



Universidad de las Ciencias Informáticas

FACULTAD 10

Título: Propuesta de un procedimiento para la planificación de proyectos con herramientas libres.

Trabajo de Diploma para optar por el título de Ingeniero Informático

Autores: Bertha Román Martínez

Dayani Flores Pérez

Tutores: Ing. Yanko Hernández Valdés

Ing. Greidys Jorda Lueges

“Año 50 de la Revolución”
Ciudad de la Habana, Cuba. Julio del 2008.



Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio, conscientes de recibir el premio de la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.

Ché

Declaración de Autoría

Por este medio declaramos ser los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas (UCI) para que haga el uso que estime pertinente con este trabajo.

Para que así conste firmamos la presente a los __ días del mes de _____ del _____.

Firma de la Autora
(Bertha Román Martínez)

Firma de la Autora
(Dayani Flores Pérez)

Firma del Tutor
(Ing. Yanko Hernández Valdés)

Firma del Tutor
(Ing. Greidys Jorda Lueges)

Agradecimientos

Compartidos:

A la Revolución, por brindarnos la oportunidad de ser mejores y por confiar su futuro en nosotros.

A Fidel, nuestro Comandante en Jefe, máximo guía y forjador de esta gran casa de estudios que es la UCI, en la que hemos hecho realidad nuestros sueños como estudiantes, y a la que hemos de llevar siempre en nuestros corazones, por aportarnos durante estos 5 años un cúmulo imperecedero e invaluable de conocimientos y experiencias.

A nuestros tutores Greidy y Yanfo por su ayuda, empeño y dedicación.

Muchas gracias.

Dayani:

A mi abuelita querida: Felicia, quien ha sido para mí un estandarte de virtudes y un ejemplo a seguir, quien a dedicado todos estos años a guiarme por el camino correcto, por su paciencia y dedicación, por sus consejos sin los cuales no habría podido convertirme en la mujer que soy hoy.

A mi madre querida, por creer siempre en mí, por estar siempre a mi lado en todo momento, por su compañía y apoyo, por ser siempre más que una madre una amiga, por sus consejos, su paciencia y su amor. Por ser la mejor madre del mundo.

A mi tío Manuel, que siempre me ha apoyado, ayudado y querido como si fuese su hija, a quien le voy a estar eternamente agradecida por ser un padre cuando lo necesité.

A mi padre, por apoyarme a pesar de la distancia, y confiar siempre en mí.

A mis hermanitos queridos: Brian, Ernestico y Efrencito, por ser su Tata.

A mi padrastro Emilio, por todo su apoyo, dedicación y consejos, mil gracias, y a Yolanda, mi segunda tía, a quien estaré eternamente agradecida por apoyarme tanto y ayudarme durante estos 5 años.

Al resto de mi familia, por confiar en mí y estar siempre cuando los necesité.

A Bertha, por haber sido una excelente compañera de tesis, y una gran amiga, por soportarme en todo momento mis intentos de que todo quedara perfecto, por estar siempre cuando la necesité.

A mi amiga Yadira, por su apoyo, compañía y comprensión, por soportarme en todo momento y escucharme cuando más lo necesité, por confiar siempre en mí.

A mis amigos (Andrés, Alexeis, Darys, Yoel, Marisleydis, Yunieski, Aenlys, Ernesto), a mis compañeros de grupo, los nuevos y los viejos, y a todos aquellos que de una forma u otra han estado siempre cuando los he necesitado.

Bertha:

Antes que todo quiero agradecerles a mis dos lindos y maravillosos padres, por haber estado a mi lado siempre, por apoyarme en todo, por ayudarme a levantarme cuando me caía, por sus consejos, su amor, dedicación, cariño, respeto, por haber hecho de mí la mujer que hoy soy. Por ser mis guías, mis límites ante mis excesos y sobre todo por darme la oportunidad de ser su hija. A ellos hoy y siempre, mil gracias.

A mi novio Aenlys, por haberme brindado tanto amor y cariño en estos 5 años, por ayudarme tanto, por su paciencia y comprensión, por haber sido más que novio compañero y amigo y por haber sacado de mis labios siempre una sonrisa cuando de mis ojos salían lágrimas. A ti, mi amor, gracias, sin ti no lo hubiese logrado.

A mis hermanos Daniel y Yoleski por haberme ayudado y aconsejado a lo largo de toda mi vida.

A mis suegros por su ayuda y cariño.

A Dayani, por haber sido una magnífica compañera de tesis y una excelente amiga, por sus consejos y por estar a mi lado cuando la necesité.

A mis amigos (Yailín, Yadira, Yunieski, Nathyara, Henry, Andrés, Axel, Piti, Orelki, Yeimi y Tony) y a todos aquellos que de una forma u otra me han ayudado en esta hermosa tarea.

Al resto de mi familia por su ayuda y preocupación.

Dedicatoria

Dayani:

A mi abuela Felicia, por ser mi gran motivo de inspiración, y mi luz y guía en todo momento.

A mi mamá, por ser la mejor y más maravillosa del mundo, por su amor, apoyo y dedicación.

A mi único tío, por ser siempre un gran padre para mí, por su paciencia, comprensión y afán por que sea cada día mejor.

A mi padre, a mis hermanos, y al resto de mi familia por confiar siempre en mí.

A todos ellos, por existir, mil gracias.

Bertha:

A mis padres por haberme dado las alas que necesité para volar, por ser la luz que guía mi camino y porque estoy convencida, que si existen padres más cariñosos, comprensivos y tiernos que los míos, no son terrestres.

A Tato, mi novio, por su amor, apoyo y estar a mi lado en los momentos bueno y malos.

A la memoria de mi abuela Bertha y mi tío Luis, que aunque hoy no estén a mi lado han sido para mi fuente de inspiración.

Resumen

La gestión de proyectos tiene como finalidad principal la planificación, seguimiento y control de las actividades y recursos humanos y materiales, que intervienen en el desarrollo del software convirtiéndose en uno de los aspectos más importantes. Es la encargada de organizar y administrar los recursos permitiendo que se pueda planificar, dirigir y controlar las tareas que conduzcan a alcanzar los requisitos de un sistema aceptable, con un costo mínimo y dentro de un período de tiempo específico.

En la Universidad de las Ciencias Informáticas (UCI), los proyectos actualmente afrontan problemas debido a la realización de incorrectas e irreales planificaciones. Esto ha afectado en gran medida en el cumplimiento de entrega de los productos en fecha, la calidad e incumplimiento de los requerimientos de los clientes. Este problema se le atribuye a que la escuela no cuenta con suficientes datos históricos de proyectos ya finalizados en los cuales se puedan basar otros para realizar sus planificaciones.

Debido a los problemas que presentan los proyectos productivos de la UCI hoy día, en cuanto a no poseer un procedimiento que ayude y guíe a los integrantes de un equipo de trabajo, a realizar planificaciones lo más reales y certeras posible; es que surge el tema de nuestro trabajo de diploma: *“Propuesta de un procedimiento para la planificación de proyectos con herramientas libres”*.

Este trabajo tiene como objetivo proponer un procedimiento que ofrezca un conjunto de pasos que guíen el proceso de planificación de los proyectos productivos en la UCI.

Palabras claves: Gestión de Proyecto, planificación, desarrollo software, procedimiento.

Tabla de Contenidos (Índice)

Introducción	1
Capítulo I: Fundamentos Teóricos	5
1.1 Procesos de Desarrollo de Software	5
1.1.1 ¿Qué es el Proceso de Desarrollo de Software?	5
1.1.2 Metodologías para el Proceso de Desarrollo del Software	6
1.1.2.1 Metodologías Tradicionales (“no ágiles”)	7
1.1.2.2 Metodologías Ágiles	9
1.2 Gestión de proyecto	14
1.2.1 Actividades de la Gestión de Proyectos.....	15
1.3 La planificación dentro de la Gestión de Proyecto.....	16
1.3.1 Importancia de realizar una buena planificación	17
1.3.2 Objetivos de la planificación	17
1.3.3 Actividades asociadas a la planificación de proyectos.....	18
1.3.3.1 Ámbito del Software.....	18
1.3.3.2 Recursos	19
1.3.4 Artefactos del proceso de planificación	20
1.3.5 Los riesgos y la planificación	21
1.4 La estimación dentro de la Gestión de Proyectos.....	23
1.4.1 Factores que influyen en la estimación.....	24
1.4.2 Técnicas de estimación	25
1.4.3 Métodos de estimación que existen.....	25
1.5 Modelos de calidad para la planificación	29
1.5.1 Modelos de Madurez de Capacidades (CMM).....	30
1.6 Herramientas de apoyo para la Gestión de Proyectos.....	31
1.7 Conclusiones parciales	36
Capítulo II: Procedimiento propuesto	37
2.1 Caracterización de la Gestión de Proyectos y el Proceso de Desarrollo en los Proyectos en la UCI	37
2.1.1 Caracterización del Proceso de Desarrollo de Software	37
2.1.2 Caracterización de la Gestión de Proyectos y el proceso de Planificación de Software.....	38
2.1.3 Caracterización general sobre las encuestas	38
2.2 La planificación según CMMI	39

2.2.1 Fases que propone CMMI para la planificación	40
2.2.2 Tareas que propone cada fase.....	40
2.3 Selección de la herramienta para la Gestión de Proyectos.....	41
2.3.1 Módulo de Usuario	42
2.3.2 Módulo de Administración	46
2.4 Presentación del procedimiento propuesto.....	48
2.4.1 Definición	48
2.4.2 Contenido a tener en cuenta para planificar	48
2.4.3 Esquema general del procedimiento	49
2.5 Definición del procedimiento propuesto.....	50
2.5.1 Selección del equipo núcleo.....	50
2.5.2 Planificación del alcance (ámbito de software)	51
2.5.2.1 Definición del alcance.....	52
2.5.2.2 Estimación del alcance	53
2.5.2.3 El alcance y la comunicación con el cliente	53
2.5.3 Creación de la Estructura de Desglose de Trabajo (EDT)	55
2.5.3.1 Estilos para realizar un EDT	56
2.5.3.2 ¿Como confeccionar un EDT?.....	57
2.5.3.3 Diccionario del EDT	58
2.5.4 Estimaciones iniciales	58
2.5.4.1 Precisiones en la estimación	58
2.5.5 Identificación de riesgos	60
2.5.5.1 Identificar los riesgos.....	61
2.5.5.2 Tipos de riesgos	62
2.5.5.3 Analizar y priorizar los riesgos	64
2.5.6 Elaborar Plan de Proyecto.....	66
2.5.6.1 Planificación de Recursos Humanos (RR.HH.)	68
Elaboración del Plan de Conocimientos y Habilidades	70
Técnicas para la selección del equipo de trabajo.....	70
Roles que se definirán.....	72
2.5.6.2 Planificación de recursos de software reutilizables	74
2.5.6.3 Planificación de recursos de entorno	74
2.5.6.4 Planificación temporal.....	75
Técnicas para la planificación temporal	76

2.5.6.5 Planificación de costos	78
Componentes de los costos	78
¿Cómo planificar costos?	79
Curvas de costes acumulados.....	79
2.5.6.6 Planificación de riesgos	80
Identificar estrategias de mitigación.....	80
Identificar estrategias de contingencia.....	82
2.5.6.7 Elaboración del Plan de Gestión de Datos.....	83
2.5.6.8 Elaboración del Plan de Participación de los Interesados.....	84
2.5.7 Planificación de comunicaciones	84
2.5.8 Replanificación y control.....	86
2.6 Conclusiones parciales	86
Capítulo III: Validación del procedimiento	87
3.1 Proceso de selección de los expertos	87
3.2 Cantidad de Expertos seleccionados	88
3.3 Guía para la validación de la propuesta	88
3.4 Conclusiones parciales	92
Conclusiones	93
Recomendaciones	94
Referencias bibliográficas	95
Bibliografía consultada	97
Anexos.....	100
Glosario de Términos	106

Índice de figuras

Figura 1: Proceso de Desarrollo de Software.....	6
Figura 2: Metodología Rational Unified Process.....	8
Figura 3: Elementos de RUP.....	8
Figura 4: Fases de la Metodología Microsoft Solution Framework.	9
Figura 5: Metodología Extreme Programing.....	11
Figura 6: Ubicación de la planificación dentro de la Gestión de Proyecto.....	15
Figura 7: Pirámide de recursos en la estimación.....	19
Figura 8: Actividades asociadas con la gestión de riesgos.....	22

Figura 9: Estimación de proyectos por método COCOMO	27
Figura 10: Estimación de Proyectos por el método de Puntos de Función.	29
Figura 11: Detalles de un proyecto.....	43
Figura 12: Generación de informes de un proyecto.....	44
Figura 13: Tareas de los usuarios.	45
Figura 14: Estado de los usuarios de un proyecto.....	47
Figura 15: Esquema general del procedimiento.	49
Figura 16: Formato indentado.	56
Figura 17: Formato gráfico.	56
Figura 18: Lista de los 10 riesgos más frecuentes.....	62
Figura 19: Diagrama de red.	76
Figura 20: Diagrama de Gantt.....	77
Figura 21: Calendario de proyecto.	77
Figura 22: Curvas de costes acumulados.....	80

Índice de tablas

Tabla 1: Forma para realizar las estimaciones. (Estos datos son hipotéticos).....	60
Tabla 2: Tipos de riesgos según RUP.....	64
Tabla 3: Clasificación de los riesgos según su impacto.....	65
Tabla 4: Clasificación de los riesgos según su probabilidad.....	65
Tabla 5: Riesgos y estrategias para su planificación.	81
Tabla 6: Indicadores y acciones para elaborar el Plan de Contingencia.....	82
Tabla 7. Resultado del trabajo de expertos	89
Tabla 8. Tabla para el cálculo de concordancia de Kendall.....	91
Tabla 9. Tabla de calificación de cada criterio.....	91

Introducción

El desarrollo informático en el mundo actual crece a pasos gigantescos, pero el desarrollo de los software sigue siendo un tema complicado. Existen actualmente grandes conflictos respecto a la gestión del software, más aún en la planificación del producto.

