

Metodología de Desarrollo de Software: MBM (Metodología Basada en Modelos)

Software Development Methodology: MBM (Methodology Based in Models)

Janeth Rozo Nader*

RESUMEN

La ingeniería del software posee técnicas y herramientas que han madurado mucho actualmente se invierten grandes cantidades de recursos para potenciar la industria del software y convertirla en uno de los sectores estratégicos de crecimiento.

Las empresas necesitan una estrategia sólida de Arquitectura Empresarial, se recomienda utilizar un software eficiente, motivo suficiente para que la Ingeniería de Software planteara metodologías para la elaboración de un producto de calidad. Hoy en día, aunque existen muchas de ellas con resultados muy buenos, no se cuenta con una estandarizada que al implementarla, permita desarrollar adecuadamente un proyecto de software que incluya la Gestión por Procesos de Negocio (BPM). Por eso se considera importante el diseño de una metodología práctica que contenga todos los procesos de creación de un producto de alta calidad, para utilizar en cualquier proyecto de software, encaminado a lograr la excelencia en la automatización de todos los procesos de negocio.

Palabras Claves: Metodología, Calidad de Software, Software, Gestión de Procesos de Negocio (BPM)

ABSTRACT

Quality engineering and software techniques and tools have matured considerably in recent years in order to make products that are made to offer high reliability and meet the needs of users.

Several countries are investing large amounts of resources in order to enhance the software industry for both domestic demand and to convert the software on one of the strategic growth sectors.

As technology advances, new implementations are required to improve the organizations, since business processes are becoming increasingly complex, full of deep interactions in all systems and dependence on collaborative activities between users. To manage successfully, companies now need a solid Enterprise Architecture strategy and a set of appropriate software tools. For this reason it is recommended to use software that is efficient and of high quality. A good software should be based on an appropriate methodology for its development, sufficient for software engineering methodologies raised for the production of a quality product reason. Today, although many of these methodologies and have shown very good results, there is no standardized to implement that, allow efficiently develop a software project that includes Business Process Management (BPM), and can be used by any software developer. Given this, it was considered important to design a practical methodology, which includes all the processes needed to create a high quality product, but mostly they can be used in any software project aimed at achieving excellence in the administration, execution, automation of all business processes.

Key Words: Methodology, Software Quality, Software, Business Process Management (BPM).

*

1. INTRODUCCIÓN

El 85 por ciento de las empresas dedicadas a la producción de software del Caribe Colombiano, no se basan en una metodología que les asegure a los empresarios la elaboración de un buen producto de software. Esto no debería ser así, los arquitectos de software deben poseer una base sólida en la que puedan apoyarse y tener en cuenta que, trabajar en un proyecto de alta calidad es difícil de controlar, si no se tiene un modelo de buenas prácticas.

Las empresas optan por no utilizar una metodología para la elaboración de sus productos porque no les interesa su uso, porque ninguna se amolda a las necesidades del software a desarrollar, o bien porque se trata de proyectos pequeños de dos o tres meses. Pero, ¿qué sucedería si el proyecto que se va a desarrollar no fuese tan pequeño? Entonces es cuando juega un papel importante la utilización de una base sólida, como una buena metodología de desarrollo de software, allí es cuando los desarrolladores empiezan a buscar cual sería la más apropiada para el caso. Con frecuencia no se encuentra la más adecuada ya que al existir variedad de metodologías, enfocadas cada una a diferentes productos pero, ninguna dirigida a los procesos de negocio de una compañía que ayude a mejorar la eficiencia y eficacia de la organización, implica entonces que se desarrollen proyectos sin metodología.

Cabe resaltar, además, la gran cantidad de recursos que se destinan a la creación de estándares y modelos para lograr software de calidad. De ahí la importancia de que existan estándares reconocidos que permitan a las empresas coordinar sus esfuerzos y reutilizar las buenas prácticas de desarrollo y gestión del software. Como el proceso de software es cambiante e irregular, los grupos de trabajo abandonan fácilmente los planes y se centran en la codificación y pruebas. Cabe notar que la ausencia de un buen software puede causar grandes desastres en las organizaciones.

