

# Ingeniería de Software e Ingeniería del Conocimiento: Reporte de las JIISIC-CEIS'2015

Omar S. Gómez  
Prometeo-Senescyt  
Escuela Superior Politécnica de Chimborazo  
Riobamba 060106, Ecuador  
ogomez@esPOCH.edu.ec

Raúl A. Aguilar-Vera  
Facultad de Matemáticas  
Universidad Autónoma de Yucatán  
Mérida 97000, México  
avera@uady.mx

José Antonio Pow-Sang  
Pontificia Universidad Católica del Perú  
San Miguel, Lima 32, Perú  
japowsang@puccp.edu.p

**Resumen**—En el presente documento se presenta el reporte de las XI Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento así como del Congreso Ecuatoriano en Ingeniería de Software, celebrados en la Escuela Superior Politécnica de Chimborazo (ESPOCH) los días 4 y 5 de junio de 2015. De un total de 28 trabajos enviados, en esta edición se han aceptado 16. En este documento se presentan los títulos y resúmenes de los trabajos aceptados.

**Palabras clave**—JIISIC'2015, CEIS'2015, ingeniería de software, ingeniería del conocimiento, ecuador.

## I. INTRODUCCIÓN

La Ingeniería de Software y la Ingeniería del Conocimiento son dos disciplinas relacionadas entre sí. Por ejemplo, enfoques basados en modelos comúnmente usados en la ingeniería del conocimiento pueden trasladarse al desarrollo de productos software como un modelo de proceso de construcción [1], puede decirse que ambas disciplinas convergen en un único ciclo de vida [2]. Con el fin de ofrecer un espacio Iberoamericano para la divulgación de investigaciones entre estas dos disciplinas surgen las JIISIC.

JIISIC o Jornadas Iberoamericanas de Ingeniería de Software e Ingeniería del Conocimiento han representado un foro de encuentro internacional de científicos y profesionales dedicados al estudio e investigación de la Ingeniería de Software y de la Ingeniería del Conocimiento. Su propósito fundamental es fomentar el contacto, la cooperación científica y profesional, así como la transferencia de tecnología en el ámbito Iberoamericano.

Desde su primera edición (2001), celebrada en Buenos Aires (Argentina), las JIISIC han convocado a un número cada vez más nutrido de asistentes, en Salvador de Bahía (Brasil), Valdivia (Chile), Madrid (España), Puebla (México), Lima (Perú), Guayaquil (Ecuador), Mérida (México), Lima (Perú), Medellín (Colombia). En esta edición 2015, regresan a Ecuador a la ciudad de Riobamba, la sultana de los Andes.

Por otra parte CEIS o Congreso Ecuatoriano en Ingeniería de Software surge de la necesidad de mantener un foro de discusión entre científicos, académicos y profesionales interesados en el estudio e investigación de la Ingeniería de Software en el Ecuador y la región.

El objetivo del CEIS es proporcionar un punto de encuentro para presentar y discutir resultados de investigaciones

realizadas en el Ecuador y en la región dentro del ámbito de la Ingeniería de Software. Como objetivo implícito, se pretende fomentar la colaboración entre las distintas partes interesadas en esta temática.

En esta edición JIISIC-CEIS'2015 se han recibido 28 trabajos de los cuales en base a las revisiones hechas por los miembros del comité de programa se han seleccionado 16 [3]; la tasa de aceptación ha sido del 57%. En esta edición se ha tenido la participación de colegas de diferentes países como Cuba, Ecuador, México, Perú, España y Canadá. El comité de programa agradece la participación de quienes eligieron JIISIC-CEIS'2015 para la difusión de sus trabajos de investigación; también agradecemos a los revisores miembros del comité científico su valiosa colaboración en el proceso de revisión. Estamos seguros que sus recomendaciones fueron valoradas positivamente por los autores de los trabajos aceptados, así como serán valoradas de forma positiva por aquellos autores cuyos trabajos no han sido seleccionados en esta edición.