La planificación de un proyecto debe afrontarse de manera adecuada para que una vez finalizado se obtenga un producto con la calidad requerida y represente un éxito para el grupo de trabajo. No se trata de una etapa independiente abordable en un momento concreto del ciclo del proyecto, pues según avance el mismo será necesario modificar tareas, reasignar recursos, etc., es decir, la planificación se lleva paralelamente al desarrollo del proyecto.

La planificación es un proceso de preparación de decisiones referentes al futuro del sistema a dirigir con lo que se condicionan y posibilitan futuras decisiones. Este proceso se basa en un pronóstico, previamente elaborado según métodos específicos, basándose en competencias especializadas, según las áreas de actividad del sistema.

La importancia que implica una adecuada planificación, como proceso que debe guiar todo el desarrollo del software, no solo está presente en las estimaciones en cuanto a costo y duración que debe tener el mismo, sino también en el hecho de garantizar una correcta organización del trabajo y una calidad superior del producto. Conocer a fondo cómo y hasta dónde se puede llegar es sin dudas una de las tareas más importantes. *“Buenos enfoques de estimación y datos históricos sólidos ofrecen la mejor esperanza de que la realidad puede vencer sobre las demandas imposibles”* [Jones C, 1998].

La planificación es entendida como el conjunto de actividades en que se prepara, por reflexión y trabajo metódico de preparación y se toman decisiones sobre alternativas de solución de problemas que posibiliten, enmarquen y ayuden a la futura toma de decisiones y lanzamiento de actividades del sistema a planificar [Smarciani, 2004].

La Universidad de las Ciencias Informáticas (UCI) es un centro donde actualmente se lleva a cabo una gran producción de software, tanto nacionales como de exportación. Pero el proceso de planificación en los proyectos productivos en la misma no se lleva a cabo o se realiza de manera inadecuada. El hecho de no tener una planificación adecuada o carecer de ella provoca resultados insatisfactorios tanto para la nuestra Universidad como para los clientes, pues no se pueden brindar fechas de

terminación exactas, incumplimiento con las fechas de entrega establecidas, afecta la toma de decisiones en el proyecto y la calidad de los entregables.

En la UCI se desarrollan diferentes tipos de software, multimedia, portales, software educativo, de gestión, entre otros, cada proyecto acumula experiencias propias en sus planificaciones y estas, a su vez, pueden servir a futuros proyectos para poder establecer comparaciones. Pero a la misma vez sucede que la UCI lleva 5 años de creada, y no cuenta con un gran número de software terminados en los cuales se puedan basar otros proyectos para realizar planificaciones más certeras y reales. A esta situación también se le atribuye que después de una investigación realizada en la Infraestructura Productiva (IP) ubicada en la UCI con personal calificado en el tema se concluye que actualmente no existe un procedimiento estandarizado que guíe el proceso de planificación de software en los distintos proyectos productivos.

De la situación anterior se identifica el siguiente **problema**: ¿Cómo facilitar el proceso de planificación de los proyectos productivos en la UCI?

El **objeto de estudio** donde se enmarca el problema se centra en los procesos de planificación de software y el **campo de acción** se limita a los procesos de planificación de software en los proyectos productivos en la UCI.

El objetivo de una planificación del software es proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables de recursos humanos, financieros, informacionales y tecnológicos; costos y plazos de ejecución.

Por estas razones se propone como **objetivo general de la investigación** crear un procedimiento para la planificación de proyectos productivos en la UCI.

Del objetivo general de la investigación se derivaron los siguientes **objetivos específicos**:

- Realizar un estudio sobre los procesos de planificación de software.
- Investigar sobre la situación actual en la que se encuentran los proyectos productivos de la UCI.
- Definir la propuesta del procedimiento para la planificación en los proyectos productivos de la UCI.
- Proponer una herramienta para el apoyo a la Gestión de Proyectos.

- Validar la propuesta por el Método de Expertos.

Con el objetivo de planificar, ejecutar, controlar y evaluar la investigación se definieron las siguientes **tareas**:

- Realizar un estudio del estado del arte sobre la Gestión de Proyectos, proceso de planificación, Proceso de Desarrollo de Software, métodos de estimación, modelos CMMI¹.
- Seleccionar de una herramienta libre para dar apoyo a la Gestión de Proyectos.
- Crear un procedimiento estándar para realizar la planificación.
- Analizar los resultados de los estudios realizados para la aplicación del procedimiento para la planificación de proyectos productivos, con el objetivo de realizar su validación.

Esta propuesta garantizaría una mayor rapidez y estandarización en el proceso de planificación, reduciría o eliminaría a gran escala los atrasos en el trabajo, mejoraría la calidad de los productos, una mayor organización del equipo de trabajo y se podría brindar a los clientes fechas de entrega más exactas y cercanas a la realidad, lo que contribuiría a elevar la satisfacción del mismo.

Actualmente, los proyectos productivos que se desarrollan en la UCI no realizan una adecuada planificación del software de ahí la importancia de siguiente investigación, la novedad y que el resultado obtenido sea aplicado en las distintas actividades productivas de la Universidad.

Los **métodos teóricos** que se han usado para dar cumplimiento a las tareas planteadas son el *Analítico-Sintético* centrándose en el análisis de los procesos y métodos de planificación que existen actualmente así como documentos, técnicas y herramientas que posibiliten el desarrollo de dicho proceso, entre otros; permitiendo la extracción de los elementos más importantes de manera que se procese la información y se puedan elaborar conclusiones, se usará además el *Análisis Histórico-Lógico* para el estudio de la evolución y desarrollo de los procesos de planificación de software.

Se utilizarán como **métodos empíricos** la *Entrevista* como técnica de recopilación de información sobre como se lleva a cabo la planificación productiva en la Universidad de las Ciencias Informáticas, y la *Encuesta* como forma de obtener la percepción del encuestado sobre el proceso de planificación en algunos proyectos productivos en la UCI, de manera escrita para garantizar que la investigación este basada en datos reales.

¹ CMMI (Por sus siglas en español): Modelo de Madurez de Capacidades Integrado.

El presente trabajo consta de una Introducción, dos capítulos, las conclusiones generales, recomendaciones, referencias bibliográficas y bibliografía utilizada durante el desarrollo del trabajo, un glosario de términos y por último los anexos que complementan el cuerpo del trabajo.

En el **Capítulo I: Fundamentos Teóricos**, se expone la fundamentación teórica del tema. Incluye una descripción detallada del objeto de estudio, con la finalidad de comprender el proceso de planificación de software. Se presenta el estado del arte de la planificación de software, y además muestra un resumen de las tendencias actuales en cuanto a planificación.

En el **Capítulo II: Procedimiento propuesto**, se describe de forma sencilla y explícita el procedimiento que se propone y que es el objetivo principal en este trabajo, la selección de la herramienta propuesta y un análisis de la situación actual de los proyectos productivos de la UCI.

En el **Capítulo III: Validación del procedimiento**, se presenta la evaluación del proceso teniendo en cuenta los resultados que arrojaron las respuestas del criterio del Grupo de Expertos, para esto se realizan un conjunto de cálculos los cuales se detallan con el objetivo de lograr un mayor entendimiento.

Capítulo I: Fundamentos Teóricos

En este capítulo se estudian los fundamentos teóricos relacionados con la Planificación del Proceso de Desarrollo de un Software. Se hará un estudio de las diferentes herramientas que existen y las que se utilizan para apoyar el proceso de gestión de proyectos productivos en la UCI, diferentes conceptos como Gestión de Proyecto, Proceso de Desarrollo de Software, donde se incluye el estudio de las actividades, los modelos y metodologías que en él se utilizan para la gestión y planificación de proyectos.

1.1 Procesos de Desarrollo de Software

El Proceso de Desarrollo de Software puede ser visto como un conjunto de técnicas y procedimientos que nos permiten conocer los elementos necesarios para definir un proyecto de software.

El desarrollo de software gana más campo en la sociedad actual. A medida que este desarrollo va creciendo ha sido necesario el desarrollo de métodos, procesos y herramientas que apoyen este proceso y que a su vez garanticen la calidad y la factibilidad de los productos.

Como ya es sabido nuestro país intenta insertarse en el mercado del software. En nuestra universidad se lleva a cabo una gran producción de software, pero no se tienen amplios conocimientos sobre el proceso de desarrollo del mismo, por lo que es necesario investigar sobre el tema para asegurar la calidad del software que se produce en la misma.

1.1.1 ¿Qué es el Proceso de Desarrollo de Software?

Existen muchas definiciones, pues son muchos los autores que han estudiado el tema y han emitido su criterio. Según [PRESSMAN, 2005], en su libro: Un Enfoque Práctico, define Proceso de Desarrollo de Software, como: *“Un marco común del proceso, donde se define un pequeño número de actividades del marco de trabajo que son aplicables a todos los proyectos de software, con independencia del tamaño o complejidad, que constituye un número de conjuntos de tareas, donde cada uno es una colección de tareas de ingeniería del software, hitos de proyectos, entregas, y puntos de garantía de calidad”*.

El Proceso de Desarrollo de Software define quién está haciendo qué, cuándo y cómo. Es la definición de un conjunto de actividades que guían el esfuerzo y trabajo de las personas que están implicadas en el proyecto, para lo cual se aplican las 4P de la Ingeniería de Software:

- *Personas*: Principales autores, seres humanos, de un proyecto, (arquitectos, implementadores, ingenieros de pruebas, clientes, usuarios así como otros interesados en el producto).
- *Proyecto*: Es el elemento organizativo a través del cual se gestiona el desarrollo de software y el resultado del mismo es una versión de un producto.
- *Producto*: Son los artefactos que se crean durante el transcurso de la vida el proyecto (código fuente, ejecutables, documentos).
- *Proceso*: Es una definición del conjunto de actividades necesarias para transformar los requisitos de los usuarios en un producto.

El mismo tiene como principal propósito producir de forma eficaz y los más eficiente posible un producto software que reúna los requisitos del cliente.



Figura 1: Proceso de Desarrollo de Software.

1.1.2 Metodologías para el Proceso de Desarrollo del Software

El desarrollo de un software es algo que puede resultar un reto difícil por muy pequeño que este sea, lo cual cambiaría si el desarrollo del mismo se hiciera basándose en una metodología ya que las mismas aparecieron por la necesidad de organizar y guiar el proceso de construcción del software.

Una metodología de desarrollo no es más que un conjunto de métodos que se emplean para el desarrollo de sistemas automatizados. Es algo más que una notación, un proceso, y herramientas. Estas proporcionan guías para estimar costos, manejo del proyecto en las tareas y entregas, medidas y métricas, formas definidas y dirección en las entregas de la construcción, políticas y procedimientos

para garantizar la calidad del software, descripciones de los roles y programas de entrenamiento detallados, técnicas para adaptar el método y técnicas definidas.

No existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Sin embargo, muchas veces no se toma en cuenta el utilizar una metodología adecuada, sobre todo cuando se trata de proyectos pequeños, pero cuando los proyectos que se van a desarrollar son de mayor envergadura, ahí toma sentido el basarse en una metodología de desarrollo, y se empieza a buscar cual será la más apropiada para el caso.

Existen dos grandes grupos de metodologías, que se pueden utilizar para el desarrollo de un producto software, las Metodologías Tradicionales entre las que podemos encontrar a *RUP (Rational Unified Process o Proceso Unificado de Desarrollo)*, *MSF (Microsoft Solution Framework)*, así como otras y las *Metodologías Ágiles*, teniendo dentro de ellas, *XP (Extreme Programming)*, *SCRUM*.

1.1.2.1 Metodologías Tradicionales (“no ágiles”)

Las metodologías tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas que se usarán. Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo, presentan cierta resistencia a los cambios, desarrollan un proceso mucho más controlado, con numerosas políticas/normas, exigen el desarrollo de más artefactos y roles y la arquitectura del software es un aspecto esencial y la misma se expresa mediante modelos. Están guiadas por una fuerte planificación durante todo el proceso de desarrollo.

Rational Unified Process (RUP)

La metodología RUP es más apropiada para proyectos grandes, dado que requiere un equipo de trabajo capaz de administrar un proceso complejo en varias etapas. Es un proceso de ingeniería de software para producir software de calidad, flexible, y en plazos y presupuestos predecibles, incorpora las mejores prácticas de desarrollo de software validadas comercialmente, proporciona un desarrollo incremental, guiado por casos de uso y centrado en la arquitectura, ayuda a alcanzar el Nivel de Madurez 2 del CMM². Divide en 4 fases el desarrollo del software:

² CMM (por sus siglas en español): Modelo de Madurez de Capacidades.