2. REVISIÓN CONCEPTUAL

Calidad de software

La obtención de un software de calidad implica la utilización de modelos o procedimientos estándares para su análisis, diseño, desarrollo y prueba que permitan uniformar la filosofía de trabajo, para lograr una mayor confiabilidad mantenimiento y facilidad de prueba. A su vez que eleven la productividad, tanto para la labor de desarrollo como para el control de la calidad del software. Se puede considerar que un software es de calidad si cumple los siguientes objetivos: concordancia con los requerimientos, desarrollo coherente y desarrollo de requerimientos implícitos en el proyecto.

Los factores que determinan la calidad del software: se clasifican en 3 grupos: Operación,

Revisión, y Transición del producto.

Ingeniería de Software

Todas las empresas desarrolladoras de proyectos o servicios software asumen que, la implantación de una metodología es necesaria si se quieren gestionar adecuadamente los proyectos o servicios software. Según el Licenciado en Informática Javier Gómez, "la metodología no debe adaptarse a la organización, sino que es la organización la que debe adaptarse a la metodología"[1].

"Metodología" se define como el enfoque de un problema de manera total, organizada, sistemática y disciplinada. Esta definición muestra una distinción entre "metodología" y "técnica". La técnica se considera como un componente de la metodología, como el medio o procedimiento que se usa para realizar la metodología misma.

CMMI

Capability Maturity Model Integration (CMMI) es un modelo para la mejora de procesos que proporciona a las organizaciones los elementos esenciales para procesos eficaces. Las mejores prácticas CMMI se publican en los documentos llamados modelos. En la actualidad hay dos áreas de interés cubiertas por los modelos de CMMI: Desarrollo y Adquisición.

Las prácticas CMMI deben adaptarse a cada organización en función de los objetivos del negocio. El objetivo del proyecto CMMI es mejorar la usabilidad de modelos de madurez integrando varios modelos diferentes en un solo marco (framework).

Los 6 niveles definidos en CMMI para medir la capacidad de los procesos son: incompleto, ejecutado, gestionado, definido, cuantitativamente gestionado, optimizante.

Metodologías de desarrollo convencionales

Una definición de metodología podría ser "el conjunto de filosofías, fases, procedimientos, reglas, técnicas, herramientas, documentos y aspectos de formación para los desarrolladores de sistemas de información" [2].

La metodología normalmente consistirá en un conjunto de fases, descompuestas en sub-fases (módulos, etapas, pasos, etc.), de forma que esta descomposición guíe a los desarrolladores en la elección de las técnicas que se deben elegir para cada estado del proyecto, facilitando la planificación, gestión, control y evaluación de los proyectos.

MÉTODOS ESTANDARIZADOS

MERISE

El Proceso MERISE es un método de concepción y de desarrollo de sistemas de información. Se Concibe el sistema de información como un objeto artificial intermediario entre el Sistema que opera y el sistema de conducción. Este enfoque mejora notablemente la Calidad de las especificaciones antes de la realización y sobre todo conduce a Sistemas que evolucionan fácilmente en caso de cambio tecnológico. Abarca los aspectos relacionados con la recopilación y validación de la información, capacitación de personal, evaluación de equipos informáticos, análisis, diseño y validación de los procesos e implementación, gestión de costos y tiempos y el desarrollo del código.

Rational Unified Process (Rup)

El Proceso Unificado es un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizació, diferentes niveles de competencia y diferentes tamaños de proyectos. Provee un enfoque disciplinado en la asignación de tareas y responsabilidades, dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible.

Metodologías ágiles

Programación Extrema (Xp)

‘Extreme Programming (XP) es en realidad una disciplina para el desarrollo de software, con muy poca edad. Ya se ha demostrado en muchas empresas de t diferentes tamaños e industrias en todo el mundo, que XP tiene éxito, ya que hace hincapié en la satisfacción del cliente. La metodología está diseñada para ofrecer el software adecuado a las solicitudes de de los clientes, cuando sea necesario. XP faculta a sus desarrolladores para responder con confianza a la evolución de las necesidades de los clientes, incluso a al final del ciclo de vida’ [3]. La programación extrema se basa en trece “prácticas básicas” que deben seguirse al pie de la letra. Las cuales son: equipo completo, planificación, test del cliente, versiones pequeñas, diseño simple, pareja de programadores, desarrollo guiado por las pruebas automáticas, mejora del diseño, integración continua, el código es de todos, normas de codificación, metáforas, ritmo sostenible.