Los trabajos presentados en este número especial de la Revista Latinoamericana de Ingeniería de Software (RELAIS) corresponden a versiones extendidas de los cinco trabajos que recibieron la mejor puntuación por parte del comité científico de las JIISIC-CEIS'2015. En el resto de secciones de este documento se presentan los títulos y resúmenes de los 16 trabajos aceptados en la edición 2015 de este evento.

## II. ESTUDIO DEL TAMAÑO DE LOS DOCUMENTOS DE REQUERIMIENTOS DE SOFTWARE COMO FACTOR PARA LA ESTIMACIÓN DEL ESFUERZO DE INSPECCIÓN DE REQUERIMIENTOS DE SOFTWARE

Autores: Carlos Monsalve, Rubén Ullón, Ricardo Maya y José Romero. Resumen: La inspección de documentos de requerimientos de software es una técnica que asegura calidad y reducción de costos en el ciclo de vida del desarrollo de software. Por otro lado, es crucial en la planificación de proyectos de software contar con métodos que permitan estimar esfuerzo basados en métricas y no únicamente en el juicio de expertos. Pese a esto, existe muy poco trabajo reportado orientado a la estimación de esfuerzo en tareas de inspección de documentos de requerimientos de software. En este artículo estudiamos la relación existente entre el tamaño de los documentos de requerimientos y el esfuerzo necesario para

su inspección. Para ello se realizó una experiencia en laboratorio con estudiantes de los últimos niveles de una carrera de grado de computación, quienes fueron entrenados en tareas de inspección. Los resultados permitieron establecer un modelo estadísticamente significativo para estimar esfuerzo en función del tamaño de los documentos inspeccionados [4].

### III. EVALUACIÓN DE LA ACCESIBILIDAD EN SITIOS WEB CUBANOS

Autores: Yenier F. Moreno y Lisandra Armas. Resumen: En los últimos años, las nuevas Tecnologías de la Información y las Comunicaciones han generado un notable impacto en nuestra vida cotidiana. El usuario puede hacer uso intensivo de estas nuevas tecnologías iniciando trámites, gestionando transacciones, consultando información o hasta participando en espacios virtuales de comunicación. El propósito de la investigación ha sido evaluar el nivel de accesibilidad de sitios web cubanos, para lo cual se utilizó la herramienta TAW, que permite analizar, de acuerdo con la norma WCAG 2.0 la existencia de barreras en ellos, luego de un análisis exhaustivo se comprobó que ninguno de los sitios evaluados la supera. Se concluye recomendando realizar re-visiones y ajustes para resolver estos problemas y facilitar la accesibilidad web [5].

### IV. EVALUACIÓN DE USABILIDAD EN HERRAMIENTAS EDUCATIVAS: UNA REVISIÓN SISTEMÁTICA

Autores: Freddy Paz, Claudia Zapata, César Olivares, Santiago Apaza y José A. Pow-Sang. Resumen: La usabilidad es un factor fundamental en la calidad de todo producto software. El sector educativo y enseñanza/aprendizaje no es la excepción. Por este motivo, varios métodos han sido propuestos con el objetivo de evaluar el nivel de usabilidad de los productos software. Sin embargo, esta gran cantidad de técnicas existentes torna compleja la elección del método más adecuado, lo cual justifica un estudio sobre las actuales tendencias, en miras de identificar los criterios más apropiados para realizar la elección correcta, en cada instancia de pruebas. El presente trabajo es una revisión sistemática de los estudios publicados entre el 2012 y el 2014, en los cuales se hayan realizado evaluaciones de usabilidad a plataformas informáticas orientadas al sector educativo y enseñanza-aprendizaje, con el propósito de identificar las actuales tendencias en el área. Se muestra un resumen de los métodos usados, y se reporta asimismo el número de usuarios o evaluadores participantes, el número de veces que se aplica el mismo método en un mismo estudio, la realización o no de una capacitación previa a la evaluación y la vinculación entre los desarrolladores de la aplicación y el equipo de evaluación [6].

