

OPTIMIZACIÓN DE MÉTRICA VERSIÓN 3 EN ENTORNOS ORIENTADOS A OBJETOS

José L. López Cuadrado, Ángel García Crespo, Belén Ruiz Mezcuca, Israel González Carrasco
Universidad Carlos III, Departamento de Informática
Avda. de la Universidad 30, 28913 Leganés, Madrid, Spain
E-Mail: jllopez@inf.uc3m.es, acrespo@ia.uc3m.es, bruiz@inf.uc3m.es, igcarras@inf.uc3m.es

Abstract: Métrica v3 methodology has been created by the Ministry of Public Administrations of Spain, to develop projects of the administration, but also it is being used in projects of the private company. The experience says that the first contact with the methodology can be complicated for managers little experienced. For that reason the methodology has been reviewed including explanations taken from the bibliography on the development of software projects, and including examples that facilitate the understanding. In addition we propose the modification of the order in some activities, as well as the suppression or fusion of others to facilitate the application of the methodology.

Resumen: La metodología Métrica v3 ha sido creada por el Ministerio de Administraciones Públicas de España, para llevar a cabo proyectos de la administración, pero también se está utilizando en proyectos de la empresa privada. La experiencia nos dice que el primer contacto con la metodología puede resultar complicado para gestores poco experimentados. Por ello se ha revisado la metodología incluyendo explicaciones tomadas de la bibliografía sobre el desarrollo de proyectos software, y se han incluido ejemplos que faciliten la comprensión. Además se propone la modificación del orden en algunas actividades, así como la supresión o fusión de otras para facilitar la aplicación de la metodología.

Palabras clave: Metodología de desarrollo, Ciclo de vida, Ingeniería del software.

1. INTRODUCCIÓN

Las Administraciones Públicas, conscientes de la importancia de seguir una metodología para desarrollar software de calidad, han promovido la utilización de metodologías para que sean usadas como referencia tanto por los organismos públicos, como por las empresas privadas. Así, la Administración Pública francesa creó la metodología MERISE, y la Administración Pública del Reino Unido creó SSADM.

Además, se siguen desarrollando otras metodologías, como el Proceso Unificado de Modelado de Jacobson, Booch y Rumbaugh, dirigido a proyectos Orientados a Objetos.

En España, el Ministerio de Administraciones Públicas desarrolló la metodología Métrica, que actualmente está en su versión 3.

Métrica Versión 3 establece tres grandes etapas:

1. Planificación de Sistemas de Información. Esta etapa no es objeto de estudio en esta revisión. Consiste en analizar las necesidades de sistemas de información que tiene la organización.
2. Desarrollo de Sistemas de Información. Esta es la etapa en la que se va a centrar la presente revisión. Se subdivide en los siguientes procesos:
 - a. Estudio de Viabilidad del Sistema (EVS)
 - b. Análisis del Sistema de Información (ASI)
 - c. Diseño del Sistema de Información (DSI)
 - d. Construcción del Sistema de

Información (CSI)

e. Implantación y Aceptación del Sistema (IAS)

3. Mantenimiento del Sistema de Información. El proceso de mantenimiento queda fuera del alcance de esta revisión.

Además Métrica Versión 3, establece las interfaces de Gestión de Proyectos, Gestión de Configuración y Gestión de Calidad que se aplican a lo largo de todos los procesos.

Los objetivos que se pretenden con la optimización son:

- Revisar la Metodología Métrica Versión 3, centrándose en proyectos Orientados a Objetos.
- Aclarar los puntos que en la realización de un proyecto supusieron dificultades.
- Reorganizar y unificar y eliminar tareas, para facilitar el desarrollo y la comprensión de la metodología. La rigidez en la interpretación y utilización de la metodología puede tener efectos no deseados en el proyecto, por lo que la metodología debe mirarse desde las necesidades del proyecto.
- Aportar ejemplos de documentación

El alcance de la optimización se centra en la etapa de Desarrollo de Sistemas de Información y las interfaces de Gestión de Calidad, Gestión de Configuración y Gestión de Proyectos.

En el presente artículo se citan las aportaciones principales que se proponen para la metodología en sus diversas fases.

2. ESTUDIO DE VIABILIDAD DEL SISTEMA

El objetivo del proceso de Estudio de Viabilidad del Sistema es analizar un conjunto concreto de necesidades que tiene una organización, para proponer una solución a corto plazo, que tenga en cuenta restricciones económicas, técnicas, legales y operativas.

