



Proceso de desarrollo de software con técnicas de creatividad, para las Pymes de la ciudad de Neiva

Software development process with creativity techniques, for Neiva city SMEs

Eduardo Martínez-Vidal¹, Sandra V. Hurtado-Gil²

¹Estudiante Maestría en Gestión y Desarrollo de Proyectos de Software, Universidad Autónoma de Manizales y Docente Corporación Universitaria del Huila, Colombia. Email: emartinez@corhuila.edu.co

²Departamento de Sistemas e Informática, Facultad de Ingenierías, Universidad de Caldas, Colombia. Email: sandra.hurtado@ucaldas.edu.co

Recibido: 28 octubre, 2017. Aceptado: 23 marzo, 2018. Versión final: 21 junio, 2018.

Resumen

El proyecto que se presenta en este artículo se origina a partir de una necesidad identificada en empresas de desarrollo de software del municipio de Neiva, relacionada con hacer uso de buenas prácticas en el proceso de desarrollo de software, que permitieran la construcción de productos de calidad con un factor diferenciador. Para dar respuesta a esta necesidad, se realizó la adaptación de un proceso de desarrollo de software, a partir del proceso DAD (*Disciplined Agile Delivery*), e incorporando técnicas creativas para la creación de productos propios. Este proceso fue aplicado en una Pyme, permitiendo no solo la validación en un contexto real, sino también contribuyendo a la construcción de un producto propio con características atractivas.

Palabras clave: atributos de calidad; DAD (*Disciplined Agile Delivery*); métodos ágiles; proceso de desarrollo de software; Pymes (pequeñas y medianas empresas); técnicas de creatividad.

Abstract

The project presented in this article born from a need identified in software development companies in Neiva city, about to the use of good practices in the software development process, allowing the construction of quality products with a differentiating factor. To respond to this need, we adapt a software development process, based on DAD (*Disciplined Agile Delivery*) process, and incorporating creative techniques for the creation of own products. This process was applied in one small enterprise, allowing not only the validation in a real context, but also contributing to the construction of one own product with attractive characteristics.

Keywords: quality attributes; DAD (*Disciplined Agile Delivery*); agile methods; software development process; SMEs (small and medium-sized enterprises); creativity techniques.

1. Introducción

La industria del software está definida en Colombia como un factor relevante en el desarrollo del país [1], pero debe enfrentar grandes retos en una industria globalizada,

principalmente relacionados con capacidades y herramientas para aprovechar las oportunidades de mercado [2].

En el municipio de Neiva, departamento del Huila, existe un incipiente grupo de empresas dedicadas al desarrollo de software, que desean ser reconocidas en la región (Suroccidente colombiano) y en el país. Sin embargo, una encuesta realizada entre los empresarios, durante la fase de diagnóstico de este proyecto, mostró que muchas de estas empresas (el 57.14 % de las empresas encuestadas) tienen un grupo de 5 o menos personas, y la gran mayoría (el 78.26 % de las empresas encuestadas) no tienen definido ningún proceso o metodología para el desarrollo de software.

Esta situación no es desconocida en nuestro país [3] [4] [5], donde las pequeñas y medianas empresas – Pymes – debido a sus limitados recursos (tanto humanos como financieros) no invierten apropiadamente en la aplicación de buenas prácticas en sus desarrollos. Sin embargo, existen diversos antecedentes que muestran como la creación o adaptación de procesos de desarrollo de software contribuye a mejorar diferentes aspectos (calidad, productividad, etc.) en las Pymes desarrolladoras de software.

En algunos de estos antecedentes [6] [7] [8] [9] se establecen procesos basados en modelos como CMMI (*Capability Maturity Model Integrated*)¹ o PSP (*Personal Software Process*)², combinados con métodos ágiles, logrando aumentar el porcentaje de cumplimiento de algunas de las áreas de CMMI o contribuyendo al logro de certificaciones de modelos de calidad, pero a veces con algunas dificultades en la implementación completa del proceso.

Otros artículos [10] [11] [12] [13] se centran en la mejora de procesos, y proponen herramientas o marcos de trabajo para ayudar a las organizaciones en la parte de apropiación y aplicación de mejoras, especialmente en los procesos de desarrollo.

Todos estos antecedentes, aunque totalmente válidos, no tenían en cuenta un elemento importante para las empresas de desarrollo de software de la ciudad de Neiva: el desarrollo de productos propios con algún factor diferenciador. Es entonces cuando se propone el objetivo de la investigación: Adaptar un proceso de desarrollo de software incorporando técnicas creativas, que permita crear productos propios en las Pymes de desarrollo de software del municipio de Neiva.

Después de analizar varios procesos de software existentes y varias técnicas creativas, se propone la adaptación del proceso DAD (*Disciplined Agile Delivery*) [14], el cual combina elementos de métodos ágiles como Scrum y XP (*eXtreme Programming*), con prácticas de un modelo más tradicional como UP (*Unified Process*)³. En esta adaptación se tienen en cuenta no solo las particularidades de las Pymes de la ciudad de Neiva, sino que también se integran técnicas creativas, presentando así un elemento diferenciador importante frente a otros procesos existentes, y que puede ser de gran utilidad para Pymes que deseen desarrollar un producto propio.

El proceso resultante se aplica en una prueba piloto en la empresa NuevasTIC, contribuyendo no solo a la mejora en la calidad del producto, el cual se desarrolla con atributos atractivos (resultado de la aplicación de las técnicas creativas), sino también al logro de la certificación en el esquema de calidad IT Mark⁴.

