Muñoz-Mata, M.A. (2017). Vulnerability detection in smartphones: A systematic literature review. In: 2017 12th Iberian Conference on Information Systems and Technologies (CISTI), (pp. 1-7). doi: 10.23919/CISTI.2017.7975914.
- Mestry, M., Mehta, J., Mishra, A., & Gawande, K. (2015). Identifying associations between smartphone usage and mental health during depression, anxiety and stress. In: 2015 International Conference on Communication, Information & Computing Technology (ICCICT). IEEE. Doi: 10.1109/ICCICT.2015.7045656.
- Moço, N. F., & Lobato-Correira, P. (2014). Mobile forensics: A smartphone-based activity logger. In: 2014 21st International Conference on Telecommunications, ICT 2014 (pp. 462–466). IEEE. Doi: 10.1109/ICT.2014.6845159.
- More, J. S., & Lingam, C. (2015). Reality Mining based on Social Network Analysis. In: 2015 International Conference on Communication, Information & Computing Technology (ICCICT). Mumbai, India: IEEE. Doi: 10.1109/ICCICT.2015.7045752.
- Mugagga, P. K. B., & Winberg, S. (2015). Sound Source Localisation on Android Smartphones. In: AFRICON 2015. Doi: 10.1109/AFRCON.2015.7331970.

- Pasillas D., J. R., & Ratté, S. (2016). An unsupervised approach for combining scores of outlier detection techniques, based on similarity measures. In: CLEI 2016 - The Latin American Computing Conference (pp. 61–77). Valparaíso, Chile: Electronic Notes in Theoretical Computer Science. Doi: 10.1016/j.entcs.2016.12.005.
- Paul, P., & George, T. (2015). An Effective Approach for Human Activity Recognition on Smartphone. In: 2015 IEEE International Conference on Engineering and Technology (ICETECH). IEEE. Doi: 10.1109/ICETECH.2015.7275024.
- Ponemon Institute LLC. (2014). Enhancing Cybersecurity with Big Data: Challenges & Opportunities. Retrieved from: <https://blogs.microsoft.com/microsoftsecure/2014/11/19/new-report-enhancing-cybersecurity-with-big-data/>.
- Spreitzenbarth, M., Schreck, T., Echtler, F., Arp, D., & Hoffmann, J. (2015). Mobile-Sandbox: combining static and dynamic analysis with machine-learning techniques. *International Journal of Information Security*, 14(2), 141–153. Doi: 10.1007/s10207-014-0250-0
- Sanou, B. (2016). ICT Facts and Figures 2016. Geneva, Switzerland. Retrieved from: <http://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx>
- Sherman, F. (2019). The Differences Between a Firewall and an Intrusion Detection System. *Small Business - Chron.com*. Retrieved from: <http://smallbusiness.chron.com/differences-between-firewall-intrusion-detection-system-62856.html>
- Su, X., Tong, H., & Ji, P. (2014). Activity Recognition with Smartphone Sensors. *Tsinghua Science and Technology*, 19(3), 235–249. Doi: 10.1109/TST.2014.6838194
- Symantec Corp. (2016a). Internet Security Threat Report. Mountain View, CA. Retrieved from: <https://www.symantec.com/security-center/threat-report>
- Symantec Corp. (2016b). Norton Cybersecurity Insights Report. Mountain View, CA. Retrieved from: http://us.norton.com/norton-cybersecurity-insights-report-global?inid=hho_norton.com_cybersecurityinsights_hero_seeglobalrpt
- Tinajero-Manjarez, G., Escamilla-Ambrosio, P. J., & Rodríguez-Mota, A. (2017). Analysis and Detection of Malware in Smartphones with Android Operating System Using Hooking Techniques. In: Conference'17, July 2017, Washington, DC, USA (p. 6). Washington, DC. USA.
- Tosi, D., Marzorati, S., & Pulvirenti, C. (2014). Vehicular Traffic Predictions from Cellular Network Data – A real world case study. In: 2014 International Conference on Connected Vehicles and Expo (ICCVE) (pp. 485–491). Doi: 10.1109/ICCVE.2014.7297594
- Tuttle, J., Walls, R. J., Learned-Miller, E., & Niel-Levine, B. (2010). Reverse engineering for mobile systems forensics with Ares. In: Proceedings of the 2010 ACM Workshop on Insider Threats - Insider Threats '10, (pp. 21–28). Doi: 10.1145/1866886.1866892
- Woods V., (2016). Gartner Says Worldwide Smartphone Sales Grew 3.9 Percent in First Quarter of 2016. Egham, UK.

Sistema inteligente para el monitoreo automatizado del transporte público en tiempo real

Yadira Quiñonez¹, Carmen Lizarraga¹, Juan Peraza¹, Oscar Zatarain¹

yadiraqui@uas.edu.mx, carmen.lizarraga@uas.edu.mx, jfperaza@uas.edu.mx, ozatarain@uas.edu.mx

¹ Universidad Autónoma de Sinaloa, Facultad de Informática, Av. Leonismo Internacional S/N y Av. de los Deportes, Ciudad Universitaria, 80000 Mazatlán, Sinaloa, México.

DOI: 10.17013/risti.31.94-105

Resumen: Este artículo propone un prototipo totalmente funcional para el monitoreo del transporte público, el principal objetivo es proporcionar información en tiempo real a los usuarios que utilizan los autobuses como medio de transporte, y al mismo tiempo, proporcionar datos históricos de cada unidad para conocer la información relacionada con los horarios y retrasos de las rutas realizadas en los diferentes circuitos; para lograr el objetivo se ha utilizado un conjunto de recursos y medios tecnológicos que se centran en las capacidades de una computadora de placa reducida Raspberry PI. De manera general, el prototipo está formado por cuatro módulos: el rastreador de GPS, que consta de dos componentes principales: una Raspberry PI y un módulo GPS NEO-6M Ublox. El servicio web y la base de datos se implementan en PostgreSQL, que almacena la información a través de un servicio web RestFul desarrollado en PHP. La aplicación web para la gestión de seguimiento de bus está desarrollada en PHP y utiliza el marco Bootstrap para la interfaz de usuario. Finalmente, la aplicación móvil para Android destinada a usuarios finales, se desarrolló en la plataforma de Visual Studio 2015 con Xamarin y se programó en C#.

Palabras-clave: Raspberry Pi, GPS, Base de Datos para el Monitoreo de Transporte, Servicio Web.

Intelligent system for automated monitoring of public transport in real time

Abstract: This article proposes a fully functional prototype for the monitoring of public transport, the main objective is to provide information in real time to users who use buses as a means of transport, and at the same time, provide historical data for each unit to know the information related to the schedules and delays of the routes made in the different circuits; to achieve the objective, a set of resources and technological means have been used that focus on the capabilities of a reduced-scoreboard computer Raspberry PI. In general, the prototype consists of four modules: the GPS tracker, which consists of two main components: a Raspberry PI and a GPS module NEO-6M Ublox. The web service and the database are

implemented in PostgreSQL, which stores the information through a RestFul web service developed in PHP. The web application for bus tracking management is developed in PHP and uses the Bootstrap framework for the user interface. Finally, the mobile application for Android intended for end users, was developed on the Visual Studio 2015 platform with Xamarin and was programmed in C #.

Keywords: Raspberry Pi, GPS, Transportation Monitoring Database, Web Service.

1. Introdução

En las últimas décadas, han surgido diferentes trabajos que abordan el desarrollo de estrategias e incorporación infraestructura para mejorar la seguridad y movilidad del transporte utilizando nuevas tecnologías. De manera colectiva, a estas tecnologías se conocen como Sistemas de Transporte Inteligente (STI), o ITS en sus siglas en inglés Intelligent Transportation Systems, que tiene como objetivo promover el desarrollo inteligente y sostenible, mejorar la eficacia y la productividad del transporte, mediante el uso e incorporación nuevas tecnologías de informática, información y telecomunicación (Lin, 2017, Piedra, 2018). De acuerdo con Chaturvedi y Srivastava (2017) proponen un sistema de STI que utiliza la red celular y la señal GPS para estimar la velocidad del vehículo bajo condiciones de tráfico heterogéneas haciendo uso infraestructura STI limitada. En otro trabajo (Quintero, Patino Alvarez & Aedo, 2016) proponen utilizar algoritmos de seguimiento de trayectorias para verificar si los vehículos de transporte están siguiendo las rutas definidas previamente, mediante el rastreo de la posición actual de la unidad de transporte.

En la actualidad, el transporte público es el medio de movilidad más utilizado, entonces, brindar información más precisa sobre horarios de llegada o retrasos de los vehículos ayuda a los usuarios a tomar buenas decisiones sobre los desplazamientos y mejorar el servicio brindado. Para lograr el objetivo de este desarrollo, la metodología empleada consistió en seleccionar los componentes adecuados que cumplieran con las características específicas y que fuera compatible para programar el algoritmo que permitiera la interacción correcta de todas las partes del sistema.

En este sentido, el objetivo de este desarrollo tecnológico es proporcionar información en tiempo real a los usuarios que utilizan los autobuses como medio transporte, y al mismo tiempo, facilitar datos históricos de cada unidad para conocer toda la información relacionada con los recorridos realizados en los diferentes circuitos y tomar decisiones inteligentes sobre los desplazamientos. Esto supone un mejor servicio atendiendo la problemática social como lo es el transporte público, en relación a la seguridad, se tendrá un control de cada vehículo en tiempo real, obteniendo la velocidad instantánea con el fin de evitar excesos de velocidad en zonas urbanas y reducir accidentes.

El presente trabajo aborda la temática de rastreo de unidades de transporte urbano mediante la utilización de diferentes dispositivos y recursos tecnológicos, tales como: la Raspberry Pi, GPS, Servicios Web y Base de Datos, que en conjunto conforman un sistema de monitoreo automatizado. Se abordan puntos técnicos del funcionamiento del sistema, así como los recursos utilizados para su desarrollo.

De manera general, la automatización de procesos busca reducir costes, incrementar la calidad y mejorar la eficiencia ante las necesidades presentadas en el mercado (González-

Filgueira, 2018). Actualmente, hay diversas aplicaciones que utilizan la tecnología raspberry pi para llevar a cabo diferentes tareas de automatización. En los trabajos (Sandeep, 2015) y (Jamil, 2015) proponen un sistema para el funcionamiento automático y control de aparatos electrodomésticos de forma remota utilizando la raspberry pi. En (Dong, 2015) presentan un método para la detección de inactividad de una alarma en tiempo real implementada en la raspberry pi. En (Ikhankar, 2015) utilizan la raspberry pi para crear un robot con un sistema de vigilancia en tiempo real. En (Nguyen, Loan, Mao & Huh, 2015) proponen la implementación de un sistema de monitoreo de bajo costo mediante una raspberry pi, reduciendo los defectos de los actuales sistemas de vigilancia, tales como: imagen borrosa o anomalías de identificación.

Este artículo se encuentra estructurado de la siguiente manera sección 2 se describen las herramientas de implementación que se han utilizado para llevar a cabo el sistema de monitoreo automatizado. En la sección 3 se describe brevemente la descripción de cada uno de los módulos del sistema. La sección 4 describe el esquema general de funcionamiento del sistema de monitoreo. En la sección 5 se presentan los resultados experimentales y las pruebas realizadas del funcionamiento del sistema en tiempo real. Finalmente, la sección 6 consta las conclusiones sobre el sistema y sobre los trabajos que se tiene planeados para el futuro.

2. Herramientas de Implementación

En la actualidad, existen diferentes tecnologías de bajo coste que permite desarrollar diferentes sistemas de una manera sencilla, algunas de estas tecnologías son: Arduino (Arduino, 2018), Raspberry Pi (RaspberryPi, 2018), OpenPicus (OpenPicus, 2018), Cubieboard (Cubieboard, 2018), Udoo (Udoo, 2018), entre otras. Sin embargo, las tecnologías más utilizadas y conocidas son Arduino, OpenPicus y Raspberry Pi.

Este trabajo se enfoca en el uso de la tecnología raspberry pi (ver Fig.1). La placa raspberry pi (RaspberryPi, 2018) fue creada en Reino Unido por la Fundación Raspberry Pique, es básicamente una microcomputadora de bajo coste que cuenta con un sistema operativo con tres diferentes distribuciones basadas en Linux, con memoria RAM, cuenta con entradas y salidas para teclado, monitor, ratón y cable Ethernet. Al ser software libre, el código fuente está accesible para cualquier usuario pueda estudiarlo o modificarlo,

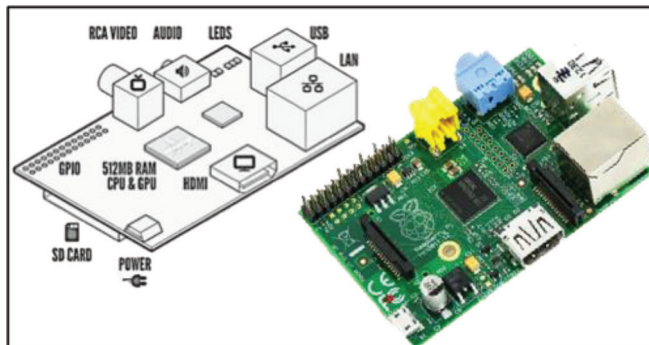


Figura 1 – Componentes de la raspberry pi

por lo tanto, la instalación de la distribución adecuada se realiza de manera sencilla a partir de una tarjeta SD con conocimientos básicos de Linux, solo es necesario seguir las instrucciones indicadas.

En la tabla 1 se muestra un resumen de las características del modelo B de raspberry pi, esta plataforma ofrece capacidades avanzadas de procesamiento debido a la potencia del CPU y GPU, recomendable cuando se va a procesar una n cantidad de datos.

Características	Descripción
<i>CPU</i>	ARM 1176JZF a 900 MHz
<i>GPU</i>	Broadcom VideoCore IV, OpenGL ES 2.0, MPEG-2 y VC-1 (con licencia), 1080p 30H.264/MPEG-4 AVC
<i>Puertos USB 2.0</i>	2 (vía hub USB integrado)
<i>Entradas de vídeo</i>	Conector MIPI CSI que permite instalar un módulo de cámara desarrollado por la RP
<i>Salidas de vídeo</i>	Conector RCA (PAL y NTSC), HDMI (rev1.3 y 1.4), Interfaz DSI para panel LCD
<i>Salidas de audio</i>	Conector de 3.5 mm, HDMI
<i>Almacenamiento Integrado</i>	SD / MMC / ranura para SDIO
<i>Conectividad de la red</i>	10/100 Ethernet (RJ-45) vía hub USB
<i>Consumo energético</i>	700 mA, (3.5 W)
<i>Dimensiones</i>	85.60mm × 53.98mm
<i>Sistemas operativos soportados</i>	GNU/Linux: Debian (Raspbian), Fedora (Pidora), Arch Linux (Arch Linux ARM), Slackware Linux. RISC OS2

Tabla 1 – Descripción de las características de la Raspberry Pi

3. Descripción de los módulos del sistema

El prototipo para el monitoreo del transporte público consta de manera general de cuatro módulos principales, a continuación hace una descripción de cada uno de los módulos.

