

Desarrollo de software aplicando una propuesta metodológica diseñada en la Universidad Católica de Pereira¹

Software development applying a methodological proposal designed at Universidad Católica de Pereira

L. E. Peláez, A. Toro, D. L. Carvajal, D. C. López,

Recibido Noviembre 10 de 2013 – Aceptado Noviembre 30 de 2013

Resumen— El presente artículo es resultado del proyecto de investigación “Validación de un modelo propio para el desarrollo de software que permita la mejora del proceso y el producto en las MyPymes y Pymes de Colombia que construyen software”, el cual se lleva a cabo en la Universidad Católica de Pereira al interior del Grupo de investigación en Innovación e Ingeniería (GIII-UCP).

El modelo comprende una serie de fases y actividades alrededor del desarrollo y la gestión: requerimientos, arquitectura, construcción, pruebas, implantación, planificación y estimación, SQA, configuración de versiones, reporte de incidencias y documentación.

Para llevar a cabo la validación del modelo se planea en un primer momento, realizar desarrollos de software siguiendo la metodología propuesta. Uno de estos desarrollos es “Sistema paz & salvo” - que permite el manejo de la información y

documentos en la nube-. El objetivo general de este trabajo es la aplicación del modelo propio en el proceso de desarrollo y gestión de dicho sistema para, posteriormente, en la ejecución de futuros proyectos, tener los medios necesarios para validar la calidad del software desarrollado frente a otros similares existentes en el mercado.

Palabras Clave— SQA, Software, ingeniería del software, gestión documental.

Abstract—This article is the result of the research project entitled “Validation of a model for the software development in order to improve the process and the product in MSMEs and SMEs which create software in Colombia “. It is carried out at Universidad Católica de Pereira by the research group Innovation and Engineering (GIII-UCP).

The model comprises a number of phases and activities around the development and management: requirements, architecture, construction, testing, implementation, planning and estimating, SQA, versions’ settings, incident reporting and documentation.

In order to validate the model, it is planned at first; to carry out software development following a specific methodology. One of these developments is “a good standing certificate system” - that allows the management of information and documents in the cloud-. The general objective of this work is the application of an own model in the process of development

¹Producto derivado del proyecto de Investigación “Validación de un modelo propio para el desarrollo y la gestión de proyectos de software en MIPYMES de Colombia que construyen software”, apoyado por la Universidad Católica de Pereira, a través del grupo de Investigación e Innovación en Ingeniería GIII.

L. E. Peláez, A. Toro, son docentes de la Universidad Católica de Pereira, L. Carvajal, D. C. López, son graduadas del programa de Ingeniería de Sistemas y Telecomunicaciones de la misma Universidad, Av. de Las Américas No. 49-95, Pereira (correos e.: luis.pelaez@ucp.edu.co, alonso.toro@ucp.edu.co, diana.carvajal@ucp.edu.co, diana.lopez@ucp.edu.co).

and management of the system. So that, the means to validate the quality of the software developed, in the implementation of future projects, contrasted to others similar existing in the market, will be ready.

Key Words— *SQA, software, software engineering, document management.*

I. INTRODUCCIÓN

En los diferentes sectores del desarrollo social y económico se pueden identificar problemas reales que pueden ser resueltos por la ingeniería del software mediante una estrategia adecuada, esto quiere decir, aplicando métodos para asegurar la calidad en el proceso y del producto; en el ámbito del software esto se conoce como SQA –Software Quality Assurance–.

La ingeniería del software es “*una disciplina que integra el proceso, los métodos, y las herramientas para el desarrollo de software de computadora*” [1]. En este sentido, si bien el proyecto no se centra en la ingeniería del software como objeto de estudio, si se hace necesario alinearse a una concepción aceptada mundialmente para, desde allí, comprender lo que propondrá el modelo a aplicar.