### V. APLICACIÓN DEL PARADIGMA DE LÍNEA DE PRODUCTO SOFTWARE PARA MODELAR UN SISTEMA FINANCIERO

Autores: Edison Espinosa, Gabriela Salguero y Paola Salguero. Resumen: El crecimiento exponencial del software y la necesidad de reducir tiempos y costos en la construcción de productos software, han hecho de la reutilización de componentes una práctica común hoy en día. El paradigma de línea de producto software se enfoca en la reutilización de componentes de forma planificada, que permite la reducción de costos tiempos y recursos en el desarrollo de software. En el proceso de línea de producto software se identifican y modelan las características comunes y variables de los productos de la línea, que permite crear un núcleo de componentes llamados core assets y establecer las características específicas que identifican a cada producto dentro de la línea. Para desarrollar

cada uno de los productos de la línea se usan los core assets y se desarrollan las características específicas que identifican a cada producto dentro de la línea. En este artículo se presenta el proceso que se llevó a cabo para identificar, modelar e implementar en una herramienta informática las características de un sistema financiero aplicando el paradigma de línea de producto software. Los resultados obtenidos hasta el momento son una matriz y algunos modelos donde se evidencian las características identificadas y el modelo de la línea del producto del sistema financiero [7].

### VI. EFECTIVIDAD DEL TEST-DRIVEN DEVELOPMENT: UN EXPERIMENTO REPLICADO

Autores: Oscar Dieste, Efraín R. Fonseca, Geovanny Raura y Priscila Rodríguez. Resumen: Los métodos ágiles y sus prácticas asociadas, e.g.: Test-Driven Development (TDD), son ampliamente utilizadas en la industria y han sido repetidamente sometidas a estudios empíricos. Antecedentes: N. Juristo y su equipo han realizado diversos experimentos en empresas y academia acerca de TDD. En general, los experimentos no muestran un efecto positivo de TDD en la calidad del código o la productividad de los programadores. Objetivo: Replicar el experimento UPM 2014 para reproducir sus resultados y secundariamente, estudiar el efecto de la experiencia del desarrollador en la efectividad de TDD. Método: Replicación experimental manteniendo similares el training y los materiales del experimento original. La replicación fue llevada a cabo en la Universidad de las Fuerzas Armadas ESPE sede Latacunga (ESPEL). Los sujetos experimentales fueron 17 estudiantes del Master en Ingeniería de Software. Resultados: Los resultados de la replicación confirman los efectos observados en UPM 2014. La efectividad de TDD ha resultado menor que ITL, aunque las diferencias no son significativas. La productividad y calidad del código producido por los estudiantes ESPEL cuando utilizan ITL y TDD es comparable a la de los estudiantes UPM, aunque menor en valores absolutos. Conclusiones: TDD no produce beneficios en calidad o productividad, o al menos no de forma inmediata. Parece necesario que los sujetos experimentales reciban training intensivo para que los efectos de TDD sean evidentes [8].

### VII. DETECCIÓN DE DEFECTOS CON Y SIN APOYO DE UN ENTORNO VIRTUAL COLABORATIVO INTELIGENTE EN CURSOS INTRODUCTORIOS DE PROGRAMACIÓN

Autores: Juan P. Ucán, Omar S. Gómez, Alejandro A. Castillo y Raúl A. Aguilar. Resumen: El uso de entornos virtuales colaborativos facilita la comunicación, coordinación y cooperación en un grupo de personas. En el ámbito de las Tecnologías de la Información y la Comunicación (TIC) se han desarrollado entornos virtuales colaborativos para apoyar el aprendizaje de la programación. En ese trabajo se examina la efectividad en la detección de defectos en programas instrumentados con y sin apoyo de un entorno virtual colaborativo inteligente (EVCI) propio. Empleando un diseño experimental cruzado se conformaron 18 equipos de 2 y 3 estudiantes (en total 46) donde en dos sesiones los estudiantes trabajaron con y sin un EVCI en la detección de defectos en dos programas instrumentados. Los resultados sugieren una efectividad equivalente en la detección de defectos para quienes emplearon el EVCI (43.45%) como para quienes trabajaron de manera tradicional (45.33%). Se observa que es igualmente efectivo para este tipo de tarea (detección de defectos) trabajar de forma virtual a través de un EVCI que

hacerlo de manera tradicional (al mismo tiempo y en el mismo sitio) [9].