El resto del artículo se desarrolla de la siguiente forma: La sección 2 incluye una corta introducción a los conceptos, en la forma de un referente teórico; en la sección 3 se presenta la metodología empleada para el desarrollo del proyecto; la sección 4 presenta los resultados del proyecto, incluyendo una visión global del proceso, y, por último, en la sección 5 se presentan las conclusiones, las recomendaciones y los planteamientos para futuros trabajos de investigación.

2. Referente teórico

2.1. Procesos de software

Uno de los logros en el desarrollo de la ingeniería de software, es la definición de una guía para el cuerpo del conocimiento: SWEBOK [15], la cual permite caracterizar la profesión y presenta de manera global las prácticas, herramientas y métodos que se usan actualmente en el escenario de la ingeniería de software. Esta guía incluye quince áreas de conocimiento, entre las cuales se encuentra el área: “Procesos de Ingeniería de Software”, que define como uno de los temas la adaptación de procesos de software, lo cual establece el contexto general para el proyecto. En esta guía se menciona que actualmente existe una amplia variedad de modelos de ciclos de vida del software, que van desde los modelos lineales (como cascada), hasta los modelos ágiles.

¹<http://cmmiinstitute.com/>

²<http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=5283>

³<http://www.methodsandtools.com/archive/archive.php?id=32>

⁴Esquema de certificación para PYMES de TI. <http://it-mark.eu/>

En la industria del software, está tomando bastante fuerza el uso de procesos de desarrollo de software basados en las metodologías ágiles [16] [17]. Los procesos de desarrollo con enfoques ágiles permiten construir aplicaciones en ciclos cortos, con el propósito de finalizar los proyectos en el menor tiempo posible y entregando valor de manera constante al cliente, siendo ambos factores determinantes en el éxito de un proyecto.

Posiblemente los métodos ágiles más conocidos sean Scrum y XP, pero existen otros igualmente interesantes como Kanban, FDD (*Feature Driven Development*), Modelado ágil o DAD (*Disciplined Agile Delivery*).

El proceso DAD (*Disciplined Agile Delivery*) es un enfoque híbrido que extiende Scrum con estrategias probadas del modelado ágil (AM), programación extrema (XP) y el proceso unificado (UP) [14]. Este proceso incluye un ciclo de vida básico, el cual se ilustra en la Figura 1.

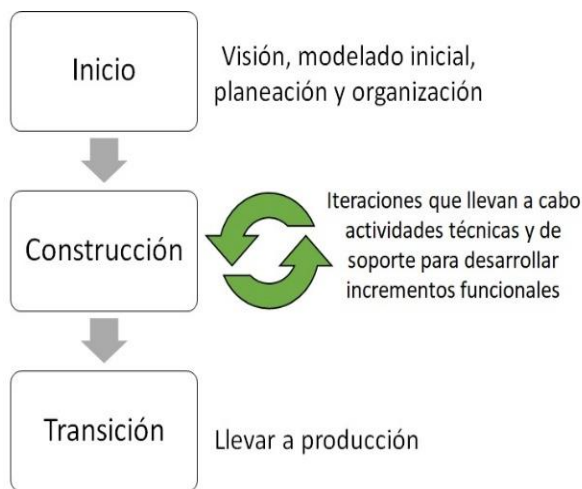


Figura 1. Ciclo de vida básico propuesto en el método DAD.

Fuente: elaboración propia, adaptado de:

<http://www.disciplinedagiledelivery.com/lifecycle/agile-lifecycle/>

En DAD se definen unos roles primarios y otros secundarios, considerando diferentes características y tamaños de proyectos. Dentro de los roles primarios, que serían los más adecuados para una Pyme, se encuentran: Product Owner, Team Member, Team Lead y Architecture Owner [14]. Además de los roles, DAD también define un conjunto de valores, principios y lineamientos para orientar el desarrollo de software, considerando a la vez un aspecto de flexibilidad para que cada organización pueda seleccionar las prácticas concretas para cada actividad del desarrollo.

2.2. Creatividad

“La creatividad solo puede atribuirse a los sujetos y no a las cosas, las personas tienen la facultad de adaptar y transformar el mundo a beneficio propio y a las necesidades en diferentes escenarios que se le pueda presentar a través de las ideas que solo este puede generar, dependiendo de la memoria, el razonamiento lógico, el aprendizaje y la percepción; el acto creativo rompe la cotidianidad obteniendo un grado más alto de innovación cuando el resultado no es conocido en lo ya existente, pudiéndose comprobar y valorar su calidad” [18].

En el proyecto se busca que las Pymes del municipio de Neiva no solamente desarrollen software con funcionalidad básica, sino que estos productos que son creados a partir de las ideas del equipo de trabajo de la organización puedan generar un impacto a un grupo objetivo mayor mediante la identificación de características atractivas que posiblemente contribuyan a la creación de productos innovadores.

Cuando se habla de características atractivas se está haciendo referencia al Modelo Kano, el cual sirve tanto como referente como técnica creativa. “El objetivo de esta técnica es la de entregar un producto funcional con atributos que van más allá de las expectativas del cliente enfocado en el posicionamiento del producto en el mercado determinado por la originalidad, innovación y utilidad” [19]. En este modelo se mencionan tres tipos de atributos de la calidad:

- **Atractiva:** Estos atributos proporcionan satisfacción cuando se logran plenamente, pero no causan insatisfacción cuando no se logran.
- **Unidimensional:** Estos atributos dan como resultado la satisfacción cuando se cumplen e insatisfacción cuando no se cumplen.
- **Obligatoria:** Estos atributos se dan por sentadas cuando se cumplen, pero dan lugar a insatisfacción cuando no se cumplen.