En esta fase se deben identificar los requisitos de usuario, a fin de tomar una decisión sobre las medidas que mejor solucionen los problemas que éste plantee. En este sentido se propone adelantar la toma detallada de requisitos de usuario, que

inicialmente la metodología ubica en el Análisis del Sistema de Información. A través de la toma detallada de requisitos la decisión que se tome, así la estimación de esfuerzo y costes para determinar la viabilidad del sistema, tendrá muchas posibilidades de ser la mejor, ya sea a través del desarrollo de un nuevo proyecto, a través de un sistema comercial existente en el mercado o una combinación de ambos.

La metodología propone dentro del EVS, la identificación de los Usuarios que participarán en el Estudio de la Situación Actual. Dentro de la actividad EVS 1 Establecimiento del Alcance del Sistema, se propone la identificación de todos los interesados del sistema. El concepto de usuarios puede llevar a equívocos, por lo que se incluye una tarea de identificación de los "stakeholders" o interesados en el sistema.

Los interesados clave en todos los proyectos incluyen [PMBOK 2000]:

- Project manager: la persona responsable de la gestión del proyecto.
- Clientes o usuarios: la persona u organización que utilizará el sistema. Existen múltiples niveles de usuarios. Por ejemplo para un producto farmacéutico, los clientes serán los doctores que prescriben los medicamentos, los pacientes que los toman y las aseguradoras que pagan por ellos. En algunos sistemas las palabras usuario y clientes son sinónimos, mientras que en otros casos el cliente se refiere a la persona u organización que paga el sistema y los usuarios son aquellos que lo utilizarán directamente.
- Organización que va a llevar a cabo el proyecto: la empresa cuyos empleados llevarán a cabo el proyecto para construir el sistema. Evidentemente la empresa también tiene interés en el proyecto pues buscará unos objetivos de beneficios.
- Miembros del equipo de proyecto: el grupo de personas que va a llevar a cabo el proyecto para crear el nuevo sistema.
- Sponsors: las personas o grupos internos o externos que proveen recursos financieros, bien sea en efectivo o de otro tipo (equipos, software, etc.) para el proyecto.

Todos estos interesados deben ser consultados y tomados en consideración tanto en la toma de requisitos como durante el desarrollo del sistema.

Una vez que se ha definido el alcance del sistema, se identifican los interesados en el sistema y se

conoce a través de ellos la situación actual se propone la toma detallada de requisitos de usuario. Métrica v3 propone la identificación de requisitos generales para el estudio y valoración de las alternativas de solución del problema. Se propone detallar más los requisitos para asegurar que la solución es la más adecuada a los problemas plantados. De esta forma se aprovechan las primeras reuniones con los interesados para conocer mejor el problema.

También se hace notar que los requisitos normalmente no son inamovibles, y conforme se avanza en el proyecto pueden surgir nuevos requisitos o modificaciones de los ya identificados, por lo que el catálogo de requisitos debe ser retroalimentado cuando se profundiza en el estudio del sistema.

3. INTERFAZ DE GESTIÓN DE CALIDAD

Actualmente la calidad es un aspecto muy importante en todas las organizaciones. Cada vez son más las que buscan y obtienen la certificación ISO 9000, para que sus clientes perciban que la organización no solo realiza su trabajo, sino que además lo realiza satisfaciendo las necesidades del cliente y realiza un trabajo de calidad. Un sistema de calidad basado en ISO tiene grandes ventajas para los clientes ya que:

- Garantiza la capacidad de la empresa para cumplir las especificaciones.
- Hay un sistema de registro que contiene la documentación de todos los procesos que se llevan a cabo para la realización de un producto, es decir, la empresa tiene los procesos documentados, con lo cuál garantiza que sabe hacer el producto, y que además lo hace bien.
- Están definidas las responsabilidades y la cualificación del personal que realiza las actividades, es decir, que cada uno sabe lo que tiene que hacer, y si falla algo se conoce quién es el responsable.

[PRESSMAN 2002] define la calidad del software como la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados, y con las características implícitas que se espera de todo software desarrollado profesionalmente.