Para el caso de este proyecto, se da una alternativa de solución para la expedición de paz y salvos en instituciones que lo requieran, con el desarrollo de un aplicativo orientado a la nube² y especializado en la gestión documental, en este caso documentos tipo paz y salvo. Además, el desarrollo de este aplicativo presenta las condiciones para la aplicación de un modelo de desarrollo y gestión de software.

El propósito de estudio se concentra en la aplicación de un modelo de desarrollo y gestión de software, en sistemas que permiten la gestión de la información y documentos en la nube, tomando la gestión documental como “*el conjunto de actividades que permiten coordinar y controlar aspectos como la creación, recepción, organización, almacenamiento, preservación, acceso y difusión de documentos.*” [2].

En el estado de la técnica de este tipo de software, se ha encontrado que la expedición de paz y salvos es una actividad recurrente y necesaria en instituciones de cualquier tipo, es por esto que se han desarrollado algunas aplicaciones como Paysa, el sistema para la gestión de paz y salvos de la Universidad de Antioquia³ o el Sistema de paz y salvos para el impuesto predial ofrecida por la casa de software

Sysman Software; si bien estas alternativas dan una solución al problema, lo hacen de una manera parcial, ignorando la variedad de instituciones que lo necesitan.

Es por lo anterior, que el proyecto apunta a una solución que abarque un universo poblacional y organizacional más amplio, con capacidad de acoplamiento a diferentes contextos y circunstancias, convirtiéndose en una nueva y mejorada alternativa al proceso acostumbrado en cuanto a la expedición de paz y salvos, que pretende dar una manejo eficaz y ordenado a la información y a los documentos. De esta manera, el proyecto considera que las instituciones lograrán un mayor nivel de eficiencia.

Para el desarrollo de este software se aplica una propuesta propia, que al tiempo ha sido basada en propuestas reconocidas en el ámbito internacional.

La metodología escogida, necesariamente debe ir acompañada de un modelo de ciclo de vida del software, el cual brinda una visión del proceso, desde la concepción hasta su fin y especifica en qué momento se debe ejecutar cada una de las etapas de la metodología.

El modelo de ciclo de vida del software utilizado para este proyecto es el modelo evolutivo incremental, que permite “*aplicar secuencias lineales en su desarrollo, obteniendo de cada una un incremento del software. El primer incremento da como resultado un producto esencial, afrontando requisitos básicos, que es evaluado por el cliente, para determinar el siguiente incremento según los requisitos establecidos, este proceso se repite hasta que se elabora el producto completo*” [1].

El desarrollo del proyecto se hace teniendo en cuenta todos los contenidos abordados anteriormente, basándose en el objetivo general, el cual plantea analizar y desarrollar una solución para documentos tipo paz y salvo aplicando un modelo de desarrollo y gestión de software formulado en la Universidad Católica de Pereira.

II. EL PROBLEMA

Para llevar a cabo ciertas actividades, tales como retiros o ingresos a servicios en instituciones y empresas, usualmente es requerido un certificado de paz y salvo, el cual da fe del estado sin deuda de los usuarios. Este trámite generalmente causa traumatismos tanto para los usuarios como para las instituciones, ya que el proceso para expedirlo por lo regular es manual y se hace necesario presentarse en cada dependencia que pueda llegar a estar involucrada para conseguir una firma, un sello o saldar deudas pendientes de las que muchas veces no se tiene conocimiento. Todo esto genera altos costos y demoras en la expedición de dicho certificado.

² Joyanes Aguilar [19] aborda el concepto como un nuevo paradigma tecnológico de gran impacto social. La Nube (*The Cloud*) es el conjunto “infinito” de servidores de información (computadores) desplegados en centros de datos, a lo largo de todo el mundo donde se almacena millones de aplicaciones Web y enormes cantidades de datos (*big data*), a disposición de miles de organizaciones y empresas, y cientos de miles de usuarios que se descargan y ejecutan directamente los programas y aplicaciones de software almacenados en dichos servidores tales como Google Maps, Gmail, Facebook, Tuenti o Flickr.