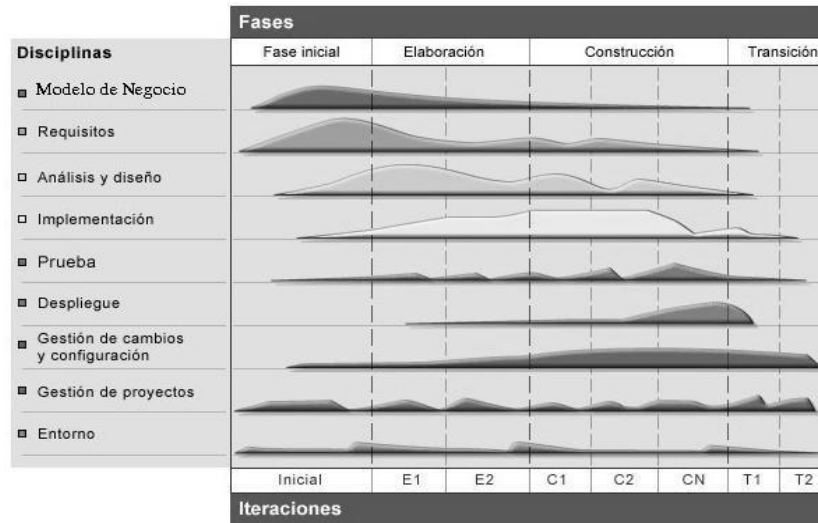


Figura 2: Metodología Rational Unified Process.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. También proporciona mitigación de riesgos mediante la planificación acertada y controlada de las iteraciones y la entrega de un producto de calidad bajo las especificaciones del cliente.

Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, trabajadores y los casos de usos con sus responsables, esto se puede observar en la figura 3, es por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.



Figura 3: Elementos de RUP.

Es posible la mezcla de RUP con otras metodologías de administración de proyectos como MSF para definir aspectos como, contratación, roles y responsabilidades, definición de grupos de trabajo y presupuesto.

Microsoft Solution Framework (MSF)

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.



Figura 4: Fases de la Metodología Microsoft Solution Framework.

Tiene las siguientes características:

- *Adaptable*: es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- *Escalable*: puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas a más.
- *Flexible*: es utilizada en el ambiente de desarrollo de cualquier cliente.
- *Tecnología Agnóstica*: porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

Consta de cinco fases distintas:

- Visión y alcances.
- Planificación.
- Desarrollo.
- Estabilización.
- Implementación.

1.1.2.2 Metodologías Ágiles

Las Metodologías Ágiles son entendidas como el conjunto de metodologías para el desarrollo de software que surgen en la década de los 90.

La filosofía principal de las metodologías ágiles de desarrollo es centrarse, principalmente en dimensiones como el factor humano o el producto software. También proporcionan un mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas. Las mismas están revolucionando la manera de producir software en el mundo actual especialmente para proyectos pequeños, con plazos reducidos y nuevas tecnologías.

Las metodologías ágiles ofrecen una solución para una gran cantidad de proyectos pequeños y con requisitos muy cambiantes, están basadas en heurísticas provenientes de prácticas de producción de código, preparada para cambios durante el proyecto, presenta un proceso menos controlado, con pocos principios, el cliente es parte del equipo de desarrollo, sus grupos deben ser pequeños (<10 integrantes) y deben trabajar en el mismo sitio, exige pocos artefacto y roles y hace menos énfasis en la arquitectura del software. Una de las cualidades más destacables en una metodología ágil es su sencillez, tanto en su aprendizaje como en su aplicación, esto es muy importante ya que reduce los costos de implantación en un equipo de desarrollo.

Sin embargo, hay que tener presente una serie de inconvenientes y restricciones para su aplicación, tales como: están dirigidas a equipos pequeños o medianos, el entorno físico debe ser un ambiente que permita la comunicación y colaboración entre todos los miembros del equipo durante todo el tiempo, el uso de tecnologías que no tengan un ciclo rápido de realimentación o que no soporten fácilmente el cambio, etc.

Las metodologías ágiles presentan diversas ventajas:

- Capacidad de respuesta a cambios de requisitos a lo largo del desarrollo.
- Entrega continua y en plazos breves de software funcional.
- Trabajo conjunto entre el cliente y el equipo de desarrollo.
- Importancia de la simplicidad, eliminado el trabajo innecesario.
- Atención continua a la excelencia técnica y al buen diseño.
- Mejora continua de los procesos y el equipo de desarrollo.

En el Manifiesto Ágil [CANÓS, 2006] se plantean los siguientes puntos:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.
- Desarrollar software que funcione más que conseguir una buena documentación.

- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

Extreme Programming (XP)

Es una de las metodologías de desarrollo de software más exitosas en la actualidad, utilizadas para proyectos de corto plazo. Esta metodología ágil está centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo así el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes. Es una metodología adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues considera esto como un principal requisito para llegar al éxito del proyecto.

El aspecto fundamental de esta metodología está dado por la comunicación entre los usuarios y desarrolladores, la simplicidad al desarrollar y codificar los módulos del sistema y la retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

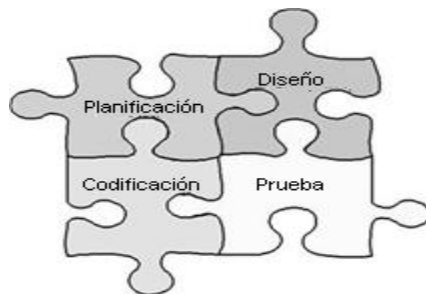


Figura 5: Metodología Extreme Programming.

XP propone:

- Empezar en pequeño y añadir funcionalidades con retroalimentación continua.
- El manejo del cambio, se convierte en parte sustantiva del proceso.
- El costo del cambio, no depende de la fase o etapa.
- No introduce funcionalidades antes que sean necesarias.
- El cliente o el usuario se convierten en miembros del equipo.

Características fundamentales de XP:

- *Pruebas Unitarias*: se basa en las pruebas realizadas a los principales procesos, de manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir. De esta manera, es como si nos adelantáramos a obtener los posibles errores.
- *Re-fabricación*: se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- *Programación en pares*: una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto, en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

SCRUM

Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. El desarrollo se realiza en forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza de software ejecutable que incorpora nueva funcionalidad y se le muestra al cliente, las iteraciones en general tienen una duración entre 2 y 4 semanas. Otra característica son la realización de reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.

Es una forma muy exitosa de gestionar proyectos de software. No es una metodología de análisis, ni de diseño, como podría ser RUP, es una metodología de gestión del trabajo. Es un proceso ágil y liviano para administrar y controlar el desarrollo de un software. SCRUM se utiliza como marco para otras prácticas de ingeniería de software como RUP o XP.

Es muy fácil de explicar y de entender, lo que ayuda mucho a su implantación. Se focaliza en priorizar el trabajo en función del valor que tenga para el negocio.

En SCRUM, el equipo se focaliza en una única cosa: construir software de calidad. Por el otro lado, la gestión de un proyecto SCRUM se focaliza en definir cuáles son las características que debe tener el

producto a construir (qué construir, qué no y en qué orden) y en remover cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo. Se busca que los equipos sean lo más efectivos y productivos que sea posible.

Por otra parte SCRUM puede ser aplicado a distintos modelos de calidad (como podría ser CMMI) puesto que estos te dicen qué tienes que hacer, es decir, te dicen que tienes que gestionar el proyecto, pero no te dicen cómo. Ahí es donde entra SCRUM como modelo de gestión del proyecto.

Adaptive Software Development (ASD)

Se enfoca en la naturaleza adaptable de las nuevas metodologías, con un énfasis particular en aplicar las ideas que se originaron en el mundo de los sistemas complejos adaptables (normalmente conocidos como teoría del caos). No proporciona el tipo de prácticas detalladas como lo hace XP, pero proporciona la base fundamental de por qué el desarrollo adaptable es importante y las consecuencias a los más profundos niveles de la organización y la gerencia.

En ASD hay tres fases solapadas, no lineales:

- Especulación.
- Colaboración.
- Aprendizaje.

James A. Highsmith el creador de esta metodología ve la planificación como una paradoja en un ambiente adaptable, ya que los resultados son naturalmente imprevisibles. En la planificación tradicional, las desviaciones del plan son errores que deben corregirse. En un ambiente adaptable, en cambio, las desviaciones nos guían hacia la solución correcta.

En este ambiente imprevisible se necesita que las personas colaboren de la mejor manera para tratar con la incertidumbre. En un ambiente adaptable, aprender desafía a todos: desarrolladores y clientes, a examinar sus presunciones y usar los resultados de cada ciclo de desarrollo para adaptar el siguiente. El aprendizaje como tal es un rasgo continuo e importante, que asume que los planes y los diseños deben cambiar conforme avanza el desarrollo.

El beneficio atropellado, poderoso, indivisible y predominante del Ciclo de Vida de Desarrollo Adaptable es que obliga a confrontar los modelos mentales que están en la raíz del autoengaño de las

personas. Obliga a las personas a estimar con realismo su habilidad. Se enfoca directamente en fomentar las partes difíciles del desarrollo adaptable, en particular cómo fomentar la colaboración y el aprendizaje dentro del proyecto.

1.2 Gestión de proyecto

La gestión de proyectos de software es una rama especializada de la Ingeniería de Software, posee además de los métodos generales de la ingeniería, metodologías propias.

Cuando hablamos de la gestión de proyecto como una rama se hace necesario plantear que:

- Emplea metodologías bien definidas.
- Realiza medidas repetibles y confiables.
- Estima costos y tiempos.
- Da elementos para la gestión de los proyectos.
- Replantea resultados para ajustar la información disponible.

Todo proceso de gestión de software debe reunir las siguientes características:

- Actividades planificadas, ejecutadas y supervisadas.
- Disponibilidad limitada de recursos.
- Limitado en el tiempo.
- Con un resultado único.

La gestión de proyectos de software persigue la misma finalidad que todas las gestiones de proyectos de ingeniería:

- Estimar que sucederá con un proyecto nuevo.
- Analizar qué sucedió con un proyecto ya finalizado.

En todos los casos se tratará de dar respuestas cuantitativas a preguntas precisas tales como:

- ¿Cuál será el plazo de entrega?
- ¿Cuántas personas necesito?

¿Cuánto costará el proyecto?

La Gestión de Proyectos tiene como finalidad principal la planificación, el seguimiento y control de las actividades de los recursos humanos y materiales que intervienen en el desarrollo de un sistema software. La planificación es la etapa donde se debe hacer las asignaciones de los recursos disponibles, para lo que es necesario estimar costos y plazos para el proyecto a realizar.



Figura 6: Ubicación de la planificación dentro de la Gestión de Proyecto.

1.2.1 Actividades de la Gestión de Proyectos

Generalmente los gestores de proyectos son responsables de algunas o todas las siguientes actividades:

Redacción de la propuesta. La propuesta describe los objetivos del proyecto y cómo se llevaría a cabo. Incluye estimaciones de costo y tiempo y justifica por qué el contrato del proyecto se debe dar a una organización o equipo en particular.

Planificación y calendarización del proyecto. Se refiere a la identificación de actividades, hitos y entregas del proyecto.

Estimación de costos del proyecto. Es una actividad relacionada con la estimación de los recursos requeridos para llevar a cabo el plan del proyecto.

Supervisión y revisión del proyecto. La supervisión es una actividad continua. El gestor debe conocer el progreso del proyecto con los costos actuales y los planificados. También, es normal tener varias revisiones formales de su gestión. Se hace una revisión completa del progreso y de los desarrollos técnicos del proyecto, teniendo en cuenta el estado del proyecto. El resultado puede dar lugar a una cancelación.

Selección y evaluación del personal. Los gestores, generalmente, seleccionan a las personas que trabajarán en su proyecto. O establecen un equipo ideal mínimo para el proyecto.

Redacción y presentación de informes. Los gestores son los responsables de informar a los clientes y contratistas sobre el proyecto. Deben redactar documentos concisos y coherentes que resuman la información crítica de los informes detallados del proyecto.

1.3 La planificación dentro de la Gestión de Proyecto

La planificación de manera general consiste en evaluar la situación entorno y la realidad en que partimos de forma sistemática, dentro de ello encontramos las variables socio-ambientales, recursos, tiempos, etcétera, para escoger entre las diferentes posibilidades de acción y trazar el camino a seguir para lograr los objetivos.

Por tanto la planificación es un proceso que ha de reunir las siguientes características:

- *Dinámico:* que no termina con el establecimiento de un plan, sino que supone un reajuste constante entre medios y fines.
- *Instrumental:* un medio dirigido al logro de los objetivos.
- *Práctico:* encaminado básicamente a la acción.
- *Integral:* donde se relacionan todos los elementos de una manera sistemática e interdependiente.
- *Anticipador:* Intenta predecir y pronosticar el futuro para acomodar la acción.
- *Facilitador:* que prepara un conjunto de decisiones que deben ser aprobadas y ejecutadas por los sectores implicados.

Comprender el ámbito del trabajo a realizar, los riesgos en los que se puede incurrir, las tareas que han de llevarse a cabo, las etapas a recorrer, los recursos y el costo del proyecto, así como el plan a seguir, previéndose las posibles situaciones de encontrar escenarios con el mejor y peor de los casos, representan acciones esenciales para conseguir un alcance exitoso en un proyecto software.

Durante la planificación se logran discernir dos grandes fases:

La *primera fase* llamada “fase de estudio” es necesaria para estudiar y establecer la viabilidad de un proyecto, ya sea interno o externo a la organización. Durante esta fase corresponde realizar estudios

técnicos, de mercado, financieros, de rentabilidad, así como una estimación de los recursos necesarios y los costos generados, siendo el elemento fundamental en el que el cliente se apoya (que puede ser la propia organización en el caso de proyectos internos) para decidir sobre la realización o no del proyecto.