Más coloquialmente se podría definir la calidad como cumplir las necesidades y expectativas del cliente, en el plazo y los costes acordados, siguiendo los estándares aplicables y manteniendo documentado todo el proceso, de forma que sea posible repetir exactamente el mismo proceso para la generación de un nuevo sistema. Es importante cumplir las expectativas del cliente, porque si el sistema cumple los requisitos pero no es lo que espera el cliente, éste no quedará totalmente satisfecho.

Normalmente las empresas tendrán definido un Plan de Aseguramiento de Calidad que la organización impone para todos sus procesos. En este caso se debe adaptar la política de la organización marcada en ese plan al proyecto concreto que se va a desarrollar, estableciendo las revisiones y pruebas que resulten oportunas. Sin embargo se puede producir el caso de que dicho plan no haya sido definido aún, por lo que surge la necesidad de crear el Plan de Aseguramiento de Calidad para el proyecto sin una base previa. Por ello se propone la utilización del estándar IEEE 730-2002 para la creación de un Plan de Calidad, describiendo los puntos principales que debe contener para facilitar su creación.

Uno de los apartados del Plan de Aseguramiento de Calidad según el estándar IEEE-730-2002, es el Plan de gestión de Riesgos. Por ello en base a [CMMI 2002] y [PMBOK 2000], se muestran una serie de pasos que se pueden seguir para planificar la gestión de riesgos que pueden afectar al desarrollo del sistema.

Para verificar que las directrices y revisiones marcadas en el Plan de Aseguramiento de Calidad se están cumpliendo, se propone la utilización de matrices de trazabilidad que faciliten las comprobaciones, especialmente en la realización de las pruebas, en las que se debe asegurar que todos los requisitos y funcionalidades marcadas han sido recogidas en el sistema final.

4. INTERFAZ DE GESTIÓN DE CONFIGURACIÓN

Según [CUEVAS 2002] la Gestión de Configuración de Software es el proceso de identificar y definir los elementos de la configuración para:

1. Controlar la liberación y los cambios durante todo el ciclo de vida.

2. Registrar e informar de su estado y de las peticiones de cambios.
3. Verificar la corrección y acabado de los elementos.

Además [CUEVAS 2002] también define configuración software como el conjunto de documentos relacionados, ficheros fuentes y herramientas utilizadas en el diseño e implantación en un sistema software.

El objetivo de la gestión de la configuración es mantener la integridad de los productos que se obtienen a lo largo del desarrollo de los sistemas de información, garantizando que no se realizan cambios incontrolados, y que todos los participantes en el desarrollo del sistema disponen de la versión adecuada de los productos que manejan. Por ello, entre los elementos de la configuración software no se encuentran solamente los ejecutables y el código fuente, sino también todos los modelos y documentos que se generan durante el ciclo de vida.

En este apartado, se han propuesto ejemplos de formatos que se pueden utilizar para la creación de un Plan de Gestión de Configuración, así como diversos ejemplos del contenido del plan, que facilitan la comprensión de los conceptos y la creación del mismo, especialmente en para el proceso de cambio.

5. GESTIÓN DE PROYECTOS

La Gestión del Proyecto tiene como finalidad principal la planificación, el seguimiento y el control de las actividades y de los recursos, tanto humanos como materiales, que intervienen en el desarrollo de un Sistema de Información, para evitar desviaciones de costes, duración o funcionalidad del proyecto. Con este control se persigue conocer en todo momento qué problemas se producen y resolverlos o paliarlos a la mayor brevedad para minimizar sus consecuencias.

Podemos agrupar las actividades de gestión en tres bloques:

- Gestión Inicial (GPI): al principio del proyecto, al concluir el Estudio de Viabilidad del Sistema, se realizarán las actividades de Estimación de Esfuerzo y Planificación de Proyecto.
- Gestión de Seguimiento y Control (GPS): comprenden desde la asignación de las tareas hasta su asignación interna por parte del

equipo del proyecto, incluyendo la gestión de incidencias y cambios en los requisitos que puedan presentarse y afectar a la planificación del proyecto. El Seguimiento y Control del proyecto se realiza durante los procesos Análisis del Sistema, Diseño del Sistema, Construcción e Implantación del Sistema y Mantenimiento.

- Actividades de Finalización del Proyecto. Al concluir el proyecto se realizan las tareas propias del Cierre del Proyecto y Registro de la Documentación de Gestión.