³ Sistema para la gestión de paz y salvos de la Universidad de Antioquia

Lo anterior, permite formular el siguiente interrogante:
¿Ofrecer la gestión de un documento tipo paz y salvo a través de la web, *ayudaría a agilizar el proceso que deben llevar a cabo las personas frente a las solicitudes que lo requieren?*

III. OBJETO DE ESTUDIO

Por la particularidad del área en la que se problematiza y para el caso que ocupa al proyecto, se reconoce como objeto puntual el estudio del modelo de desarrollo de software en sistemas que permiten la gestión de información en la nube.

Se pretende al final reconfigurar el aseguramiento de la calidad para compararlo con las metodologías en vigencia. Sin embargo, dicha comparación no hace parte del estudio, solo la aplicación del modelo.

IV. DELIMITACIONES Y ALCANCES DEL PROYECTO

Este proyecto tiene como objetivo llegar a la implantación, a nivel de prueba, de la solución desarrollada para la generación de certificados tipo paz y salvo, realizando las correcciones a los posibles errores que se presenten, con el fin de lograr que el sistema cumpla con las políticas de calidad propuestas.

Para la realización de dichas pruebas se consideraron como limitantes aspectos tales como:

- Poca capacidad de almacenamiento en el servidor de pruebas.
- Errores de red, herramientas, equipos o plataforma tecnológica sobre la cual se va a montar la solución.
- Inconvenientes en la migración de la información.

De acuerdo con lo anterior, la implementación de este proyecto está delimitada a un servidor que cuente con todos los requerimientos técnicos necesarios.

Finalmente, es importante verificar el nivel de calidad logrado con el modelo, sin embargo sólo se llegó a la aplicación de éste, para, durante la ejecución de futuros proyectos, tener los medios necesarios para validar la calidad del software desarrollado frente a otros similares existentes en el mercado.

V. ENFOQUE METODOLÓGICO O MODELO TEÓRICO

El enfoque metodológico está basado en los modelos internacionales PSP⁴ (SEI, 1995) [3], TSP⁵ (Watts

⁴ PERSONAL SOFTWARE PROCESS: metodología desarrolla por el SEI, diseñada para ayudar a los ingenieros de software a producir software de calidad; es un proceso de auto mejoramiento, diseñado para ayudar a controlar, administrar y mejorar la forma en que se trabaja individualmente.

⁵ TEAM SOFTWARE PROCESS: Guía para ingenieros desarrolladores de productos software. El uso de TSP ayuda a las organizaciones a establecer una práctica de ingeniería madura y disciplinada que produce software seguro y fiable en menos tiempo y en costos más bajos. <http://www.sei.cmu.edu/tsp/> [Consultado en 25-11-2013].

Humphrey, 1996) [4], SCRUM⁶ (Takeuchi & Nonaka, 1986) [5], Métrica⁷ (Ministerio de Administración Pública, 2001) [6], MOPROSOFT⁸ (Secretaría de Economía de México 2005) [7], MPS-BR⁹ (SOFTEX, 2003) [8], RUP¹⁰ (1999) [9] que han servido como soporte para la formulación del modelo propio (en forma de metodología) para el desarrollo y gestión de proyectos del software propuesto.

La *Figura 1* muestra la forma en que el enfoque metodológico hace despliegue durante todo el proceso de desarrollo y gestión [10].



PUNTOS CLAVE EN EL ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE

LEPV,2012

Fig. 1. Modelo para el desarrollo y gestión de proyectos de software. (Peláez, 2012)

A continuación se realiza una breve descripción de cada una de las fases y procesos que componen la metodología. Es importante tener en cuenta que los procesos de desarrollo y gestión se realizan de manera paralela, retroalimentado el proceso.