También es importante reflexionar sobre las características “cotidianas” de las cosas. A menudo, se da por hecho ideas preconcebidas sin pensar si son apropiadas o no en una situación particular. Es aquí donde contribuye la técnica del listado o lista de atributos. Esta es una técnica creativa que consiste en ir desmenuzando el problema en partes cada vez más pequeñas y buscar soluciones alternativas a esas partes [20], se van describiendo todos los componentes o atributos de un producto, un proceso o un problema y después se intenta hacer una lista de posibles alternativas a estos atributos dados.

Con estos dos pilares teóricos, de procesos de software y creatividad, se puede proceder con el desarrollo del proyecto.

3. Metodología

Para dar cumplimiento a los objetivos específicos del proyecto se aplicó una metodología soportada en etapas, cada una a su vez basada en procedimientos que permitieran alcanzar cada objetivo específico; el fundamento que se usó en la ejecución del proyecto fue el tipo de investigación aplicada experimental. A continuación, se presentan las etapas de la metodología.

3.1. Diagnóstico

Se analizaron los datos de la encuesta aplicada a los empresarios de la industria de software de la ciudad de Neiva interesados en la mejora continua de sus organizaciones mediante la implementación de procesos que les permita crear productos de calidad y al mismo tiempo desarrollar un producto propio, que los identifique en el mercado.

La muestra para la encuesta fue por conveniencia, tomada a partir de un conjunto de empresas convocadas a participar en un evento en Corhuila, mostrando así su interés en mejorar sus procesos de desarrollo de software. De 51 empresas registradas en Neiva como desarrolladoras de software, se convocaron a 12 que contaran con experiencia de más de dos años en desarrollo de software y un equipo de trabajo conformado por más de una persona. De estas empresas, 10 participaron en el evento y 7 respondieron la encuesta.

La aplicación del instrumento permitió identificar el contexto de las Pymes de desarrollo de software de la ciudad de Neiva, logrando establecer los fenómenos que influyen en la carencia de un proceso definido para el desarrollo de productos software. Se ubicó la problemática estableciendo las vulnerabilidades y sugerencias por parte de los empresarios, estos elementos se tuvieron en cuenta en la adaptación del proceso de desarrollo.

3.2. Diseño del proceso

En esta etapa se escogieron algunos procesos de desarrollo de software como son: UP, TSP (*Team Software Process*)⁵, ICONIX⁶, Scrum, XP y DAD,

seleccionados en el marco de las metodologías ágiles y tradicionales, para su caracterización, tomando de ellos las actividades que tuvieran participación con el cliente y al tiempo aporte al fortalecimiento de las fases de análisis de requisitos y diseño; de igual manera se hizo un estudio de algunas técnicas creativas como Canvas, Lluvia de ideas, Lista de atributos, Modelo Kano y Enfoque analógico, seleccionando aquellas de fácil aplicación y que estuvieran relacionadas con conceptos de satisfacción de los clientes.

Esta actividad se llevó a cabo mediante una revisión bibliográfica a través de artículos y libros, con el propósito de seleccionar los componentes que pudieran aportar al fortalecimiento de la productividad en las Pymes de la ciudad de Neiva, identificando técnicas que no sobrecargaran el proceso de desarrollo, pero conservando la calidad del producto e incentivando la participación del equipo de trabajo.

3.3. Prueba piloto

Esta parte de la metodología se implementó el proceso de desarrollo propuesto en la empresa de desarrollo NuevasTIC de la ciudad de Neiva, en la fabricación de un producto propio, que fue una aplicación web para hacer seguimiento de las contravenciones de acuerdo con la ley 1801 del 29 de Julio de 2016 del nuevo código de policía; siendo los clientes identificados las secretarías de gobierno de los municipios del departamento del Huila.

En esta actividad se realizó el respectivo seguimiento para identificar los aciertos y posibles mejoras en el proceso.

4. Resultados

De acuerdo con la metodología, y en concordancia con los resultados esperados en la investigación, se describen las actividades que se realizaron para la definición del proceso de desarrollo de software con técnicas creativas, y su aplicación en una empresa de desarrollo de software de la ciudad de Neiva del departamento del Huila.

4.1. Diagnóstico

El objetivo de esta etapa fue identificar el estado actual en las Pymes de desarrollo de software del municipio de Neiva; para alcanzar dicho propósito se aplicaron siete encuestas a empresarios del ámbito del desarrollo de

⁵<http://www.sei.cmu.edu/tsp/>

⁶Se define como un proceso de desarrollo de software práctico entre UP y XP - <http://www.informit.com/articles/article.aspx?p=167902>

software, en el marco de la actividad realizada por la institución de educación superior Corhuila perteneciente a la ciudad.

Las variables más relevantes que se tomaron en cuenta en la aplicación de la encuesta, para conseguir identificar el estado actual de las empresas de desarrollo de software en la ciudad de Neiva, se presentan a continuación.

4.2. Diseño del proceso

Se diseñó un proceso de desarrollo basado en las metodologías ágiles, considerando lo siguiente: el tamaño de la organización, la preparación del recurso humano y el tiempo de entrega de los productos.

Para el diseño del proceso se utiliza como base DAD, el cual, como se mencionó en el referente teórico, es un enfoque híbrido que presenta la estructura general del desarrollo, pero también es flexible en la selección de las prácticas específicas.

