

Automatización de la Medición de Software para Flotas Dinámicas mediante un Modelo de Calidad Mixto para la Movilidad en Smart Cities

Alejandro Rivoira (+); Giselle Cavallera (+); Carlos Salgado (+); Alberto Fernández Gil (*); Alberto Sánchez (+); Mario Peralta (+)

(+) Departamento de Informática - Facultad de Ciencias Físico-Matemáticas y Naturales - Universidad Nacional de San Luis
Ejército de los Andes 950 – C.P. 5700 – San Luis – Argentina

e-mail: {rivoira.ale, giselle.cavallera}@gmail.com, {csalgado, alfanego, mperalta}@unsl.edu.ar

(*) Universidad Rey Juan Carlos

(*) Grupo de Inteligencia Artificial (GIA). Centro para las Tecnologías Inteligentes de la Información y sus Aplicaciones (CETINIA) – Madrid – España

e-mail: alberto.fernandez@urjc.es

RESUMEN

La movilidad se ha convertido en un asunto central del desarrollo urbano. Su relación con los temas de sostenibilidad y su capacidad para generar competitividad y calidad de vida nos sitúa ante la necesidad de replantear su futuro. Los nuevos sistemas de uso compartido de vehículos son parte del desafío que plantean las Smart Cities (Ciudades Inteligentes). Ya que el objetivo de los mismos es la mejora de la movilidad humana y la reducción de costos, su análisis y evaluación es de gran utilidad para lograr su optimización. Para ello, se debe pensar en un planteo más crítico de infraestructura y mejoras en los servicios de transporte público.

Por tanto, se ha definido un modelo de calidad mixto para software responsable de la gestión de la movilidad urbana con el objetivo de producir información cuantitativa sobre ciertas características de calidad para poder tomar decisiones de cambio o mejoras si fuesen necesarias. Como soporte a este modelo se ha completado el desarrollo de una herramienta que posibilita la medición y evaluación de calidad de este tipo de software. En estos momentos nos encontramos ampliando el conjunto de métricas e indicadores para aplicaciones móviles teniendo en mente las características de infraestructura, nuevas tecnologías e impactos en el medioambiente.

Palabras clave: Modelo de Calidad Mixto, Smart Cities, Flotas Dinámicas, Medición y

Evaluación de Calidad Software.

CONTEXTO

El presente trabajo se enmarca en el proyecto de investigación: Ingeniería de Software: Conceptos, Prácticas y Herramientas para el desarrollo de Software con Calidad – Facultad de Ciencias Físico-Matemáticas y Naturales, Universidad Nacional de San Luis. Proyecto N° P-031516. Además, se encuentra reconocido por el programa de Incentivos.

La tarea se efectúa en forma colaborativa con el grupo de investigación de Inteligencia Artificial (GIA), de la Universidad Rey Juan Carlos de Madrid- España. Dicho proyecto es la continuación de diversos proyectos de investigación sobre la gestión de flotas dinámicas y calidad aplicada a sistemas software, a través de los cuales se ha logrado un importante vínculo con la mencionada universidad internacional.

1. INTRODUCCIÓN

El concepto ciudades inteligentes nace como idea global de la gestión de los recursos de una ciudad dirigidos a mejorar la calidad de vida de los ciudadanos. También se puede describir como aquella ciudad que aplica las tecnologías de la información y de la comunicación (TIC) con el objetivo de proveerla de una infraestructura que garantice:

- Un desarrollo sostenible.

- Un incremento de la calidad de vida de los ciudadanos.
- Una mayor eficacia de los recursos disponibles.
- Una participación ciudadana activa.

Para determinar si una ciudad es o no inteligente, generalmente, se toma como punto de partida un modelo en torno a seis características: *Economía, Gestión de gobierno, Ciudadanía, Entorno, Calidad de vida y Movilidad*. A partir de este modelo, se han desarrollado objetivos agrupados en una serie de servicios para los que se han descrito las principales tecnologías que disponemos en la actualidad. De esta forma, se obtuvo una serie de plataformas de servicios, que se deben integrar y conectar entre sí, facilitando que el ciudadano forme parte activa del proceso de gestión de su ciudad [1]. En otras palabras, se busca modernizar la gestión de las ciudades, fomentando una mayor interacción entre las instituciones, los ciudadanos y los sistemas informáticos [2].