A. Proceso de gestión

1) *Planificación y estimación (inicio)*: la planificación es entendida como “la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades de un proyecto para cumplir los requisitos del mismo” [11], lo que quiere decir que debe existir un control y monitoreo al personal, al producto, al proceso y al proyecto desde

⁶ SCRUM: Metodología ágil para equipos de trabajo desarrolladores de productos, basada en pequeños pasos y en la integración de cada pieza. <https://www.scrum.org/Resources/What-is-Scrum> [Consultado en 25-11-2013].

⁷ METRICA: Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información. <http://administracionelectronica.gob.es> [Consultado en 25-11-2013].

⁸ MOPROSOFT: es un modelo de procesos diseñado en México como parte del programa PROSOFT de la Secretaría de Economía, dirigido a elevar el nivel de madurez de capacidades de las pequeñas y medianas empresas mexicanas dedicadas al desarrollo y mantenimiento de software. <http://www.moprosoft.com.mx> [Consultado en 27-11-2013].

⁹ MPS-BR: es un programa creado en 2003 por Softex para mejorar la capacidad de desarrollo de software en las empresas brasileñas. <http://www.softex.br> [Consultado en 27-11-2013].

¹⁰ RUP: Mejores prácticas para el diseño, la implementación y la gestión eficaz de proyectos de software. <http://www01.ibm.com/software/rational/rup/> [Consultado en 25-11-2013].

su concepción preliminar hasta su despliegue operativo completo.

Con base en lo anterior, se llevan a cabo los siguientes pasos para la planificación del proyecto:

- **Iniciación y alcance del proyecto:** en este punto se elabora la carta del proyecto, se hace un estudio de viabilidad, se definen los alcances y se determinan los objetivos.

- **Planificación del proyecto:** se realiza estimación de costos, tiempo y esfuerzo. Para este proceso es necesario tener claros los objetivos, que los requerimientos estén bien especificados y disponibilidad de toda la información recaudada hasta el momento; dentro del proceso se contemplan diferentes técnicas y métodos de estimación, razón por la cual es conveniente utilizar el que más se acomode al contexto del proyecto y sus variables.

2) *SQA*: la garantía de la calidad del software es una actividad de protección que se aplica a lo largo de todo el proceso del software.

Dentro de la gestión de la calidad se asegura que tanto el producto como el proceso de producción de software cumplan con los requerimientos de calidad establecidos. Por tanto, el aseguramiento de la calidad es definido como “*un patrón sistemático y planificado de todas las acciones necesarias para afirmar con certeza que un producto es conforme con los requisitos técnicos establecidos*”. [12]

Esta actividad se debe realizar a lo largo de todo el proceso y considera que “*la calidad de un producto viene en gran medida determinada por la calidad de sus procesos*”. [13]

3) *Configuración de versiones*: desarrollo y aplicación de procedimientos para gestionar la evolución del software. Se entiende también como el proceso de registrar y procesar los cambios, considerando cómo se relacionan estos con los componentes del sistema y los métodos utilizados para identificar las diferentes versiones. En este proceso se define:

- Lo que se debe gestionar
- El responsable de este procedimiento
- Las políticas de gestión de la configuración
- Otras

Se hace conveniente entonces comprender que “*una versión es cada uno de los siguientes estados diferentes, en los que un elemento de configuración se encuentra durante el desarrollo o durante la evolución del software*” [13].

Por su parte,

“*la gestión de la configuración es la disciplina que aplica dirección y control técnico y administrativo para: identificar y documentar las características físicas y funcionales de los*

elementos de configuración, controlar los cambios de esas características, registrar e informar del procedimiento de los cambios y el estado de la implementación, y verificar la conformidad con los requisitos especificados” [14]

Por lo tanto, la gestión de la configuración de versiones no solo abarca el producto final entregado sino también las innumerables versiones de los diferentes elementos que se van produciendo como resultado de la actividad de la ingeniería, tales como código fuente, documentación, especificaciones, notas técnicas, entre otros.

Este proceso no es tenido en cuenta en el desarrollo del proyecto ya que sólo se dispone de una versión, tanto del software como de su documentación.