En cada una de los bloques que se enumeran se han propuesto distintas tareas y técnicas que facilitan y mejoran la gestión del proyecto.

5.1 Inicio de proyecto

En el inicio del proyecto se estimará el esfuerzo y tiempo necesario para llevarlo a cabo y se planificarán las actividades a realizar a lo largo del mismo, estableciendo los hitos que se consideren necesarios, teniendo en cuenta las revisiones y pruebas marcadas en el Plan de Aseguramiento de Calidad.

Para la planificación se propone la utilización de técnicas de diagramación WBS (Work Breakdown Structure), OBS (Object Breakdown Structure) y RBS (Resource Breakdown Structure) para facilitar la identificación de tareas y productos resultantes que se deben tener en cuenta en la estimación y planificación del proyecto.

5.2 Seguimiento y control del proyecto

El objetivo de la Gestión del Proyecto es que el proyecto se entregue en la fecha convenida, con el coste acordado y con la funcionalidad y calidad requerida por el cliente. El seguimiento y control del proyecto tiene como objetivo fundamental la vigilancia de todas las actividades de desarrollo del sistema. Es una de las labores más importantes en todo desarrollo de sistemas, ya que un adecuado control hace posible evitar desviaciones en costes y plazos, o al menos detectarlas cuanto antes.

En esta fase se deben prestar atención a los indicadores identificados en el plan de gestión de riesgos que indiquen que se ha producido o se va a producir alguno de los riesgos que se han identificado. Cuando se detecta esta situación, es

necesario llevar a cabo medidas correctoras destinadas a evitar que el riesgo se produzca, o bien paliar sus consecuencias a partir de los planes de contingencia que se hayan establecido.

Dentro del seguimiento del proyecto, se propone, además del seguimiento de tareas que propone Métrica Versión 3, el seguimiento de los indicadores definidos en [MONTENEGROb 2004]:

- Actividad GPS 3: Seguimiento de Tareas.
- Seguimiento de Recursos.
 - Recursos Humanos.
 - Recursos Materiales.
- Seguimiento de Costes
 - Costes derivados del personal.
 - Costes derivados por recursos materiales.
 - Costes derivados de las actividades.
 - Costes acumulados.
- Seguimiento de Clientes.
 - Satisfacción Respecto al Equipo de Trabajo.
 - Satisfacción del Cliente Respecto a la Tecnología.
 - Participación del Cliente en el Proyecto.
 - Participación del Cliente en el Incremento de Requisitos.
- Seguimiento de Usuarios.
 - Interfaz de Usuario.
 - Satisfacción del Usuario Respecto al Equipo del Proyecto.
 - Participación del Usuario.

5.3 Gestión de finalización de tareas y finalización del proyecto.

En la gestión de finalización se identifican las tareas que se deben realizar cuando se finaliza una tarea o cuando se cierre el proyecto. Principalmente se refiere a la documentación y revisiones que se realizan en la conclusión de cada tarea y del proyecto en general. En este sentido se ha definido el Documento Histórico de Proyecto, que se detallará más adelante.

6. ANÁLISIS DEL SISTEMA DE INFORMACIÓN

El objetivo del Análisis del Sistema de Información es obtener una especificación detallada del sistema que se va a construir. Así

mismo, el análisis servirá como base para el posterior diseño del sistema.

La fase de análisis es la fase de definición del problema. En esta fase no se pretende solucionar el problema del que surge la necesidad del sistema. Se trata de captar las necesidades que debe resolver, y modelar el problema utilizando distintas técnicas, en función de las características del proyecto, para posteriormente, en el diseño del sistema, resolverlo.

En determinadas tareas del proceso de análisis deben participar también los interesados del sistema. Es necesario definir los interesados que deben participar en este proceso, determinando los perfiles, las responsabilidades que tendrán en el proceso y la dedicación al proceso necesaria. La participación de los interesados es especialmente relevante para el establecimiento de requisitos y la especificación detallada del nuevo sistema.

La principal propuesta que se realiza en esta fase es la identificación de los requisitos software. Al hilo del adelanto de la toma de requisitos de usuario al EVS, se adelanta a la fase de Análisis la definición de requisitos software para profundizar en la definición del problema.

Antes de la toma de requisitos software se definirán los Diagramas Casos de Uso derivados de los requisitos de usuario que ya se habían identificado. A partir de ellos se avanza en la identificación de los requisitos software.