La *segunda fase*, llamada “fase de ejecución”, tiene lugar una vez se ha decidido ejecutar el proyecto. Este es el momento de realizar una planificación detallada punto por punto. Uno de los errores más importantes y graves en gestión de proyectos es querer arrancar con excesiva premura la obra, sin haber prestado la atención debida a una serie de tareas previas de preparación, organización y planificación que son imprescindibles para garantizar la calidad de la gestión y el éxito posterior.

1.3.1 Importancia de realizar una buena planificación

La planificación es la primera función del proceso administrativo, por tanto, realizarla de manera adecuada conlleva a tener una buena organización, dirección y control de la empresa lo cual se traduce en una administración cien por ciento efectiva. Prepara a la empresa para hacer frente a las contingencias que se presenten, con las mayores garantías de éxito.

Reduce al mínimo los riesgos y aprovecha al máximo las oportunidades.

La planificación permite decidir cuánto tiempo se necesita para hacer un trabajo y controla el progreso mientras se está haciendo el mismo. Para la realización de un software se necesita planificar el trabajo del personal, lo cual permite cumplir con las fechas de realización de las tareas asociadas al equipo de trabajo. También permite llevar el control de lo que se está haciendo, y ver si se ha avanzado o no con el software.

1.3.2 Objetivos de la planificación

La planificación como proceso rector de la gestión de proyectos tiene como objetivo principal conocer y definir el entorno del proceso de desarrollo y del producto a desarrollar, así como los componentes que el sistema en si exige para un exitoso ciclo de vida. El hacer estimaciones de recursos, costos y planificación temporal se incluye dentro de este proceso. Estas actividades se hacen en un tiempo limitado al comienzo del proyecto y luego deberán ser actualizadas y revisadas durante el progreso

del mismo. Además se deberán definir los escenarios del *mejor caso* y del *peor caso* de forma que los resultados del proyecto puedan limitarse.

Más específicamente estos son los objetivos que se buscan durante la planificación [Teruel, 2001]:

- *¿Qué se desarrolla?* Esto no es más que comprender a grandes rasgos el sistema a desarrollar, pues este no solo incluye el software como resultado final sino también otros componentes que de una forma u otra en algún momento durante su ciclo de vida interactuarán con él, tales como otros software, bases de datos, hardware, recursos humanos y recursos organizacionales.
- *¿Para qué se desarrolla?* Esto incluye las razones que llevan al cliente a costear este desarrollo, lo que llevara a aclarar las metas de negocio del sistema.
- *¿Para quién se desarrolla?* El no conocer a veces quién es realmente el cliente puede traer en ciertas ocasiones una pérdida de la responsabilidad y el comprometimiento con el éxito del producto a desarrollar. Es por esto que es importante identificar al cliente, distinguiendo claramente entre los usuarios y los patrocinantes del sistema. Estos participantes deben quedar dispuestos a involucrarse en el desarrollo del sistema y sentirse comprometidos a él.
- *¿Qué permitirá hacer el sistema? ¿Qué condiciones afectan el desarrollo?* Elicitar los requerimientos del software y de su proceso de desarrollo, es decir, se busca que el cliente colabore con el desarrollador para que ambos puedan aprehender clara y precisamente los requerimientos que aplican.
- *¿Qué problemas amenazan al desarrollo? ¿Qué oportunidades deben aprovecharse?* Identificar los riesgos y oportunidades asociados al desarrollo, como por ejemplo, las fuentes de volatilidad de los requerimientos.

1.3.3 Actividades asociadas a la planificación de proyectos

1.3.3.1 Ámbito del Software

El ámbito del software es la primera actividad que se lleva a cabo durante la planificación de proyectos de software [Pressman, 1998]. Esta es una etapa de evaluación de la función y el rendimiento que se asignaron al Software para establecer un ámbito de proyecto que no sea ambiguo, e incomprensible. Este ámbito se centra principalmente en el contexto, los objetivos de información y las funciones y el rendimiento.

De manera general según [Nova, 2005] durante en el ámbito del software se describe las interfaces, el rendimiento, la función, las restricciones y la fiabilidad, se evalúan las funciones del ámbito y en algunos casos se refinan para dar más detalles antes del comienzo de la estimación. Las restricciones de rendimiento abarcan los requisitos de tiempo de respuesta y procesamiento, identifican los límites del software originados por el hardware externo, por la memoria disponible y por otros sistemas existentes.

1.3.3.2 Recursos

La estimación de los recursos requeridos para acometer el esfuerzo de desarrollo de Software es sin dudas una tarea muy importante, esto lo podemos ver como una pirámide donde las Herramientas (hardware y software), son la base que proporciona la infraestructura de soporte al esfuerzo de desarrollo, en segundo nivel de la pirámide se encuentran los Componentes reutilizables. Y en la parte mas alta se encuentra el recurso primario, las personas (el recurso humano).



Figura 7: Pirámide de recursos en la estimación.

Recursos humanos

La cantidad de personas requeridas para el desarrollo de un proyecto de software solo puede ser determinado después de hacer una estimación del esfuerzo de desarrollo (por ejemplo personas mes o personas años), y seleccionar la posición dentro de la organización y la especialidad que desempeñará cada profesional, es decir el rol.

Recursos o componentes de software reutilizables

Cualquier estudio sobre recursos de software estaría incompleto sin estudiar la reutilización. Esto es la creación y la reutilización de bloques de construcción de software. Tales bloques se deben establecer

en catálogos para una consulta más factible, estandarizarse para una fácil aplicación y validarse además para una integración.

Recursos de entorno

El entorno es donde se apoya el proyecto de software, llamado a menudo entorno de Ingeniería de Software, incorpora hardware y software.

El hardware proporciona una plataforma con las herramientas (software) requeridas para generar los productos, que son el resultado de la buena práctica de la Ingeniería del Software.

En esencia, el intento a la hora de planificar para predecir cuánto dinero, esfuerzo, recursos y tiempo; llevará realizar un producto software determinado, implica estimar constantemente, siendo base también en futuras planificaciones.

1.3.4 Artefactos del proceso de planificación

Durante el proceso de planificación se desarrollan algunos artefactos que no es posible dejar de pasar por alto, pues son una prueba del rigor con que tiene que ser llevado a cabo dicho proceso. Los artefactos según [Teruel, 2001] son los siguientes:

- *Panorama general del sistema:* Una breve descripción del sistema que se desea desarrollar.
- *Las metas del sistema* (qué va a lograr el sistema para el negocio):
 - Mejorar la calidad y variedad de servicios para captar y/o retener usuarios (incrementar participación de mercado).
 - Mejorar imagen del negocio.
 - Reducir costos y/o aumentar ganancias.
 - Mejorar procesos del negocio.
- *Descripción de clientes.*
- *Glosario de términos del dominio del sistema.*
- *Las funciones (requerimientos funcionales) que desempeñará el software.*
 - *Los atributos (requerimientos no funcionales) del software.*

- Tiempo de respuesta
- Filosofía de la interfaz
- Tolerancia a fallas
- Rendimiento
- Seguridad
- Plataforma tecnológica (restricciones)
- *Atributos asociados a cada función.*
- *Requerimientos sobre el proceso de desarrollo.*
- *Lista de los riesgos (y oportunidades) más serios.*

Estos artefactos permiten que:

- El cliente sienta que el desarrollador ha entendido, en forma preliminar, sus necesidades.
- El desarrollador entienda las necesidades, riesgos y el entorno del desarrollo.
- El cliente y el desarrollador evalúen posteriormente si el sistema desarrollado cumple con sus metas.
- El desarrollador desarrolle criterios de evaluación para sus diseños con alta probabilidad que el cliente se las apruebe.
- El cliente y el desarrollador se comuniquen efectiva y eficientemente.
- El desarrollador tenga una base para empezar a analizar si es factible desarrollar el producto sujeto a las condiciones exigidas.

1.3.5 Los riesgos y la planificación

Al hilo de lo señalado al principio, la planificación de los proyectos debe estar afectada de un notable grado de *agilidad y dinamismo*: no es razonable planificar un proyecto y pensar que esa planificación es ya definitiva e inmutable. En casi todos los casos, la realidad no coincide exactamente con lo previsto, por lo que es necesario ir haciendo ajustes periódicos. La planificación es una tarea dentro de la gestión de proyecto que posibilita la toma de decisiones, no para imaginar en un primer momento una evolución que posteriormente el tiempo se encargará de demostrar que estaba equivocada.

Durante la planificación se debe tener en cuenta los riesgos que podrán o no surgir durante el desarrollo del proyecto. Un riesgo puede ser cualquier obstáculo o suceso que se interponen en el

camino hacia el éxito del cumplimiento de un objetivo que afecte negativamente el desarrollo del proyecto y el mismo es desconocido o incierto.

En la figura 8, se muestran algunas de las actividades inherentes a los riesgos que deberán tenerse en cuenta para realizar una adecuada identificación de los mismos.



Figura 8: Actividades asociadas con la gestión de riesgos

Identificar los riesgos:

- Riesgos relativos a la planificación, control y seguimiento del proyecto.

Ejemplo: Planificación demasiado optimista, Se pierde mucho tiempo en reuniones.

- Riesgos relativos a requerimientos y relación con el cliente.

Ejemplo: Requerimientos demasiado ambiciosos para el tiempo disponible, Cambios en los requerimientos

- Riesgos relativos al diseño e implementación.

Ejemplo: Cambios en los requerimientos, el diseño es demasiado sencillo y hay que volver a hacerlo.

- Riesgos relativos al equipo.

Ejemplo: Los miembros del equipo no logran trabajar bien juntos, El equipo no confía en uno de sus miembros.

Controlar los riesgos:

- Reaccionar ante problemas: Monitorear el desarrollo para poder detectar rápidamente un riesgo que se presente. En cuanto se detecte se empieza a planificar cómo tratarlo.
- Mitigar el riesgo: Planificar con antelación (p. ej. planes de contingencia) cómo tratar con riesgos si se presentan, monitorear para detectarlos rápidamente.
- Prevenir el riesgo: Determinar con antelación los posibles orígenes de riesgos y preparar y llevar a cabo planes para evitar que se presenten los riesgos.

1.4 La estimación dentro de la Gestión de Proyectos

La estimación es la primera actividad dentro del proceso de Gestión de Proyectos, ya que el mismo comienza con un conjunto de actividades que, globalmente se denominan planificación del proyecto. La primera de estas actividades es la estimación de costes y tiempos, por lo que se entiende por estimación al conjunto de técnicas que permiten dar un valor aproximado. La estimación proporciona valores aproximados de costos, tiempos y esfuerzos que se necesitarán para el desarrollo del producto a construir. Esa aproximación, requiere experiencia, acceder a una buena información histórica y el coraje de confiar en predicciones (medidas) cuantitativas cuando todo lo que existe son datos cualitativos, sin embargo, contar con dicha información en etapas tempranas de desarrollo, permite tomar decisiones importantes antes llevarlo a cabo el desarrollo del producto.

Pese a que la planificación y la estimación son actividades muy relacionadas dentro de la Gestión de Proyectos, se diferencian en cuanto su alcance y propósito, la estimación normalmente está orientada al proyecto en su conjunto, mientras que la planificación está dirigida a los individuos. La estimación, de forma general, se realizará al principio del proyecto, precisamente para saber cuánta gente se necesita o cuánto durará una etapa en el proceso y la planificación es la etapa donde se asigna exactamente quién hace qué y en cuánto tiempo. Las etapas de estimación y planificación se harán de forma continua a todo lo largo del proceso de desarrollo de software al inicio de cada etapa. La estimación requiere de experiencia, buena información histórica y coraje de confiar en las métricas y la experiencia.

Como la estimación es la base de la planificación, hay que prestarle especial atención. Como diría Sun Tzu en El arte de la guerra, siglo IV a.c.: *“... si las estimaciones efectuadas antes de las hostilidades indican victoria, es porque los cálculos muestran que la fortaleza propia es superior a la del enemigo; si*

indican derrota, es porque los cálculos muestran que es inferior. Con muchos cálculos se puede ganar; con pocos, no. ¡Cuántas menos posibilidades de victoria tiene quien no hace ninguno!"

Según [Sommerville, 2002] "...la estimación y calendarización (planificación) del proyecto se llevan a cabo de forma conjunta".

1.4.1 Factores que influyen en la estimación

Una vez que se ha decidido realizar la estimación se deben tener en cuenta cuatro factores que influyen significativamente en las estimaciones:

- La complejidad del proyecto: La complejidad es relativa a la experiencia en proyectos anteriores. Existen medidas sobre la complejidad de proyectos basadas en el diseño y código (métricas técnicas). En la fase de estimación no son aplicables porque no hay ni diseño ni código. Por eso hay que utilizar medidas más subjetivas.
- El tamaño del proyecto: En los proyectos grandes, crece la interdependencia entre los componentes del proyecto. La interdependencia conlleva a la descomposición del producto en elementos muy grandes lo que dificulta el proceso de estimación.
- El grado de incertidumbre estructural: La estructura no es más que el grado en que los requisitos se han definido, la facilidad con la que se pueden compartimentalizar funciones y la naturaleza jerárquica de la información a procesar. Por tanto una incertidumbre estructural baja proporciona una mejor descomposición del producto y por ende una mejor estimación.
- Disponibilidad de información histórica: La existencia de métricas del software de proyectos anteriores facilita la estimación.

En cualquier caso, la estabilidad de los requisitos por parte del cliente es fundamental para la estimación. Si los requisitos cambian, la estimación no vale, por eso son buenos los modelos de procesos evolutivos.