DAD incluye prácticas que no se encuentran incluidas en el Scrum como es el modelado y la documentación, ofreciendo la posibilidad de adaptar estrategias funcionales. Este proceso se identifica como la mejor opción para plantear un proceso de desarrollo ágil sin dejar a un lado la documentación, que es una actividad relevante en la entrega de productos de calidad, que a su vez es uno de los propósitos que se desea lograr en las Pymes de desarrollo de software del municipio de Neiva. Como este proceso ya integra buenas prácticas de otros procesos de desarrollo, coincide con el propósito buscado por el proyecto. Además, aunque no se encontraron antecedentes de la aplicación de DAD en Colombia, está basado en Scrum, del cual sí hay varios antecedentes de aplicación, permitiendo tener en cuenta las recomendaciones encontradas en estos antecedentes.

La definición del proceso contempla:

- Definición de roles.
- Definición de fases y actividades.
- Definición de técnicas y artefactos.

Los roles para el nuevo proceso se definieron a partir de los roles establecidos en Scrum, más un rol para la parte de gestión del proyecto, que se encuentra definido en DAD. Aunque son cuatro roles, dos de ellos pueden ser desempeñados por personas externas a la empresa, considerando que existen Pymes de menos de cuatro personas.

Tabla 1. Análisis de variables de las encuestas de la fase de diagnóstico.

Variable	Análisis
Tamaño de la organización	Tomando el número de integrantes que conforman las organizaciones encuestadas, se pudo establecer que se encuentran en el rango de dos (2) a un máximo de diez (10) personas entre el gerente y desarrolladores; pero el promedio de general es de tres (3) personas, indicando que las empresas de desarrollo de la ciudad de Neiva son relativamente pequeñas. Además, el talento humano utilizado en el área de programación está conformado principalmente por estudiantes de último semestre de carreras relacionadas con el desarrollo de software.
Roles	No hay roles asignados de carácter especializado en el equipo de trabajo, lo que se debe principalmente al poco recurso humano. A esto se suma que no hay un monitoreo y evaluación precisa del trabajo desarrollado por los mismos miembros del grupo.
Aplicación de metodologías en el proceso de desarrollo	Las empresas encuestadas conocen o han escuchado temas relacionados con las metodologías, calidad de software y procesos de desarrollo, pero estos no se aplican por el poco plazo que tienen para entregar los productos. La cultura de estas empresas es la de hacer entregas periódicamente de funcionalidades al cliente, y durante el proceso de funcionamiento en entorno de producción se van localizando las fallas de la aplicación, que son reportadas en el periodo de garantía, para la respectiva solución.
Actividades de mayor importancia en el proceso de desarrollo	Todos los encuestados coincidieron en que el levantamiento de información es importante y de mayor atención en el momento de construir una aplicación, seguido del análisis y diseño. Por otra parte, las de menos interés o relevancia para las empresas fueron las actividades de documentación y pruebas. Afirman los encuestados que apoyan las investigaciones alrededor de técnicas, métodos y metodologías que se puedan utilizar en la fabricación de software, siempre con el fin de crear productos de calidad en el menor tiempo posible, y sin dejar a un lado la documentación

Fuente: elaboración propia.

Tabla 2. Roles del proceso.

Rol	Descripción
Scrum Master	Encargado de la orientación del proceso, es quien contribuye al cumplimiento de los objetivos facilitando la labor del equipo de trabajo. Este rol puede ser desempeñado por una persona externa a la empresa, a manera de asesor.
Product Owner	Representante de los intereses del cliente, recibiendo las preguntas e inquietudes que son comunicadas a los <i>Team Member</i> ; es quien garantiza que el producto entregado sea de valor para el negocio. Este rol puede ser desempeñado tanto por una persona de la empresa de desarrollo que tenga conocimiento del negocio y amplio contacto con el cliente, o por alguien externo que sea designado por el mismo cliente.
Team Member	Responsable por la construcción del producto, aplicando las actividades técnicas del proceso de desarrollo.
Team Leader	Es el director del equipo de trabajo, encargado de verificar que las herramientas, condiciones y el clima laboral se establezcan satisfactoriamente para el grupo de talento humano interno y externo de la organización.

Fuente: elaboración propia.

Cada rol tiene unas responsabilidades definidas en el proceso. Por ejemplo, el rol *Team Leader* tiene las siguientes responsabilidades:

- Fijar políticas y normas institucionales.
- Dirección estratégica de la empresa y del proyecto.
- Elaborar las propuestas y actas de inicio y cierre (y otros documentos formales o de comunicación con el cliente que sean necesarios).
- Verificar el cumplimiento de tiempos y el uso adecuado de recursos, y definir acciones correctivas si es necesario.
- Diseñar y hacer seguimiento al plan de riesgos.

Como lo establece el proceso DAD, se proponen tres grandes fases que definen el avance global del proyecto:

inicio, construcción y transición. Cada una se explica seguidamente.

4.2.1. Fase de inicio

Es la fase de preparación y adecuación del ambiente de trabajo, se alistan los elementos requeridos del proceso de desarrollo como:

- Identificar el producto a desarrollar a través de las especificaciones del cliente.
- Elaboración de la propuesta y cronograma de actividades.
- Reuniones con el cliente para la presentación de la propuesta.
- Realizar ajustes en la propuesta.
- Firma de contratos y acta de inicio.
- Asignación de roles al equipo de trabajo.
- Configuración del ambiente de desarrollo (Equipos y Software).
- Instalación de las herramientas que permitirán automatizar algunas actividades del proceso de desarrollo.
- Capacitación del talento humano, en temas específicos que se necesitan en el desarrollo del producto.