4) *Incidencias y requisitos*: en esta etapa se hace un proceso de revisión y evaluación del proyecto, mediante la validación y verificación del cumplimiento de los requisitos, teniendo en cuenta funcionalidades, costos y plazos definidos inicialmente.

5) *Documentación*: actividad en la cual se lleva registro del proceso de implantación del software con el fin de tener los soportes y la información necesaria del procedimiento. La información es entendida como “*Los datos o materia informacional relacionada o estructurada de manera actual y potencialmente significativa*” [15].

Dentro de los documentos a considerar están los siguientes:

- plan de implantación
- requisitos del ambiente para la implantación
- manual de configuración del sistema
- evaluación del proceso de implantación
- acta de cierre del proceso de implantación

B. Proceso de desarrollo

1) *Requerimientos*: los requisitos entendidos como “*la propiedad que un software desarrollado o adaptado debe tener para resolver un problema concreto*” [16], son obtenidos mediante un proceso de educación, el cual consiste en la aplicación de diferentes técnicas, para obtener de los usuarios, los requerimientos que consideran necesarios para el sistema. Posteriormente, los requerimientos son analizados para lograr su definición, clasificación y ponderación; seguidamente, es necesario que el cliente los revise y retroalimente para que finalmente sean aprobados, y así obtener los requisitos del proyecto.

2) *Arquitectura*: definida como el “*proceso fundamental donde se establece un marco estructural del sistema con sus componentes, relaciones y entorno*” [17]. La arquitectura del software desde el punto de vista de diseño se hace respetando los siguientes principios:

- Abstracción
- Acoplamiento y cohesión
- Descomposición y popularización
- Encapsulamiento y ocultación de información
- Separación de interfaz e implementación
- Suficiencia, compleción y sencillez
- Principios de diseño de software

La arquitectura del sistema es “*la organización fundamental de dicho sistema plasmada en sus componentes, las relaciones entre estos y con el entorno, y los principios que guían su diseño e implementación*”. [18]

Dentro de la especificación de la arquitectura se hace necesaria la descripción de los actores del sistema, requerimientos de hardware y de software, especificación de la interacción con la tecnología, procesos, subprocesos y modularidad. Es por esto que se hace necesaria la utilización de diferentes modelos y diagramas que especifican de manera concreta los puntos anteriores. Algunos de ellos son los siguientes:

- Modelo del problema
- Modelo de la solución
- Modelos de casos de uso
- Diagramas de secuencia
- Diagramas de actividades
- Prototipos no funcionales
- Diagrama de navegación
- Interfaces de usuario
- Diagramas de clases
- Diagramas de objetos

3) *Construcción*: en este apartado se define un conjunto de actividades que engloban fundamentalmente la codificación, verificación de código, depuración y criterios de prueba; la construcción está directamente relacionada con las demás actividades, pero especialmente con el diseño, las pruebas y el mantenimiento; para construir software es necesario un esfuerzo previo en el diseño y otro posterior para comprobar que funciona bien.

Al respecto, la IEEE menciona que “*construir consiste en crear software mediante una combinación de codificación, verificación, pruebas unitarias y depuración*.” [13]

En esta etapa se implementan las bases de datos, se codifica los módulos, se documenta el código, se hacen pruebas unitarias por parte del desarrollador y se elabora el manual de usuario; todo esto teniendo en cuenta la reutilización de código, la minimización de la complejidad y las buenas prácticas en el desarrollo de software.

4) *Pruebas*: una vez generado el código fuente el software debe ser probado para descubrir y corregir el

máximo de errores posibles, puesto que el éxito de la prueba radica en la detección de errores tanto en el propio proceso como en el software obtenido. Así pues, “*la prueba de software es todo proceso orientado a comprobar la calidad del software mediante la identificación de fallos en el mismo, la prueba implica necesariamente la ejecución del software*” [13].