El propósito de la definición de requisitos software es analizar los requisitos de usuario y producir un conjunto de requisitos que debe cumplir el software, tan completo, constante y correcto como sea posible. Los requisitos software son una visión del problema por parte del desarrollador. La documentación de requisitos debe seguir unas normas de estilo y redacción que posibiliten y facilite su comprensión. Las características de dicha documentación son [CUEVAS 2002]:

- Los requisitos son entendibles de una sola manera, es decir, no son ambiguos.
- Completa: la documentación recoge todos los requisitos que debe cubrir el sistema.
- Verificable: se puede comprobar que el sistema cumple con los requisitos, es decir, se puede establecer una acción para cada requisito que compruebe que el sistema le incorpora.
- Coherente: no hay requisitos contradictorios.

- Fácil de modificar. El procedimiento de cambio se especifica en el Plan de Gestión de Configuración.
- Fácil de identificar el origen y las consecuencias de los requisitos.
- Fácil de utilizar en el resto de fases.

En la optimización se han incluido formatos para la representación de requisitos.

Seguidamente la metodología propone el análisis de los casos de uso, la creación del modelo de clases y su análisis. Pese a que para desarrollos orientados a objetos la metodología no aplica la creación de un modelo de datos, en la optimización se ha tomado en cuenta, pues puede facilitar la identificación de necesidades de migración y carga inicial.

La definición de la interfaz de usuario en el análisis, según algunos autores, es cuestión de gustos, pudiendo dejar esta actividad para el Diseño pues forma parte de él. Sin embargo, en la optimización se ha considerado que puede ser útil la definición en el Análisis, pues facilita la visión del sistema a los interesados que no son informáticos. Para la definición de las interfaces se parte de la descripción de cada escenario en los casos de uso. [BRAUDE, 2001] define los siguientes pasos que se pueden seguir para definir la interfaz de usuario:

- Paso 1: Conocer al usuario.
- Paso 2: Entender la función de negocio en cuestión.
- Paso 3: Aplicar principios de diseño de diálogos.
- Paso 4: Seleccionar el tipo adecuado de ventanas.
- Paso 5: Definir los menús del sistema.
- Paso 6: Seleccionar los controles adecuados al tipo de interfaz.
- Paso 7: Elegir los controles apropiados a la pantalla.
- Paso 8: Organizar y presentar las ventanas.
- Paso 9: Elegir los colores apropiados.
- Paso 10: Crear iconos significativos.
- Paso 11: Proporcionar mensajes y retroalimentación al resto de procesos.

Estos pasos se desarrollan en las tareas que componen la actividad de Definición de Interfaces de Usuario y se completarán en el proceso de diseño.

Finalmente se completa el proceso de análisis con ejemplos de las revisiones de consistencia entre las especificaciones del sistema y los requisitos, utilizando matrices de trazabilidad, así como de la definición del plan de pruebas del sistema.

7. DISEÑO DEL SISTEMA DE INFORMACIÓN

El objetivo del Diseño del Sistema de Información es resolver el problema que se ha descrito y modelado en el Análisis del Sistema de Información. En este proceso se va a definir cómo se soluciona el problema planteado, detallando los algoritmos y técnicas concretas a utilizar. También se realiza una especificación detallada de los componentes del sistema.

Además, en este proceso se define completa y detalladamente la arquitectura del sistema y el entorno tecnológico en el que va a funcionar.

Por arquitectura del sistema no solo nos vamos a referir a arquitectura física, sino también a la arquitectura de componentes software que van a formar parte del sistema. Es por ello que proponemos como nombre Diseño de Arquitectura Software. Según [RUMBAUGH 1996], en todas las aplicaciones, salvo en las más pequeñas, el primer paso para diseñar un sistema consiste en dividir el sistema en un pequeño número de componentes. Cada uno de los componentes principales de un sistema se llama subsistema. Cada subsistema abarca aspectos del sistema que comparten alguna propiedad común, como puede ser una funcionalidad similar, la misma ubicación física, o la ejecución en el mismo tipo de hardware.

Como los requisitos software ya han sido definidos en el proceso de análisis, en la fase de Diseño solo se van a definir los procedimientos de operación y seguridad derivados de los requisitos ya identificados, completando aquellos que sea necesario a partir de la arquitectura software propuesta.