Rastreador GPS: el sistema para el monitoreo automatizado de transporte público que tiene como objetivo principal proporcionar información en tiempo real a los usuarios que utilizan los autobuses como su medio transporte, y al mismo tiempo, facilitar datos históricos de cada unidad para conocer toda la información relacionada con los recorridos realizados en los diferentes circuitos y tomar decisiones inteligentes sobre los desplazamientos; todo esto se logra utilizando un conjunto de recursos y medios tecnológicos que se centran en las capacidades de una computadora de placa reducida raspberry pi.

El rastreador GPS funciona bajo un algoritmo desarrollado en Python, el cual obtiene, analiza, procesa y almacena los datos emitidos por el módulo GPS en el servidor. Además, de llevar a cabo una serie de controles con el fin de brindar estabilidad al sistema. Básicamente está compuesto por una computadora de placa reducida raspberry pi (Fig.2), la cual se encuentra cargada con un sistema operativo Rasbian en su versión

Jessie en conjunto con un módulo GPS Ublox modelo neo-6m. Está implementado en distintas plataformas de desarrollo, cada una de ellas atendiendo a la necesidad de cada aplicación. Se utilizaron lenguajes como PHP, JavaScript, C#, Python y Frameworks como Bootstrap.

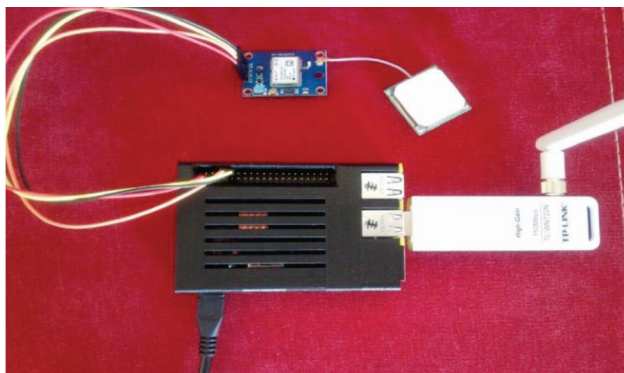


Figura 2 – Dispositivo raspberry pi con módulo GPS Ublox conectado

Con la tecnología GPS se logra un alto grado de precisión y aplicado a diversas áreas (Linli & Zhangyi, 2011). El rastreo por GPS es una de las herramientas indispensables para distintos usos como encontrar un automóvil extraviado, rastrear las unidades de paquetería e inclusive los dispositivos móviles cuentan con la opción de rastreo con el fin de saber tu ubicación o encontrar el dispositivo en caso de extravío. En este sentido, el módulo GPS está enlazado a la raspberry mediante los pines de conexión que esta posee y mediante la instalación de paquetes GPS en la raspberry se obtiene la interfaz donde se puede apreciar los datos de rastreo como latitud, longitud, altitud, velocidad, entre otros datos.

En resumen, el dispositivo raspberry pi tiene como tarea interpretar los datos obtenidos por el dispositivo GPS, esto se realiza mediante un algoritmo desarrollado en Python, la funcionalidad de este algoritmo radica en obtener los datos del GPS y enviarlos al servidor de base de datos, la transmisión de la información se realiza mediante una comunicación inalámbrica mediante un módulo Gsm/Gprs utilizando datos móviles, esto es, una vez que el algoritmo procesa la información, es enviada a la base de datos para almacenarlo. En caso de que los datos móviles fallen el dispositivo de monitoreo está preparado para estar guardando de manera local un archivo de texto con las posiciones que le proporcione el GPS, esto para que no se pierda ningún dato y en el momento en el que se restablezca la señal se envían los datos al servidor.

Servicio web y servidor de base de datos: el funcionamiento básico del servicio web se da a partir del intercambio de datos o archivos en formato XML alojados en el servidor a través de servicios computacionales en la nube (Palos-Sánchez, 2017). Utiliza el protocolo HTTP para establecer la comunicación entre el servidor y el navegador. En este sentido, los servicios RestFul han demostrado que tienen buen potencial para el funcionamiento con este tipo dispositivos con un menor consumo de energía en los

servicios web (Nunes, 2014). En el sistema se ha utilizado RestFul como servicio web el cual se encuentra desarrollado en PHP, que permite obtener los datos del servidor de base de datos realizando una petición, donde devuelve los datos obtenidos en formato JSON y estos pueden ser utilizados por distintas aplicaciones.

Existen diversas alternativas a utilizar como bases de datos, las más utilizadas son: MySQL (MySQL, 2018), SQLite (SQLite, 2018) y PostgreSQL (PostgreSQL, 2018), entre otras. Para este trabajo se ha optado por utilizar la base de datos de tipo PostgreSQL porque es de código abierto, utiliza un modelo cliente/servidor y usa multiprocesos en lugar de multi-hilos con fin de garantizar la estabilidad, robustez y facilidad de administración. Entonces, una vez que se obtiene la información por el dispositivo de monitoreo, se envía al servidor de base de datos alojado en PostgreSQL que es un motor de base de datos muy estable, en él se guardan datos como latitud, longitud, velocidad entre otras cosas, con fin de que la información requerida para el funcionamiento del sistema es almacenada y obtenida por los distintos dispositivos y aplicaciones mediante un Servicio Web RestFul desarrollado en PHP.

Aplicación para la administración del rastreo de autobuses: Es necesario que cada unidad de transporte cuente con un dispositivo de rastreo, para que pueda ser monitoreada desde una aplicación web, esta aplicación web está desarrollada en PHP y se utiliza el framework Bootstrap para la interfaz de usuario. La aplicación web está segmentada en dos partes Autobuses Históricos y Monitoreo en tiempo real. Cabe mencionar, que las rutas de los autobuses se encuentran definidas y están guardadas en la base de datos y las cuales se encuentran divididas en IDA y VUELTA.

Autobuses Histórico. En la Figura 3 se muestra un listado de los autobuses registrados en el sistema realizando una petición al servicio web y las rutas realizadas por cada unidad. Al seleccionar un autobús se solicita la fecha y después muestra un mapa que

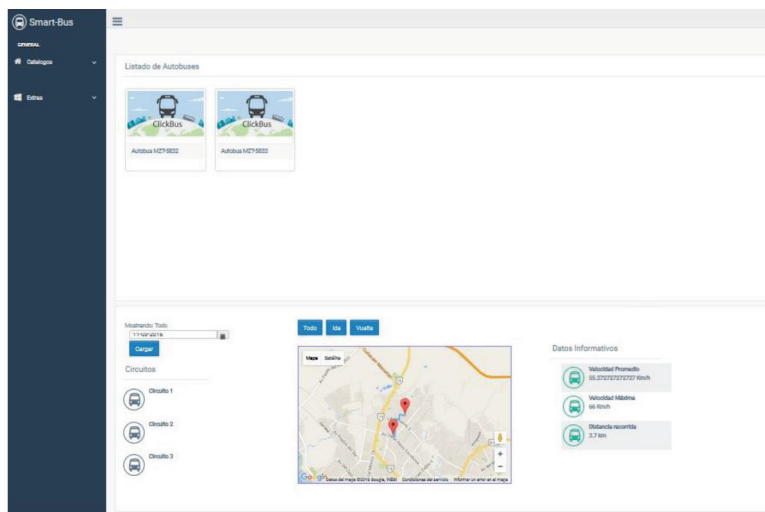


Figura 3 – Autobuses histórico

está implementado con la API de Google Maps con la ruta trazada del recorrido del autobús en el día especificado, se puede seleccionar la visualización del circuito completo del autobús, solo la IDA o en su caso solo la VUELTA, cada una de las opciones muestra la velocidad promedio, velocidad máxima y la distancia recorrida en km. Esto con el fin de saber si el autobús se desvió de la ruta o excedió el límite de velocidad permitido.

Monitoreo en Tiempo Real. Al igual que Autobuses Histórico muestra un listado de autobuses, pero en este caso solo los que se encuentren en servicio, al seleccionar algún autobús se carga el mapa con la posición actual de la unidad y se actualiza cada 5 segundos, al mismo tiempo, muestra la velocidad que lleva el autobús en ese momento.

Aplicación móvil para usuarios finales: se cuenta con una aplicación móvil llamada Smart BS destinada para usuarios finales, esta aplicación está desarrollada en la plataforma de Visual Studio 2015 utilizando Xamarin y programada en C#. Esta aplicación (Fig. 4) se encuentra estructurada en dos partes la primera es la visualización de las rutas de autobuses en tránsito y la segunda es activación de alarmas.

La visualización de rutas aparecen las rutas de autobuses de acuerdo a la ubicación geográfica, a partir de ahí, se selecciona la ruta de la cual se quiere obtener información. Una vez seleccionada la ruta se muestran las paradas oficiales del autobús, al seleccionar alguna de las paradas se carga un mapa mostrando la unidad próxima que va a pasar por el punto especificado y el tiempo estimado para la llegada del autobús.

La aplicación cuenta con alertas, que tienen como propósito avisarle al usuario cuando el autobús está próximo a llegar (Fig.5). Esto se realiza seleccionando la ruta y la parada del autobús, posteriormente, se tiene que especificar la hora que desea tomar vehículo y el tiempo de anticipación, para que el dispositivo le avise mediante una alerta cuando el vehículo le falte el lapso de tiempo especificado para pasar por la parada de autobús.



Figura 4 – Aplicación móvil para Android

Esto con el fin de que el usuario de los transportes públicos tenga una mejor organización del tiempo, de este modo, podrá conocer los tiempos de horarios y retrasos de los vehículos, de esta forma, decidir los tiempos en que desea salir a la parada de autobús.

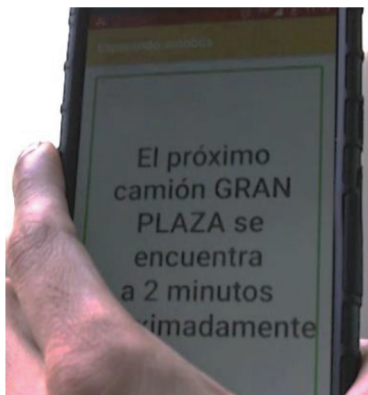


Figura 5 – Alertas para usuarios

4. Esquema general del desarrollo del sistema

El desarrollo del sistema está compuesto por tres etapas: la investigación para el desarrollo de la sistema, estructura o módulos del sistema y la planeación de procesos para su correcto desarrollo. A continuación, en la Figura 6 se muestra el diagrama de desarrollo del sistema para el monitoreo automatizado del transporte público y una breve explicación sobre cada segmento.

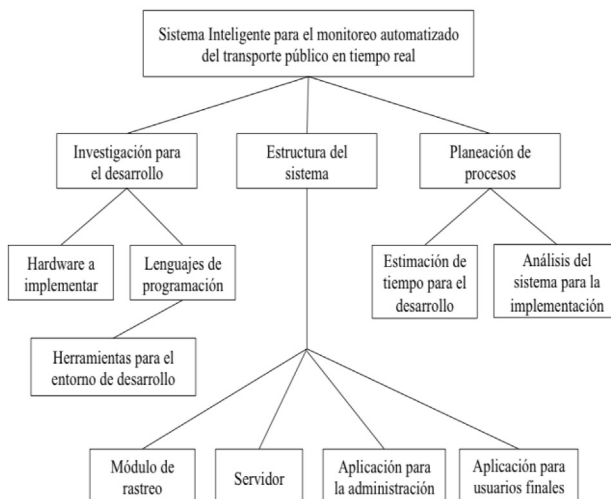


Figura 6 – Esquema general

La investigación del desarrollo se refiere a los dispositivos y al entorno de desarrollo a utilizar, es decir, al estudio del Hardware y al Software que se han implementado, esto se realiza mediante la definición del problema y las herramientas adecuadas a utilizar que cubran las necesidades de la manera más acorde de acuerdo al requerimiento.

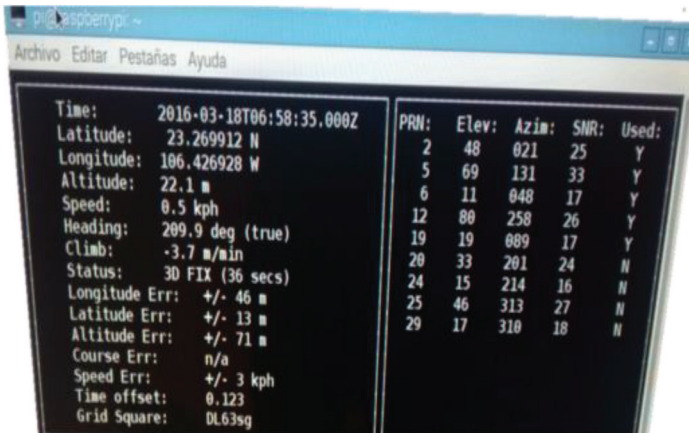
La estructura de sistema se refiere a los módulos que lo conforman, ya que el sistema esta segmentado en diversas aplicaciones, cada una de ellas se encargan de realizar los procesos y funciones necesarias para el correcto funcionamiento.

Al final, se encuentra la planeación de procesos que consiste en estimar tiempos de desarrollo para cada módulo, y su respectiva asignación a la persona que se encarga del desarrollo, al igual que el análisis del sistema, como la funcionalidad deseada y que es lo que se quiere obtener al finalizarla.

5. Resultados experimentales

Para conseguir el correcto funcionamiento del sistema y visualizar la información del trayecto en tiempo real del autobús, así como velocidad y frecuencia aproximada (Fig. 7); se han realizado diversos experimentos para evaluar estabilidad del algoritmo implementado y la respuesta del servidor a las aplicaciones que consumen sus servicios, con el fin de conocer el rendimiento del sistema.

Los primeros experimentos realizados fueron dentro del campus universitario con vehículo personal, posteriormente se hicieron en la ciudad con las rutas propias de los autobuses para medir el rendimiento del algoritmo utilizado y comprobar la viabilidad del proyecto.



Time: 2016-03-18T06:58:35.000Z	PRN: Elev: Azim: SNR: Used:
Latitude: 23.269912 N	2 48 021 25 Y
Longitude: 106.426928 W	5 69 131 33 Y
Altitude: 22.1 m	6 11 048 17 Y
Speed: 0.5 kph	12 80 258 26 Y
Heading: 209.9 deg (true)	19 19 009 17 Y
Climb: -3.7 m/min	20 33 201 24 N
Status: 3D FIX (36 secs)	24 15 214 16 N
Longitude Err: +/- 46 m	25 46 313 27 N
Latitude Err: +/- 13 m	29 17 310 18 N
Altitude Err: +/- 71 m	
Course Err: n/a	
Speed Err: +/- 3 kph	
Time offset: 0.123	
Grid Square: DL63sg	

Figura 7 – Funcionamiento del GPS en la placa raspberry pi

Primero, se han realizado pruebas para corroborar el funcionamiento correcto del dispositivo raspberry pi en conjunto con el rastreador GPS (Fig. 7), en este sentido, se conectó una pantalla a la raspberry pi para comprobar que los datos obtenidos de la señal

GPS correspondían a la posición en la que se encontraba el dispositivo, para visualizar de manera gráfica la posición se ha utilizado Google Maps (Fig. 8) y se ha verificado la posición actual era la correcta.