Actualmente vivimos en la convergencia de dos fenómenos importantes en la historia de la humanidad: la aceleración de la urbanización a nivel mundial y la revolución digital. De hecho, todas las aglomeraciones urbanas presentan diversos retos. Uno de sus retos consiste en abordar el proyecto de diseñar una ciudad inteligente manteniendo su ADN. Cada urbe tiene sus propias señas de identidad y sus propios valores que se expresan en las formas arquitectónicas, la configuración de los espacios y, sobre todo, las relaciones entre sus habitantes, y entre los ciudadanos y su gobierno [3]. Por ello, cada vez son más importantes la planificación urbana y el desarrollo de mecanismos de decisión dinámicos que tomen en cuenta el crecimiento y la inclusión de procesos de participación ciudadana.

El empleo masivo de productos software a escala global, prácticamente en todos los ámbitos del desempeño humano, favorecido en el último lustro por la enorme tasa de crecimiento de los servicios relacionados a la red Internet, han convertido a los usuarios en potenciales referentes para evaluar la calidad

de productos de software. Además, una extraordinaria demanda de aplicaciones produjo un nuevo cambio en el desarrollo de software. La respuesta a esta demanda, por parte de la industria y la comunidad de desarrolladores fue importante en tecnología y escala de producción. Sin embargo, no es extraño encontrar hoy que, las cuestiones relacionadas a la calidad -especialmente la calidad que perciben los usuarios reales respecto a las aplicaciones software que emplean- han sido postergadas o directamente ignoradas, entre otras razones por las urgencias de entregar los productos al mercado, eventuales ahorros o falta de recursos humanos capacitados [4].

La preocupación por la calidad de los sistemas basados en software ha aumentado a medida que éste se integra en cada aspecto de la vida cotidiana. La dependencia de las organizaciones respecto de este recurso es crítica. La automatización de las actividades y la generación y disponibilidad de información para la toma de decisiones, logradas a través del software, son claves para el logro de los objetivos y supervivencia de las mismas [5]. Los fallos de software afectan a todos los sectores y a todos los países, como puede constatar en la recopilación de incidentes documentados por Peter Neumann desde 1987 [6]. Lamentablemente, estos problemas pueden ser mucho más graves si ocurren en sistemas críticos, es decir, aquellos cuyo fallo puede provocar terribles pérdidas económicas o problemas ambientales o sociales e, incluso, la pérdida de vidas humanas.

Basados en lo que dice Tom Demarco [7], “No se puede controlar lo que no se puede medir”, que después complementan Fenton y Pfleeger [8] “No se puede predecir lo que no se puede medir”, es indispensable y altamente necesario para la vida de una ciudad inteligente poder realizar una gestión de calidad integral en los sistemas informáticos que interactúan en ella. Según la norma ISO 9000 [9], la calidad es el “grado en el que un conjunto de características inherentes cumple con los requisitos”. Como los requisitos dependerán de las diferentes partes interesadas (stakeholders), la calidad es un

concepto multidimensional, ya que puede ser sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, facilidad de mantenimiento, portabilidad, facilidad de uso, seguridad, integridad, etc. En la evolución experimentada por la calidad de los sistemas informáticos se ha pasado de un tratamiento centrado fundamentalmente en la inspección y detección de errores en los programas, a una aproximación más sistemática, dada la importancia que ha adquirido la calidad en la ingeniería de sistemas y en la ingeniería del software. La demanda de software por parte de las organizaciones y, en general, de la sociedad ha crecido mucho más deprisa que la capacidad de la industria para producir software de calidad, haciendo crónica la denominada "crisis del software" [10].

Una de las razones principales del incremento masivo en el interés por las métricas software ha sido la percepción de que son necesarias para la mejora de la calidad del proceso [11]. Para poder asegurar que un proceso o sus productos resultantes son de calidad, o poder compararlos, es necesario asignar valores, descriptores, indicadores o algún otro mecanismo mediante el cual se pueda llevar a cabo dicha comparación. Para ello, es necesario llevar adelante un proceso de medición del software, que en general, persigue tres objetivos fundamentales: ayuda a entender qué ocurre durante el desarrollo y el mantenimiento, permite controlar qué es lo que ocurre en los proyectos y otorga la posibilidad de mejorar los procesos y productos [8]. En efecto, las métricas son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo de software y los proyectos de mantenimiento [12] y pueden ser utilizadas por profesionales e investigadores para tomar mejores decisiones.