Gestión de proyectos

Según el autor Domínguez Fernández Bari, ‘un proyecto consiste en utilizar los recursos para lograr un objetivo específico, siguiendo un esquema planificado y organizado. La gestión de proyectos se enfoca en planificar las actividades, programarlas y luego, mientras se está en

el proceso de ejecución, controlar dichas actividades.

En un proyecto, el alcance es una parte clave, en él se deben definir claramente todos los entregables. Lo importante es visualizar todas las posibles acciones para lograr los objetivos del proyecto.

BPM (Gestión de Procesos de Negocio)

Es el enfoque que consiste en usar un equipo para modelar los procesos de negocio de una compañía, en el que se involucran aplicaciones y personas.

El ciclo de vida de un enfoque BPM puede dividirse en las siguientes etapas: estudio de la compañía, modelar los procesos del negocio, implementación de una solución, ejecución, control, optimización.

3. METODOLOGÍA

La investigación realizada fue de tipo exploratoria- propositiva, ya que partiendo de las metodologías existentes a través de un estudio de estado del arte, se planteó una nueva metodología, que cumple con las características requeridas en este contexto. Se llevaron a cabo las fases de diagnóstico y estudio del estado del arte, en la que se realizó un estudio del estado actual de los procesos de desarrollo de software en las empresas PYMES de la industria del software en el Caribe Colombiano; y la fase de Definición, en la que, a partir del estudio realizado en la primera fase y del conocimiento del estado del arte, se definió una metodología de creación de software, que tiene en cuenta la gestión por procesos de negocio (BPM).

4. METODOLOGÍA DE DESARROLLO DE SOFTWARE: MBM (METODOLOGÍA BASADA EN MODELOS)

Esta metodología permite desarrollar un software en un ambiente agradable y de forma fácil, que se basa en modelos, y cumple con los estándares de calidad y se enfoca hacia los procesos del negocio.

Esta metodología está estructurada en 8 fases, las cuales tienen procesos, técnicas, tareas y actividades que conducen a la buena implementación de la metodología. Inicia con la fase de Modelado de Procesos de Negocio (BPM), sigue con las fases de Gestión de Requisitos, Análisis de requisitos, Desarrollo e Implementación, Pruebas, Despliegue, Operación y Mantenimiento, y Gestión de Proyecto.

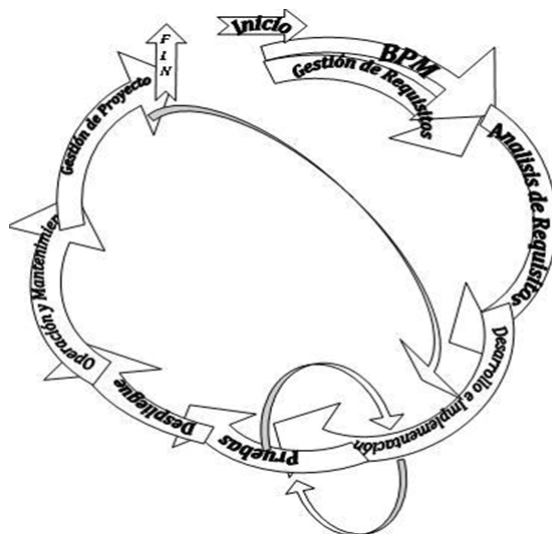


Figura 1. Ciclo de vida de la Metodología Basada en Modelos (MBM)

Fuente: Elaboración de los autores

MODELADO DE PROCESOS DE NEGOCIO (BPM).

En esta fase o nivel se realizan dos procesos: conocer procesos actuales de la organización y crear modelos de procesos.