La siguiente actividad propuesta en la metodología es el Diseño de Subsistemas de Soporte. Se propone en este punto identificar también los mecanismos genéricos de diseño, como la identificación de patrones, resultando una única actividad que se debería denominar Diseño de Arquitectura Software. Citando a [JACOBSON 2000], se trata de realizar un diseño de alto nivel del sistema que se va a desarrollar: es el primer

nivel de descomposición, previo al diseño detallado. La arquitectura software se centra tanto en los elementos estructurales significativos (subsistemas específicos y subsistemas de soporte identificados en la Actividad DSI 1: Definición de la Arquitectura del Sistema), como en las colaboraciones que tienen lugar entre estos elementos a través de las interfaces.

En paralelo al Diseño de la Arquitectura Software se van a definir los Casos de Uso Reales y se va a realizar el Diseño de las Clases, a partir de los resultados que provienen del Análisis del Sistema.

Tras las actividades anteriores se realiza el Diseño Físico de Datos, a partir de las clases que se han identificado, especificando las formas de acceder a los datos y la ubicación de los mismos.

El siguiente paso que propone la metodología, previo a la definición detallada de los componentes del sistema, es verificar que se han cumplido con los estándares y criterios de diseño que se han marcado para el diseño, así como comprobar que el diseño que se ha realizado hasta ese momento es coherente con el análisis previo.

Tras la verificación y aceptación de la arquitectura del sistema se realiza el diseño detallado de los componentes. Esta actividad constituye el diseño detallado del sistema de información. También se generan las especificaciones para la construcción del sistema de información, a partir del diseño de la arquitectura, los casos de uso y las clases identificadas durante el proceso de diseño. Se debe especificar hasta el pseudocódigo que define la funcionalidad de cada uno de ellos. En la optimización se propone un formato que se puede seguir en la definición de cada componente.

Seguidamente se diseña la migración y carga inicial de datos, generando los componentes que resulten necesarios.

En la especificación técnica del plan de pruebas se describen los distintos niveles de prueba que se pueden llevar a cabo y se ilustra con ejemplos cada uno de ellos.

Finalmente, se establecen los requisitos que deberá cumplir la documentación de usuario y así como el entorno en el que el sistema deberá ser implantado.

8. CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN

En este proceso se genera el código de los componentes del sistema de información, se desarrollan todos los procedimientos de operación y seguridad y se elaboran todos los manuales de usuario final y de explotación, con el objetivo de asegurar el correcto funcionamiento del Sistema para su posterior implantación.

En la construcción se parte del resultado del diseño y se implementa el sistema en términos de componentes, es decir, ficheros de código fuente, scripts, ficheros de código binario, ejecutables y similares [JACOBSON 2000].

La metodología propone en primer lugar preparar el entorno de construcción, cargando las bases de datos o ficheros necesarios, e instalando los equipos y el software necesario para llevar a cabo la construcción, de acuerdo con los requisitos especificados.

Tras la preparación del entorno de construcción, se lleva a cabo la codificación de todos los componentes que se han diseñado en la fase anterior.

Para la documentación de las pruebas se propone un formato de control para la realización de las pruebas en el que se debe incluir la firma de cada uno de los responsables de la realización de las pruebas.

Durante la construcción de los componentes se llevan a cabo las pruebas de integración e integración, y una vez finalizada la construcción se llevan a cabo las pruebas del sistema (las que verifican que todos los subsistemas funcionan conjuntamente), incluyendo las pruebas de volumen que se hayan establecido.

La siguiente actividad propuesta por la metodología es la generación de los componentes necesarios para la migración y carga inicial de datos, si son necesarias. La generación de la documentación de usuario, que la metodología ubica en esta fase, en la presente optimización se ha decidido trasladar al proceso de Implantación y Aceptación del sistema, pues es cuando se forma a los usuarios.

Finalmente, una vez realizadas todas las pruebas descritas para esta fase, el equipo del proyecto da su aprobación al sistema.

En el proceso de construcción no se han propuesto cambios significativos respecto a las actividades propuestas en la metodología. Simplemente se han añadido algunos ejemplos y formatos para el control de las pruebas que se realizan.

9. IMPLANTACIÓN Y ACEPTACIÓN DEL SISTEMA

En este proceso se realiza la entrega al cliente del Sistema de Información ya construido.