4.2.2. Fase de construcción.

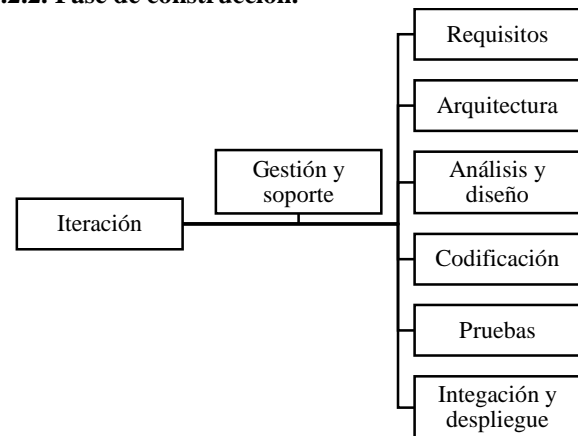


Figura 2. Actividades en cada iteración del proceso de desarrollo. Fuente: elaboración propia.

La fase de construcción tendrá una o más iteraciones (*sprints*), cada una de las cuales terminará con un incremento del software. Cada iteración consta de diferentes tipos de actividades, como se ilustra en la Figura 2.

El proceso incluye para cada actividad una descripción general y una explicación más detallada, para orientar al equipo de desarrollo en su aplicación. A manera de

ejemplo se muestra la descripción de una actividad (requisitos).

Requisitos: Actividad relevante en el proceso de desarrollo, porque de ella depende en gran medida el éxito del producto, si de forma clara se logra identificar explícitamente las necesidades del cliente, para luego ser interpretadas en diagramas de casos de uso, historias de usuario y prototipos de pantalla. En esta fase se aplican las técnicas creativas, con el objetivo de poder establecer los atributos Atractivos, Obligatorios y Unidimensionales; se espera que el equipo de trabajo pueda identificar por lo menos un atributo unidimensional, ya que es la característica que aportará el factor diferenciador en la aplicación.

También se identificaron las principales técnicas y artefactos que se usarían en cada una de las actividades, incluyendo, por supuesto, algunas técnicas creativas.

Tabla 3. Técnicas y artefactos de cada actividad del proceso.

Actividad	Técnica/artefacto
Gestión y entorno	Lista de requisitos Lista de tareas <i>Burndown chart</i> Retrospectiva (final de cada <i>sprint</i>) Control de versiones Reuniones diarias (15 minutos)
Requisitos	Diagrama de casos de uso Especificación casos de uso Prototipos de pantalla Lista de atributos Modelo Kano
Arquitectura	SAD (<i>Software Architecture Document</i>)
Análisis y diseño	Diagramas de clases Diagramas de secuencia Modelo relacional Diseño de las pruebas Lista de atributos
Codificación	Pruebas unitarias Repositorio de versiones
Pruebas	Pruebas funcionales (casos de uso) Pruebas no funcionales
Integración y despliegue	Manual de usuario Manual técnico Entrega del producto

Fuente: elaboración propia.

Las técnicas creativas (en negrita en la tabla) fueron incluidas en las actividades de requisitos y diseño, por ser

estas actividades que propician la búsqueda de alternativas y el trabajo colaborativo.

4.2.3. Fase de transición

Esta fase comprende la entrega del producto al cliente, cerrando el proyecto de desarrollo mediante la entrega del software instalado y la documentación necesaria para su operatividad.

Es importante en esta fase poder establecer el grado de satisfacción de los usuarios y realizar un análisis de la calidad de la aplicación identificando posibles mejoras o cambios para la próxima versión y hacer seguimiento o acompañamiento de la solución.

El proceso de desarrollo implementado tiene un buen margen de flexibilidad, estableciendo una diferencia con otros procesos que son más rígidos para la ejecución de las tareas; además, el aporte de la creatividad de las personas en el proceso es vital para el desarrollo de un producto con características creativas. Una visión global del proceso se puede ver en la Figura 3.

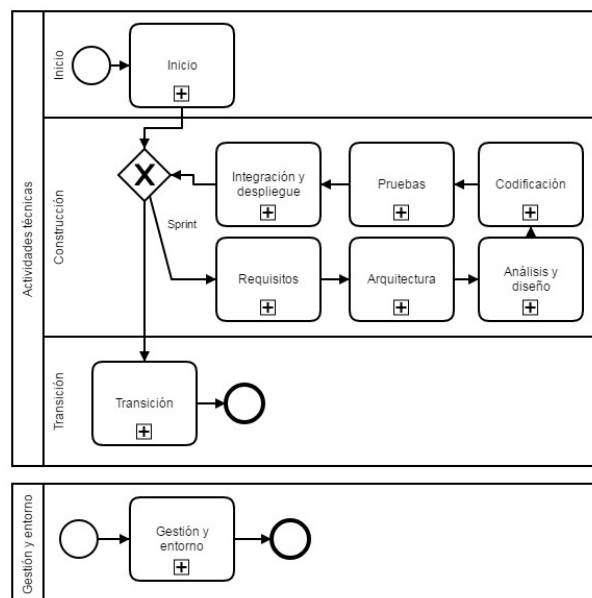


Figura 3. Visión global del proceso definido (adaptado a partir de DAD). Fuente: elaboración propia.

La dinámica del proceso de desarrollo está basada en iteraciones (*sprints*), que permiten evaluar el progreso del proyecto a través de reuniones diarias con el equipo de trabajo y ejecutando actividades que ayudan a la mejora continua, y al fortalecimiento de las buenas prácticas en el desarrollo de software.