Proceso: conocer procesos actuales de la organización

• Crear una perspectiva general de todos los procesos de negocio de la compañía y sus interacciones para poder optimizarlos. Este proceso consta de la técnica Estudio preliminar. Técnica donde se analizan los procesos actuales de la empresa u organización del cliente, para determinar una posible solución del problema. Esta técnica consta de las siguientes tareas:

- Definir los procesos de la empresa
- Análisis de los procesos actuales

Proceso: crear modelos de procesos

Diseñar los modelos que ilustren los procesos de la empresa, anteriormente definidos. Y que se deben tener en cuenta en todo proceso de una organización. Consta de un inicio, una planificación, un control y una finalización de procesos con una aceptación total de los procesos del proyecto. Este proceso consta de la siguiente técnica:

• Modelar, donde se crea un modelo, lo más realista posible de los procesos actuales de la organización. Esta técnica consta de la siguiente tarea:

DISEÑO DE MODELOS

Gestión de requisitos

En esta fase o nivel se realiza el proceso de Toma de requisitos.

Proceso: toma de requisitos

Obtener los conocimientos necesarios para definir los requisitos que proporcionarán los límites, metas y finalidad del software a construir. Es necesario que estos requisitos obtenidos queden bien claros, ya que de estos dependerá el éxito del software. Este proceso consta de las siguientes técnicas:

Entrevistas (cliente)

Esta técnica se concentra en recolectar la mayor cantidad de datos posibles sobre el cliente, y a su vez, el cliente conocerá sobre la entidad desarrolladora. Será una entrevista formal en donde cada parte entenderá la misión de su empresa. Se programan reuniones puntuales que permitan definir los requisitos que deberán cumplirse al finalizar el proyecto software. Esta técnica consta de las siguientes tareas, definir el ámbito de trabajo y recolectar datos básicos.

- Definir el ámbito de trabajo
- Recolectar datos básicos

Reunión (cliente)

Esta técnica permite definir, formalizar y documentar todos los requisitos posibles del cliente, ya sean funcionales o no funcionales del proyecto software. Es sumamente importante que estos requisitos sean comprensible para poder especificar los requisitos del sistema. La especificación de los requisitos del sistema deberá describir funciones y capacidades del sistema. Se deberá documentar toda la especificación de los requisitos. Esta técnica consta de las siguientes tareas:

- Capturar y definir requisitos
- Formalizar requisitos
- Documentar requisitos

ANALISIS DE REQUISITOS

Esta fase o nivel consta de dos procesos: Identificar requisitos y Modelar requisitos. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, aseguradores de calidad, analistas, diseñadores y documentadores.

Proceso: identificar requisitos

Detectar y resolver posibles conflictos entre requisitos, definir los límites y alcance del sistema, cómo debe interactuar con su entorno y elaborar el paso de requisitos del sistema, a requisitos software. Este proceso consta de la siguiente técnica: agrupar requisitos y definir grupos de trabajo.

- Agrupar requisitos y definir grupos de trabajos

Con esta técnica se deben identificar cada uno de los requisitos capturados, y describir las tareas a realizar para cumplir con los requisitos. Se agruparán aquellos requisitos que dependen el uno del otro, se deben crear grupos de trabajos para que se les asignen requisitos en común y, por último, se define la relación que tiene cada grupo de trabajo. Esta técnica consta de las siguientes tareas: propuesta de la solución global, identificar cada requisito con sus correspondientes tareas, agrupar requisitos en común, definir grupos de trabajo, y definir las relaciones de los grupos de trabajo (Entrada y Salida), establecer estimaciones de trabajo.

Propuesta de la solución global: Identificar cada requisito

- Agrupar requisitos en común.
- Definir grupos de trabajo.
- Definir las relaciones de los grupos de trabajo. (Entrada y Salida).
- Definir el Plan de Gestión.

PROCESO: MODELAR REQUISITOS.

Modelar los requisitos que fueron capturados y definidos en la reunión con el cliente. Este proceso consta de la siguiente técnica: Diseño.