#### VIII. SOFTWARE MEASUREMENT FOR UNDERGRADUATES: SUGGESTIONS FOR A SOFTWARE ENGINEERING CURRICULUM

Autores: Mónica Villavicencio y Alain Abran. Resumen: Over the past decade, efforts have been made to develop an updated version of the Software Engineering Body of Knowledge (SWEBOK). Although this is a great contribution to the field, there is still a need for updating the software engineering curriculum guidelines. Building upon previous research, the present study addresses this need by providing a set of suggestions regarding the topics of software measurement to be emphasized in undergraduate programs. A revision of the bodies of knowledge and curriculum guidelines together with an examination of opinions from practitioners and teachers were performed. Results indicate that five software measurement topics are essential in a software engineering curriculum [10].

#### IX. CERTIFICACIÓN DE LA MANTENIBILIDAD DEL PRODUCTO SOFTWARE. UN CASO PRÁCTICO

Autores: Moisés Rodríguez, Óscar Pedreira y Carlos M. Fernández. Resumen: La calidad del software está adquiriendo durante los últimos años una gran importancia, principalmente debido a que el software está presente en prácticamente todo lo que nos rodea y se hace necesario asegurar su correcto funcionamiento. Sin embargo, hasta ahora la mayor parte de los estudios se han centrado en evaluar la calidad de los procesos de desarrollo, y aunque existen trabajos centrados en la calidad del producto software, no existe todavía una propuesta completa para la evaluación y certificación de la calidad del producto software, basada en la nueva familia de normas ISO/IEC 25000. El presente artículo expone un caso de estudio de evaluación, mejora y certificación de la mantenibilidad de un producto software, en el que han participado una empresa desarrolladora de software, un laboratorio de evaluación acreditado y una entidad de certificación. Para ello, se presentarán las características de este tipo de evaluaciones, el rol que cumple cada una de las entidades participantes y el proceso seguido desde la evaluación inicial hasta la certificación definitiva [11].

#### X. EVALUACIÓN DE LA CALIDAD EN PRODUCTOS SOFTWARE. EXPERIENCIA CUBANA HACIA LA ACREDITACIÓN

Autores: Yanet Brito y Tayché Capote. Resumen: Actualmente existe una tendencia generalizada a la evaluación de productos software a través del esquema de tercerización del servicio u outsourcing de pruebas, tanto para desarrolladores como para clientes, que exigen en sus productos un elemento diferenciador de la competencia. Para proveer servicios especializados de testing, el laboratorio debe emplear normas y modelos que le permitan certificar en dichos productos las características de calidad establecidas a escala internacional. Un requisito indispensable para ello es que el laboratorio esté acreditado según la ISO/IEC 17025:2006 "Requisitos generales para la competencia de los laboratorios de ensayo y de calibración". El siguiente trabajo esboza el proceso de evaluación de software que se realiza en un laboratorio de pruebas al software que brinda servicios outsourcing y la implementación de normas internacionales establecidas para la acreditación de ensayos. La propuesta considera las normas, NC-ISO/IEC 9126-1:2005, ISO/IEC 14598 y NC-ISO/IEC 17025:2006 y la adecuación de esta última de manera novedosa

en el área del software, presente en el documento: Guía de requisitos complementarios. Como resultados la aplicación de la propuesta en un entorno real, evidenciándose excelentes resultados para su validación [12].