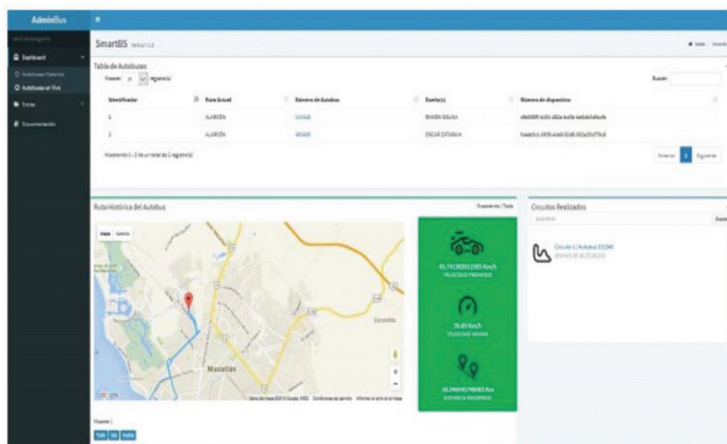


Figura 8 – Visualización de las rutas en Google Maps

Se ha realizado un análisis sobre el consumo de los datos móviles del dispositivo de rastreo con el fin de conocer el promedio de datos (megabyte) utilizados por el rastreador GPS, en la Fig. 9 se muestran los resultados obtenidos en relación a los megabytes consumidos por horas, como se puede observar, el consumo de datos del dispositivo es variable porque depende totalmente del nivel de uso, es decir, el consumo promedio en megabyte depende de las peticiones que se hacen al servidor.

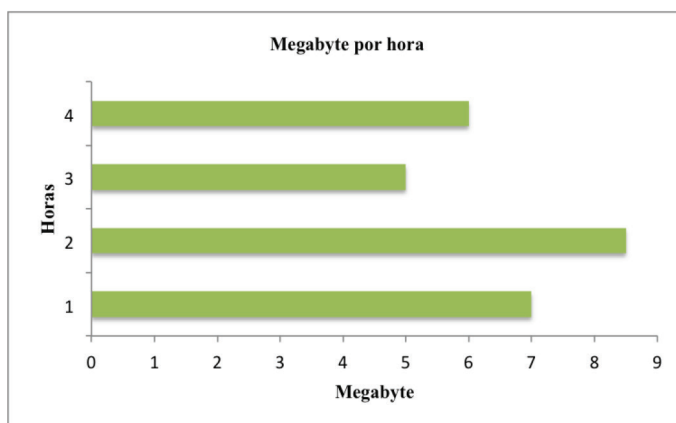


Figura 9 – Megabytes consumidos por hora.

6. Conclusiones

En la actualidad, el transporte público en México y en otras partes del mundo se está convirtiendo en un medio de transporte cada vez más importante y complejo, en este sentido, existe la necesidad de desarrollar tecnología adecuada para proporcionar información de los tiempos de desplazamiento a los usuarios en tiempo real para que puedan trasladarse por las ciudades utilizando el transporte público. Consideramos que este prototipo de monitoreo automatizado es de gran utilidad para los usuarios ya que se tendrá acceso a la red de autobuses en lo que respecta a trayectos, horarios y retrasos.

Con este sistema de monitoreo automatizado se intenta contribuir a llevar un control sobre cada unidad activa con el fin de poder brindar un mejor servicio, es importante mencionar que los componentes utilizados son de bajo coste y se obtiene buen rendimiento, solo que es fundamental llevar a cabo pruebas exhaustivas para poder determinar la estabilidad del sistema a implementar. Todo esto con la finalidad de optimizar los largos periodos de espera de los usuarios, donde los principales beneficiarios serán todas aquellas personas que utilizan los autobuses públicos como medio de transporte.

Smart BS es una aplicación móvil de desarrollo propio bajo la plataforma Android, la cual utiliza un diseño estándar de acuerdo a los lineamientos de Google Material Design para que los usuarios identifiquen los elementos de la interfaz de manera fácil e intuitiva. Al utilizar este estándar de diseño se garantiza que la aplicación sea muy familiar para los usuarios tales como las interfaces gráficas de aplicaciones más populares como WhatsApp y Facebook se encuentran diseñadas bajo estas reglas.

Referencias

- Arduino. (2018). Recuperado de: <https://www.arduino.cc/>.
- Chaturvedi, M., & Srivastava, S. (2017). Multi-Modal Design of an Intelligent Transportation System. *Transactions on Intelligent Transportation Systems*, 18(8), 2017-2027.
- CubieBoard. (2018). A series of open source hardware. Recuperado de <http://cubieboard.org/>.
- Dong, Q., Yang, Y., Hongjun, W., & Jian-Hua, X. (2015). Fall alarm and inactivity detection system design and implementation on raspberry pi. In: *International Conference on Advanced Communication Technology*, (pp. 382-386). Seoul, South Korea: IEEE publishing.
- González-Filgueira, G., & Rodríguez, F.G. (2019). Automatización de una planta industrial de alimentación mediante control distribuido. *Revista Ibérica de Sistemas y Tecnologías de Información*, 27(6), 1-17.
- Ikhankar, R., Kuthe, V., Ulabhaje, S., Balpande, S., & Dhadwe, M. (2015). Pibot: The raspberry pi controlled multi-environment robot for surveillance and live streaming. In: *International Conference on Industrial Instrumentation and Control*, (pp. 1402-1405). Pune, India: IEEE publishing.

- Jamil, M. M. A., & Ahmad, M. S. (2015). A pilot study: Development of home automation system via raspberry Pi. In: International Conference on Biomedical Engineering, (pp. 1-4). Penang, Malaysia: IEEE publishing.
- Lin, Y., Wang, P., & Ma, M. (2017). Intelligent Transportation System (ITS): Concept, Challenge and Opportunity. In: International Conference on Big Data Security on Cloud, (pp. 167-172). Beijing, China: IEEE publishing.
- Linli, W. & Zhangyi, S., (2011). Difference analysis of GPS data base sources based on vehicle location system. In: International Conference on Computer Research and Development, (pp. 421-425). Shanghai, China: IEEE publishing.
- MySQL. (2018). Recuperado de: <https://www.mysql.com/>.
- Nguyen, H.Q., Loan, T. T. K., Mao, B. D., & Huh, E.N. (2015). Low cost real-time system monitoring using raspberry pi. In: International Conference on Ubiquitous and Future Networks, (pp. 857-859). Sapporo, Japan: IEEE publishing.
- Nunes, L.H., Nakamura, L.H.V., Vieira, H.F., Libardi, R.M.O., de Oliveira, E.M., Estrella, J.C., & Reiff-Marganiec, S. (2014). Performance and energy evaluation of RESTful web services in Raspberry Pi. In: International Performance Computing and Communications Conference, (pp. 1-9). Austin, Texas, USA: IEEE publishing.
- OpenPicus Microchip Technology. (2018). Recuperado de: <http://www.microchip/openpicus.com/>.
- Palos-Sánchez, P.R., Arenas-Márquez, F.J., & Aguayo-Camacho, M. (2017). La adopción de la tecnología cloud computing (SaaS): efectos de la complejidad tecnológica vs formación y soporte. *Revista Ibérica de Sistemas y Tecnologías de Información*, 22(6), 89-105.
- Piedra, N., & Suárez J.P. (2018). Hacia la interoperabilidad semántica para el manejo inteligente y sostenible de territorios de alta biodiversidad usando SmartLand-LD. *Revista Ibérica de Sistemas y Tecnologías de Información*, 26(3), 104-121.
- PostgreSQL (2018). PostgreSQL: The world's most advanced open source database. Recuperado de: <https://www.postgresql.org/>.
- Quintero, Y. A., Patino Alvarez, G. A., & Aedo, J. E. (2016). Path-Tracking Algorithm for Intelligent Transportation System. *IEEE Latin America Transactions*, 14(6), 2934-2939.
- Sandeep, V., Gopal, K. L., Naveen, S., Amudhan, A., & Kumar, L. S. (2015). Globally accessible machine automation using raspberry pi based on Internet of Things. In: International Conference on Advances in Computing, Communications and Informatics, (pp. 1144-1147). Kochi, India: IEEE publishing.
- SQLite (2018). SQLite Home Page. Recuperado de: <http://www.sqlite.org/>.
- Raspberry Pi (2018). Teach, Learn, and Make with Raspberry Pi. Recuperado de: <https://www.raspberrypi.org/>.
- UDOO (2018). UDOO: All you need mini PC Android + Linux + Arduino. Recuperado de: <https://www.udoo.org/>.

El Desarrollo del Hardware para la Realidad Virtual

Graciela Lara¹, Alexis Santana¹, Andrés Lira¹, Adriana Peña¹

graciela.lara@academicos.udg.mx,
Adriana.peña@cucei.udg.mx

alexis.s.g@hotmail.com,

eliteandryod@gmail.com,

¹ 1CUCEI, Universidad de Guadalajara, Blvd. Marcelino García Barragán 1421, C.P. 44430, Guadalajara, Jalisco, México.

DOI: 10.17013/risti.31.106–117

Resumen: La Realidad Virtual (RV) es una simulación de un Ambiente Virtual (AV) creado por computadora en el que el usuario tiene la sensación de estar dentro de ese entorno, al mismo tiempo que puede interactuar usando tecnología de hardware llena de posibilidades, con aplicaciones en múltiples áreas tales como la medicina, entretenimiento, educación, psicología, comercio electrónico, entre muchas otras. Con el apoyo de dispositivos como cascos de Realidad Virtual, gafas, y posicionadores, entre otros es posible alcanzar una inmersión total y una mayor interacción dentro de estos ambientes. Asimismo, el usuario puede experimentar que desaparece del mundo real. En este trabajo profundizamos sobre el desarrollo y la evolución de las herramientas de hardware que han permitido la interacción entre los usuarios y los entornos virtuales en 3D. Al mismo tiempo, proponemos una clasificación en eras para evolución de los dispositivos de realidad virtual.

Palabras-clave: Entorno Virtual; Dispositivos de RV; IHC; Inmersión.

The Hardware Development for Virtual Reality

Abstract: Virtual Reality (VR) is a simulation of a Virtual Environment (VE) created by computer in which the user has the sensation of being inside that environment, at the same time that can interact using hardware technology full of possibilities, with applications in multiple areas such as medicine, entertainment, education, psychology, e-commerce, among many others. With the support of devices such as Virtual Reality helmets, glasses, and positioners, among others it is possible to achieve total immersion and greater interaction within these environments. Also, the user may experience that it disappears from the real world. In this work, we go deeper into the development and evolution of hardware tools that have allowed the interaction between users and virtual environments in 3D. At the same time, we propose a classification in eras for the evolution of virtual reality devices.

Keywords: Virtual Environment; VR devices; HC; Immersion

1. Introducción

Dentro de la Realidad Virtual, no sólo basta disponer de las herramientas de software, sino que además es indispensable el uso de dispositivos (hardware), que logren introducir

al usuario en un entorno virtual. Hardware es una palabra de origen inglesa que la Real Academia Española define como el conjunto de aparatos o dispositivos auxiliares e independientes de una computadora.

Los dispositivos periféricos (hardware) en la Realidad Virtual, permiten al usuario estimular sus procesos sensoriales, principalmente el de la vista, el oído y el tacto (Islas, Zabre & Pérez, 2004). En este sentido, dentro de las aplicaciones de Realidad Virtual es importante incluir el manejo del hardware de interacción que sea adecuado, simple de manejar e intuitivo para conseguir el vínculo entre lo real y lo que se quiere percibir como real.

Disponer de hardware conveniente para el uso de aplicaciones de RV, ha permitido alcanzar una retroalimentación e interpretación del comportamiento de usuario. Un ejemplo de estas aplicaciones es el que presentan (Arango, Mazo, Palacio, 2013), donde es posible detectar señales neuronales, para conocer la intención de cerrar o abrir un modelo virtual de una mano o prótesis pegada al muñón real de un usuario. Para reconocer este proceso, se utilizó un dispositivo BCI (por sus siglas en inglés *Brain Computer Interfaces*), conocido como Emotiv EPOC. La disposición de hardware brinda rapidez y rendimiento necesario para obtener la mejor experiencia de RV.

Evidentemente, la disposición de hardware viene vinculada desde la capacidad gráfica de los ordenadores y sus componentes. Estos componentes contemplan un procesador con una frecuencia y número de núcleos capaces, para atender la cantidad de procesos que puedan ejecutarse de manera sincrónica; la memoria RAM necesaria para ejecutar otras aplicaciones en segundo plano; las unidades de almacenamiento, donde se montarán las aplicaciones de RV; así como otro software necesario para su ejecución. Finalmente, es indispensable disponer de una tarjeta gráfica poderosa, para que el rendimiento de las aplicaciones de RV sea eficaz.

Dentro de este contexto, los tipos de dispositivos periféricos se han clasificado en *dispositivos de entrada y dispositivos de salida (E/S)*. Los primeros se ocupan de captar los movimientos de los usuarios en los EVs y de dirigirlos al computador; como parte de estos dispositivos se encuentran los posicionadores (sistemas de tracking), guantes, micrófonos, ratones, andadores, mesas de visualización, entre muchos otros. Los segundos tienen la función de convertir las señales de audio, video y demás derivados del computador en un estímulo, para los sentidos de los usuarios en Ambientes Virtuales (AVs); aquí se pueden identificar dispositivos como los displays basados en proyector (CAVE), plataformas móviles, sistemas de sonido, altavoces, entre otros. Ambos tipos de dispositivos logran la comunicación entre el usuario y los AVs.

Desde una perspectiva más práctica y como parte de los efectos que estos dispositivos pueden provocar en los usuarios, también se pueden catalogar los dispositivos para la RV en de *visión*, de *audición*, y para *interactuar*. Los primeros facilitan una visión similar a la real, a través de displays con una vista estereoscópica. Los segundos emiten sonidos propios de los EVs que dependiendo de punto de ubicación estimulan el sentido del oído de los usuarios. Finalmente, los últimos consideran dispositivos tanto de entrada como de salida, debido a que proporcionan al usuario la interacción con el AV, a través de movimientos que pueden ser rastreados.

Como antecedentes se reconoce que el teclado y el ratón fueron los dispositivos básicos, para la comunicación e interacción entre humanos y computadoras desde la invención

del primer ordenador personal manejado a ratón por Charles Thacker (Wadlow, 1981) considerandos, por lo tanto, los dispositivos que permitieron a los usuarios sumergirse en AVs. Sin embargo, antes el uso del ratón como un dispositivo elemental de interacción, se diseñó y desarrollo hardware para la interacción con ambientes específicos, creados principalmente para proyecto militares, siendo estos dispositivos extensamente caros y no multipropósito.

Sin duda, una característica en común de estos dispositivos de interacción con la computadora y los AVs ha sido la inmersión. Estaban diseñados principalmente para crear la interacción humano-computador tan natural como fuese posible. Esto a su vez, mejoro altamente una de las principales características de los AVs, la cantidad de presencia que evocan en sus usuarios (Meehan, Insko, Whitton & Brooks, 2002).

A través de los años, estos y otros dispositivos han sido incorporados y aceptados de forma masiva entre los usuarios de AVs, para proporcionales la ilusión de presencia física. Por otro lado, las tecnologías de fabricación has evolucionado, permitiendo ser capaces de competir entre fabricantes en términos de calidad, rendimiento y costos.

La mayoría de estos dispositivos, también han avanzado con mejoras en el aspecto de ergonomía. Aspecto importante en el diseño del hardware, el cual se centra en la protección de la salud de los usuarios, desde distintos enfoques tales como: el agrado, la comodidad, la eficiencia, la eficacia, la seguridad, y el bienestar de los usuarios (Ver figura 1). Asimismo, esta evolución ha generado un cambio en la adaptación dentro de un mundo constantemente variable, donde las demandas de los usuarios no se dejaban esperar.