El modelo de calidad representa la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto. Cada modelo cuenta con características, subcaracterísticas y métricas asociadas. Las métricas son una medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo dado

[13]. Los indicadores, por su parte, son métricas o combinación de métricas que proporcionan una visión profunda, del proceso, del proyecto o del producto software [14]. Las métricas e indicadores son activos claves de una organización que proveen datos e información útiles para los procesos de análisis [15]. A su vez, también, son un buen medio para entender, monitorizar, controlar, predecir y probar el desarrollo del software y los proyectos de mantenimiento [16].

Para implementar las mediciones sobre el software de movilidad en una ciudad inteligente, fue necesario definir un método de evaluación y medición acorde a las necesidades propias del dominio de trabajo e investigación. El método es una combinación de buenas prácticas, métodos de evaluación y estándares para abordar lo que refiere puntualmente al paradigma de las ciudades inteligentes. Consta de 5 fases, las cuales se describen a continuación.

FASE 1: Seleccionar un modelo de calidad para Software de Movilidad: para poder hacer el análisis, estudio, medición y evaluación del software de movilidad había que adoptar un modelo de calidad, métricas e indicadores que permitiesen tener el marco donde poder hacer el estudio. Este modelo puede perfeccionarse, adaptarse, actualizarse o hasta cambiarse completamente si es necesario.

FASE 2: Aplicación/Definición de Métricas e Indicadores: una vez establecido y definido el Modelo de Calidad para aplicaciones de movilidad, se debe, como segundo paso:

- Calcular las métricas de cada atributo especificado en el Modelo de Calidad.

- Calcular los indicadores elementales.

FASE 3: Generación de la estructura de Agregación utilizando la lógica de operadores de la lógica continua para:

- Permitir detallar el peso y operador para cada atributo.

- Calcular los indicadores derivados.

FASE 4: Interpretar los datos/información y generar reportes.

FASE 5: Generar repositorios de información de cada proyecto de Medición y Evaluación (M&E) para análisis y

comparación posterior. Es decir, ir hacia la mejora continua.

2. LÍNEAS DE INVESTIGACIÓN, DESARROLLO E INNOVACIÓN

En base a lo explicado anteriormente, en esta línea de investigación se viene trabajando en la definición de un modelo de calidad mixto, contextualizado y adaptado, para medir y evaluar la calidad de un producto software responsables de la gestión de la movilidad, y por ende del tráfico, en una ciudad inteligente (específicamente Madrid). El modelo de calidad mencionado se denomina mixto porque, si bien tiene como base principal el Modelo ISO/IEC 25010, también se utilizaron indicadores o métricas del Modelo de Pressman, y algunas propias del dominio del problema. Además, se viene trabajando en el desarrollo de una herramienta software que permite la recolección de datos desde una aplicación de bicicletas compartidas llamada Ecobike, desarrollada dentro de un proyecto denominado Ecobike Solutions [17], perteneciente a alumnos en Ingeniería Informática de la Universidad Rey Juan Carlos de Madrid, España.

3. RESULTADOS Y OBJETIVOS

Hoy la tecnología nos brinda muchas herramientas para solucionar las problemáticas de las ciudades. Pero la preocupación está puesta en la calidad de esos sistemas basados en software, que ha aumentado a medida que estos se integran en cada aspecto de nuestras vidas cotidianas. Por ello, para poder gestionar y mejorar las aplicaciones software se deben controlar a través de mediciones y evaluación continua. Es así que se hace indispensable contar con herramientas que den soporte tecnológico a este proceso de medición y evaluación, para alivianar tareas, disminuir errores y obtener resultados adecuados de manera eficiente y eficaz.

Entre los resultados obtenidos que se proponen, se encuentra un modelo de calidad mixto que intenta determinar el impacto de las deficiencias en un agente software dirigido a

la gestión de la movilidad en ciudades inteligentes, teniendo como base la ciudad de Madrid. En relación a esto, se propondrán algunos elementos de solución que ayuden a crear un pensamiento consciente en las personas/usuarios.

A su vez se propone la herramienta que mide este tipo de software aplicando dicho modelo.