#### XI. USO DE HERRAMIENTAS DE COMPUTACIÓN EN LA NUBE PARA DEFINIR, PLANIFICAR, CONTROLAR PROYECTOS DE INNOVACIÓN Y GENERACIÓN DE CONOCIMIENTO

Autores: Pablo Quezada, Liliana Enciso y Juan Garbajosa. Resumen: La Computación en la Nube se está convirtiendo en la mejor manera de entregar soluciones que satisfagan la necesidad actual de una mayor colaboración entre empresas, educación y sociedad. En este contexto el aprendizaje de la Innovación como instrumento de desarrollo de productos software es importante durante la vida profesional de un Ingeniero en Sistemas Informáticos y Computación ya que conlleva a transformar tareas rutinarias en tareas automatizadas, de igual forma le permitirá gestionar procesos, enfocados en alinear los servicios de Tecnologías de la Información (TI) a las necesidades de las empresas con énfasis en los beneficios que puede percibir el cliente final adicionado el factor de I+D+I para crear un producto innovador. El presente trabajo se enfoca en el uso de herramientas de Computación en la Nube (CN) en sus fases: definición, planificación y control de proyecto de software con base en la innovación como herramienta de desarrollo y generación de conocimiento. Las herramientas de Computación en la Nube analizadas fueron: Alfresco, Redbooth, DotProject, Dealines, Redmine. El uso de estas herramientas permitió obtener información para formular un cuerpo de conocimiento en el contexto de proyectos [13].

#### XII. MODELO NEURONAL DE ESTIMACIÓN PARA EL ESFUERZO DE DESARROLLO EN PROYECTOS DE SOFTWARE (MONEPS)

Autores: Mario G. Almache, Jenny A. Ruiz, Geovanny Raura y Efraín R. Fonseca. Resumen: La estimación temprana del esfuerzo para la construcción de un producto software, es crucial en la previsión del costo y tiempo necesarios para el desarrollo de software. Los modelos y técnicas para la estimación del esfuerzo presentan, como principal inconveniente, la poca precisión en las predicciones realizadas, y generalmente se hace una mínima consideración de los aspectos no funcionales del software. Se propone la construcción de un modelo de estimación para el esfuerzo en el desarrollo de software, denominado MONEPS, que pretende mejorar la precisión en la estimación del esfuerzo, utilizando una Red Neuronal Artificial (RNA) en Backpropagation, cuya capa de entrada se estructura sobre la base de un conjunto de características y atributos tomados de la norma ISO 25000 de la calidad del software. La RNA fue entrenada con datos recopilados de aplicaciones desarrolladas en el ámbito académico, de las cuales se conocían sus tiempos de desarrollo y costos asociados. Las estimaciones de tiempo y costo, para dos casos de prueba, muestran más precisión en el modelo neuronal, en comparación con los modelos Cocomo-81 y COCOMO-II. MONEPS ha logrado la convergencia de aspectos funcionales y no funcionales para mejorar la precisión en la estimación de dicho esfuerzo [14].

#### XIII. INTEGRACIÓN DE METODOLOGÍAS ÁGILES EN EL DESARROLLO DE UN SISTEMA DE MONITOREO INALÁMBRICO PARA MEDIR LA CONTAMINACIÓN DEL AIRE EN TIEMPO REAL

Autores: Walter Fuertes, Diego Carrera, César Villacís, Fernando Galárraga, Theofilos Toulkeridis y Hernán Aules.

Resumen: Este artículo presenta el proceso de desarrollo de un sistema de monitoreo inalámbrico de bajo costo, basado en un modelo distribuido multicapa, que en combinación con la plataforma Arduino, permite medir parámetros referenciales de calidad de aire. Para llevarlo a cabo, se combinaron metodologías ágiles como Scrum y Extreme Programming con el propósito de garantizar la calidad del software. El dispositivo electrónico está equipado con tres sensores que miden la concentración de monóxido de carbono, dióxido de carbono y densidad del polvo, con la capacidad de conectarse al Internet para la transmisión, en tiempo real, de la información recolectada utilizando una API desarrollada en C/C++. La prueba de concepto se la aplicó en las ciudades de Quito, Amaguaña y el Tena de Ecuador, con resultados positivos, tanto en el funcionamiento del software y hardware que componen el prototipo, como en la medición referencial de la concentración de los contaminantes del aire [15].