Para llevar a cabo este proceso es indispensable la creación de algunos documentos y el seguimiento de pasos, como los que a continuación se especifican:

Plan de pruebas: documento formal que guía el proceso; debe contemplar:

- Encabezado: contiene un resumen de la funcionalidad, los responsables por parte del cliente y el líder de proyecto de pruebas.
- Alcance: el alcance y los supuestos indicados por el cliente.
- Matriz de riesgos: herramienta de control y gestión que identifica actividades y sus prioridades dentro de las pruebas, esta es elaborada con las funcionalidades, impacto y probabilidad de fallo.
- Estrategia de prueba: define el qué, cómo y cuándo de las pruebas.
- Requerimientos para las pruebas: recursos necesarios, su descripción y responsables.
- Cronograma: calendario de actividades donde se deben especificar las actividades a realizar, su peso (en porcentaje), fecha de inicio, fecha de finalización y tiempo estimado en días.

Casos de prueba: documentos que deben ser diseñados antes de iniciar la prueba. Contienen el conjunto de condiciones y variables bajo las cuales se realizarán las pruebas. En él debe contemplarse lo siguiente:

- Encabezado: nombre del cliente, nombre del proyecto, versión, tipo de prueba, fecha de creación, creado por y modificado por.
- Descripción de los pasos: es necesario tener una explicación detallada y específica de cada uno de los pasos necesarios para probar un elemento.
- Funcionalidad a probar
- Resultado esperado: comportamiento deseado del sistema bajo las condiciones de la prueba.
- Número de Bug (incidencia): identificador de errores encontrados en la ejecución de la prueba.
- Proceso de reporte de incidencias: proceso o protocolo que se sigue para documentar, reportar y dar solución a las incidencias encontradas.

Informe de cierre de las pruebas: cuando ya no se registran errores sea porque no son encontrados o ya han sido corregidos se considera que las pruebas han terminado. En este punto es necesario documentar la finalización de las pruebas.

Carta de aceptación del cliente (en caso de ejecutarse las pruebas de aceptación)

5) *Implantación*: transferencia del software producido al entorno para el que fue desarrollado, en este punto se hace necesario contar con una estrategia de implantación en la que se debe especificar:

- Equipo que va a llevar a cabo la implantación
- Configuración del ambiente
- Capacitación de los usuarios, administradores y técnicos.
- Conversión de datos y cambios en el sistema

Este paso no es tenido en cuenta dentro del desarrollo del proyecto ya que en los objetivos planteados sólo se llega hasta la implementación.

Mantenimiento del producto: actividad post entrega, donde se realizan cambios al producto inicialmente acordado ya sea para corregir fallos, mejorar su rendimiento o adaptar el producto a un entorno modificado. En las actividades de mantenimiento se encuentran las siguientes:

- Entender los cambios que se deben realizar
- Modificar el software y actualizar la documentación
- Volver a realizar las pruebas del software (pruebas de regresión), además de probar específicamente las partes añadidas o modificadas.

Este último paso no es tenido en cuenta dentro del proyecto ya que solo se llega a la implementación del sistema.

IV. RESULTADOS

Como resultado de este proyecto se generó una solución que permite la gestión de paz y salvos en instituciones educativas por medio del seguimiento detallado de la metodología propuesta por el grupo de investigación, que tiene como fin presumir la calidad del producto ya que se desarrolló y gestionó de manera disciplinada.

Este desarrollo tiene por nombre “*Sistema paz & salvo*”. En dicho sistema se pueden encontrar tres roles para el manejo de la información denominados *Solicitante*, *Administrativo* y *Administrador del sitio* (Figura 2); el primero de ellos tiene la posibilidad de consultar su estado en el sistema y en caso de estar a paz y salvo este podrá exportar su certificado (Figura 3). En el módulo administrativo las dependencias de la institución podrán registrar los usuarios y asignar el estado

de deuda con cada uno de ellos y por último, el administrador del sitio tiene la autoridad de administrar los usuarios, las dependencias y la institución (Figura 4).