Existen algunos pasos generales que se deben seguir para poder realizar estimación de un proyecto de software:

- 1) Obtener toda la información posible acerca del proyecto.

- 2) Seleccionar las métricas a estimar para el proyecto y los métodos mediante los que realizaremos esa estimación.
- 3) Recolectar los datos necesarios para analizar las estimaciones.
- 4) Se realizan las estimaciones para evaluar si están siendo cumplidas o no. Los resultados son retroalimentados tanto para el desarrollo del sistema como a la realización de futuras estimaciones.
- 5) Evaluación de las métricas usadas para hacer las estimaciones aprendiendo de la experiencia.
- 6) Guardar la experiencia adquirida a fin de que sea usada para el futuro.

1.4.2 Técnicas de estimación

Se han desarrollado varias técnicas de estimación para el desarrollo de software, aunque cada una tiene sus puntos fuertes y sus puntos débiles, todas tienen en común los siguientes atributos.

- Se han de establecer de antemano el ámbito del proyecto.
- Como bases para la realización de estimaciones se usan métricas del software de proyectos pasados.
- El proyecto se desglosa en partes más pequeñas que se estiman individualmente.

Algunas de las técnicas ya existentes son:

- *Retrasar* la estimación lo máximo posible. Cuanto más la retrasemos, más precisa será.
- *Estimación por analogía*. Utilizar el coste de proyectos similares ya terminados.
- *Ley de Parkinson*. *El trabajo se extiende para rellenar el tiempo disponible.*
- *Precio para ganar*. *El coste se estima en todo el dinero que el cliente puede gastar en el proyecto.*
- *Técnicas de descomposición*. *Estiman el coste descomponiendo el producto y/o el proceso.*
- *Modelos empíricos*. *Modelos de regresión que relacionan esfuerzo con tamaño o funcionalidad*

1.4.3 Métodos de estimación que existen

Algunas características que en todo modelo de estimación deben existir son las siguientes:

- Poder adaptarse a la productividad de la organización.
- Considerar el costo de comunicación entre personas.

- Incorporar guías útiles para estimar aquellos parámetros que son subjetivos o no se deducen en forma explícita a partir del modelo.
- Usable.
- Constar de etapas simples de entender y definidas en forma precisa.
- Objetivo.
- Costo efectivo.
- Proveer medios para adaptarse a cambios en el ambiente de desarrollo.
- Permitir una estimación temprana.

Factores para del éxito de un método:

- La estimación inicial no se desvía más del 30% de la real.
- El método permite refinar la estimación durante el ciclo de vida.
- Facilidad de uso.
- Las reglas son entendidas por todos.
- El método es soportado por herramientas y está documentado.

Modelo COCOMO³

Es el modelo de estimación de software más utilizado en estos tiempos. Permite la estimación del esfuerzo, costo y tiempo que requiere un proyecto software partir de una medida del tamaño del mismo la cual es indirecta del tamaño del código fuente.

Este modelo es una jerarquía de modelos de estimación que aborda:

- *Modelo de composición de la aplicación*, al inicio del ciclo de vida, cuando se realiza la recolección de requerimientos.
- *Modelo de etapa de diseño temprano*, se utiliza cuando se realiza el diseño de la aplicación, en la evaluación de la tecnología.
- *Modelo de etapa posterior*, cuando ya se encuentra en la fase de construcción del software.

³ COCOMO (por sus siglas en inglés): Constructive Cost Model.



Figura 9: Estimación de proyectos por método COCOMO

COCOMO II

Es una modificación del COCOMO, pues a medida que fue pasando el tiempo la complejidad de los software fue aumentando, esta nueva versión permite estimar el coste, esfuerzo y tiempo cuando se planifica una nueva actividad de desarrollo software a partir de los puntos de función sin ajustar. Es objetivo, repetible, eficiente, pero a la vez también presenta entradas subjetivas, no adaptable a circunstancias excepcionales. Consiste básicamente en la aplicación de ecuaciones matemáticas sobre los Puntos de Función sin ajustar o la cantidad de líneas de código.

Puntos de Casos de Usos

Se trata de un método de estimación del tiempo de desarrollo de un proyecto mediante la asignación de "pesos" a un cierto número de factores que lo afectan, para finalmente, contabilizar el tiempo total estimado para el proyecto a partir de esos factores.

El método utiliza los actores y casos de uso identificados para calcular el esfuerzo que costará desarrollarlos, por lo que requiere la existencia de un modelo de casos de uso, y se podrá comenzar a aplicar, una vez que se tenga algún entendimiento del dominio del problema.

La estimación por Puntos de Caso de Uso resulta muy efectiva para estimar el esfuerzo requerido en el desarrollo de los primeros Casos de Uso de un sistema, si se sigue una aproximación iterativa como el Proceso Unificado de Rational. En éste tipo de aproximación, los primeros Casos de Uso a

desarrollar son los que ejercitan la mayor parte de la arquitectura del software y los que a su vez ayudan a mitigar los riesgos más significativos.

Puntos de función

Es un método para medir el tamaño del software. Pretende medir la funcionalidad entregada al usuario independientemente de la tecnología utilizada para la construcción y explotación del software, y también ser útil en cualquiera de las fases de vida del software, desde el diseño inicial hasta la explotación y mantenimiento.

La aplicación de la técnica de los puntos función comprende los siguientes pasos:

- Definición de los límites del sistema.
- Definición de parámetros externos.
- Valoración de la complejidad.
- Análisis de las características generales del sistema.

Los objetivos de los puntos de función son:

- Medir lo que el usuario pide y lo que el usuario recibe.
- Medir independientemente la tecnología utilizada en la implantación del sistema.
- Proporcionar una métrica de tamaño que dé soporte al análisis de la calidad y la productividad.
- Proporcionar un medio para la estimación del software.
- Proporcionar un factor de normalización para la comparación de distintos software.

El análisis de los puntos de función se desarrolla considerando cinco parámetros básicos externos del sistema:

1. Entrada (EI, del inglés External Input).
2. Salida (EO, del inglés External Output).
3. Consultas (EQ, del inglés External Query).
4. Ficheros Lógicos Internos (ILF, del inglés Internal Logic File).
5. Ficheros Lógicos Externos (EIF, del inglés External Interface File).

Con estos parámetros, se determinan los puntos de función sin ajustar (PFsA). A este valor, se le aplica un factor de ajuste obtenido en base a unas valoraciones subjetivas sobre la aplicación y su entorno; es decir, las características generales del sistema.

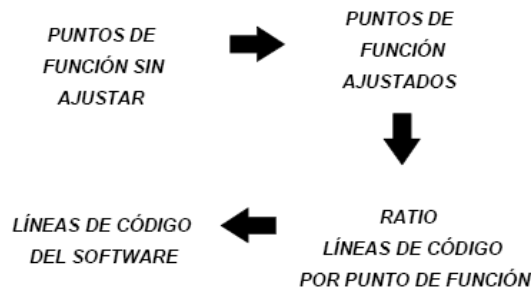


Figura 10: Estimación de Proyectos por el método de Puntos de Función.

Juicio del experto

Las opiniones de los expertos se basa principalmente en juicios emitidos por uno o varios expertos avalados por su experiencia en entornos similares y apoyados en datos objetivos obtenidos de proyectos anteriores y almacenados aunque no se incluyen dentro del marco de trabajo para seleccionar métodos de estimación, ya que estos métodos no se pueden caracterizar fácilmente.

Tiene como ventaja que indica que las estimaciones parciales son neutralizadas y se presenta una estimación global.

Con la utilización de este método las estimaciones suministradas por este grupo de expertos difícilmente pueden ser obviadas gracias a la trascendencia que la organización otorga a este proceso, al proporcionar costosos recursos a esta tarea. Otra desventaja de este método es el alto costo en tiempo y recursos humanos necesarios para su implantación, así como la subordinación al nivel de experiencia y conocimientos en el entorno que puedan aportar los técnicos.

1.5 Modelos de calidad para la planificación

En la industria del software cada día se hace más necesario elevar la calidad de los productos, para esto existen modelos, normas y estándares que se definen a nivel mundial.

La calidad del software es “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 el software desarrollado profesionalmente”. [Pressman 1998].

La calidad del software es un aspecto que debe contemplarse a lo largo de todo el ciclo de vida del proyecto y al mismo tiempo debe correr paralela desde la planificación del producto hasta la fase de producción del mismo.

1.5.1 Modelos de Madurez de Capacidades (CMM)

“El CMM es un modelo de calidad del software que clasifica las empresas en niveles de madurez. Estos niveles sirven para conocer la madurez de los procesos que se realizan para producir software” [Gracia, 2005].

CMMI (Modelo de Capacidad y Madurez Integrado) es un modelo que constituye un marco de referencia de la capacidad de desarrollo de software de diferentes empresas, siendo una base para evaluar la madurez de las mismas y una guía para llevar a cabo una estrategia de mejoras continuas. Este modelo surgió a partir de CMM (Modelo de Capacidad y Madurez) como resultado de la integración de varios modelos que definían la madurez en diferentes disciplinas, pero que dada la variedad constituían un problema para las empresas por lo que fue necesario agruparlos en el CMMI.

CMMI sirve como guía para mejorar los procesos de desarrollo de software en una empresa y da un punto de referencia para realizar una evaluación de los procesos actuales.

Este modelo está organizado en cinco niveles, ayudando a definir la prioridad de las acciones para aumentar la madurez y mejorar los procesos de ingeniería de software y proporciona una forma de ir mejorando progresivamente.

El nivel 2 de CMMI es donde se definen todos los procesos de gestión y control del desarrollo del software. Las áreas de procesos de este nivel son:

- Gestión de Requisitos.
- *Planificación de proyectos.*
- Monitorización y Control de proyectos.

- Medición y Análisis.
- Aseguramiento de la calidad.
- Gestión de la configuración.

En la planificación de proyecto, CMMI plantea lo siguiente: el objetivo de la planificación de proyectos es establecer y mantener planes que define las actividades del proyecto, lo que incluye establecer estimaciones, desarrollar un Plan de Proyecto y obtener compromisos con el plan.

1.6 Herramientas de apoyo para la Gestión de Proyectos

OpenProj

Según [Fernández, 2008] OpenProj es multiplataforma, está desarrollado en Java y funciona bajo Windows, Linux, Unix y Mac OS. Es una excelente alternativa a Microsoft Project. De hecho, una de las ventajas de OpenProj es que permite exportar e importar archivos .mpp y .mpx (extensiones de Microsoft Project). Además es completamente gratuito y está traducido a 8 idiomas incluyendo el español.

Presenta funcionalidades básicas y avanzadas de gestión de planificaciones, como pueden ser:

- Gestión de calendarios de trabajo
- Existencia de diferentes vistas del proyecto (Gantt, diagramas de red, de recursos, histogramas, etcétera).
- Gestión de tareas, con niveles de jerarquía y todas las posibilidades de dependencia necesarias.
- Gestión de recursos (tantos humanos como materiales) asignados a un proyecto. Gestión de costes, aunque de forma básica.
- Gestión de líneas base para la replanificación controlada del proyecto. Funciones de seguimiento de proyectos, ya sea por dedicación de recursos o simplemente por avance por porcentaje (introducción manual).

Es posible que en futuras versiones vaya mejorando en algunos aspectos, como puede ser una mayor usabilidad y el añadido de algunas características adicionales (columnas personalizadas, funciones de estilo, etc.). De todas formas, como principal defecto podemos destacar la falta de documentación

(existe pero realmente limitada). Tiene una interfaz muy intuitiva y ligera, sin embargo, según [Carlos, 2008] tiene algunas carencias, como errores en la traducción o que la estética de las ventanas es la de Java por defecto y no se integra con la del sistema.

OpenProj está disponible en dos opciones. Para descarga gratuita (OpenProj), o para contratación mediante la modalidad Software como Servicio (SaaS), denominado Project-ON-Demand, con lo que se tiene la herramienta disponible en Internet para compartir con empleados, clientes y proveedores sin necesidad de complejas instalaciones.

DotProject

Está construido por aplicaciones de código abierto. Es una aplicación basada en Web, multiusuario, soporta varios lenguajes. Básicamente es un programa al que se le van añadiendo proyectos, y dentro de estos proyectos las tareas que lo conforman. Tiene todas las opciones básicas de cualquier gestor de proyectos:

- Gestor de empresas
- Proyectos
- Tareas
- Calendario
- Ficheros
- Contactos
- Usuarios
- Sistema

Es una herramienta que permite gestionar las distintas fases y tareas que componen un proyecto. Según [Team, 2007] se perfila como una interesante herramienta para trabajar en entornos colaborativos, permitiendo a los integrantes del equipo trabajar compartiendo información relativa a los proyectos.

De forma general las características de la herramienta son:

- Permite la gestión y planificación de proyectos en entornos colaborativos.
- Basado en plataforma Web permite la participación online de los miembros de un proyecto.

- Permite la asignación de recursos (equipamientos, mobiliario...) a un proyecto o varios, así como la descomposición en tareas
- Permite Vista de eventos y tareas en calendario.
- Permite la visualización de informes y estadísticas sobre los proyectos registrados.
- Gestiona las actividades de la empresa al permitir la parcelación del proyecto en tareas.
- Permite a su vez la generación de gran cantidad de informes

GanttProject

Es una herramienta libre y fácil de usar. Está basada en la creación de diagramas de Gantt y de los recursos de carga gráfica. Puede organizar el proyecto en un árbol de tareas y asignar recursos humanos que tienen que trabajar en cada tarea, con la posibilidad de crear dependencias entre estas últimas. Según la información consultada en el Sitio Oficial de la herramienta GanttProject es posible generar informes en HTML⁴ y PDF⁵, el intercambio de datos con Microsoft Project y aplicaciones con hojas de cálculo.