Figura 1 – Dispositivos usados en la Realidad Virtual

Cabe mencionar, algunos dispositivos desarrollados nunca fueron bien recibidos por la audiencia o su tecnología no era apreciada hasta hace algunos años. La rueda de desplazamiento, inventada para la armada británica para manipular una computadora utilizada para atacar aeronaves en 1946 (Engineering and Technology History Wiki, 2015), es un ejemplo de dichas innovaciones, trascendiendo como concepto al hardware de hoy en día como el SpaceTraveler de 3Dconnexion (3DConnexion, 2003) y SpaceNavigator (3DConnexion, 2007), entre otros más productos fabricados por la empresa 3Dconnexion, quien en la actualidad sigue desarrollando tecnología de navegación para AVs 3D y aplicaciones CAD (Diseño Asistido por Computadora) como SpanceNavigator, Cadmouse, SpaceMouse Pro. Dispositivos reconocidos como controladores de movimiento compatibles con varias aplicaciones tales como: Blender, Photoshop, Second Life, Virtual Earth 3D, entre otras (3dconnexion, 2018). Podemos observar otro gran ejemplo en la invención del joystick, creado en 1926 para pilotar aeronaves no tripuladas (NRL review 2011, 2011), pero no fue completamente utilizado hasta 1944 para dirigir cohetes y misiles (Wolf, 1997), dicha tecnología se sigue utilizando hoy en día en una variedad de joysticks utilizados principalmente en juegos.

A través de los años han sido varias las características consideradas en el diseño, desarrollo y uso de los dispositivos de E/S, para interactuar con los AVs, las cuales nos permiten clasificarlos en tres eras: la era Precursora, era Prototipo y la era Del Consumidor.

2. Era Precursora

La era Precursora toma lugar de 1940 a 1968 y se caracterizaba por el inmenso equipo de cómputo utilizado para interactuar con la realidad virtual. Sus costos de producción altamente caros y la prioridad militar y privada de los proyectos son otras características consideradas.

Durante este tiempo la mayoría de dispositivos E/S eran creados con una meta militar. Tal era el caso del joystick, creado con la intención de dirigir misiles a sus objetivos (Wolf, 1997) y manipular aeronaves no tripuladas. Este dispositivo se caracterizaba por ser un controlador en el que un par de controladores expedían ráfagas de señales con un mismo canal de comunicación, a través de un rango de frecuencia de rayos infrarrojos, con una transmisión electromagnética sin pérdida de datos reconocibles (Leung, 1990). Esta tecnología al final de este mismo periodo, fue adaptada para funcionar como un control para una consola de televisión llamada Magnavox Odyssey (Huhtamo, 2007) y después fue producida para una audiencia más amplia.

Otro ejemplo de los dispositivos E/S desarrollados en este periodo es el proyecto Super Cockpit creado por Thomas Furness III (Furness, 1986). El cual era esencialmente una réplica de la cabina de una aeronave pilotada remotamente mediante gestos con las manos. Furness a finales de la década de los 70 ya comenzaba el desarrollo de interfaces virtuales, para tener el mando de vuelos. En este proyecto incorpora la tecnología HDM como simulador de entrenamiento, y agrega además el seguimiento de la cabeza, los ojos, así como el reconocimiento de voz (Enciclopedia Británica, 2019).

Durante esa época un considerable número de proyectos surgieron, aunque la mayoría de ellos eran altamente secretos y no se considera que hayan tenido una especial relevancia.

Sin embargo, todos estos crearon dispositivos de interacción únicos y personalizados. Estos dispositivos E/S intentaron asegurar un alto nivel de inmersión para los soldados involucrados, ayudándolos a interactuar en un ambiente semi-real con el objetivo de ayudarlos en su preparación y entrenamiento para situaciones en la vida real.

Por otra parte, la industria del entretenimiento observó las invenciones de Morton Heilig quien, en 1962, inventó el Sensorama la cual era una máquina en forma de caja con una silla motorizada y una cabina que cubría el rostro del usuario. Era un dispositivo mecánico grande, voluminoso (Arce, 2013), que contaba con una pantalla estereoscópica a color, ventiladores, emisores de olor, sistema de sonido estéreo y poseía instalados cinco cortos cinematográficos sincronizados con el hardware para estimular los sentidos del usuario mediante sus diferentes métodos (Heilig, 1962). Este dispositivo tuvo un alto costo y dificultad para su fabricación, por lo cual no fue puesta para producirse en serie (Martín-Herrera, 2012). Heilig además inventó la Telesphere Mask, una máscara redonda que soportaba imágenes con soporte de visión periférica del 100% y sonido binaural (Heilig, 1998).

No muy lejos, en 1968, Ivan Sutherland y Bob Sproull crearon The Sword of Damocles una inmensa máquina con una pantalla que colgaba del techo debido a que poseía un gran peso, este se colocaba en la cara del usuario. Una computadora utilizaba la posición de la cabeza para computar los elementos del escenario en base a la matriz de rotación y traslación y generar una vista apropiada en relación al punto de vista particular (Sutherland, 1968). Esta máquina también era capaz de rastrear tanto la posición del usuario como la posición de sus ojos a fin de actualizar la vista estereoscópica de acuerdo a la posición del usuario (Boas, 2013).

La mayoría de estos dispositivos requerían una gran parte de un cuarto y un inmenso equipo de cómputo para poder ser utilizados. Sus costos de producción eran alrededor de los cientos de millones de dólares, haciendo prácticamente imposible el ser usado con propósitos personales, con la excepción del joystick (Huhtamo, 2007).

3. Era Prototipo

Prototipo abarca de 1970 hasta 1995 y es caracterizada por equipo de cómputo más accesible, cambiando todos los equipos y máquinas especializadas por computadoras de escritorio y reduciendo los costos de producción permitiendo a las universidades y grandes industrias experimentar e intentar crear sus propios dispositivos E/S, para una interacción con un mayor factor de inmersión.

Durante este período, los beneficios de los ambientes virtuales comenzaron a llamar la atención de diferentes campos como tratamientos médicos, apoyo psicológico, tratamientos de recuperación, diseño de automóviles y videojuegos caseros. Por extensión, esto significó la evolución para los dispositivos de interacción en los ambientes virtuales.

Para algunos diseños y creaciones emergentes se tomó una aproximación médica como el Sayre Glove desarrollado por Thomas DeFanti y Daniel Sandin en la Universidad de Illinois en Chicago, este es un económico y ligero guante para monitorear movimiento de la mano, basado en la idea de Rich Sayre a quien se le acredita el nombre. Estos

se utilizaron como un método efectivo de control multidimensional, además de una interacción más humana dentro de ambientes virtuales, utilizando literalmente sus manos como método primario de interacción (Sturman & Zeltzer, 1994).

Otro dispositivo que hizo su debut en este periodo fue el primer gamepad moderno por la Nintendo TM Company Ltd. para su consola casera en 1983. Poseía un grupo de cuatro botones en forma de cruz utilizados para propósitos direccionales en el AV además de dos botones utilizados como botones de acción. El cual se volvió un estándar para gamepads como dispositivos de interacción en épocas recientes. Combinando los joysticks con sets de botones como el gamepad original de Nintendo creó gamepads que permitieron un movimiento más natural dentro de un AV. Esto facilitó una inmersión más profunda dentro de los AVs.

En 1993 la compañía LogiCad3D TM (ahora 3Dconnexion TM) creó su primer ratón especial utilizado para una mejor navegación dentro de espacios tridimensionales mediante un mango sensible a la presión (LogiCad3D, 2000). Durante este periodo de tiempo múltiples variaciones de los dispositivos de interacción básicos fueron creados incluyendo una versión específica de dispositivos basados en guantes como el Data Glove de VPL TM (Zimmerman, Lanier, Blanchard, Bryson, & Harvill, 1987) o el Power Glove de Mattel TM (Quek, 1994). Las diferentes compañías de videojuegos tenían sus propios diseños y orientaciones de gamepad, y comenzó la aparición de dispositivos volante con propósitos experimentales; las simulaciones y videojuegos de arcade comenzaron a ganar audiencia, etc. Durante estos años la principal mejora para una inmersión más profunda fueron las versiones hápticas de los dispositivos ya mencionados recreando el sentido del tacto mediante la aplicación de fuerza, vibraciones o movimientos al usuario.

Estos dispositivos también observaron una gran mejora en la ergonomía. Los equipos de computación en los cuales estos dispositivos eran conectados se volvieron significativamente más pequeños, pero más poderosos y veloces. Los costos de producción se encontraban alrededor de los millones de dólares, lo cual permitió un precio más accesible a los gamepads y ratones permitiendo su producción y venta al público.

4. Era del Consumidor

La era del consumidor cubre del 2010 hasta el día de hoy y está fuertemente inspirado en la tecnología de percepción electrónica (Roeber, Bacus & Tomasi, 2003), tal es el caso del dispositivo Kinect de Microsoft TM (Microsoft, 2013) y la tecnología Head-Mounted Display o HMD (Blanchard, Burgess, Harvill, Lanier, Lasko, Oberman, & Teitel, 1990). La última de estas tecnologías ofrece con la ayuda de sensores detectar la posición y orientación de objetos reales, esta información es utilizable al resto del sistema. El uso más común de esta información es la monitorización de la cabeza y las manos del usuario. Esta tecnología se ofrece desde dos enfoques: óptico o inercial. El primero capta imágenes, a través de cámaras ópticas con reflectores infrarrojos. El segundo obtiene información relativa de la aceleración ejercida por una masa pequeña que inclina al moverse, a través de dispositivos piezoeléctricos sensibles a los cambios de orientación (Duarte, 2018).

En 2010 uno de los dispositivos de interacción más perceptivos de la historia fue lanzado por Microsoft el Kinect. Este es un dispositivo que posibilita a los usuarios a reconocer e interactuar con una consola sin necesidad de tener contacto físico con un controlador de videojuegos habitual mediante una interfaz natural de usuario que reconoce gestos, comandos de voz, objetos e imágenes (Arce, 2013). La tecnología del Kinect dispone de una cámara con sensores infrarrojos y micrófono multi array capaz de separar voces provenientes de frente al dispositivo de los otros sonidos del ambiente (Microsoft, 2013). Este dispositivo fue creado para videojuegos pertenecientes a la consola casera Xbox 360 pero como resultado de su gran desempeño y mayores posibilidades, se volvió un dispositivo preferido para la interacción en AVs en múltiples campos y diferentes ambientes como por ejemplo rehabilitación física generalizada (Chang, Lange, Zhang, Koenig, Requejo, Somboon & Rizzo, 2012), navegación de Google Earth (Boulos, Blanchard, Walker, Montero, Tripathy & Gutierrez-Osuna, 2011) y detección de caídas en hogares de adultos mayores (Stone & Skubic, 2015).

El año 2016 se vio el lanzamiento de no solo uno sino tres de los más avanzados y ergonómicos dispositivos HMD en la historia. El Oculus Rift TM, HTC Vive TM y PlayStation VR™. Originalmente la tecnología de HMD se caracterizaba por ser dispositivos muy pesados, para la cabeza del usuario, por lo que había que tenerlos colgados del techo del laboratorio. Las nuevas versiones fueron mejorando, logrando con ello alcanzar dispositivos que permite reproducir imágenes creadas por computadora, muy cercanas a las de los ojos humanos (Arce, 2013). Dos de estos dispositivos fueron ya mejorados con controladores de seguimiento de movimiento, el Oculus Touch de Oculus Rift y los controles de HTC Vive, además el PlayStation VR™ fue complementado con los ya existentes PlayStation Move™. Estos dispositivos marcaron una revolución para la interacción con los AVs con grandes niveles de inmersión y accesibilidad haciendo al 2016 el año de la RV.

Estos dispositivos fueron creados para el entrenamiento, pero sus aplicaciones en otros campos van más allá de lo que imaginamos. Estos dispositivos están siendo actualmente estudiados e investigados, siendo aplicados lentamente a diferentes campos alrededor del mundo. Las compañías Oculus y HTC se encuentran actualmente enfocando sus esfuerzos en mejorar su tecnología, de tal manera que se puedan crear dispositivos más económicos y precisos que permitan a la realidad virtual ser accesible, para todos como nunca antes lo ha sido y proveerla de una experiencia completamente inmersiva. Dentro del aspecto de la precisión, se espera alcanzar una mejor resolución para cada ojo, y colocar mejores sensores de posición y orientación.

En esta misma era, otro dispositivo llamado The Leap surge en la empresa Leap Motion, la cual deja ver el alcance del futuro de la RV y de la Realidad Aumentada (RA). Esta tecnología se caracteriza por ser un sensor de movimientos, capaz de capturar el movimiento realizado con las manos de los usuarios. El seguimiento manual se ejecuta en procesadores móviles con baja latencia y alta precisión. El artefacto logra una interacción entre el usuario y el ordenador, a través de gestos haciendo posible una nueva experiencia del usuario, la cual difiere con la práctica tradicional donde la experiencia es, a través del teclado y ratón (LeapMotion, 2018).

Un avance complementario con los cascos de RV, que permite acercar más a los usuarios con los EVs son los andadores de RV. Estos se reconocen por ser como caminadoras de

gimnasio, que permiten a los usuarios desplazarse de forma segura y libre en cualquier dirección, lo que facilita movimientos de 360°. El Virtuix Omni fue la primera opción, con una forma cóncava en donde los pies de los usuarios se apoyan. Los movimientos de los usuarios pueden verse limitados por un arnés de seguridad con el que cuenta, impidiendo que puedan inclinarse. Sin embargo, éste puede evitar cualquier accidente (VirtuixOmni, 2019). Otro andador más ligero fue Cyberith Virtualizer, un nuevo dispositivo con un diseño más ligero. El dispositivo los conforma una alfombra que se desliza, de acuerdo con los pasos que el usuario realiza sobre la superficie en cualquier dirección (Katvr, 2018).

La mayoría de los dispositivos de este periodo requieren solo de una computadora personal. Su rendimiento experimenta una gran mejora cuando son conectados a un equipo de escritorio equipado con una tarjeta gráfica preparada para RV. Además, su costo de producción fue drásticamente reducido permitiendo así que logren alcanzar al público abierto. Con un precio minorista de 249 dólares para el Kinect, 399 dólares para el Oculus Rift, 499 dólares para el HTC Vive y 299 dólares por el PlayStation VR (PlayStation, 2016), haciendo de la realidad virtual una experiencia alcanzable como nunca antes, y logrando una realidad la inmersión profunda para muchas personas.

En la Figura 1 se presenta una línea de tiempo que cubre las tres eras descritas. En esta Figura 1 se puede observar las invenciones ya mencionadas localizadas en su respectivo período de tiempo, con la excepción de la tecnología de percepción electrónica, debido a que su invención fue previa a su desarrollo llevado a cabo varios años después.