La aplicación del modelo en el caso de estudio sirvió para validar el modelo en un caso práctico real y para ver una tendencia en el resultado de la evaluación que sirva de análisis global

Se espera cumplir algunos objetivos adicionales en trabajos futuros:

- 1) Ampliar el conjunto de métricas e indicadores para aplicaciones móviles teniendo en mente las características de infraestructura, nuevas tecnologías e impactos en el medioambiente.
- 2) Dotar a la herramienta de M&E de un usuario administrador que pueda redefinir y adaptar métricas nuevas provenientes de los nuevos escenarios por venir.
- 3) Proveer una base de información, a partir de las definiciones, mediciones y evaluaciones logradas, para automatizar la medición en tiempo real a través de un sistema informático.
- 4) Brindar satisfacción a usuarios actuales, y generar atracción a nuevos usuarios a través de:
 - a. Protección y seguridad.
 - b. Adaptación y personalización.
 - c. Mejoras en usabilidad, rendimiento, eficiencia.
- 5) Incremento de la productividad y satisfacción al trabajo de los profesionales afines al desarrollo de software.
- 6) Mejora en la planeación eficaz de los agentes software.
- 7) El acercamiento a cero defectos.
- 8) Finalmente, servirá para concientizar sobre la importancia de la calidad en la creación, desarrollo y uso del software.

4. FORMACIÓN DE RECURSOS HUMANOS

Bajo esta línea, en el grupo de investigación, se han desarrollado dos tesis de maestría en Calidad de Software. En una de ellas se ha definido el modelo de calidad que cumple con los requerimientos de una aplicación software de movilidad para una Ciudad Inteligente. La otra consiste en la definición e implementación de una herramienta para la medición y evaluación de la calidad de software de gestión de tráfico. Además, se están llevando a cabo trabajos de grado con relación a la temática por alumnos de la carrera de Ingeniería Informática de la Universidad Rey Juan Carlos, Madrid.

Como así también se están llevando a cabo algunas tesinas de grado para la Licenciatura en Computación, y trabajos finales de carrera de la Ingeniería en Informática e Ingeniería en Computación de la Facultad de Ciencias Físico-Matemáticas y Naturales, Universidad Nacional de San Luis.

5. REFERENCIAS

- [1]. Sergio Colado, Abelardo Gutiérrez, Carlos J. Vives, Eduardo Valencia. SMART CITY: Hacia la gestión inteligente. Editorial: S.A. MARCOMBO.
- [2]. Fundación País Digital. País Digital: Smart Cities. <http://www.paisdigital.org/PD/smart-cities/>.
- [3]. Eduard Martin. Economía Digital: <http://www.expansion.com/economia-digital/protagonistas/2017/12/15/5a0b3846ca4741123c8b45fe.html>. España. 2017.
- [4]. R. Baskerville, L. Levine, J. Pries-Heje, S. Slaughter. How Internet Software Companies Negotiate Quality. IEEE Computer, 2001.
- [5]. Roger S. Pressman. "Ingeniería del Software: un enfoque práctico. Séptima Edición". Ed: McGraw-Hill, 2006.
- [6]. Peter G. Neumann. Illustrative Risks to the Public in the Use of Computer Systems and Related Technology desde 1987 de sitio web <http://www.csl.sri.com/neumann/illustrative.html>
- [7]. Tom Demarco. Software Project Management: A Guide for Service Providers. Editorial: Pearson Education (Us), 1982.
- [8]. Fenton y Pfleeger. "Software Metrics: A Rigorous and Practical Approach". Ed: PWS PUBLISHING COMPANY, 1997.
- [9]. UNE-EN ISO 9000:2000. Sistemas de gestión de la calidad Fundamentos y vocabulario. España. 2000.
- [10]. Piattini, M., García F., Caballero, I. "Calidad de los Sistemas Informáticos". Editorial Alfaomega. España. 2007.
- [11]. Fenton, N. "Viewpoint Article: Conducting and Presenting Empirical Software Engineering". 2001.
- [12]. Briand, L., Morasca, S. y Basili, V. "Property-Based Software Engineering Measurement". 1996.
- [13]. IEEE - Software Engineering Standards, Standard 610.12-1990, 1993.
- [14]. Ragland, B. Measure, Metric or Indicator: What is the Difference? Ed: Crosstalk. 1995.
- [15]. Olsina L., Pesotskaya E., Covella G., Dieser A. Bridging the Gap between Security/Risk Assessment and Quality Evaluation Methods. Publicada en ASSE Argentine Symposium on Software Engineering. 2012-2013.
- [16]. Ian Sommerville. "Ingeniería de Software Séptima edición". Ed: PEARSON Addison Wesley. 2005.
- [17]. Alberto Fernandez, Holger Billhardt. Dynamic Co-ordination of Open Fleets in Urban Environments. SURF: Intelligent System for integrated and sustainable management of URban Fleets. Centre for Intelligent Information Technologies (CETINIA). Proyecto: Ecobike Solutions: Gómez Pérez, Manuel y López González, Julio. Gestión de sistemas de alquiler de bicicletas. 2016.