Las principales ventajas que posee esta herramienta son:

- **Facilidad de aprendizaje:** no necesita manuales de espesor que empezar a trabajar con GanttProject. Si eres un usuario familiarizado con el concepto de tareas, las cesiones y las dependencias, puede convertirse en un experto en GanttProject en un par de horas.
- **Excelente precio:** las herramientas comerciales de gestión de proyectos cuestan mucho dinero. GanttProject es libre para cualquier fin.
- **Crossplatform:** GanttProject es una aplicación Java y se ejecuta en Windows, Linux, MacOSX y otros sistemas operativos de apoyo de Java.
- **Abrir el código fuente:** GanttProject puede ajustarse a sus necesidades, implementar nuevas características, añadir sus propios informes, etcétera.

TaskJuggler

Es una potente herramienta de código abierto para la gestión de proyectos. Con el enfoque de planificación de proyectos que posee es superior y más flexible que las herramientas de edición de

⁴ HTML (por sus siglas en español): Lenguaje de Marcado de Hipertexto.

⁵ PDF (por sus siglas en inglés): Portable Document Format.

diagramas de Gantt. Puede sustentar cientos de miles de tareas y gran cantidad de recursos lo que es aplicable a proyectos de gran escala.

Para mayor garantía de los jefes o administradores de proyectos permite la asimilación de tareas a partir de la primera idea de realización del proyecto, por lo que es de gran ayuda para el alcance del mismo, la asignación de recursos, gastos e ingresos de planificación, gestión y comunicación de riesgo.

TaskJuggler ofrece una planificación óptima pues calcula el proyecto en líneas de tiempo y de recursos basado en el perfil del proyecto y las limitaciones o posibles riesgos que se han identificado y proporcionado. Además posee una alarma si el proyecto no cumple con el tiempo establecido, lo que permite obtener avisos respecto a esto. Es también ideal para las nuevas estrategias de gestión, tales como Extreme Programming y Agile Project Management.

Según [Freitag, 2008] alguna de sus principales características son:

- Gestiona tareas, recursos y costos en un solo paquete.
- Nivelación automática de los recursos, las tareas de resolución de conflictos, y la tarea de filtrado.
- Completa de los informes y opiniones para que usted pueda encontrar la información que necesita cuando lo necesita.
- Construir plantillas para comenzar (también puede guardar sus propias plantillas de los horarios para el uso futuro).
- Seguimiento y presentación de informes.
- Interfaz gráfica de usuario convenientemente al editar sus proyectos, ver y generar informes y diagramas de Gantt
- Número ilimitado de los escenarios (líneas de base) de un mismo proyecto para utilizarlo en caso de análisis.
- CSV para el intercambio de datos de exportación con Office Suites.
- Análisis de riesgos.
- Desglose de proyectos y la gestión del equipo de apoyo.
- Horarios flexibles de trabajo y manipulación de vacaciones.
- Apoyo a cambio de trabajo.

- Múltiples zonas horarias.
- Las tareas pueden tener costos iniciales y costos de acabado.
- Los recursos pueden tener gastos de funcionamiento.
- Apoyo de análisis a las pérdidas y ganancias.
- Generación de informes en HTML y XML.
- Potente sintaxis con la descripción de apoyo de macro proyectos.
- Apoyo a la asignación de recursos de base de datos.
- Generación de archivos de iCal para el intercambio de datos con las herramientas de productividad.

Planner

Es una herramienta libre de aplicación de escritorio para la gestión y seguimiento de proyectos de software. Permite la descomposición en tareas y subtareas, dependencias, identificación de la ruta crítica, diagramas de Gantt. Entre sus funcionalidades permite definir un calendario, marcar eventos importantes, asignar recursos tanto humanos y materiales, establecer prioridades, entre otras más. En un principio fue desarrollada para Linux, pero consta con una versión beta disponible para Windows. Además de estas funcionalidades, Planner ofrece:

- Cuadro Intuitivo: Se puede conocer la planificación completa de todo el proyecto, aquí se pueden establecer diferentes colores, diferentes estilos para cada planificación y para cada grupo de personal.
- Acceso a Redes: Otros usuarios pueden abrir el mismo archivo que se está editando con acceso de sólo lectura a través de redes.
- Planificación relacionada a tareas: Cada planificación debe ser adjuntada a una tarea que lo ayuda a rastrear y visualizar todas las planificaciones para una tarea específica.
- Cada archivo puede configurarse con una contraseña para abrir y una contraseña para modificar.
- Puede exportar a archivo de Texto (*.txt), archivo script SQL (*.sql), archivo HTML (*.htm), archivo XML (*.xml), archivo de hoja de cálculo (*.xls), portapapeles de MS Windows, Formato de Texto Enriquecido (*.rtf), formato Word (*.doc)

1.7 Conclusiones parciales

La planificación de software en los proyectos productivos en la UCI es un aspecto de gran importancia para obtener una mayor organización y calidad del producto que se produce.

El proceso de planificación debe tener en cuenta a todo lo largo del desarrollo del software.

En este capítulo se trató conceptos generales del Proceso de Desarrollo del Software, la Gestión de Proyectos y dentro de esta ultima la planificación, como actividad primordial de esta gestión.

Se realizó un estudio de las herramientas más usadas actualmente para la Gestión del Software en el mundo lo que permitió establecer una comparación para la selección de una de ellas que sirva de apoyo al procedimiento propuesto que se describirá más detalladamente en uno de los epígrafes del próximo capítulo.

Capítulo II: Procedimiento propuesto

2.1 Caracterización de la Gestión de Proyectos y el Proceso de Desarrollo en los Proyectos en la UCI

En el transcurso del curso escolar 2007 – 2008, se llevó a cabo la aplicación de encuestas y entrevistas a Jefes de proyectos productivos, directivos y personal capacitado en el tema de las 10 facultades. Se tomo una población de 50 personas.

La realización de estas encuestas y entrevistas sirvió para realizar una valoración, de la situación por la que atraviesan los proyectos productivos actualmente en la UCI. La información obtenida permitió basar la investigación en las principales necesidades que sufren los proyectos en la Universidad actualmente.

Con los resultados arrojados por las encuestas, *ver Anexo 1*, se obtuvo una caracterización del Proceso de Desarrollo de Software, la Gestión de Proyectos y dentro de ella los procesos de planificación, asociados a los proyectos productivos que se desarrollan actualmente en la UCI.

2.1.1 Caracterización del Proceso de Desarrollo de Software

De las conclusiones que se pudieron obtener de las encuestas realizadas, está en primer lugar, que la metodología más usada en el Proceso de Desarrollo de Software en los proyectos productivos en la UCI es RUP donde más del 74% de los encuestados, *ver Anexo 2*, usan esta metodología de desarrollo y más del 70% sugieren la misma para llevar a cabo el desarrollo de software, *ver Anexo 3*.

No existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Pese a los resultados obtenidos en la realización de encuestas y entrevistas en la Universidad existen proyectos que utilizan otras metodologías, *ver anexo 2*, este trabajo no propone ninguna metodología para el desarrollo del software, es el equipo de desarrollo el que selecciona la metodología de desarrollo a utilizar.

2.1.2 Caracterización de la Gestión de Proyectos y el proceso de Planificación de Software

La Gestión de Proyectos puede caracterizarse, según los resultados obtenidos durante la investigación, es que la misma no se realiza de una forma eficiente, en los proyectos productivos que se desarrollan actualmente en la UCI, no se tienen en cuenta todos los pasos que se deben realizar en para la gestión de proyectos. Esto puede provocar que el desarrollo del proyecto no sea el más eficiente posible, puede influir también de manera significativa en la calidad de los entregables, los plazos y costos del proyecto pueden ser alterados y por tanto repercute en la factibilidad del mismo y afecta en gran medida a la planificación del software.

En cuanto al proceso de planificación no se le da la importancia que la misma requiere y exige. Aunque más del 80% de los encuestados aseguran conocer los procesos de planificación, las entrevistas demuestran que a pesar de que se quiere abogar por la realización de una planificación eficaz, esto casi nunca se logra. En la mayoría de los casos el rol de planificador esta obsoleto en el proyecto pues no tiene los medios ni la preparación necesaria para realizar una correcta planificación.

2.1.3 Caracterización general sobre las encuestas

Preguntas 2 y 3.

En cuanto a la Gestión de Proyectos y planificación de software, más del 80% de los encuestados evaluaron a la Gestión de Proyectos en la Universidad de regular, el 60% evaluaron la planificación regular, un 30% de mala y solo un 10% de buena. Esto indica que no se lleve a cabo una buena Gestión de Proyectos, y que las planificación es que se realizan en la mayoría de las actividades productivas no sean lo más correctas posibles, por la razón de que no existe una guía o procedimiento que proporcione un orden lógico para la realización los pasos principales para realizar una panificación lo más real permisible, *ver Anexo 4*.

Pregunta 6.

También sobre la planificación se concluyó que la mayoría de los proyecto realicen las planificaciones de los mismo las iteraciones, *ver Anexo 5*.

2.2 La planificación según CMMI

Según [CMMI, 2006] la planificación incluye la estimación de los atributos de los productos y de trabajo, las tareas, determinar los recursos necesarios, la negociación de compromisos, la elaboración de un calendario, así como identificar y analizar los riesgos del proyecto.

Estas actividades deberán ser necesarias para establecer el Plan de Proyecto. El mismo proporciona la base para llevar a cabo el control de las actividades del proyecto que se ocupan de los compromisos con el cliente. El Plan de Proyecto por lo general tiene que ser revisado a medida que el proyecto avanza para enfrentar los cambios en los requisitos y compromisos, estimaciones inexactas, acciones correctivas y cambios en el proceso.

La planificación incluye de forma general lo siguiente:

- El desarrollo del Plan de Proyecto.
- Interacción con los interesados
- Obtener compromiso con el Plan
- Mantener el Plan

En la UCI se está abogando por alcanzar el nivel 2 de CMMI. Inicialmente todas las empresas poseen el nivel 1 por el solo hecho de existir como empresas. Para alcanzar el nivel 2, CMMI propone prácticas, que están agrupadas en las diferentes áreas de trabajo, a las cuáles se hicieron referencias en el Capítulo I, enmarcándose para este momento solamente en la planificación de proyectos. Alcanzar este nivel resulta una tarea difícil pues pretende conseguir que en los proyectos de la organización haya una gestión de los requisitos y que los procesos estén planeados, ejecutados, medidos y controlados. De una forma más explícita, con el nivel 2 de CMMI se pretende:

- Ejecutar y gestionar los proyectos de acuerdo con los planes de proyecto.
- Saber cuánto trabajo está hecho y cuánto queda por hacer.
- Revisar y controlar los elementos de trabajo con las personas involucradas.

2.2.1 Fases que propone CMMI para la planificación

Sus tres fases son:

1. *Establecer estimaciones:* A partir de las necesidades iniciales del cliente, se establece una serie de lineamientos y actividades iniciales, que deben ser acordes a las necesidades del cliente y las capacidades generales de la empresa.
2. *Desarrollar un Plan de Proyecto:* A partir de las estimaciones iniciales y un análisis más detallado del proyecto, se genera un plan suficientemente completo y delimitado con actividades y responsables, todo con el fin de tener una organización lo más completa posible del proceso de desarrollo.
3. *Compromiso con el Plan:* Una vez planteado el plan, se obtiene el compromiso por parte de cada uno de los responsables de las actividades contempladas en el plan, con el fin de que si existen discordancias con las fechas y actividades propuestas, se adapte el plan a la realidad lo más correcto posible.

2.2.2 Tareas que propone cada fase

Fase 1: Estimaciones

La idea principal es analizar el alcance y nivel de complejidad del proyecto para poder establecer una serie de lineamientos en el tiempo.

- Estimar el alcance del proyecto: en base a las habilidades y recursos de la empresa, se deben analizar los requerimientos que debe contemplar.
- Establecer las tareas y productos de trabajo.
- Definir el ciclo de vida del proyecto.
- Determinar las estimaciones de esfuerzo y costo.

Fase 2: Plan de proyecto

A partir de las estimaciones, la idea es estructurar el Plan del Proyecto, con la idea de planificar lo más completo posible el proceso de desarrollo, de acuerdo con las capacidades y necesidades de la empresa.

- Establecer el presupuesto y cronograma.
- Identificar los riesgos del proyecto.
- Plan para la gestión de los datos del proyecto.
- Plan para los recursos del proyecto.
- Plan para las habilidades y conocimiento necesarios.
- Plan para involucrar a los participantes.
- Establecer el plan del proyecto.

Fase 3: Compromiso con el Plan

La idea es que a partir del plan, se revise la factibilidad de las actividades con los encargados de cada área, evaluando las perspectivas y permitiendo ajustar el mismo de ser necesario. Es importante que todos los responsables estén de acuerdo con las actividades, fechas y recursos asignados para cada una de sus responsabilidades, para así evitar conflictos de tiempo y calidad del producto.

- Revisión de los planes que afectan al proyecto
- Reconciliar el trabajo y niveles del recurso
- Obtener el compromiso sobre el plan

2.3 Selección de la herramienta para la Gestión de Proyectos

Después de una investigación realizada sobre las herramientas libres para la Gestión de Proyectos más usadas actualmente, se seleccionó a DotProject como la herramienta propuesta para dar apoyo a la Gestión de Proyectos en las actividades productivas en la UCI.