Diseño

Con esta técnica se realizan tareas que cuidan el óptimo desarrollo del sistema con los modelos de los procesos y, de igual forma, minimizan y controlan los riesgos de la construcción del sistema. Para ello se implementará la arquitectura MDA (Model Driven Arquitectura). Esta técnica consta de las siguientes tareas:

- Escoger herramienta para modelar.
- Modelar los procesos.
- Agrupar los Modelos y Formalizar el conjunto modelado (Reuniones).

DESARROLLO E IMPLEMENTACIÓN

Esta fase o nivel consta del proceso: Construcción del sistema de información. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, aseguradores de calidad, diseñadores, programadores, administrador de la configuración y documentadores.

PROCESO: CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN.

Generación del código por cada grupo de trabajo y definir bajo que estándares de programación se va a trabajar. Este proceso consta de la siguiente técnica, Codificar requisitos.

Codificar Requisitos.

Dada la gran cantidad de productos generados en este proceso y, según las características del proyecto, la codificación debe ser sometida a controles de seguridad, que eviten filtraciones indeseables de datos relativos al sistema de información. Por esto se trabaja la fase de pruebas en paralelo con esta actividad. Esta técnica consta de las siguientes tareas:

- Escoger los estándares de programación.
- Generación del código por los grupos de trabajo.
- Agrupar y formalizar el código.
- Realizar los manuales de usuario.

PRUEBAS

Esta fase se realiza en paralelo con la construcción del sistema de información, debido a que la terminación del código por parte de cada grupo de trabajo debe pasar por pruebas, antes de ser fusionado. Esta fase o nivel consta de dos procesos: ejecución de pruebas individuales y ejecución de pruebas del sistema. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, aseguradores de calidad, programadores, tésters, ingeniero de mantenimiento, ingeniero de validación y verificación, administrador de la configuración y documentadores.

PROCESO: EJECUCIÓN DE PRUEBAS INDIVIDUALES

Verificar si los componentes o subsistemas interactúan correctamente a través de sus interfaces, cubren la funcionalidad establecida, y se ajustan a los requisitos especificados. Este proceso consta de la siguiente técnica: simulación de módulos individuales.

Simulación de Módulos Individuales

Con esta técnica se realizan las pruebas de cada uno de los componentes del sistema de información, una vez codificados, con el objeto de comprobar que su estructura es correcta, y que se ajusta a la funcionalidad establecida. Esta actividad consta de las siguientes tareas:

- Simular la operación del módulo
- Analizar el comportamiento del módulo
- Documentar fallos y mejoras

PROCESO: EJECUCIÓN DE PRUEBAS DEL SISTEMA.

Verificar si el sistema funciona correctamente a través de sus interfaces, cumple con la funcionalidad establecida, y se ajusta a los requisitos especificados. Este proceso consta de la siguiente técnica:

Simulación del Sistema

El objetivo de las pruebas del sistema es comprobar la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen, y con el resto de sistemas de información con los que se comunica. Dentro de esta técnica están las tareas:

- Simular la operación del sistema
- Analizar el comportamiento del sistema
- Documentar fallos y mejoras

Despliegue

Esta fase o nivel consta del proceso entrega del sistema. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, cliente, programadores y documentadores.

Proceso: Entrega del Sistema

Entrega del producto terminado. Una vez realizado todo el proceso de pruebas y las modificaciones necesarias, se entrega el producto al cliente con los manuales de usuarios respectivos. Este proceso consta de la siguiente técnica: presentación del producto terminado al cliente.

Presentación del producto terminado al cliente

Con esta técnica se le hace entrega al cliente del producto, con toda la documentación. Esta técnica consta de las siguientes tareas:

- Entrega de soportes al cliente
- Asesoría de manejo del sistema
- Cierre de contratos.
- Documentar la entrega

Operación y Mantenimiento

Esta fase consta del proceso del seguimiento y control. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, aseguradores de calidad, administrador de la Configuración, ingeniero de validación y verificación cliente, tésters y documentadores.

Proceso: Seguimiento y Control

Es un seguimiento que se realiza al producto con el fin de supervisar y controlar el funcionamiento de éste. Este proceso consta de las siguientes técnicas: gestión de configuración, validación y verificación.