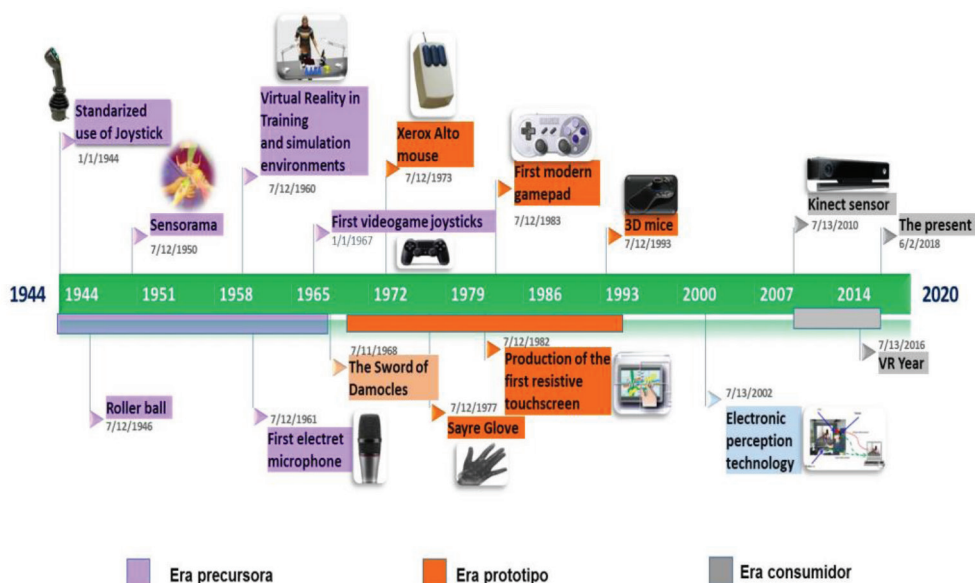


Figura 2 – Eras de los dispositivos de interacción para ambientes virtuales

La Figura 2 muestra los predecesores de las tecnologías mostradas en la Figura 1, viajando a través de la historia para localizar como cada tecnología evoluciono desde su invención.

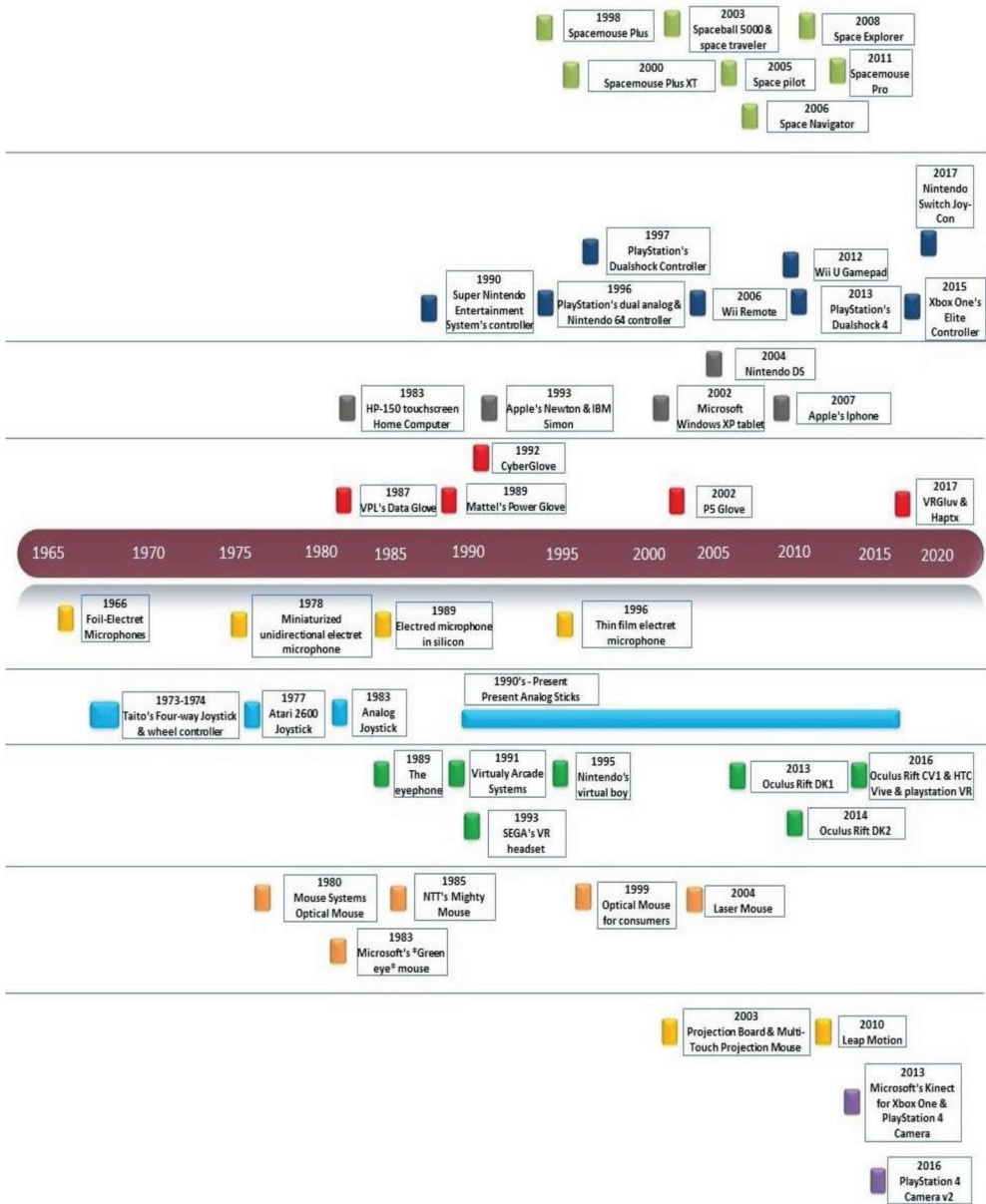


Figura 3 – Línea de tiempo de las tecnologías descritas, mostrando su desarrollo a través de los años

5. Conclusiones

Para poder ser parte de la Realidad Virtual con un buen nivel de inmersión, no es suficiente disponer de una herramienta de software, para ello es indispensable contar con un conjunto de dispositivos clave que permitan introducirse en los AVs. Sin duda, el avance de los dispositivos de interacción para los AVs está unido a los avances tecnológicos en el hardware, lo cual ha permitido mayores niveles de inmersión. Esto, expandió la ya abundante posible aplicación de la realidad virtual y otros ambientes virtuales.

Son las aplicaciones de software las encargadas de generar y enviar la información necesaria hacia el equipo de cómputo desde los AVs, pero para llevar a cabo este proceso se requiere un equipamiento adecuado conformado por varios elementos, para montar un sistema completo de Realidad Virtual.

En este trabajo, se identificaron tres épocas utilizando factores de clasificación como los costos de producción y las metas de los dispositivos de interacción.

Para la primera era, puede observarse fácilmente que se requerían una inmensa cantidad de investigación y recursos para desarrollar los diferentes dispositivos de interacción y la mayoría poseían un objetivo de entrenamiento y/o misión militar o aeronáutica.

La determinada segunda era puede caracterizarse por el alto costo requerido para la fabricación de dispositivos de interacción, requiriendo así que múltiples compañías desarrollan sus propios dispositivos, algunos de ellos sirvieron como estándares para múltiples dispositivos en el futuro. Aquí, los objetivos de los AV y sus dispositivos de interacción comenzaron a explorar diferentes campos y fueron exitosamente aplicados a los mismos.

En lo que respecta a la era del Consumidor, puede observarse a los diferentes dispositivos de interacción encontrados en la era anterior, propiamente establecidos y estandarizados, suficiente para establecer un mercado alrededor de ellos, lo cual sirve de evidencia de la reducción drástica en los costos de producción al punto de permitir la existencia de equipos caseros para la realidad virtual con altos niveles de inmersión al alcance de la mano.

En resumen, dentro de este estudio también se analizó, que la evolución tecnológica del hardware para la RV, ha mejorado a través de los años desde diferentes aspectos tales como: rendimiento, costo, ergonomía, resolución, entre otros, llegando a despertar el interés de los fabricantes y de los usuarios. Son muchas las novedades que podríamos esperar a futuro de la Realidad Virtual.

Referencias

- Islas, E., Zabre, E., & Pérez, M. (2004). Evaluación de herramientas de hardware y software para el desarrollo de aplicaciones de realidad virtual. *Bulletin Electrical Research Institute*, 28, 61–67.
- Arango, J. E., Mazo, J. C., & Palacio, A. P. (2013). Sistema para rehabilitación del síndrome del miembro fantasma utilizando interfaz cerebro-computador y realidad aumentada. *RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação*, (11), 93–106.

- Wadlow, T. A. (1981). The Xerox alto computer. *Byte Magazine*, 6(9), 58–68.
- Meehan, M., Insko, B., Whitton, M., & Brooks Jr, F. P. (2002). Physiological measures of presence in stressful virtual environments. *ACM Transactions on Graphics (TOG)*, 21(3), 645–652.
- Engineering and Technology History Wiki (2015), Ralph Benjamin, an oral history conducted in 2005 by Peter Hill. Hoboken, NJ, USA: IEEE History Center. Retrieved from: https://ethw.org/Oral-History:Ralph_Benjamin.
- Naval Research Laboratory (2011). NRL review 2011. Washington, DC: NRL. Retrieved from: https://www.nrl.navy.mil/content_images/2011_NRL_Review.revised.pdf.
- Wolf, W. (1997). German Guided Missiles: Henschel HS 293 and Ruhrstahl SD 1400X, (Vol. 53). Merriam Press.
- Leung, A. M. (1990). U.S. Patent No. 4,924,216. Washington, DC: U.S. Patent and Trademark Office.
- Huhtamo, E. (2007). Slots of fun, slots of trouble. *Artnodes*, 7.
- Furness III, T. A. (1986). The super cockpit and its human factors challenges. In: *Proceedings of the Human Factors Society Annual Meeting* (Vol. 30, No. 1, pp. 48–52). Los Angeles, CA: SAGE Publications.
- Enciclopedia Británica. (2019). Retrieved from: <https://www.britannica.com/technology/virtual-reality>.
- Arce, C. (2013). Realidad Aumentada. Retrieved from: <http://jeuazarru.com/wpcontent/uploads/2014/10/RA2013.pdf>.
- Heilig, M. L. (1962). U.S. Patent No. 3,050,870. Washington, DC: U.S. Patent and Trademark Office.
- Heilig, M. (1998). Beginnings: sensorama and the telesphere mask. In: *Digital illusion* (pp. 343–351). ACM Press/Addison-Wesley Publishing Co.
- Martín-Herrera, B. (2012). El móvil en la educación: un nuevo paradigma. Ventajas y desventajas de su uso.
- Sutherland, I. E. (1965). The ultimate display. *Multimedia: From Wagner to virtual reality*, (pp. 506–508).
- Roeber, H., Bacus, J., & Tomasi, C. (2003). Typing in thin air: the canesta projection keyboard—a new method of interaction with electronic devices. In: *CHI’03 extended abstracts on Human factors in computing systems* (pp. 712–713). ACM.
- Chang, C. Y., Lange, B., Zhang, M., Koenig, S., Requejo, P., Somboon, N., ... & Rizzo, A. A. (2012). Towards pervasive physical rehabilitation using Microsoft Kinect. In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth)*, 2012 6th International Conference on (pp. 159–162). IEEE.
- Boulos, M. N. K., Blanchard, B. J., Walker, C., Montero, J., Tripathy, A., & Gutierrez-Osuna, R. (2011). Web GIS in practice X: a Microsoft Kinect natural user interface for Google Earth navigation.

- Stone, E. E., & Skubic, M. (2015). Fall detection in homes of older adults using the Microsoft Kinect. *IEEE journal of biomedical and health informatics*, 19(1), 290–301.
- Boas, Y. A. G. V. (2013). Overview of virtual reality technologies. In: *Interactive Multimedia Conference* (Vol. 2013).
- Sutherland, I. E. (1968). A head-mounted three-dimensional display. In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part I* (pp. 757-764). ACM.
- Sturman, D. J., & Zeltzer, D. (1994). A survey of glove-based input. *IEEE Computer graphics and Applications*, 14(1), 30–39.
- Duarte, M., C. (2018). Hacia el espacio. Retrieved from: <https://haciaelespacio.aem.gob.mx/revistadigital/articul.php?interior=796>.
- VirtuixOmni. (2019). Retrieved from: <http://www.virtuix.com/>.
- Katvr. (2018). Retrieved from: <http://katvr.com/about/>.
- LeapMotion. (2019). Retrieved from: <https://developer.leapmotion.com/#101>.
- Blanchard, C., Burgess, S., Harvill, Y., Lanier, J., Lasko, A., Oberman, M., & Teitel, M. (1990, February). Reality built for two: a virtual reality tool. *ACM SIGGRAPH Computer Graphics* 24(2), 35–36.
- Wolf, M. J. (1997). Inventing space: Toward a taxonomy of on-and off-screen space in video games. *Film Quarterly* (ARCHIVE), 51(1), 11.
- Zimmerman, T. G., Lanier, J., Blanchard, C., Bryson, S., & Harvill, Y. (1987). A hand gesture interface device. *ACM SIGCHI Bulletin* 18(4), 189–192.
- Quek, F. K. (1994). Toward a vision-based hand gesture interface. In: *Virtual reality software and technology* (pp. 17-31).
- LogiCad3D. (2000). *Magellan/SpaceMouse Classic/Plus/Plus XT.: User's Manual*. Seefeld, Germany: LogiCad3D.
- 3DConnexion. (2003). *SpaceTraveler™ Small size. Big performance: User's Manual*. Seefeld, Germany: 3DConnexion. Retrieved from: http://www.surface3d.com.tw/spacetraveler/SpaceTraveler_En.pdf.
- 3DConnexion. (2007). *SpaceExplorer™ Navigate Your 3D World™: User's Manual*. Seefeld, Germany: 3DConnexion. Retrieved from: http://www.crystaltech.ca/Pdfs/SpaceNavigator_Datasheet.pdf.
- 3DConnexion. (2018). Retrieved from: <https://www.3dconnexion.com/service/drivers.html>.
- Microsoft. (2013). *Xbox one and kinect sensor product manual: Product's Manual*. Redmond, WA, USA: Microsoft.
- Sony. PlayStation. (2016). *PlayStation® Camera: Instruction Manual*. Zeventem, Belgium: Sony. Retrieved from: <https://www.playstation.com/es-mx>.

Una aplicación móvil para el monitoreo de cultivos: caso de estudio campaña contra el pulgón amarillo del sorgo

Alejandra Perez-Mena¹, José Alberto Fernández-Zepeda¹, Juan Pablo Rivera-Caicedo², Himer Avila-George³

perezmena@cicese.edu.mx, fernan@cicese.mx, jprivera@uan.edu.mx, himer.avila@academicos.udg.mx

¹ Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, 22860, Baja California, México.

² CONACYT-UAN, Centro Nayarita de Innovación y Transferencia Tecnológica, Tepic, 63173, Nayarit, México.

³ Centro Universitario de los Valles, Universidad de Guadalajara, Ameca, 46600, Jalisco, México.

DOI: 10.17013/risti.31.118-133

Resumen: En este artículo, se introduce PulAm, una aplicación móvil basada en el sistema operativo móvil Android. Esta aplicación se diseñó como una herramienta de apoyo en el proceso de monitoreo de plagas de diferentes cultivos. Como caso de estudio, se introduce el monitoreo de cultivos de sorgo, en el estado de Nayarit, México; específicamente para la campaña contra el “pulgón amarillo” del sorgo.

Palabras-clave: aplicación móvil; sorgo; plaga del pulgón amarillo.