#### XIV. PROPUESTA DE DIRECTRICES PARA EL ÁREA DEL CONOCIMIENTO “GESTIÓN DE PROCESOS ORGANIZACIONALES” DEL MCDAI

Autores: Danay Ramírez y Olga L. Rodríguez. Resumen: En la actualidad, la Industria Cubana del Software (ICSW) está compuesta mayormente por micro, pequeñas y medianas empresas (PYMES), las cuales no implementan los modelos y estándares existentes a nivel mundial. La aplicación de estos modelos y estándares existentes en las PYMES es difícil, ya que están enfocados generalmente a las grandes empresas de software, y no tienen la orientación o el nivel de detalle que les permita adoptarlos. Es por este motivo que surge la necesidad de crear el Modelo de Calidad para el Desarrollo de Aplicaciones Informáticas (MCDAI), tomando como referencia los existentes pero adaptándolos al contexto cubano. Su principal objetivo es guiar y estandarizar la producción de software en el país. El mismo cuenta con 12 procesos bases, los cuales actualmente se están definiendo en una 1ra versión. Uno de ellos es el proceso base Gestión de Procesos de la Organización (GPO), el cual consiste en garantizar un marco adecuado para la gestión de los procesos organizacionales, posibilitando definir sus objetivos, procesos, repositorios, estándares del entorno de trabajo y modelos de ciclo de vida, así como gestionar los recursos de la organización. En esta investigación se presenta una propuesta del proceso GPO, donde se proponen 11 directrices para alcanzar una correcta gestión de procesos, las cuales fueron validadas en Grupos Técnicos de Trabajo (TWG) conformados por especialistas en el tema pertenecientes a diferentes empresas de la ICSW [16].

#### XV. ADQUISICIÓN DEL CONOCIMIENTO EN EL PROCESO DE COMPOSICIÓN MUSICAL EN BASE A TÉCNICAS DE INTELIGENCIA ARTIFICIAL

Autores: Efraín Astudillo, Pedro Lucas y Enrique Peláez. Resumen: Generalmente, la composición musical es ejecutada por un proceso creativo en base a un amplio conocimiento de la teoría musical. Sin embargo, es posible crear piezas musicales extendiendo ese conocimiento aprendido en la academia. El propósito de este artículo es analizar y proponer representaciones abstractas, basadas en técnicas de Inteligencia Artificial, para la extracción del conocimiento en este proceso de composición e ir más allá de los tradicionales conceptos musicales. Los resultados de esta investigación contribuirán a entender la interacción de los elementos involucrados en el proceso de composición permitiendo el desarrollo de sistemas inteligentes para esta tarea y brindando soporte a los

compositores en la realización de música experimental. Además, se presentan los resultados de las pruebas realizadas con una de las representaciones pertenecientes al modelo completo propuesto para el conocimiento musical, donde se generaron melodías que fueron evaluadas estéticamente y sometidas a una Prueba de Turing [17].

#### XVI. PATRONES DE USO DOCENTE DE EVAS EN LA ESCUELA DE INGENIERÍA EN SISTEMAS INFORMÁTICOS DE LA ESPOCH

Autores: Gonzalo Samaniego y Luis Marqués. Resumen: El presente trabajo encuentra y valora patrones de uso docente de los Entornos Virtuales de Aprendizaje (EVAs) de los profesores de la Escuela de Ingeniería en Sistemas (EIS) de la Facultad de Informática y Electrónica (FIE) de la Escuela Superior Politécnica de Chimborazo (ESPOCH). Se utiliza un procedimiento en base a los logs del apartado “Informes” de los EVAs, que permite determinar y analizar variables tales como el lugar, horario y fecha frecuente de trabajo, los diferentes niveles de interacción y los recursos y actividades generados a lo largo de nueve semestres académicos. Los resultados evidencian la transición del lugar habitual de trabajo de los docentes en los EVAs desde la ESPOCH hacia la CASA, el aumento progresivo del uso e interacción en los EVAs, el uso reiterativo de un grupo de recursos y actividades, entre otros [18].