En primer lugar se lleva a cabo la definición del plan de implantación y se forma a los encargados de llevar a cabo dicha implantación. Se ha trasladado a esta fase la redacción de la documentación de usuario, pues es en esta fase en la que se lleva a cabo la formación de los usuarios finales. Se complementa la documentación de usuario propuesta por la metodología con la redacción de dos documentos distintos:

- Manual de usuario: Expone los procesos que el usuario puede realizar con el sistema implantado. Para lograr esto, es necesario que se detallen todas y cada una de las características que tienen los programas y la forma de acceder e introducir información. El manual de usuario permite a los usuarios conocer el detalle de qué actividades deberán desarrollar para la consecución de los objetivos del sistema. Reúne la información, normas y documentación necesaria para que el usuario conozca y utilice adecuadamente la aplicación desarrollada.
- Manual de Referencia: en el manual de referencia se especifican todas las funcionalidades que se pueden hacer con el sistema. No todo lo que aparece en el manual de usuario es lo que se puede hacer con el sistema. En el manual de usuario se especifican las funcionalidades que va a necesitar el usuario, y cómo puede utilizarlas. El manual de referencia recoge la funcionalidad completa del sistema, incluyendo llamadas a funciones que puedan realizar los programadores de otro sistema, detalles de administración, modos de depuración, etc.

Tras la generación de la documentación de usuario se lleva a cabo la formación de los usuarios finales y se lleva a cabo la instalación del sistema en el

entorno de operación.

Una vez se ha llevado a cabo la instalación en el entorno de operación se realizan las pruebas de implantación y de aceptación del sistema.

Tras las pruebas se prepara el plan de mantenimiento del sistema, se acuerdan los servicios que en adelante se deberán prestar por parte del equipo de desarrollo y finalmente se presenta el sistema para su aprobación.

Finalmente, una vez se ha aprobado y aceptado el sistema por parte del cliente, se pasa producción, comenzando la explotación del sistema de información.

10. DOCUMENTO HISTÓRICO DE PROYECTO

En la optimización se aporta la idea de crear al final del proyecto el documento Histórico de Proyecto. El objetivo del Documento Histórico de Proyecto es recoger en un solo documento toda la información relevante que se ha ido recogiendo en diversos documentos a lo largo del ciclo de vida del proyecto. Al incluir dicha información en un único documento, se facilita su consulta para aprovechar la experiencia adquirida en proyectos posteriores.

Se propone el siguiente contenido para el documento:

- Descripción del proyecto.
- Ejemplos de funcionamiento.
- Gestión del Proyecto.
 - Organización del Proyecto.
 - Estructura Organizativa.
 - Límites Organizativos y Relación Entre los Participantes
 - Métodos Empleados.
 - Tecnologías, equipos y herramientas empleados.
- Planificación.
 - Producción Software.
 - Estimación Inicial.
 - Estimación Final.
 - Datos Reales.
 - Conclusiones.
 - Documentación. Listado de todos los documentos que se han generado.
 - Esfuerzo Estimado vs Esfuerzo Real.
 - Recursos Computacionales.

- o Análisis de los Factores de Productividad.

Revisión del Aseguramiento de la Calidad. En este punto se resumen de las revisiones que se han realizado y los resultados que se han obtenido.

11. MEJORAS PROPUESTAS

Tras la revisión del proceso propuesto por Métrica V3, hemos identificado la necesidad de adelantar la identificación de requisitos de usuario a la fase de estudio de viabilidad, y la identificación de requisitos software a la fase de análisis. Por otro lado la identificación de casos de uso y subsistemas queda separada de los requisitos, creando dos caminos que se recorren en paralelo. Por un lado el sistema se define a través de los requisitos y por otro a través de casos de uso, diagramas de secuencia, para al final, converger en una única solución que no pase nada por alto. El proceso, comenzaría en la definición de requisitos de usuario en el estudio de viabilidad. A partir de los requisitos de usuario se definen los requisitos de software en el análisis, y a partir de estos se identifican componentes de arquitectura y componentes de bajo nivel en el diseño. En paralelo se realiza un análisis de casos de uso, identificando clases de análisis, modelo de datos, y un diseño de clases detallado. Al final, ambos procesos concluyen en una definición de componentes de bajo nivel, que en la fase de construcción serán implementados.