A mobile application for crop monitoring: case study campaign against the yellow sorghum aphid

Abstract: In this manuscript, PulAm is presented, a mobile application based on the Android mobile operating system. This application was designed as a support tool in the process of monitoring pests of different crops. As a case study, its use is introduced in the monitoring of sorghum crops in the state of Nayarit, Mexico; Specifically, for the campaign against the “yellow aphid” of sorghum.

Keywords: mobile app; sorghum; yellow aphid.

1. Introducción

La agricultura es una actividad primaria y de gran relevancia para el ser humano desde el punto de vista social, económico y ambiental. Lo anterior debido a que genera un gran número de trabajos y productos alimenticios; además, ayuda a conservar el suelo y la biodiversidad.

El sorgo es el segundo cultivo más importante para la economía mexicana. La superficie destinada para la producción de sorgo es 1,720,208 hectáreas (SIAP, 2017). Según el Marco Censal Agropecuario, a excepción de la Ciudad de México, todos los estados del país tienen un porcentaje de superficie destinada al cultivo del sorgo (INEGI, 2016).

En la Figura 1 se puede apreciar un mapa de México donde se resalta en tonos naranja las zonas donde se cultiva el sorgo (el mapa se generó con la herramienta mapa digital de México (INEGI, 2019)), donde los estados de Tamaulipas y Sinaloa son los mayores productores de sorgo (SAGARPA, 2014).

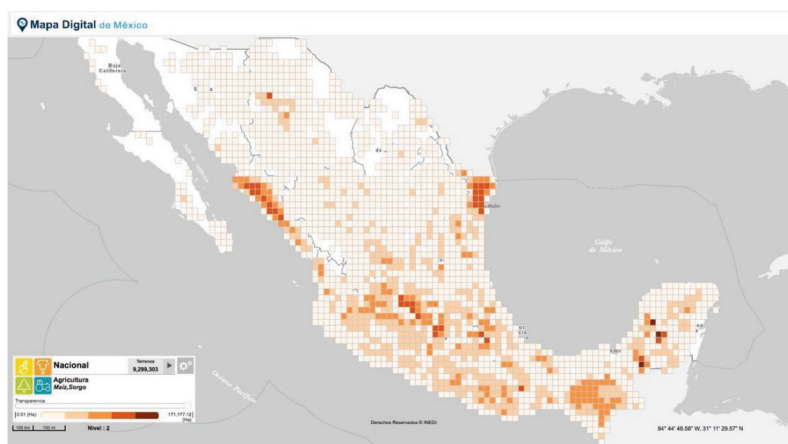


Figura 1 – Zonas de cultivos de sorgo en México

En el año 2013 apareció por primera vez en el territorio mexicano el virus del pulgón amarillo (*Melanaphis sacchari*), dañando severamente los cultivos de sorgo. Desde entonces, la producción de sorgo ha decrecido entre 30% y 100% en muchas áreas (Rodríguez-del-Bosque & Terán, 2015). Esta plaga se ha establecido en prácticamente todo el territorio nacional debido a su alta capacidad de dispersión y elevado potencial de reproducción.

El pulgón amarillo es un insecto clasificado en la familia *Aphididae*, dicho insecto tiene hábitos gregarios, i.e., vive en colonias abundantes, específicamente en el lado posterior de las hojas de las plantas invadidas. Inicialmente, las colonias de pulgón amarillo aparecen en la base de la planta del sorgo y gradualmente se mueve a la parte alta, en donde pueden infestar la panícula del sorgo en floración (Rodríguez-del-Bosque & Terán, 2015).

Una planta de sorgo con la plaga del pulgón amarillo muestra un cambio de coloración en la hoja, ya que este insecto se alimenta de la savia de la planta. Este proceso causa pérdida de nutrientes y estrés en la planta, ver Figura 2. El insecto también secreta una sustancia azucarada en la superficie de las hojas que propicia la aparición de moho y promueve la expansión de algunos virus como el *Sugarcane mosaic (ScMV)* o *Sugarcane yellow leaf (ScYLV)*. Sin embargo, el daño más crítico en el cultivo es la reducción de la calidad del cultivo y del rendimiento de la cosecha (SENASICA, 2014).



Figura 2 – Hojas de sorgo infestadas por pulgón amarillo

Dada la importancia económica del cultivo del sorgo en México y los daños significativos causados por la plaga del pulgón amarillo, en 2015, el Servicio Nacional de Sanidad, Inocuidad y Calidad Agroalimentaria (SENASICA, 2014), a través de los Comités Estatales de Salud vegetal, autorizaron la campaña en contra del pulgón amarillo en sorgo. Esta campaña incluye acciones como el monitoreo, aplicación de control biológico, químico y cultural con énfasis en el entrenamiento para los productores.

En Nayarit, el Comité Estatal de Sanidad Vegetal (CESAVENAY) es responsable de la implementación de esta campaña. Las actividades llevadas a cabo por los técnicos del CESAVENAY durante estas inspecciones son las siguientes:

1. Se realiza la evaluación de la presencia de la plaga en el cultivo, si es el caso, se determina el nivel de infestación.
2. Se recolecta información relacionada con la parcela y el cultivo (fecha de siembra, etapa fenológica (ver Tabla 1), nivel de daño e información geográfica).
3. Se capacita a los productores de sorgo acerca del manejo del cultivo. Dicha capacitación se enfoca en la prevención y tratamiento de la plaga.

Los técnicos del CESAVENAY recolectan manualmente toda la información relacionada con el proceso de monitoreo de cultivos de sorgo en bitácoras. Después, capturan esta información en una plataforma web. Este proceso ha causado inconsistencias en la integridad de la información, principalmente en el registro de las coordenadas geográficas de la parcela evaluada, las etapas fenológicas del cultivo y los niveles de infestación reportados. Dichos problemas podrían ser atenuados si se automatiza el proceso de levantamiento de información haciendo uso de las nuevas tecnologías.

Etapas fenológicas		
Nombre según Vanderlip and Reeves (1972)	Nombre según CESAVERNAY	No. de Días
Siembra	Siembra	1
Emergencia	Crecimiento	20
Collar de tercera hoja visible	Desarrollo vegetativo	30
Collar de quinta hoja visible	Elongación de tallo	5
Diferenciación de punto de crecimiento	Floración	6
Última hoja en espiral visible	Elongación de panícula	8
Arranque de fruto	Formación del fruto	10
Media floración	Desarrollo de panícula	10
Masa blanda	Grano lechoso	10
Masa dura	Grano masoso	15
Madurez fisiológica	Madurez fisiológica	25
Cosecha	Finalización del periodo de cosecha	1
Total		141

Tabal 1 – Etapas fenológicas del sorgo

Las nuevas tecnologías y paradigmas computacionales pueden ser empleados en diversos campos (Quezada-Sarmiento & Mengual Andrés 2017), entre ellos la agricultura; el uso y manejo de dispositivos móviles constituyen una de las opciones que permiten agilizar el trabajo en diversas áreas (Razo Salas 2016).

Previamente, algunos investigadores han llevado a cabo esfuerzos de investigación relacionados con la automatización del proceso de monitoreo de cultivos y levantamiento de información. En algunos de estos trabajos se han desarrollado aplicaciones móviles; como el *FarmBeats* (Vasisht, y otros, 2017). Esta aplicación es parte de un sistema informático a través del cual el usuario puede visualizar los resultados del análisis de suelo y variables de salud en un cultivo. Este sistema contempla escenarios donde no hay conexión de Internet. El trabajo de Liopa-Tsakalidi, et al. (2013) describe un sistema que incluye diferentes herramientas (servicios web, una aplicación móvil basada en Android y una base de datos). Con esta aplicación móvil, y a través de sensores, el productor puede añadir información relacionada con el cultivo a una base de datos. Esta aplicación ayuda a tomar decisiones relacionadas con el desarrollo del cultivo.

En el presente trabajo proponemos una aplicación móvil llamada PulAm, la cual tiene como finalidad la automatización del monitoreo de diferentes cultivos, principalmente durante la captura de información de campo. Con esta aplicación, un técnico puede identificar parcelas, capturar automáticamente su información geográfica, y basado en su experiencia, determinar el nivel de infestación de un cultivo. Finalmente, como caso de estudio, proponemos el monitoreo de cultivos de sorgo afectados por la plaga del pulgón amarillo en ciertas regiones de Nayarit, México. Cabe mencionar que este trabajo es una extensión del trabajo presentado por Pérez-Mena et al. (2018) en el Congreso Internacional de Mejora de Procesos de Software 2018.

2. Área de estudio y proceso tradicional de monitoreo

El área de estudio seleccionada es el estado mexicano de Nayarit, ver Figura 3, siendo éste el quinto productor de sorgo a nivel nacional. En Nayarit, el cultivo de sorgo es el segundo en orden de importancia en cuanto a la superficie sembrada. Anualmente se siembran más de 63,000.00 hectáreas de este grano en condiciones de humedad residual y riego. En agosto de 2014, la plaga del pulgón amarillo atacó por primera vez a los cultivos de sorgo en Nayarit, específicamente en el municipio de Amatlán de las cañas (CESAVENAY, 2015).

El Comité de Sanidad Vegetal de Nayarit (CESAVENAY) tiene un equipo de técnicos que se encargan del monitoreo de la plaga del pulgón amarillo en cultivos de sorgo. Planifican semanalmente jornadas de inspección en diferentes zonas cultivadas de sorgo. Hacen este diseño de acuerdo con los datos históricos de los cultivos inspeccionados, la presencia de la plaga y su grado de infestación.

En un día típico de inspección, el técnico llega en la mañana a su zona de inspección y distribuye control químico y/o biológico a los productores de sorgo de acuerdo con el nivel de infestación en su parcela. Después, los técnicos inspeccionan los cultivos de acuerdo con su planeación. Durante esta actividad, los técnicos registran nuevas parcelas si los productores lo requieren y evalúan el grado de infestación muestreando un cierto número de plantas en una parcela. Después registran en sus bitácoras la siguiente información: coordenadas del punto de inspección (obtenidas con un GPS), nombre de la plaga, nivel de infestación, número de plantas infestadas, etapa fenológica del cultivo inspeccionado, variedad de sorgo, fecha de inspección y otros datos¹. Finalmente, el

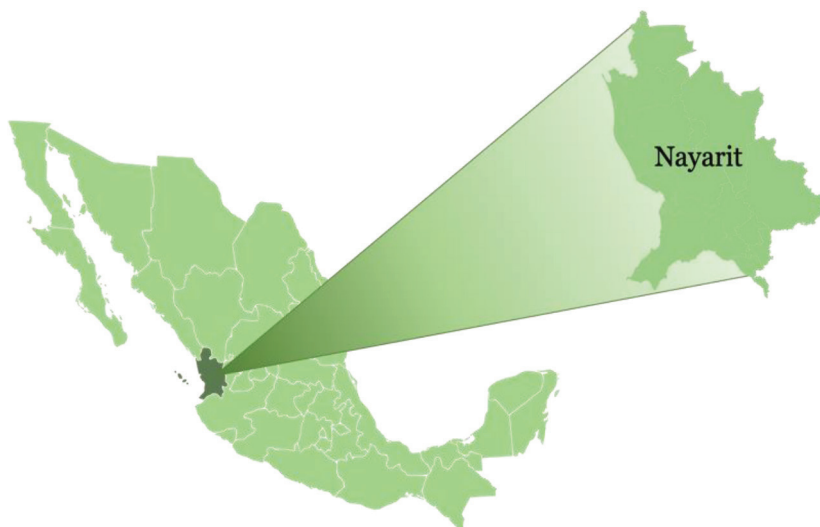


Figura 3 – Área de estudio

¹ Para realizar este estudio, personal del CESAVENAY proporcionó las bitácoras de la campaña del pulgón amarillo en Nayarit correspondientes a los años 2016 al 2018

técnico ingresa manualmente la información obtenida durante su jornada a una plataforma Web propiedad del CESA VENAY.

Los problemas más frecuentes al realizar este proceso son los siguientes:

- Los técnicos pueden incurrir en errores durante el proceso de captura de la información de las inspecciones en sus bitácoras y después a la plataforma computacional donde almacenan esta información. Estos errores pueden ser de escritura o de inexactitud de datos.
- Los técnicos no siempre registran las nuevas coordenadas del punto de inspección, ya que éste a veces lo asocian a las coordenadas de un punto previamente obtenido de la parcela, no reflejando el punto exacto de la inspección.
- No hay forma de comprobar que los técnicos llevaron a cabo las inspecciones registradas en la bitácora.
- Los técnicos no tienen ninguna forma rápida para consultar el historial de información antes recabada.
- No hay suficientes técnicos para cuantificar los daños de todos los cultivos de sorgo.

3. Propuesta

En este trabajo, se introduce una aplicación móvil llamada PulAm para mitigar los problemas descritos en las secciones anteriores y para automatizar el proceso de monitoreo de los cultivos. Esta sección describe el objetivo de esta aplicación, su descripción técnica y sus componentes principales.

3.1. Objetivos de *PulAm*

El propósito principal de PulAm es dar soporte al personal técnico que se encarga de capturar toda la información relacionada con las inspecciones de campo, sin preocuparse de los detalles técnicos del proceso, como validaciones geográficas e integridad de los datos. PulAm tiene los siguientes objetivos particulares:

- Permitir la captura automática y confiable de la información de inspección.
- Mantener actualizada la información de las inspecciones para poder clasificar los daños causados por las plagas con información reciente. Esta actualización es automática cuando el dispositivo móvil, donde se encuentre instalada la aplicación, se conecte a Internet.
- Evitar la generación de errores al capturar información de la inspección.
- Consultar y editar la información relacionada con las inspecciones.

3.2. Descripción técnica de *PulAm*

PulAm se desarrolló para la versión de Android 4.0 (KitKat). El dispositivo móvil en el que PulAm se ejecute debe tener un GPS y conexión a Internet. La aplicación móvil necesita conexión a Internet porque debe conectarse a un servidor web para actualizar la información modificada, eliminada y agregada en la base de datos. En la Figura 4, se muestra la arquitectura de PulAm. Además de esta aplicación, el dispositivo Android hospeda una base de datos SQLite, de la que PulAm obtiene su información. El dispositivo

se conecta al servidor a través de Internet y un servicio PHP para actualizar la información modificada o añadida a la base de datos MySQL descrita en la Sección 3.3.