Fig. 2. , Pantalla de inicio Sistema Paz & Salvo



Fig. 3. Pantalla de administración de la institución



Fig. 4. Certificado Paz y Salvo

C. Marco propuesto para la validación del modelo

Para la validación del modelo es necesaria la comparación y evaluación del sistema desarrollado con sistemas similares a él, desarrollados con diferentes metodologías o sin ninguna de ellas; para este proceso se creó al interior del grupo un instrumento basado en los atributos de calidad planteados por la ISO en la norma 9126. Este instrumento fue aplicado a los usuarios que utilizan sistemas similares a este.

Con estos recursos y con los resultados obtenidos, de la aplicación de la encuesta a usuarios de sistema Paz & salvo, se podrá comparar, y por tanto validar la calidad del producto desarrollado con la metodología propia planteada.

V. CONCLUSIONES

Aunque no es una evidencia propia del problema de investigación, se confirma que desarrollar un proyecto de software siguiendo adecuadamente una metodología para la gestión y el mismo desarrollo, permite tener procesos estandarizados, equipos de trabajos acoplados, desarrollos con precisión, presunción de mayor calidad y fácil adaptabilidad.

Lo anterior, contrastado en las causas por las que las Mipymes en Colombia no acostumbran seguir modelos, guías o metodologías¹¹, puede reafirmar que una razón de peso es que los profesionales encargados de construir software no se sientan identificados con la imposición de metodologías internacionales.

Por otro lado, el seguimiento preciso de un modelo de este tipo, permite asegurar mayor efectividad en la relación esfuerzo-tiempo-costos. Por ejemplo, cuando se hace una definición detallada de la arquitectura del sistema, se tiene una alta probabilidad de realizar un proceso ordenado, con definiciones exactas en cuanto a todo lo necesario para lograr un producto que cumpla con los requerimientos del cliente con mejores resultados que cuando no se cumple el proceso ordenado.

Dentro de la ejecución de la metodología uno de los puntos más relevantes es el aseguramiento de la calidad, ya que dentro de esta actividad se asegura que tanto el producto como el proceso cumplen con los requerimientos de calidad establecidos.

En la misma línea, se comprueba la importancia de la documentación para un proyecto de ingeniería de este tipo. La documentación en todas las etapas del proyecto es indispensable para una buena gestión del proceso, teniendo en cuenta los cambios que pueden llegar a realizarse en cada etapa, ya que estos son el soporte de todas las actividades realizadas.

La creación de software es una actividad que puede ser entendida como aquella que aporta considerablemente al desarrollo económico y social; podría inclusive, ser un producto clave en el avance de la sociedad, ya que con este se puede llegar a suplir necesidades que redundan en menos tiempo, menores costos financieros y mayor efectividad de respuesta; todo ello mejoraría la calidad de vida de las personas. Es por esto que se debe seguir un proceso responsable y detallado que permita generar desarrollos significativos, serios y tratados responsablemente.

Es relevante que se pueda iniciar, en el marco de otro proyecto, la medición de la calidad de los productos que se elaboren con el modelo aplicado en este proyecto; siempre

¹¹ Según estudio hecho por el mismo grupo de Investigación en la Universidad Católica de Pereira durante los años 2009 y 2012.

teniendo en cuenta la mirada del cliente y la del usuario, que en algunos casos, no son las mismas.