Gestión de configuración

Llevar un control detallado de todos los elementos de la infraestructura que sirven de apoyo a la realización del software, y así tener información precisa sobre la configuración de cada recurso, con el fin de monitorizar los procesos. Esta técnica consta de las siguientes tareas:

- Identificar los elementos de configuración.
- Llevar un control de los elementos de configuración.
- Monitorizar periódicamente la configuración de la infraestructura.

Validación y verificación (CMMI)

Asegurar que el software cumple con su especificación y satisface las necesidades del cliente. Esta técnica consta de las siguientes tareas:

- Encontrar los fallos del sistema.
- Revisiones formales

Gestión de Riesgos

Maximizar la probabilidad y consecuencias de eventos positivos y minimizar las de eventos negativos. Esta técnica consta de las siguientes tareas:

- Planificación de Gestión de Riesgos
- Identificación de Riesgos
- Análisis cualitativos de los riesgos.
- Análisis cuantitativo de los riesgos.
- Planificación de respuestas ante los riesgos

GESTION DE PROYECTOS

Esta fase se compone de cuatro procesos: evaluación de funcionalidad, ejecución de pruebas individuales, ejecución de pruebas del sistema, gestión de proveedores. Las personas que intervienen en esta fase deberían ser: administrador del proyecto, aseguradores de calidad, cliente, analistas, diseñadores ingenieros de manutención, programadores, administrador de la configuración, ingeniero de validación y verificación y documentadores.

Proceso: Evaluacion de Funcionalidad

Gestionar el software de tal forma que se evalúe su funcionalidad por medio de métricas y surjan nuevas versiones. Este proceso se realiza por medio de las técnicas: Análisis, y optimización.

Análisis

En esta técnica se implementan métricas que evalúen el producto, y así determinar si el servicio ha mejorado o no y, si es necesario, se realizan estrategias de optimización del producto para mejorar. Esta técnica consta de las siguientes tareas:

- Definir y aplicar métricas.
- Estrategias de optimización.

Optimización

Realización de mejoras del producto y nuevas versiones. Esta técnica consta de las siguientes tareas.

- Brainstorming. (Lluvia de ideas)
- Identificar ideas
- Analizar ideas
- Modelar nuevos procesos
- Agrupar los modelos y Formalizar el conjunto modelado
- Generación del código
- Agrupar y formalizar el código
- Realizar nuevos manuales

Proceso: Gestion de Alcances y Tiempos

Gestionar que el software incluya todos los trabajos requeridos y en el tiempo estipulado, es decir, en los plazos definidos para el desarrollo de éste. Consta de las siguientes tareas:

- Planificación del alcance y tiempo
- Definir alcance y tiempo
- Verificación del alcance y tiempo

Proceso: Ejecución de Pruebas Individuales

Verificar si los componentes o subsistemas interactúan correctamente a través de sus interfaces, cubren la funcionalidad establecida, y se ajustan a los requisitos especificados. Este proceso consta de la siguiente técnica:

Simulación de Módulos Individuales

Con esta técnica se realizan las pruebas de cada uno de los componentes del sistema de información, una vez codificados, con el objeto de comprobar que su estructura es correcta, y que se ajusta a la funcionalidad establecida. Esta actividad consta de las siguientes tareas:

- Simular la operación del módulo
- Analizar el comportamiento del módulo
- Documentar fallos y mejoras

Proceso: Ejecución de Pruebas del Sistema

Verificar si el sistema funciona correctamente a través de sus interfaces, cubren la funcionalidad establecida, y se ajustan a los requisitos especificados. Este proceso consta de la siguiente técnica:

Simulación del Sistema

El objetivo de las pruebas del sistema es comprobar la integración del sistema de información globalmente. Se verifica el funcionamiento correcto de las interfaces entre los subsistemas que lo componen, y con el resto de sistemas de información con los que se comunica. Dentro de esta técnica están las tareas:

- Simular la operación del sistema
- Analizar el comportamiento del sistema
- Documentar fallos y mejoras