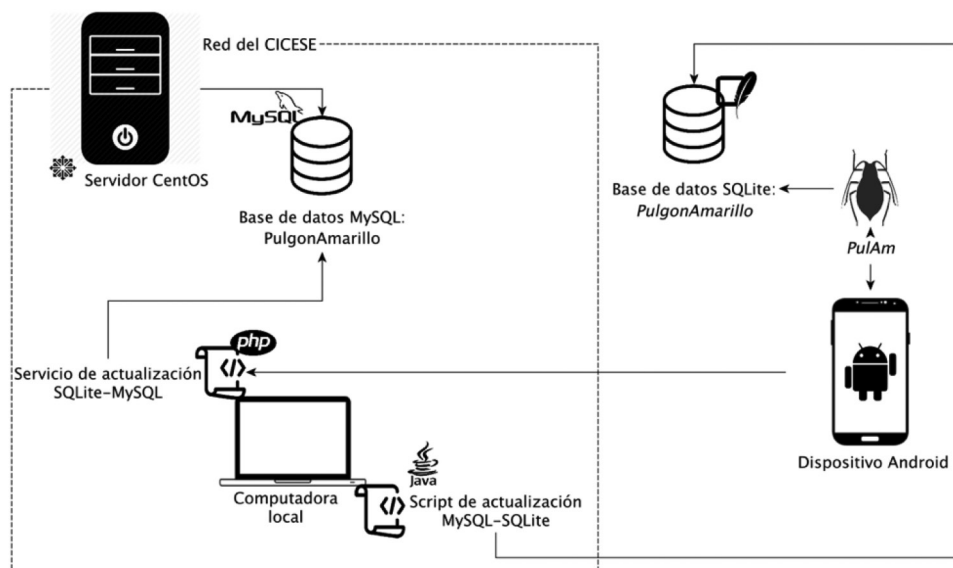


Figura 4 – Arquitectura de PulAm

Para el desarrollo de esta aplicación se utilizaron las siguientes librerías con su respectiva función en la aplicación:

- Librería Google maps. Se utiliza para desplegar los mapas de las capas de cultivos e inspecciones. Además, se usan funcionalidades de ubicación actual y obtención de coordenadas.
- Librería sqlite. Maneja las conexiones a la base de datos SQLite hospedada en el dispositivo Android.
- Librería volley. Maneja la conexión con el servicio web PHP para la sincronización de las bases de datos.

3.3. Módulos de PulAm

PulAm consiste de cinco módulos: (1) administración de la base de datos; (2) verificación de predios; (3) captura de la inspección; (4) personalización y (5) sincronización de la base de datos.

3.3.1. Administración de la base de datos

A partir de las bitácoras de la campaña del pulgón amarillo proporcionadas por el CESAVENAY, se extrae la información necesaria para crear la base de datos del proyecto, para lo cual se creó en el servidor una base de datos en MySQL, véase la Figura 4.

Para la transmisión de esta información a la base de datos MySQL, se generó un programa en Java para agregar y validar de las inspecciones en las bitácoras. En la Figura 5, se pueden ver las tablas utilizadas en el proceso de almacenamiento de inspecciones. PulAm cuenta con una base de datos SQLite hospedada en el dispositivo móvil.

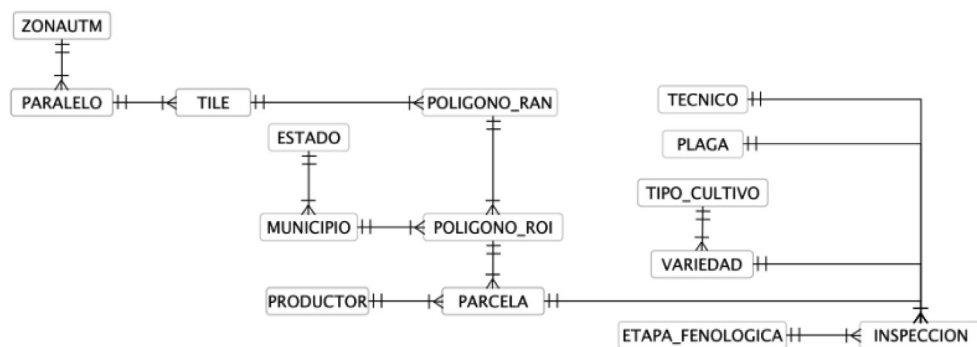


Figura 5 – Diagrama de la base de datos

En principio, los registros de inspección de las bitácoras no contienen información sobre la localización de las parcelas en donde se realizaron las inspecciones. Por lo que para obtener su información geográfica se siguieron los siguientes pasos:

1. Se obtuvo la información geográfica de las regiones agrarias de Nayarit a través del Registro Agrario Nacional (polígonos RAN). Las regiones agrarias son zonas de cultivos de especies vegetales.
2. Se obtuvieron las coordenadas de inspección en formato shapefile de puntos.
3. Se conjuntaron la capa de coordenadas y la capa de polígonos RAN para obtener únicamente los polígonos que contienen inspecciones.
4. Se hizo la sobreposición de la imagen del polígono RAN con la capa de coordenadas de inspecciones con la ayuda del software QGIS.
5. Por cada inspección, se dibujó el polígono de la región de interés de la parcela o polígono ROI, guardándose en un shapefile.

El módulo de manejo de la base de datos es responsable de la consulta, edición, eliminación y adición de información en las siguientes tablas de la base de datos: inspección, plaga, tipo de cultivo, productor, técnico y cultivos (ver Figura 6 (a)-(f)). El manejo de cada tabla tiene restricciones relacionadas con la prioridad de la información y cómo la modificación afecta a otras tablas.

Cabe destacar que el técnico no puede agregar inspecciones desde esta sección, sólo las puede consultar y editar. Con el fin de minimizar problemas de conectividad, se implementó una base de datos local en el dispositivo móvil en lugar de un servicio web; lo cual permite que la aplicación funcione incluso donde no hay señal de Internet.

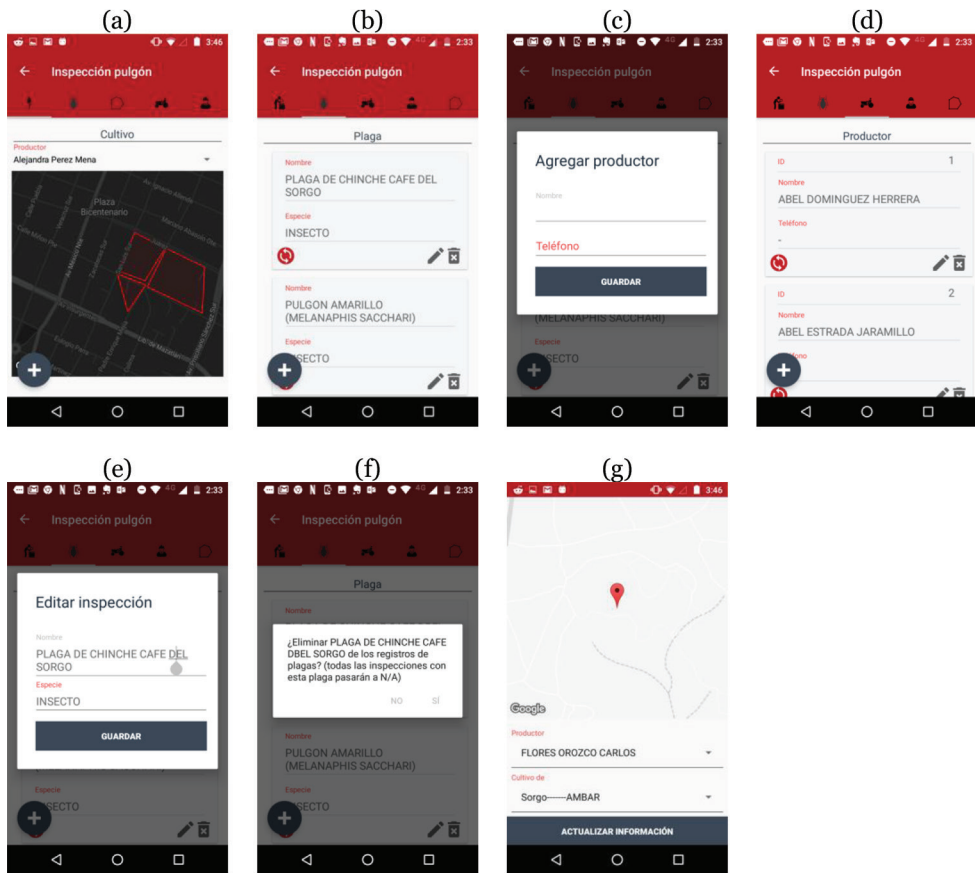


Figura 6 – Módulo de administración de base de datos y verificación de predio

3.3.2. Verificación de cultivos

En el módulo de verificación de cultivos, el usuario puede seleccionar a quién le pertenece una parcela. Además, muestra quién se encuentra actualmente trabajando en la parcela y qué tipo de cultivo está sembrado. Se implementó este módulo debido a que existen predios cuyo productor actual no corresponde al registrado en la base de datos (ver Figura 6 (g)).

3.3.3. Captura de la inspección

En el módulo de captura de inspección (ver Figura 7), el técnico puede capturar el nivel de infestación de la plaga en el cultivo en el que se encuentre en ese instante. PulAm automáticamente captura las coordenadas, fecha y hora de la inspección, y completa el registro de inspección con la información de la última inspección añadida. El técnico

puede modificar la información de la inspección que está por añadir, con excepción de las coordenadas, fecha y hora de la inspección.

El módulo de captura de la inspección consiste de tres pantallas: (a) el mapa (donde se visualiza la ubicación del técnico en el cultivo); (b) la adición rápida de la inspección (donde si el usuario ingresa el nivel de infestación y da clic en Guardar, la aplicación almacena la información restante de la última inspección añadida); y (c) la pantalla donde el técnico puede modificar la información de la inspección capturada. En caso de que el usuario esté ubicado en un punto donde no exista un cultivo registrado en la base de datos, debe crear un nuevo cultivo para agregar una inspección. Esto se hace por medio de la sección de administración de la base de datos.

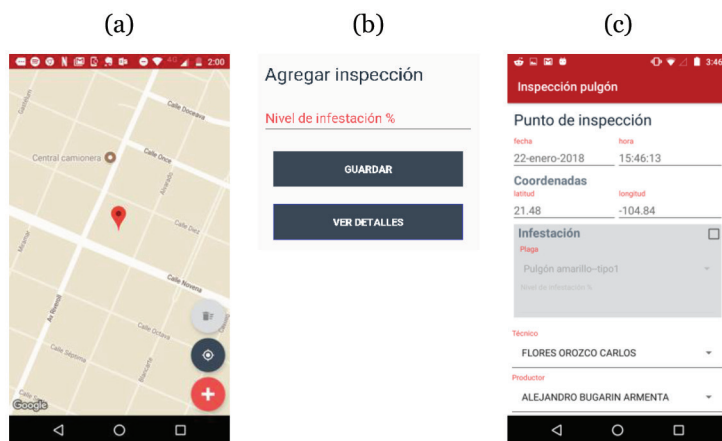


Figura 7 – Módulo captura de inspección

3.3.4. Personalización de PulAM

PulAm permite a los usuarios personalizar las siguientes características: tipo de mapa usado (por defecto el mapa cambia de color de acuerdo con el momento del día) y el técnico que usa la aplicación (ver Figura 8).

3.3.5. Sincronización de la base de datos

El módulo de sincronización de la base de datos es responsable de la sincronización de todos los registros eliminados, editados y añadidos por medio de PulAm cuando el dispositivo se conecta a Internet. Cada vez que un usuario cambia algo en la base de datos, la consulta entra a una cola de consultas. Una vez que el dispositivo se conecta a Internet, este módulo manda la información al servidor mediante el protocolo HTTP que captura las consultas con un servicio PHP.

Finalmente, el servidor ejecuta las consultas en la base de datos MySQL. Para la conversión de la base de datos MySQL a SQLite se pueden utilizar distintos programas de conversión; sin embargo, éstos requieren supervisión o tienen un costo por uso, por lo que se generó un programa en Java que convierte la base de datos de MySQL a SQLite antes de la generación en producción de la aplicación. Además, este módulo agrega nuevos

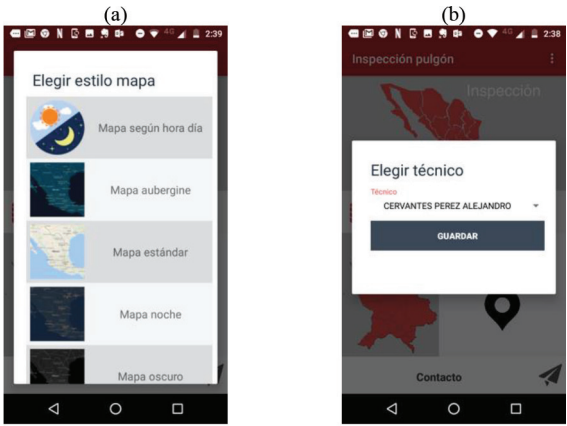


Figura 8 – Módulo de personalización

datos de clasificación en la base de datos. No se implementó una base de datos MySQL en el dispositivo móvil debido a que demanda muchos recursos computacionales en comparación con una base de datos SQLite. Este módulo es útil cuando no hay conexión a Internet disponible; la información permanece en la cola hasta que se puede sincronizar.

4. Inspecciones a cultivos de Nayarit y pruebas funcionales de la aplicación

Para probar PulAm, llevamos a cabo una campaña de monitoreo, la cual consistió en visitar 40 plantaciones de sorgo. Para realizar esta actividad se solicitó al CESAVENAY copia de las bitácoras de las inspecciones a cultivos de sorgo y que el primer autor de este



Figura 9 – Campaña de monitoreo de campos de sorgo

trabajo pudiera estar presente en las campañas fitosanitarias del pulgón amarillo, ver Figura 9. Las inspecciones se llevaron a cabo en los municipios de nayaritas de Rosamorada, Santiago Ixcuintla y Tuxpan.

Durante la fase de inspección a cultivos, se recibió capacitación para reconocer el grado de infestación en el cultivo y en las plantas. Para evaluar el daño, se eligen plantas al azar y por planta se hace un estimado del número de pulgones que existen en ella. Como resultado, se obtuvo la información de cultivos de sorgo en cinco de sus etapas fenológicas: madurez fisiológica, cosecha, grano lechoso, grano masoso y panoja cerrada.

5. Resultados

Se obtuvo como producto una base de datos llamada *pulgonamarillo* la cual almacena información histórica de inspecciones (8411 inspecciones). En la Tabla 2, se muestra el total de inspecciones por año.

Año	Inspecciones
2016	3717
2017	4085
2018	610
Total	8412

Tabal 2 – Inspecciones

En la Figura 10, se observa la cantidad de inspecciones por etapa fenológica que existe en la base de datos. La etapa fenológica con más inspecciones registradas es la de cosecha,

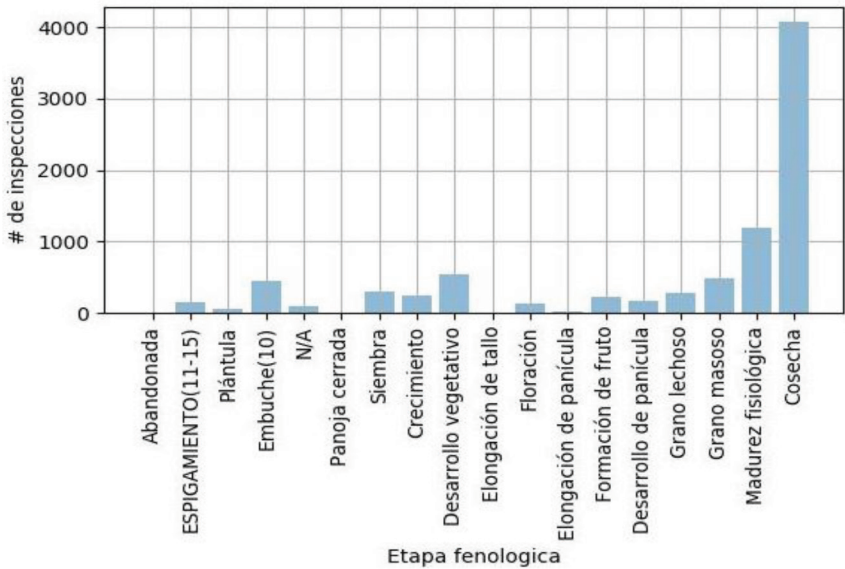


Figura 10 – Inspecciones por etapa fenológica

seguida por la de madurez fisiológica. Esto se debe a que los técnicos del CESAVERNAY marcan como cosecha toda aquella parcela en la que no hay nada cultivado.