#### XVII. PREDICTOR BASADO EN PROTOTIPOS DIFUSOS Y CLASIFICACIÓN NO SUPERVISADA

Autores: Aníbal Vásquez, Enrique Peláez y Xavier Ochoa. Resumen: La construcción de prototipos difusos es un método que permite describir a los elementos más representativos de un clúster, a través de su tipicidad. Los prototipos, como los datos más representativos de cada clúster, pueden ser usados en un proceso de clasificación como datos de entrenamiento. Estos prototipos y los clusters pueden ser construidos mediante algoritmos de clustering difuso; los clusters representados por los prototipos poseen variables descriptivas y atributos que pueden ser asociados a nuevos datos. El siguiente trabajo propone una arquitectura que utiliza herramientas de clustering y prototipado difuso, para clasificación no-supervisada y predicción a través de la extracción de variables descriptivas. El desarrollo de un caso de estudio permitió validar el modelo de clasificación para predecir el riesgo de falla en el rendimiento académico de estudiantes, basado en su carga académica y rendimiento, en la selección de cursos antes de registrarse, con un porcentaje de certeza significativo [19].

#### REFERENCIAS

- [1] R. Studer, V. R. Benjamins, and D. Fensel. *Knowledge engineering: Principles and methods*. Data & Knowledge Engineering, Volume 25, Issues 1–2, March 1998, p. 161–197.
- [2] F. Alonso, N. Juristo, J. L. Maté, and J. Pazos. *Software engineering and knowledge engineering: Towards a common life cycle*. Journal of Systems and Software, Volume 33, Issue 1, April 1996, p. 65–79.
- [3] O. S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors. *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas*. 11th (JIISIC'2015). Curran Associates, Inc., May 2015. ISBN:978-1-5108-0208-7.
- [4] C. Monsalve, R. Ullón, R. Maya, and J. Romero. *Estudio del tamaño de los documentos de requerimientos de software como factor para la estimación del esfuerzo de inspección de requerimientos de software*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento*. Jornadas