La selección de esta herramienta se basa fundamentalmente en las características expuestas anteriormente en el Capítulo I, además después de analizar los resultados arrojados en las encuestas realizadas, ver Anexo 6, más del 66% de los proyectos productivos en la UCI utilizan a esta herramienta la gestión de los mismos.

DotProject es una herramienta de gestión de proyectos, creada por la comunidad de Software Libre internacional y un sistema integrado de gestión.

Esta herramienta fue construida con aplicaciones de código abierto, es una aplicación basada en la Web, un sistema multiusuario y puede integrarse a LDAP (*Lightweight Directory Access Protocol*) protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red.

Dada esta característica de código abierto en nuestra Universidad un grupo de desarrollo, dada algunas deficiencias que traía consigo esta herramienta le hicieron algunas mejoras a ciertas funcionalidades, ejemplo de esto fue la integración con LDAP, esto ha facilitado en gran medida que se cuente con sistema de automatización centralizado, además que el administrador de DotProject no se vea en la obligación de crear los usuarios manualmente ya que cuando este último se loguea se adiciona de forma automática en la Base de Datos que la herramienta tiene. También se hizo mejoras en el módulo correspondiente a generar los diagramas de Gantt ya que el mismo no se visualizaba normalmente, esto fue de gran valor pues la generación de estos diagramas son de vital importancia para la planificación de un proyecto tanto para los Jefes del mismo como para los usuarios en general.

Esta herramienta esta compuesta por dos grupos de módulos:

- Usuarios.
- Administración.

2.3.1 Módulo de Usuario

- *Empresas*: Permite crear y manipular nuevas empresas.
- *Proyectos*: Permite crear y manipular nuevos proyectos.
- *Tareas*: Permite crear y manipular nuevas tareas.
- *Calendario*: Visualiza tareas y eventos.
- *Ficheros*: Permite gestionar ficheros de los proyectos.
- *Contactos*: Muestra información de los usuarios del sistema.
- *Foros*: Permite crear foros y asociarlos a los proyectos.

Módulo de Empresas

Entre las opresiones generales de este módulo se encuentran:

- ✓ Visualizar las empresas.
- ✓ Crear nuevas empresas.
- ✓ Editar una empresa.
- ✓ Eliminar una empresa.

Módulo de Proyectos

Este módulo permite realizar operaciones sobre un proyecto en específico así como la visualización de los distintos proyectos, *ver figura 11*, que existan en diferentes vistas y como aspecto importante permite generar un gráfico de Gantt con toda la planificación de los proyectos.

Este módulo es muy importante permite ver el estado en el cual se encuentran todos los proyectos, visualiza la fecha de inicio y fin, ordenar los mismo así como filtrarlos por dueño y por empresas, lo cual otorga una mayor organización.

The screenshot shows a web interface for project management. At the top, there's a navigation bar with 'Ayuda | Mis datos | A realizar |'. Below it, a search bar and buttons for 'nueva tarea', 'new event', and 'nuevo'. The main content area is titled 'Ver Proyecto' and shows details for a project named 'Mi proyecto'.

Detalles:

- Empresa: Empresa Citimatel
- Alias: M proyect
- Fecha Inicio: 02/05/2007
- Fecha de Finalización Tentativa: 25/07/2007
- Fecha de finalización Real: **31/08/2007**
- Presupuesto Tentativo: \$0.00
- Responsable de Proyecto: Matinez Fernandez, rixal
- URL: [empty]
- URL de la Organización: [empty]

Resumen:

- Estado: En Progreso
- Prioridad: High
- Tipo: Unknown
- Progreso: 1.2%
- Horas Trabajadas: 0.00
- Horas Programadas: 109
- Horas Proyecto: 213

	Correo Electrónico	Teléfono	Dapar
rixal rixal	rixal@gc.oe.ce.cu		
rixal Matinez Fernandez	rixal@gc.oe.ce.cu		
eder despaigne	eder@gc.oe.ce.cu		

Below the details, there are tabs for 'Tareas', 'Tareas (inactivas)', 'Foros', 'Gráficos de Gantt', 'Historiales de Tarea', and 'Ficheros'. The 'Tareas' tab is active, showing a list of tasks:

Pincho	Nuevo	Historial	Trabajo	Progreso	Nombre de la Tarea	Creador de la Tarea	Usuarios Asignados	Fecha Inicio	Duración	Fecha de Finalización	Ultima
			85%		Mi tarea	rixal	admin (100%)	02/05/2007 11:30 am	1 horas	23/05/2007 05:00 pm	
		Historial	20%	↑	Subir los servidores	rixal	rixal (100%)	19/05/2007 02:00 pm	2 horas	19/05/2007 05:00 pm	
		Historial	0%		arrglar gannt	eder	eder (100%) (+1)	11/06/2007 05:45 am	104 horas	27/05/2007 05:00 pm	
		Historial	0%		Segunda tarea	rixal	admin (50%)	14/06/2007 08:00 am	1 horas	31/05/2007 02:00 pm	

At the bottom, there's a legend for task status: [empty] =Tarea Futura, [light gray] =Empezado y a tiempo, [medium gray] =O ebí haber comenzado, [dark gray] =Atrazado, [white] =Hecho. Open : Close All Tasks

Figura 11: Detalles de un proyecto.

Con la generación de informes, ver figura 12, operación que permite este módulo, se puede obtener estadísticas de cada proyecto y de los usuarios (tareas terminadas, atrasadas, en progreso, etc.) así como la generación de un pdf con el informe final.

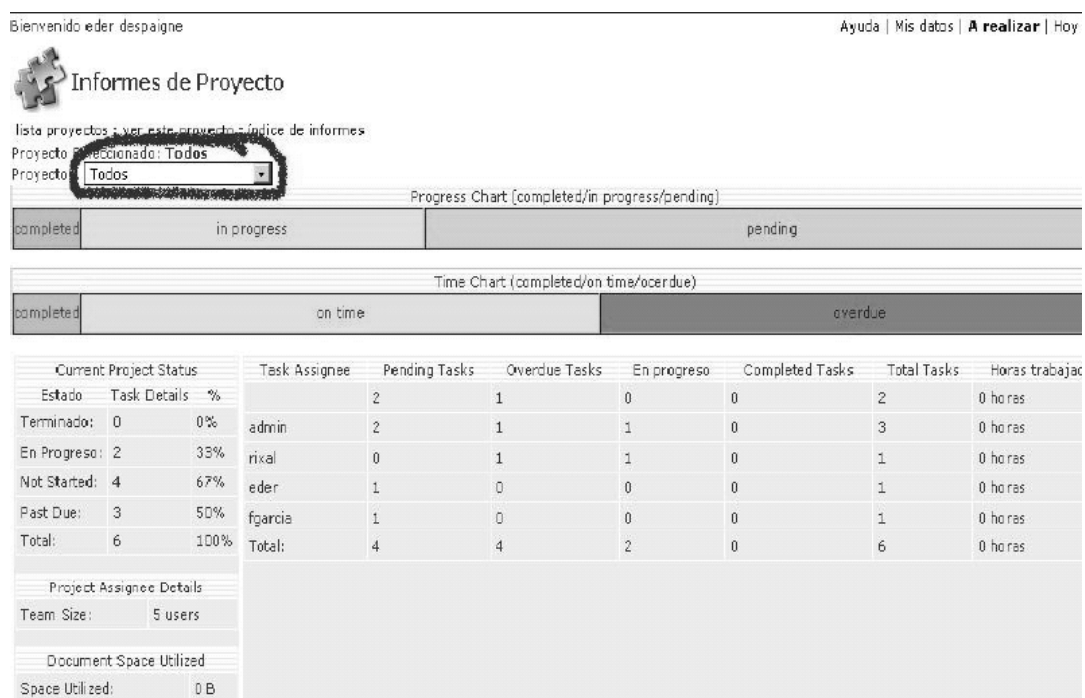


Figura 12: Generación de informes de un proyecto.

Entre las operaciones generales están:

- ✓ Visualizar los proyectos.
- ✓ Crear nuevos proyectos.
- ✓ Generar informes.
- ✓ Editar un proyecto.
- ✓ Eliminar un proyecto.

Módulo de Tareas

Este módulo permite ver el estado de cada tarea, mostrando de la misma la información completa así como la gráfica de Gantt de las tareas del usuario, ver figura 13. También proporciona tener el conocimiento de la carga de trabajo de cada usuario, ver la relación gráfica de dependencia de las

tareas, visualiza los detalles, fecha inicio y fin propuesta para las tareas y los recursos humanos que requieren las mismas.

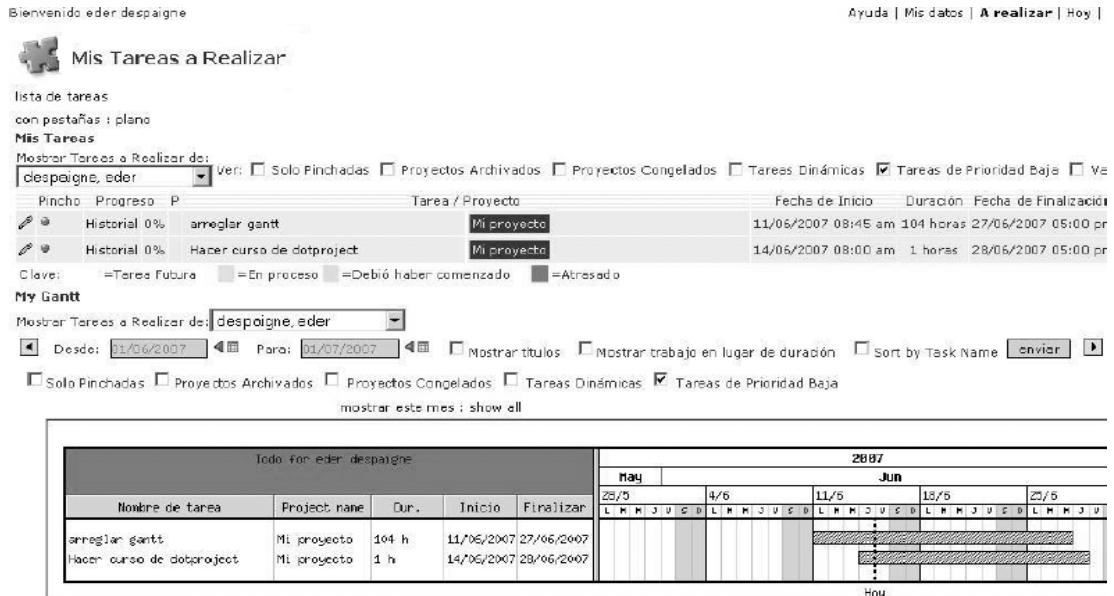


Figura 13: Tareas de los usuarios.

Entre las operaciones generales de este módulo se encuentran:

- ✓ Visualizar las tareas.
- ✓ Crear nuevas tareas.
- ✓ Editar una tarea.
- ✓ Eliminar una tarea.

Módulo de Calendario

Este módulo permite ver las tareas y eventos de cada empresa, las tareas y eventos de un mes y un año en específico y muestra la planificación por semanas o la planificación de las tareas por cada día de la semana marcada.

Módulo de Contactos

Entre las operaciones generales de este módulo se encuentran:

- ✓ Visualizar el personal registrado en el dotProject.

- ✓ Añadir un nuevo contacto.
- ✓ Editar un contacto.

Módulo de Ficheros

Entre las operaciones que podemos realizar con este módulo están:

- ✓ Visualizar los ficheros subidos al sistema.
- ✓ Subir ficheros.
- ✓ Editar información sobre ficheros.
- ✓ Eliminar ficheros.
- ✓ Trabajo con carpetas.

Módulo de Foros

Las operaciones que permite este módulo son:

- ✓ Visualizar los foros creados.
- ✓ Crear nuevo foro.
- ✓ Agregar un nuevo tema.
- ✓ Responder en un foro.
- ✓ Borrar y editar un foro.
- ✓ Borrar y editar un tema del foro.

2.3.2 Módulo de Administración

- *Usuarios*: Permite gestionar información acerca de los usuarios.
- *Sistema*: Permite configurar todos los parámetros del sistema.

Módulo de Usuarios

Este módulo de forma general permite ver la lista de los usuarios y el estado actual de cada usuario, ver figura 14, y hacer un seguimiento de los mismos. También muestra los datos de los usuarios y permite el cambio de roles de los mismo.

Sistema de gestión de proyectos del Grupo Creativo dotProject.net
FREE SOFTWARE

Empresas | Proyectos | Tareas | Calendario | Ficheros | Contactos | Foros | Tickets | Usuarios | Sistema - Nuevo Elemento -

Bienvenido eder despaigne Ayuda | Mis datos | **A realizar** | Hoy | Salir

Gestión de Usuarios

Ver: Todos A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

con pestañas : plano agregar usuario

ordenar por:	Historial Conexión	Nombre Conexión	Nombre Real	Empresa
	Desconectado	admin	Person, Admin	
	1.23 hrs. (0.00 hrs. listo) - Conectado	eder	despaigne, eder	
	Desconectado	fgarcia	Garcia, Fernando	
	Desconectado	rixal	Matinez Fernandez, rixal	

Figura 14: Estado de los usuarios de un proyecto.

Operaciones que permite este módulo:

- ✓ Visualizar el personal registrado en el dotProject.
- ✓ Añadir, Editar, borrar y cambiar permisos de usuario.
- ✓ Editar preferencias del usuario.