En la Figura 11, se muestran los histogramas de las inspecciones por nivel de infestación obtenidas durante el llenado de la base de datos. En estos histogramas se puede ver cuál es la distribución de los niveles de infestación en las diferentes etapas fenológicas del cultivo del sorgo.

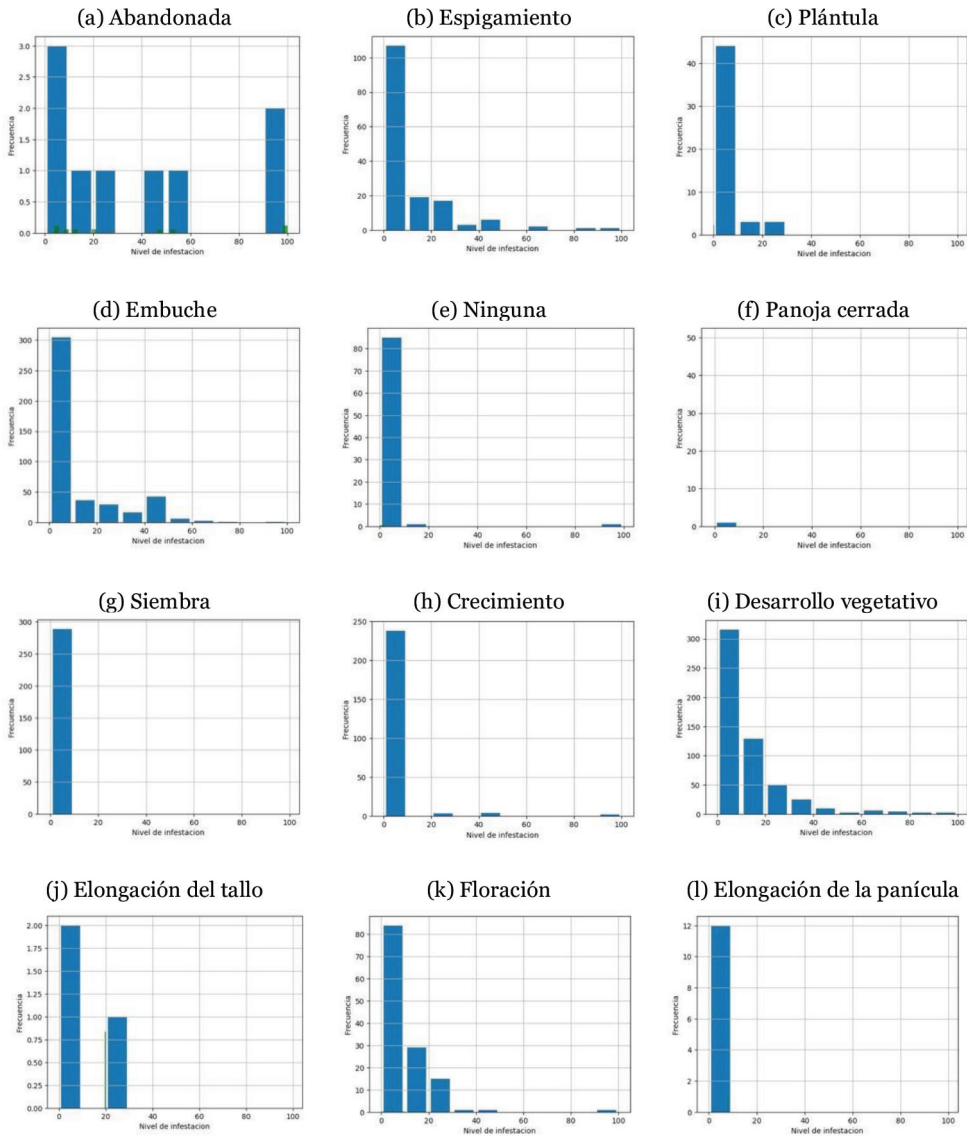


Figura 11 – Histogramas por etapa fenológica

Del análisis de estos datos se obtiene lo siguiente:

- En la mayoría de las inspecciones, el nivel de infestación es 0%, excepto en la etapa fenológica de formación de fruto, donde la mayoría de las inspecciones reportan niveles del 20 al 30%.
- Las inspecciones con una distribución más variada de niveles de infestación son las de abandonada, embuche, desarrollo vegetativo y formación de fruto.

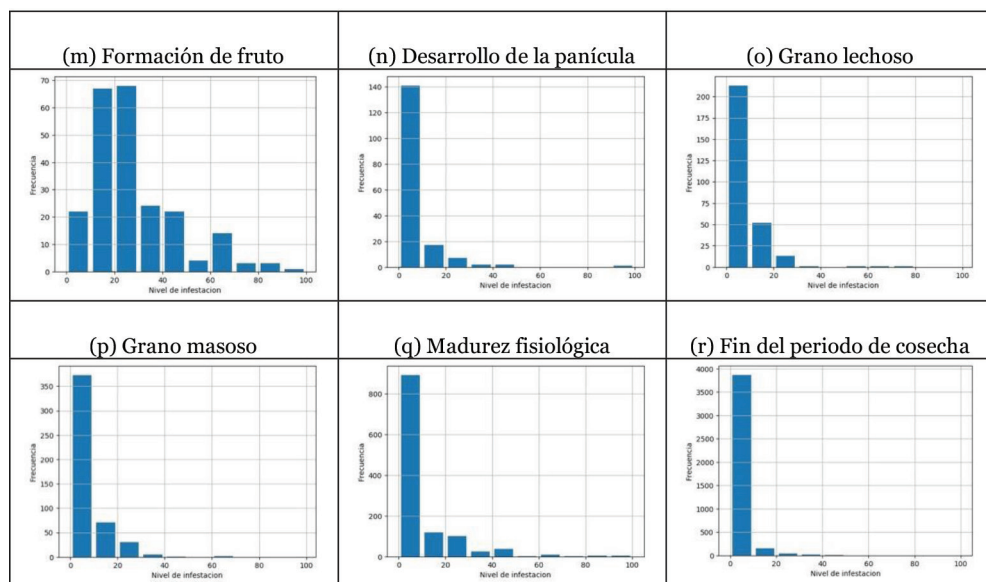


Figura 12 – Histogramas por etapa fenológica (*continuación*)

Con la información recopilada, se identificaron algunas mejoras necesarias para la aplicación. Posteriormente, se realizaron pruebas de funcionalidad a los cuatro módulos principales de PulAm. A partir de estas pruebas, se pudieron realizar las siguientes mejoras:

- El usuario tuvo que seleccionar su nombre manualmente cada vez que quería agregar nuevas inspecciones. Posteriormente, se agregó una opción en el menú de preferencias, por lo que el usuario debe hacerlo sólo una vez.
- Se configuró el dispositivo móvil de tal manera que, en ausencia de una señal GPS, el dispositivo puede solicitar los datos más precisos disponibles a través de Wi-Fi y las torres celulares del proveedor de la red.
- Se agregó un cuadro de diálogo en el módulo de captura de inspección para facilitar la adición de una nueva inspección. Ahora, el usuario sólo debe agregar el grado de infestación mientras la aplicación toma el resto de la información de la última inspección agregada, información que se almacena automáticamente en el archivo de preferencias compartidas. Finalmente, el usuario puede editar esta información de acuerdo con los datos de la inspección actual.

- PulAm inicialmente usó una serie de puntos almacenados en la base de datos que juntos formaron un polígono representado por una plantación. La aplicación tuvo que dibujar todos los puntos para formar el polígono, usando recursos computacionales altos. Ahora, PulAm lee los polígonos como una capa geográfica que dibuja en el mapa rápidamente.
- Se añadieron funciones adicionales en el módulo de gestión de bases de datos.

6. Conclusiones

PulAm es una aplicación móvil basada en el sistema operativo Android, la cual es muy útil para el levantamiento de datos en campo. Como se mostró en el presente trabajo de investigación, PulAm es un prototipo a través del cual los técnicos de CESAVERNAY pueden registrar el levantamiento de datos de sus campañas fitosanitarias, de forma ágil y minimizando los errores. Además, pueden acceder a información histórica sobre inspecciones realizadas previamente.

Como trabajo futuro, se planea realizar pruebas de aceptación de PulAm con el personal técnico de CESAVERNAY. También se planea diseñar un módulo en el que el usuario pueda tomar una o más fotografías de las plantas de algún cultivo con su dispositivo móvil y que la aplicación pueda determinar automáticamente el grado de infestación. Con este módulo, el usuario tendría una medición más estandarizada del nivel de infestación.

Agradecimientos. Agradecemos al personal de CESAVERNAY por toda su ayuda, tanto técnica como logística para la recolección de datos en el campo. Alejandra Perez-Mena agradece al Consejo Nacional de Ciencia y Tecnología (CONACYT) por la subvención financiera otorgada a través de la beca #613382.

Referencias

- CESAVERNAY. (2015). *Campaña contra el pulgón amarillo*. México: Comité Estatal de Sanidad Vegetal de Nayarit.
- Liopa-Tsakalidi, A., Tsolis, D., Barouchas, P., Chantzi, A.-E., Koulopoulos, A., & Malamos, N. (2013). Application of Mobile Technologies through an Integrated Management System for Agricultural Production. *Procedia Technology*, 8, 165–170.
- INEGI. (2016). *Actualización del Marco Censal Agropecuario 2016*. Obtenido de <https://www.inegi.org.mx/proyectos/agro/amca/>
- INEGI. (2019). *Mapa Digital de México*. Obtenido de <http://gaia.inegi.org.mx/mdm6/>
- Montoya, F. G., Gómez, J., Cama, A., Zapata-Sierra, A., Martínez, F., De La Cruz, J., & Manzano-Agugliaro, F. (2013). A monitoring system for intensive agriculture based on mesh networks and the android system. *Computers and Electronics in Agriculture*, 99, 14–20.

- Pérez-Mena, A., Fernández-Zepeda, J. A., Rivera-Caicedo, J. P., & Avila-George, H. (2019). PulAm: An App for Monitoring Crops. En: J. Mejia, M. Muñoz, Á. P. Rocha, & M. Pérez-Cisneros (Eds.), *Proceedings of the 7th International Conference on Software Process Improvement (CIMPS 2018)* (Vol. 865, págs. 196-205). Guadalajara, México: Springer.
- Quezada-Sarmiento, P. A. & Mengual Andrés, S. (2017). Implementación de una solución web y móvil para la gestión vehicular basada en Arquitectura de Aspectos y metodologías ágiles: Un enfoque educativo de la teoría a la práctica. *RISTI-Revista Ibérica de Sistemas y Tecnologías de Información*, 27, 97–111.
- Razo Salas, M. A., Vázquez Reyes, S. & Solís Robles, R. (2016). Mejoras del proceso endoscópico usando aplicaciones móviles. *RISTI-Revista Ibérica de Sistemas y Tecnologías de Información*, 17, 115–127.
- Rodríguez-del-Bosque, L. A., & Terán, A. P. (2015). *Melanaphis sacchari* (Hemiptera: Aphididae): A New Sorghum Insect Pest in Mexico. *Southwestern Entomologist*, 40(2), 433–434.
- SAGARPA. (2014). *Atlas agroalimentario México*. México: Secretaría de Agricultura, Ganadería, Desarrollo Rural, Pesca y Alimentación.
- SENASICA. (2014). *Pulgón amarillo Melanaphis sacchari (Zehntner)*. México: Servicio Nacional de Sanidad, Inocuidad y Calidad Agroalimentaria.
- SIAP. (2017). *Servicio de Información Agroalimentaria y Pesquera*. Obtenido de: <http://infosiap.siap.gob.mx>
- Tezer, O. S. (2014). *SQLite vs MySQL vs PostgreSQL: A Comparison Of Relational Database Management Systems*. Obtenido de: <https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems>
- Vanderlip, R. L., & Reeves, H. E. (1972). Growth Stages of Sorghum (*Sorghum bicolor*, (L.) Moench.). *Agronomy journal*, 64(1), 13–16.
- Vasisht, D., Kapetanovic, Z., Won, J.-h., Jin, X., Chandra, R., Kapoor, A., . . . Stratman, S. (2017). Farmbeats: an IoT platform for data-driven agriculture. En: A. Akella, & J. Howell (Ed.), *Proceedings of the 14th USENIX Symposium on Networked Systems Design and Implementation* (págs. 515-529). Boston, MA, USA: USENIX Association.

Crítérios Editoriais

A RISTI (Revista Ibérica de Sistemas e Tecnologias de Informação) é um periódico científico, propriedade da AISTI (Associação Ibérica de Sistemas e Tecnologias de Informação), que foca a investigação e a aplicação prática inovadora no domínio dos sistemas e tecnologias de informação.

O Conselho Editorial da RISTI incentiva potenciais autores a submeterem artigos originais e inovadores para avaliação pelo Conselho Científico.

A submissão de artigos para publicação na RISTI deve realizar-se de acordo com as chamadas de artigos e as instruções e normas disponibilizadas no sítio Web da revista (<http://www.risti.xyz/>).

Todos os artigos submetidos são avaliados por um conjunto de membros do Conselho Científico, não inferior a três elementos.

Em cada número da revista são publicados entre cinco a oito dos melhores artigos submetidos.

Criterios Editoriales

La RISTI (Revista Ibérica de Sistemas y Tecnologías de la Información) es un periódico científico, propiedad de la AISTI (Asociación Ibérica de Sistemas y Tecnologías de la Información), centrado en la investigación y en la aplicación práctica innovadora en el dominio de los sistemas y tecnologías de la información.

El Consejo Editorial de la RISTI incentiva autores potenciales a enviar sus artículos originales e innovadores para evaluación por el Consejo Científico.

El envío de artículos para publicación en la RISTI debe hacerse de conformidad con las llamadas de los artículos y las instrucciones y normas establecidas en el sitio Web de la revista (<http://www.risti.xyz/>).

Todos los trabajos enviados son evaluados por un número de miembros del Consejo Científico de no menos de tres elementos.

En cada número de la revista se publican cinco a ocho de los mejores artículos enviados.

Os asociados da AISTI recebem a RISTI gratuitamente, por correio postal. Torne-se associado da AISTI. Preencha o formulário abaixo e envie-o para o e-mail aistic@gmail.com

Los asociados de la AISTI reciben la RISTI por correo, sin costo alguno. Hazte miembro de la AISTI. Rellena el siguiente formulario y remítelo al e-mail aistic@gmail.com



Formulário de Associado / Formulario de Asociado

Nome/Nombre: _____

Instituição/Institución: _____

Departamento: _____

Morada/Dirección: _____

Código Postal: _____ Localidade/Localidad: _____

País: _____

Telefone/Teléfono: _____

E-mail: _____ Web: _____

Tipo de Associado e valor da anuidade:

☐ Individual - 35€

☐ Instituição de Ensino ou I&D/Institución de Educación o I&D - 250€

☐ Outro (Empresa, etc.) - 500€

NIF/CIF: _____

Data/Fecha: __/__/____ Assinatura/Firma: _____



Revista Ibérica de Sistemas e Tecnologias de Informação
Revista Ibérica de Sistemas y Tecnologías de Información

©AISTI 2019 <http://www.aisti.eu>