BIBLIOGRAFÍA

- [1] R. S. Pressman, Ingeniería del software, un enfoque práctico, Quinta ed., Madrid: MC Graw Hill, 2001.
- [2] P. Russo Gallo, Gestión documental en las organizaciones, Barcelona: Editorial UOC, 2009.
- [3] SEI, PSP: Personal Software Process, Software Engineering Institute, 1995.
- [4] SEI, TSP: Team Software Process, Software Engineering Institute, 2000.
- [5] H. Takeuchi y I. Nonaka, «What is Scrum?», 1986. [En línea]. Available: http://www.iei.liu.se/fek/svp/723g18/articles_and_papers/1.107457/TakeuchiNonaka1986HBR.pdf. [Último acceso: 25 Noviembre 2013].
- [6] Ministerio de Hacienda y Administraciones Públicas de España, Métrica V.3, España: Ministerio de Administraciones Públicas, 2001.
- [7] Secretaría de Economía de México, MoProSoft, México: Secretaría de Economía de México, 2005.
- [8] SOFTEX: Asociación para promoción de la excelencia del software brasileño, MPS-BR, Brasil: SOFTEX, 2003.
- [9] IBM, «Rational Unified Process (RUP)», 1999. [En línea]. Available: <http://www-01.ibm.com/software/rational/rup/>. [Último acceso: 2013 Noviembre 25].
- [10] Grupo de investigación TICS de la UCP, «Modelo para el aseguramiento de la calidad en la construcción de software en las PYMES colombianas: Desarrollo y gestión», Pereira, 2013.
- [11] PMI, PMBOK: Project Management Body of Knowledge, Project Management Institute, 2008.
- [12] IEEE, Std. 730: Standard for Software Quality Assurance Plans, New York: The Institute of Electrical and Electronics Engineers, 2002.
- [13] M. Silicia, D. Rodríguez y S. Sánchez, Ingeniería del Software, un enfoque desde la guía SWEBOK, México: Alfa Omega, 2012.
- [14] IEEE, Std. 828: Standard for Configuration Management Systems and Software Engineering, International Organization for Standardization, 2012.
- [15] I. Páez Urdaneta, Gestión de la inteligencia, aprendizaje tecnológico y modernización del trabajo informacional., Caracas: Univerdiad Simón Boívar, 1992.
- [16] IEEE, «Guide of the Software Engineering Body of Knowledge», 2004. [En línea]. Available: <http://www.computer.org/portal/web/swebok>. [Último acceso: 20 08 2013].
- [17] I. Sommerville, Ingeniería del software, España: Pearson Educación S.A, 2005.
- [18] IEEE, Std. 42010: Systems and software engineering, Architecture description., The Institute of Electrical and Electronics Engineers, 2007.
- [19] L. Joyanes Aguilar, «Computación en la Nube e innovaciones tecnológicas: el nuevo paradigma de la sociedad del conocimiento», p. 21, 2011.

Luis Eduardo Peláez Valencia, actualmente se desempeña como Vicerrector académico de la Universidad Católica de Pereira. Magister en Ingeniería del software, Master en en Gestión y Desarrollo de Proyectos Software, Especialista en propiedad intelectual: propiedad industrial, derechos de autor y nuevas tecnologías. Ingeniero de sistemas. Investigador del Grupo de Investigación en Innovación e ingeniería (GIII) de la Universidad Católica de Pereira. Profesor invitado de posgrado en la Universidad Pontificia Bolivariana (UPB). Asesor de la Comisión Nacional de Aseguramiento de la Calidad de la Educación Superior CONACES 2013-2015.

Alonso Toro Lazo, actualmente se desempeña como docente en la Universidad Católica de Pereira. Estudiante de Maestría en Gestión y Desarrollo de Proyectos de Software. Ingeniero de Sistemas y Telecomunicaciones. Investigador del Grupo de Investigación e Innovación en Ingeniería (GIII-UCP) de la misma universidad.

Diana Lizeth Carvajal Portilla, estudiante de décimo semestre de Ingeniería de Sistemas y Telecomunicaciones, residente en línea del Grupo de Investigación en Innovación e Ingeniería (GIII) de la Universidad Católica de Pereira.

Daniel Carolina López López, estudiante de décimo semestre de Ingeniería de Sistemas y Telecomunicaciones, residente en línea del Grupo de Investigación en Innovación e Ingeniería (GIII) de la Universidad Católica de Pereira.