Los interfaces de gestión de proyectos y gestión de configuración constituyen ejes básicos para que el proceso reorganizado funcione correctamente. En la gestión del proyecto se deben tener en cuenta los indicadores que muestran que el proyecto evoluciona correctamente, en especial aquellos relacionados con la satisfacción del cliente y su grado de participación en el proyecto. Una métrica adecuada para estos parámetros proporcionará un índice de modificación de requisitos más bajo. En cuanto a la gestión de configuración, al utilizar dos caminos en paralelo, es fundamental el control de los elementos resultado de cada fase de refinamiento. La trazabilidad entre los elementos de las dos vías se torna crítica, y la utilización de herramientas de gestión de configuración y de gestión de requisitos facilita la labor.

Las metodologías no siempre proponen una técnica o una guía de documentación concreta a utilizar. Por ello hemos propuesto formatos de documentos

que permitan centrar el esfuerzo en aplicar la metodología, en lugar de pensar en el formato adecuado.

12. CONCLUSIONES

En este trabajo se han revisado las actividades y tareas de la metodología Métrica Versión 3 del Ministerio de Administraciones Públicas de España. Se han completado y aclarado las definiciones de las tareas, facilitando su comprensión, añadiendo ejemplos para ayudar en la documentación de cada una de las fases.

Con los cambios de las tareas con respecto a la organización propuesta en la metodología, se pretende mostrar que la metodología no es una secuencia inamovible de tareas que deben ser ejecutadas en su totalidad y en el orden propuesto: las características concretas de los proyectos pueden hacer que determinadas tareas no se apliquen, y que otras puedan ser ejecutadas en un orden distinto.

Esta es una aportación que pretende facilitar la aplicación y comprensión de la metodología, pero como siempre se puede mejorar. Cada experiencia de aplicación que se comparta ayudará a mejorar el proceso de desarrollo de software y esto repercutirá en la construcción de un software de mayor calidad.

REFERENCIAS

A guide to the project management body of knowledge : PMBOK guide. Project Management Institute, 2000 [PMBOK 2000]

Capability Maturity Model® Integration (CMMISM), Version 1.1 Software Engineering Institute, 2002 [CMMI 2002]

Metrica. Versión 3. Ministerio de Administraciones Públicas. [MAP]

BRAUDE, Eric J. SOFTWARE ENGINEERING An Object-Oriented Perspective. John Wiley & Sons, INC. [BRAUDE 2001]

BURGESS, T.F.; McKee D., Kidd C. Configuration Management in the Aerospace Industry: a review of industry practice. International Journal of Operations & Production Management Volume 25, No 3, 2005. [BURGESS 2005].

CUEVAS AGUSTÍN, Gonzalo. Gestión del

Proceso Software. Editorial Centro de Estudios Ramón Areces, S.A. [CUEVAS 2002].

JACOBSON, BOOCH, RUMBAUGH. El Proceso Unificado de Desarrollo de Software. Addison Wesley. [JACOBSON 2000].

MONTENEGRO, Marilú y GARCÍA, Ángel. Representación visual de la Gestión de Requisitos en la Gestión de Proyectos Informáticos. 30ma Conferencia Latinoamericana de Informática (CLEI2004) pp.402—413. [MONTENEGROa 2004].

MONTENEGRO, Marilú; GARCÍA, Ángel; URQUIZA, Alfonso. Information Technology Graphics in the Software Engineering. WSEAS Trans. on INFORMATION SCIENCE and APPLICATIONS, December 2004. [MONTENEGROb 2004].

PRESSMAN, Roger S. Ingeniería del Software. Un Enfoque práctico. Adaptado por Darle Ince. Mc Graw Hill. Quinta Edición. 2002 [PRESSMAN 2002]

RUMBAUGH, James. Modelado y Diseño Orientado a Objetos. Blaha Michael, William Premerlani, Eddy Frederick, Lorenzen William. Prentice Hall, 1996. [RUMBAUGH 1996]

RUMBAUGH, James. El Lenguaje Unificado de Modelado. Addison Wesley, 2000. [RUMBAUGH 2000]

SOMMERVILLE, Ian. Integrated Requirements Engineering: A Tutorial. Software, IEEE Volume 22, Issue 1, Jan.-Feb. 2005 Page(s):16 - 23 [SOMMERVILLE 2005].