El aporte de la creatividad de las personas en el proceso es vital para el desarrollo de un producto novedoso. El proceso con sus tres fases enmarca todas las actividades que son relevantes en un proyecto de desarrollo de software, partiendo de la conceptualización de la idea e identificación de una necesidad, pasando por la construcción y validación del producto y finalizando con un software con atributos que permitirán aumentar la satisfacción del usuario. Esto con el propósito de establecer un factor diferenciador por el cual sea reconocida la organización en el medio de la industria del software a nivel local y nacional mediante un producto propio.

4.3. Prueba piloto

El propósito de esta etapa fue el de llevar a la práctica el proceso de desarrollo propuesto, logrando de esta manera la validación (parcial) del proceso e identificando puntos de mejora a partir de la experiencia en una empresa.

NuevasTIC es una empresa que siempre se ha preocupado por desarrollar productos propios de alta calidad; para el logro de este fin la organización participó en las convocatorias del año 2015 del MinTIC, para recibir consultoría en la norma ISO/IEC 20000 y el modelo ITMark.

NuevasTIC fue una de las cuatro empresas seleccionadas del país por el MinTIC, en recibir consultoría de la empresa mexicana INNEVO y COLCIENCIAS, encontrándose comprometida en el logro de estas certificaciones. Esto permitió que la empresa participara en el proyecto, dada la necesidad de contar con un proceso de desarrollo de software que permitiera aplicar buenas prácticas, especialmente para cumplir con las exigencias del modelo ITMark.

El modelo ITMark exige cumplir con tres elementos importantes para aquellas empresas de desarrollo de software que deseen obtener la certificación del modelo, que son: la mejora de procesos de desarrollo de software, la gestión del negocio y la seguridad de la información. El proceso de desarrollo de software con técnicas creativas aportó a dos de los tres elementos exigidos por el modelo, que fueron: la mejora del proceso de desarrollo de software y en la gestión del negocio.

El proceso de desarrollo con técnicas creativas, en la aplicación en cada una de sus fases, logró identificar roles y responsabilidades dentro del equipo de trabajo, lo cual es uno de los requisitos del modelo ITMark. Por otra parte, el modelo exige ambientes de desarrollo, pruebas, calidad y producción; en este aspecto el proceso de desarrollo de la investigación contempla las fases de

desarrollo y pruebas, utilizando repositorios de versiones y considerando los ambientes necesarios para las pruebas unitarias, pruebas funcionales y no funcionales; en estas actividades se capacitó el talento humano de la organización y fue uno de los puntos positivos a la hora de mostrar evidencias al auditor del modelo. ITMark también requiere realizar un análisis del cliente, obteniendo un conocimiento más amplio en relación a las necesidades del negocio; lo cual se pudo lograr en parte al momento de aplicar las técnicas creativas, pues se identificaron necesidades que el cliente no había podido describir de manera clara.

Previo a la implementación del proceso de desarrollo, se llevó a cabo un reconocimiento del contexto de la organización, a través de la observación y entrevistas, con el objetivo de conocer el punto de vista del equipo de trabajo sobre la pertinencia en la aplicación de un proceso de desarrollo de software incorporando técnicas creativas. De este reconocimiento se resalta que tanto el gerente como el equipo de trabajo (seis personas) consideraron totalmente pertinente la aplicación de un nuevo proceso, aunque no se sentían muy cómodos con las técnicas creativas, principalmente por desconocimiento de las mismas.

Una de las primeras actividades en el piloto fue la preparación del equipo de trabajo a través de capacitaciones y asesorías en la adaptación del proceso de desarrollo, logrando incentivar al talento humano en la ejecución de las tareas establecidas. Las orientaciones se dieron en la explicación del proceso de desarrollo, las técnicas creativas y pruebas unitarias. Las estrategias usadas en las capacitaciones incluyeron clases magistrales, debates y aplicación de técnicas creativas.

Para la capacitación sobre el Modelo Kano se realizó inicialmente una lluvia de ideas sobre posibles funcionalidades del producto, y luego se usaron colores para identificar los distintos tipos de atributos de calidad. A partir de argumentaciones y debates se seleccionaron las funcionalidades que tendría el producto. Esta capacitación se resalta porque obtuvo muy buenos resultados en el equipo, obteniendo comentarios como:

- “Es interesante y útil antes de iniciar un desarrollo, porque permite dar una visión de forma más profunda del servicio que ofrecerá la aplicación al cliente”
- “Es de novedad y se debería hacer no solo con productos propios sino en todos los proyectos de desarrollo de software”.

Después de las primeras capacitaciones, se empieza la aplicación del proyecto, identificando la apropiación del

mismo por parte del equipo de trabajo y solucionado las inquietudes presentadas. Cabe resaltar que el estudiante de maestría autor de este artículo participó en el rol de *Scrum Master*, para poder ser el mentor en el proceso.

Durante la aplicación del proceso se obtienen algunas recomendaciones, que son tenidas en cuenta para la mejora de la propuesta. A partir de estas recomendaciones se hacen algunos ajustes, como por ejemplo la inclusión del rol *Team Leader*, y se observa la necesidad de incluir las guías de capacitación como parte del proceso.