Proceso: Gestión de Costes

Asegurar que el proyecto sea terminado con el presupuesto previsto. Consta de las siguientes tareas:

- Estimación de recursos y costos
- Realización de presupuesto de costos
- Control de costos

Proceso: Gestión de la Calidad

Asegurar que el proyecto satisfaga los requisitos. Consta de las siguientes tareas:

- Planificación de la calidad
- Aseguramiento de la calidad
- Control de calidad

Proceso: Gestión de Proveedores

El objetivo es establecer acuerdos con proveedores y aseguramiento del cumplimiento de dichos acuerdos. Este proceso consta de las siguientes técnicas:

Establecer acuerdo con proveedores

En esta tarea se procede a determinar los tipos de adquisición que se desea, a seleccionar los proveedores y a establecer acuerdos con ellos.

Cumplir acuerdos con proveedores

El objetivo de esta tarea es realizar un plan para asegurar el cumplimiento de los acuerdos con los proveedores, monitoreando el proceso de selección y los productos ofrecidos, para lograr la aceptación total del producto adquirido.

5. CONCLUSIÓN

Debido a la importancia que los sistemas de información tienen para el funcionamiento de la sociedad actual, así como para el bienestar de las personas, hay que enfatizar, aún más, en la importancia de la calidad en la construcción de software.

En los años setenta la productividad de software era la preocupación de moda, sustituida en los ochenta por la calidad y en los noventa por el time to-market y el desarrollo rápido y con calidad.

Debido a la importancia del proceso de creación de software es necesario tener una metodología para la elaboración de un producto software que pueda cumplir con las necesidades de las organizaciones. Es importante conocer e identificar los procesos esenciales de las metodologías y estándares existentes para la construcción de los productos de software y se debe tener en cuenta el modelado de procesos de negocios (BPM).

Conociendo cada uno de los estándares y los procesos que tienen las metodologías y sabiendo también los requerimientos de un software con calidad, se construyó una nueva metodología, con base en los procesos más importantes y destacados de cada una de las metodologías existentes.

Se tuvieron en cuenta también los estándares de calidad de software, una metodología que a su vez sea más práctica, clara, eficiente y fácil de usar, para la elaboración de los productos software de las empresas del Caribe Colombiano, desarrolladoras de Software.

REFERENCIAS BIBLIOGRAFICAS

- [1] J. Gomez, [En línea]. <http://www.versioncero.com/articulo/469/metodologias-pero-cual> (Consultado en Abril de 2008)
- [2] F. Garcia. (2008, maz). [En línea]. Crescencio Bravo Santos Ingeniería del Software. Ingeniería del Software de Gestión. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/tema9_2xh.pdf
- [3] Extreme Programming: A gentle introduction. (2008, jun). [En línea]. What is Extreme Programming?. Disponible en: <http://www.extremeprogramming.org/what.html> (Consultado en 2 de Junio de 2008)

BIBLIOGRAFIA DE CONSULTA

1. L. Moliner. Informáticos de la Generalitat Valenciana, grupos A y B, Bloque Específico.
2. L. Garcia. Informática, Temario Específico.
3. Modelos de gestión de la calidad del software: Normas ISO 12207 Información. Disponible en: <http://juanmarcosteoria2.blogspot.com/2008/01/normas-iso-12207.html>.
4. (ISO/IEC 12207:1995 Amd 1:2002, Amd 2:2005 INFORMATION TECHNOLOGY. Software life cycle processes). Norma Técnica Peruana NPT-ISO/IEC 12207. 2ª Edición. 2006. Información disponible en: <http://www.bvindecopi.gob.pe/normas/isoiec12207.pdf> (Consultado en 10 de Marzo de 2008)
5. Metodologías de desarrollo software <http://www.scribd.com/doc/2050925/metodologias-de-desarrollo-software> (Consultado en Marzo de 2008)
6. Ministerio de Industria Francés. Primera versión: 1972-1976, <http://www.escet.urjc.es/~gtazon/ISMetodologias.pdf> (Consultado en Abril de 2008)