- Iberoamericanas. 11th (JIISIC'2015). Curran Associates, Inc., Riobamba, 2015, p. 1–12.
- [5] Y. F. Moreno, and L. Armas. *Evaluación de la accesibilidad en sitios web cubanos*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 13–26.
- [6] F. Paz, C. Zapata, C. Olivares, S. Apaza, and J. A. Pow-Sang. *Evaluación de usabilidad en herramientas educativas: una revisión sistemática*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 27–38.
- [7] E. Espinosa, G. Salguero, and P. Salguero. *Aplicación del paradigma de línea de producto software para modelar un sistema financiero*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 39–52.
- [8] O. Dieste, E. R. Fonseca, G. Raura, and P. Rodríguez. *Efectividad del test-driven development: un experimento replicado*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 53–64.
- [9] J. P. Ucán, O. S. Gómez, A. A. Castillo, and R. A. Aguilar. *Detección de defectos con y sin apoyo de un entorno virtual colaborativo inteligente en cursos introductorios de programación*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 65–78.
- [10] M. Villavicencio, and A. Abran. *Software measurement for undergraduates: Suggestions for a software engineering curriculum*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 79–92.
- [11] M. Rodríguez, O. Pedreira, and C. M. Fernández. *Certificación de la mantenibilidad del producto software. Un caso práctico*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 93–106.
- [12] Y Brito, and T. Capote. *Evaluación de la calidad en productos software. Experiencia cubana hacia la acreditación*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 107–120.
- [13] P. Quezada, L. Enciso, and J. Garbajosa. *Uso de herramientas de computación en la nube para definir, planificar, controlar proyectos de innovación y generación de conocimiento*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 121–132.
- [14] M. G. Almache, J. A. Ruiz, G. Raura, and E. R. Fonseca. *Modelo neuronal de estimación para el esfuerzo de desarrollo en proyectos de software (MONEPS)*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 133–146.
- [15] W. Fuertes, D. Carrera, C. Villacís, F. Galárraga, T. Toulkeridis, and H. Aules. *Integración de metodologías ágiles en el desarrollo de un sistema de monitoreo inalámbrico para medir la contaminación del aire en tiempo real*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 147–158.
- [16] D. Ramírez, and O. L. Rodríguez. *Propuesta de directrices para el área del conocimiento "Gestión de Procesos Organizacionales"*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 159–170.
- [17] E. Astudillo, P. Lucas, and E. Peláez. *Adquisición del conocimiento en el proceso de composición musical en base a técnicas de inteligencia artificial*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 171–184.
- [18] G. Samaniego, and L. Marqués. *Patrones de uso docente de EVAs en la escuela de ingeniería en sistemas informáticos de la ESPOCH*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 185–196.
- [19] A. Vásquez, E. Peláez, and X. Ochoa. *Predictor basado en prototipos difusos y clasificación no supervisada*. In O.S. Gómez, G. Arcos, L. Aguirre, E. Villa, and R. H. Rosero, editors, *Ingeniería de Software e Ingeniería del Conocimiento. Jornadas Iberoamericanas. 11th (JIISIC'2015)*. Curran Associates, Inc., Riobamba, 2015, p. 197–206.



**Omar S. Gómez** es Ingeniero en Computación por la Universidad de Guadalajara (México, 2001), Maestro en Ingeniería de Software por el Centro de Investigación en Matemáticas (México, 2005), Doctor en Software y Sistemas por la Universidad Politécnica de Madrid (España, 2012), Post-Doctorado en la Universidad de Oulu (Finlandia, 2014). Actualmente es investigador Prometeo-Senescyt adscrito a la Escuela Superior Politécnica de Chimborazo (Ecuador). Su trabajo de investigación se centra en experimentación en ingeniería de software así como en temas relacionados con la calidad y el diseño de software.



**Raúl A. Aguilar-Vera** es Licenciado en Ciencias de la Computación por la Universidad Autónoma de Yucatán, Doctor en Informática por la Universidad Politécnica de Madrid, España. Actualmente es profesor de tiempo completo en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán. Su trabajo de investigación incluye las siguientes áreas: Ingeniería de Software e Informática Educativa.



**José Antonio Pow-Sang** es Doctor en Ingeniería Informática y Máster en Ingeniería del Software por la Universidad Politécnica de Madrid, España, y es Ingeniero Informático por la Pontificia Universidad Católica del Perú (PUCP). Es profesor principal de la PUCP y actualmente se desempeña como Director Ejecutivo de la Escuela de Posgrado y Director de la Maestría en Informática en la PUCP. Fue por cinco años coordinador de la especialidad de Ingeniería Informática, y durante ese periodo participó en la acreditación de este programa con ICACIT (Perú), ABET (EEUU) y CEAB (Canadá). Ha participado en un proceso de evaluación de ICACIT y actualmente está habilitado

por esta institución como evaluador. Representó al país en el área de Informática en el proyecto Alfa Tuning América Latina, proyecto en el que se definieron las competencias genéricas y específicas que deberían tener los profesionales en el área. Ha participado como conferencista, ponente, organizador, miembro del comité científico de conferencias internacionales en sus áreas de investigación. Es actualmente el segundo investigador en el país con más publicaciones en Ciencias de la Computación según Scopus®. Sus áreas de investigación son Experimentación en Ingeniería del Software, Métricas y Estimación en Proyectos de Software, Interacción Persona-Computador y Educación en Informática.