Después de aplicar técnicas creativas en la etapa de requisitos, se obtuvieron 55 atributos de interés (adicionales a las funcionalidades básicas del sistema), los cuales se clasificaron en 16 atributos atractivos, 30 obligatorios y 9 unidimensionales. A manera de ejemplo, se presentan tres de los atributos atractivos (simplificados por efectos de confidencialidad):

- Los datos modificados deben recuperarse en menos de un día.
- Alertas interactivas con asignación en calendario.
- Identificar coordenadas de la localización del sitio del registro de la información.

En el caso del modelo de diseño, se trabajó principalmente en la revisión de los diagramas de clase y de secuencia haciendo uso de las técnicas creativas, lo cual permitió analizar diferentes alternativas de solución para aspectos de seguridad, auditoría, disponibilidad y facilidad de uso.

Algunos de los cambios notorios en la empresa con la aplicación de este nuevo proceso son:

- Se fortalecen algunas actividades técnicas del desarrollo, como los requisitos, el diseño y las pruebas. Estas últimas representan un reto interesante para los miembros del equipo, quienes deben tratar de diseñar las pruebas antes de codificar, lo cual no es algo a lo que están acostumbrados, pero que muestra ser de valor para aumentar la calidad del producto final.
- Se establecen ambientes de pruebas y de producción por separado; y se trabaja con un servidor de versiones.
- Mejora la comunicación y el seguimiento, gracias a las reuniones diarias de 15 minutos y a las entregas periódicas de funcionalidad.

Una dificultad que se presentó fue que el tiempo estimado se extendió, debido a los cambios introducidos por el proceso y a la adición de nuevas funciones en el producto. Sin embargo, se considera que este desfase en

la estimación no es propio del proceso sino de la curva de aprendizaje.

Algunos de los logros que se evidenciaron con la aplicación del proceso, fueron:

- Gracias a la aplicación de las técnicas creativas se identificaron nueve características unidimensionales en el producto propio, que no estaban contempladas en el alcance inicial, y que constituyen un valor agregado para los clientes.
- El proceso de desarrollo aplicado aportó al proceso de certificación de la norma ISO/IEC 20000 y el modelo ITMark de la organización.
- El aporte más significativo fue el trabajo en equipo y aprovechamiento de la creatividad de las personas participantes en el proceso de desarrollo.

4.4. Resultados obtenidos

La presente investigación deja un proceso que se puede aplicar en Pymes que emprenden procesos de certificación en modelos de calidad y buenas prácticas de desarrollo de software, como también en aquellas que buscan crear aplicaciones propias en mercados específicos. Este proceso permite desarrollar productos sencillos y atractivos para el cliente sin incurrir altos costos, y simultáneamente fortalece la comunicación del equipo de la organización. Hay que aclarar que la aplicación de un proceso de desarrollo con técnicas creativas no asegura el mercado del producto propio desarrollado, pero permite disminuir la incertidumbre e incrementar la posibilidad de obtener un producto innovador.

Con este proyecto también se realiza una contribución a la comunidad científica, con la adaptación de un proceso en el contexto de la ingeniería de software; esto teniendo en cuenta el estudio realizado de los antecedentes que se llevó a cabo en el presente trabajo, en el cual no se encontró un proceso que haga énfasis en desarrollar productos propios, orientado al descubrimiento de algún factor diferenciador que dé lugar a características atractivas para el cliente.

5. Conclusiones, recomendaciones y trabajos futuros

5.1. Conclusiones

Para la prueba piloto llevada a cabo, el proceso permitió organizar y estandarizar las actividades de fabricación del software, cumpliendo con la exigencia de los procesos de certificación en modelos de calidad, y sin perder la flexibilidad ni la capacidad creativa.

La aplicación de técnicas creativas mediante actividades lúdicas en el equipo de trabajo logra que las personas expongan sus ideas con mayor libertad, y que después de ser discutidas entre los miembros del grupo y en consenso, pueden ser consideradas como atributos de calidad, para ser incluidos en la construcción de un software y de esta forma ser considerado como un producto diferenciador.

Es relevante para las pequeñas y medianas empresas de desarrollo de software identificar un proceso de desarrollo que ofrezca la oportunidad de la participación del equipo de trabajo en la generación de opiniones y de ideas, permitiendo la construcción de productos con un sello personalizado, y con el cual los integrantes del equipo se sientan identificados y comprometidos.

Por otra parte, es necesario que el proceso de desarrollo permita que la gerencia pueda hacer seguimiento al proyecto, dado que la visión de negocio está en cabeza de esta área de la organización.

El interés de crear productos propios es una estrategia que permite a las Pymes mantenerse dentro de un mercado cada vez más exigente, y donde se compite no solo con empresas locales sino también nacionales e internacionales.

5.2. Recomendaciones

Se recomienda a las organizaciones de desarrollo de software que están interesadas en ser reconocidas en el mercado a través de un producto propio, la aplicación de técnicas creativas en sus procesos, especialmente en las actividades de requisitos y diseño, pero sin dejar de lado otras actividades donde éstas puedan ser aplicadas.

Para una adecuada aplicación del proceso se recomienda:

- Aplicar las técnicas creativas antes del desarrollo del producto, identificando las funcionalidades y características creativas en las actividades de requisitos, análisis y diseño, esto para no afectar el cronograma del proyecto, implicando mayores costos para su ejecución.
- Tener cuidado de no hacer funcionalidades extras que no requiere el cliente y terminar desarrollando un producto diferente a las condiciones del negocio. No perder el objetivo del proyecto.
- Realizar capacitaciones al talento humano, incentivando el desarrollo de la creatividad, debido a que es muy valioso el aporte de las personas. Cuando los empleados aportan ideas y se les gratifica de alguna manera su ingenio, se fortalece la competitividad de la empresa.

- Realizar una autoevaluación de la organización para identificar en qué condiciones se encuentra la empresa antes de comenzar el desarrollo de un producto propio con un nuevo proceso de desarrollo.
- Para minimizar el riesgo financiero al invertir tiempo, costos y talento humano en la fabricación de un producto que no cumpliera con las expectativas de los clientes y capacidad del entorno, se debe definir un nicho de mercado concreto. Esto se puede hacer con técnica como el Modelo Canvas.

5.3. Trabajos futuros

Un trabajo que se puede realizar a partir de este proyecto es la aplicación del proceso en otras Pymes de la ciudad de Neiva, e incluso en otras ciudades del país, para tener una población de estudio mayor que permita la validación estadística del proceso. También se pueden construir guías de ajuste del proceso, para diferentes tipos de productos o servicios.

Por otra parte, el deseo de las Pymes de desarrollo de software de desarrollar productos propios que ofrezcan una propuesta valor en un mercado competitivo da lugar a proponer una línea de investigación que se encuentre direccionada al estudio de técnicas creativas que contribuyan al aumento de la satisfacción de los usuarios, basados en modelos o técnicas nuevas que pueden ser aprovechados al máximo en un grupo de desarrollo de software. La línea de investigación propuesta debe ser fundamentada en nuevas técnicas que sean fáciles, que permitan desmenuzar un problema en partes cada vez más pequeñas e ir buscando soluciones alternativas a esas partes, describiendo los atributos según el impacto y funcionalidad sin que afecten en mayor parte el presupuesto y tiempo en un proyecto de desarrollo de software.

Referencias

- [1] M. C. Cuéllar, «Los desafíos de las empresas de software frente a la globalización,» Revista de Ingeniería. Universidad de los Andes, pp. 86-90, 2013.
- [2] MinTIC y Fedesoft, «Informe de caracterización del sector de software y tecnologías de la información en Colombia,» 2015.
- [3] C. A. Ardila y F. J. Pino, «Panorama de gestión cuantitativa de procesos de desarrollo de software en pequeñas organizaciones,» Sistemas & Telemática, pp. 29-46, 2013.
- [4] L. E. Peláez Valencia, L. Cardona Benjumea y A. Toro Lozano, «Estado del arte que soporta el proceso de

desarrollo de software en las PYMES colombianas: Una mirada desde las organizaciones nacionales que tienen que ver con la disciplina.» Entre Ciencia e Ingeniería, pp. 93-107, 2011.

[5] MinTIC y Fedesoft, «Informe sectorial de la industria de software y servicios asociados de Colombia 2012,» Bogotá, 2012.

[6] J. A. Britto Montoya, «Adaptación de un proceso de desarrollo de software basado en buenas prácticas,» Universidad Autónoma de Manizales, 2014.

[7] J. A. Holguín Barrera, «Definición de un proceso de desarrollo de software con control de calidad del producto en una empresa PYME de la región,» Universidad Autónoma de Manizales., 2014.

[8] L. F. Londoño, «Recomendaciones para la formación de una empresa de desarrollo de software competitiva en un país como Colombia,» Avances en Sistemas e Informática, 2005.

[9] L. Merchán, A. Urrea y R. Rebolgar, «Definición de una metodología ágil de ingeniería de requerimientos para empresas emergentes de desarrollo de software del sur-occidente colombiano,» Revista Científica Guillermo de Ockham, 2008.

[10] L. Pantoja, C. A. Collazos y V. M. Penichet, «Entorno colaborativo de apoyo a la mejora de procesos de software en pequeñas organizaciones de software,» Dyna, pp. 40-48, 2013.

[11] S. Barbieri, «Trabajo de Tesis Framework de Mejora de Procesos de Desarrollo de Software,» Universidad de la Plata Facultad de Informática, La Plata Argentina, 2007.

[12] G. Solarte, L. Muñoz y B. Arias, «Modelos de calidad para procesos de software,» Scientia Et Technica, vol. 2, n° 42, pp. 172-176, 2009.

[13] A. Mas y E. Amengual, «La mejora de los procesos de software en las pequeñas y medianas empresas (PYME). Un nuevo modelo y su aplicación a un caso real,» REICIS Revista Española de Innovación, Calidad e Ingeniería del Software., vol. 1, n° 2, 2005.

[14] S. Ambler y M. Lines, Disciplined Agile Delivery: A practitioner's guide to agile software delivery in the enterprise, IBM Press, 2012.

[15] IEEE Computer Society, SWEBOK v3.0: Guide to the Software Engineering Body of Knowledge, P. Bourque y R. Fairley, Edits., IEEE, 2014.

[16] S. Stavru, «A critical examination of recent industrial surveys on agile method usage,» Journal of Systems and Software, vol. 94, pp. 87-97, Agosto 2014.

[17] B. Fitzgerald, K.-J. Stol, R. O'Sullivan y D. O'Brien, «Scaling Agile Methods to Regulated Environments: An Industry Case Study,» de International Conference on Software Engineering (ICSE), San Francisco, 2013.

[18] E. Martínez Jáuriguí, «Creatividad e innovación : Juntas pero no revueltas,» de Juntas pero no revueltas, 2010.

[19] E. Yacuzzi y F. Martín, «Aplicación del método de Kano en el diseño de un producto farmacéutico,» Revista UCEMA, 2002.

[20] InnoSupport, Guía de Apoyo a la Innovación en Pequeñas y Medianas Empresas, 2009.