Módulo de Sistema

Este módulo muestra todos elementos configurables que presenta DotProject, permite activar los gráficos de Gantt, integrar el DotProject a LDAP, establecer nombre, dominio, tiempo de duración de trabajo, días laborables.

También permite visualizar los módulos que están presentes y su estado, ver si los módulos están visibles o no, instalar, desinstalar, configurar y posicionar los módulos.

Entre las operaciones que podemos realizar con este módulo se encuentran:

- ✓ Visualizar las opciones a configurar.
- ✓ Configurar el sistema, preferencias de usuario predeterminadas, los módulos y los roles de usuario.

2.4 Presentación del procedimiento propuesto

2.4.1 Definición

Nombre del procedimiento: Procedimiento para la planificación de proyectos con herramientas libres.

Objetivo: proporcionar una línea de trabajo a los planificadores de los proyectos productivos, por la necesidad de obtener plazos reales y aceptables para la terminación de software, y mayores garantías en todo lo referente a la organización del trabajo en equipo y la calidad de los entregables, que es por lo que se aboga en la UCI; además está encaminado a aportar un punto de inicio al proceso de planificación y proporcionar, sino en su totalidad, gran parte de los conocimientos adquiridos durante la investigación para todos aquellos interesados en el tema de forma organizada y flexible para su uso. El procedimiento además propone el uso de una herramienta libre para la gestión del proyecto que ayuda durante la planificación y seguimiento del proyecto.

Alcance: constituye una guía para planificar los proyectos productivos en la UCI.

Referencia: Se fundamenta en el marco teórico descrito en el **Capítulo I**, donde se argumenta todo lo referente al proceso de planificación, estimaciones, modelo CMMI, algunas de las metodologías ágiles y las herramientas de apoyo a la planificación que actualmente son más utilizadas; por la importancia que debe otorgársele a estos temas cuando se propone un procedimiento de este tipo.

Responsables:

Autoras: Dayani Flores Pérez y Bertha Román Martínez

Tutor: Yanko Hernández Valdés

Cotutor: Greidys Jorga Luegues

2.4.2 Contenido a tener en cuenta para planificar

El procedimiento es una guía que debe garantizarle a todos aquellos que vayan a usarlo tener en cuenta algunos aspectos que no deben pasar por alto. En este caso para planificar proyectos software existen varios temas que deben estar presentes y adecuarlos a las características específicas de la UCI. Los mismos son:

- ¿Qué tipo de proyecto se va a planificar?

- ¿En qué metodología de desarrollo basarse?
- ¿Qué modelo o norma se tendrá en cuenta para la calidad?
- ¿Qué documentos o entregables deben generarse durante el proceso?
- ¿Qué herramientas de gestión se utilizarán para apoyar el proceso?

2.4.3 Esquema general del procedimiento

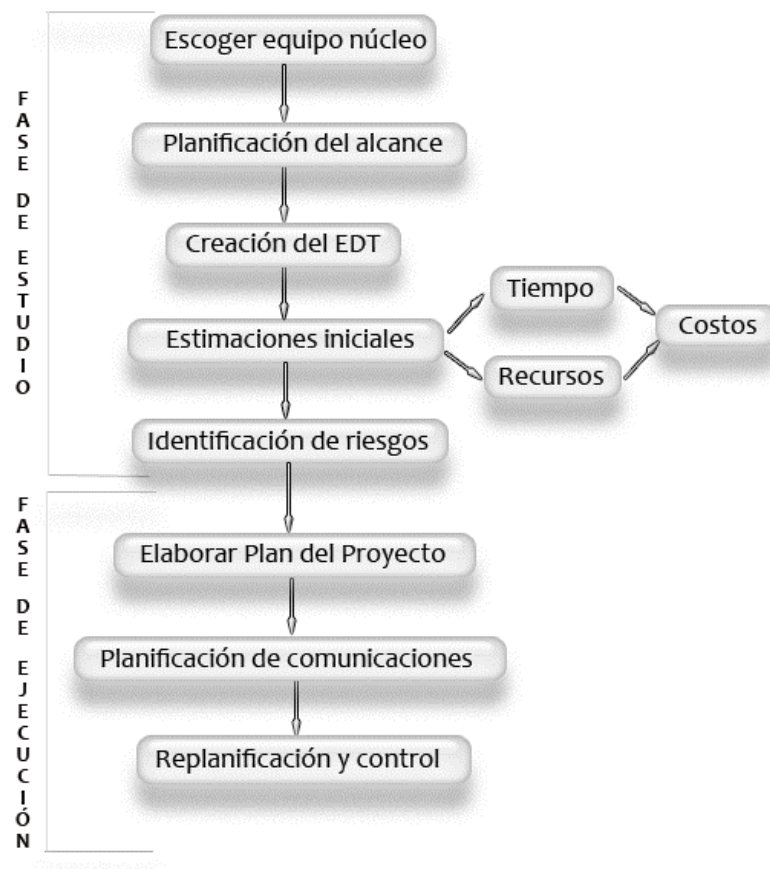


Figura 15: Esquema general del procedimiento.

Fase de estudio: En la UCI generalmente un proyecto de software es asignado a una persona en específico, la cual llevará el mismo a una realización completa, con un inicio y un final bien delimitados. Lo correcto sería que esta persona asignase un planificador y que este se encargase de realizar todos los pasos que se proponen para planificar en conjunto con el Jefe de proyecto. Esta fase se divide en:

Selección del Equipo Núcleo: Primer paso que se propone para lograr una planificación eficaz y certera.

Planificación del Alcance: Planificación más importante pues delimita el desarrollo del proyecto.

Creación del EDT: Organización general de las tareas. Este es un paso sumamente importante, pues de él se apoyarán otros pasos para su realización.

Estimaciones iniciales: Se busca estimar de manera inicial el tiempo, recursos y costos con el objetivo de obtener cierta garantía en cuanto a la factibilidad del proyecto.

Identificación de riesgos: Reconocimiento de los primeros riesgos que puedan aparecer en el proyecto, cuestión que no debe obviarse por las graves consecuencias que traería sino se toman en cuenta.

Fase de ejecución: Esta fase se lleva a cabo una vez que se ha decidido continuar con el desarrollo del proyecto. Esta fase se divide en:

Elaborar Plan de Proyecto: Es la elaboración del documento principal para la gestión de proyecto. Se le ha incluido la planificación temporal, costos, recursos y riesgos por la dependencia que tiene su elaboración con cada una de ellas.

Planificación de comunicaciones: Planificación de las reuniones del proyecto.

Replanificación y control: Momento en que debe actualizarse el Plan de Proyecto, y hacerse una revisión de todo el procedimiento para actualizar la planificación y reordenarla.

2.5 Definición del procedimiento propuesto

2.5.1 Selección del equipo núcleo

No es difícil imaginar que para una sola persona, el planificador, resultará un tanto engorroso tener en cuenta todos y cada uno de los detalles que deberán analizarse durante el proceso de desarrollo.

Durante la investigación realizada se concluye que se hace necesaria la selección de un Equipo Núcleo (EQ) por la importancia que traería consigo contar con personal de apoyo a las tareas de iniciación y control del proyecto y proporcionen conocimientos en las áreas claves del mismo.

Las personas implicadas en el EQ deberán ser capaces de actuar sin supervisión externa, ser emprendedoras. A veces estas personas no se escogen, ya vienen impuestas, por lo que sería conveniente lograr una gran comunicación con éstas y entre ellas para que el proyecto mantenga un ánimo de trabajo coordinado y colaborativo.

El presente trabajo propone que deberá seleccionar un equipo compuesto por un:

- Arquitecto de software
- Analista del sistema
- Líder de desarrollo
- Líder de prueba
- Gestor de gestión de la configuración
- Representante del cliente

La selección puede variar de acuerdo a la metodología utilizada.



Este paso será realizado por el **Planificador** junto al **Jefe de Proyecto**.

2.5.2 Planificación del alcance (ámbito de software)

La planificación principal que debe llevarse a cabo es la del alcance, pues de ella dependen el resto de las planificaciones (costes, tiempo, recursos, etcétera). Según [Proyecto, 2004] la planificación del alcance es el "*proceso de elaborar y documentar progresivamente el trabajo del proyecto, que incluye redactar una definición del alcance que incluye la justificación del proyecto, los entregables principales, y los objetivos del proyecto*".

La etapa de la planificación del alcance es el momento de centrarse en los detalles del proyecto, pues esta será la base para la creación de la Estructura de Desglose de Trabajo (EDT), que será explicado en el próximo acápite. Esta planificación se relaciona, principalmente, con lo que está o no incluido en el proyecto. La planificación del alcance del proyecto es uno de las fases de la Gestión del Alcance del

Proyecto que agrupa además la Definición del Alcance, la creación del EDT, verificación del alcance y control del alcance. Estas dos últimas fases se realizan una vez que el proyecto esté en marcha, por eso se tendrá en cuenta las tres primeras.

2.5.2.1 Definición del alcance

Se refiere principalmente al enunciado del alcance que describe en detalle los entregables del proyecto y el trabajo necesario para crear esos entregables. Proporciona un entendimiento común entre los interesados del proyecto y el equipo de desarrollo, y describe los principales objetivos del proyecto. Permite al equipo de trabajo realizar una planificación más detallada guiándolo durante la ejecución del proyecto y proporciona una línea base para determinar los límites del proyecto respecto a las solicitudes de cambio o de trabajo adicional.

Según [Proyecto, 2004] gestionar el alcance del proyecto puede determinar el nivel de éxito con que el equipo de dirección del proyecto podrá planificar, gestionar y controlar la ejecución del proyecto. El enunciado del alcance del proyecto detallado incluye, ya sea de forma directa o mediante referencia a otros documentos, lo siguiente:

- *Objetivos del proyecto*
- *Descripción del alcance del producto.* Describe las características del producto, servicio o resultado para el cual se creó el proyecto. Generalmente, estas características serán menos detalladas en las fases iniciales y más detalladas en las fases posteriores, a medida que las características del producto se van elaborando gradualmente.
- *Requisitos del proyecto*
- *Límites del proyecto.* Identifica generalmente qué está incluido dentro del proyecto.
- *Productos entregables del proyecto.*
- *Criterios de aceptación del producto.* Definen el proceso y los criterios para aceptar los productos completados.
- *Restricciones del proyecto.* Enumeran y describen las restricciones específicas del proyecto asociadas con el alcance del proyecto que limitan las opciones del equipo del proyecto.
- *Organización inicial del proyecto.* Se identifica a los miembros del equipo del proyecto, así como a los interesados. También se documenta la organización del proyecto.
- *Riesgos iniciales definidos.* Identifica los riesgos conocidos.

- *Hitos del cronograma.* El cliente o la organización ejecutante pueden identificar hitos y pueden colocar fechas impuestas en dichos hitos del cronograma. Estas fechas pueden tratarse como restricciones del cronograma.
- *Limitación de fondos.* Describe cualquier limitación aplicada a la financiación del proyecto, ya sea en cuanto al valor total o durante períodos específicos.
- *Especificaciones del proyecto.* Identifica aquellos documentos de especificaciones con los que debería cumplir el proyecto.
- *Requisitos de aprobación.* Identifica los requisitos de aprobación que pueden aplicarse a aspectos tales como objetivos, productos entregables, documentos y trabajo del proyecto.

2.5.2.2 Estimación del alcance

Según lo planteado en [CMMI, 2006] en la Fase 1: Establecer estimaciones, para la estimación del alcance del proyecto se debe tener en cuenta:

- Desarrollar el EDT.
- Identificar con detalle paquetes de trabajo para especificar las estimaciones de las tareas del proyecto, responsabilidades y calendarios.
- Identificar los componentes de productos que serán adquiridos desde el exterior con los que se van a trabajar.
- Identificar los productos de trabajo que se reutilizarán.

Para la estimación del alcance se debe tener en cuenta la definición del ciclo de vida del proyecto, es decir las fases con las que contará el proyecto, definidas en función del alcance de las necesidades, las estimaciones del proyecto y las características del mismo. Dentro de estas fases a su vez pueden existir subfases compuestas por tareas. En proyectos de gran envergadura pueden constar de múltiples fases.

2.5.2.3 El alcance y la comunicación con el cliente

El cliente representa un factor importante para definir el alcance, pues solo él es capaz de mostrar el camino a seguir sin ambigüedades y de la forma más sencilla. Sin embargo a veces sucede, que por mucho que indagemos con él no se logra concretar nada.

Según [Navarro, 2005] la comunicación con el cliente es fundamental pues es quién puede ayudar a determinar el ámbito del software de forma correcta y enfocada a los objetivos principales. Esta comunicación se puede iniciar con las preguntas de contexto libre, de las que existen tres tipos:

- El primer grupo se centra en el cliente, los objetivos globales y los beneficios. Algunas de las preguntas que se realizarían son:

- ¿Quién está detrás de la solicitud de este trabajo?
- ¿Quién utilizará esta solución?
- ¿Cuál será el beneficio económico de una buena solución?
- ¿Hay otro camino para la solución?

- El segundo grupo permite comprender mejor el problema y que el cliente exprese sus percepciones sobre una solución:

- ¿Cómo caracterizaría (el cliente) un resultado correcto que se generaría con una solución satisfactoria?
- ¿Con qué problemas se enfrentará esta solución?
- ¿Puede mostrarme (o describirme) el entorno en el que se utilizará la solución?
- ¿Hay aspectos o limitaciones especiales de rendimiento que afecten a la forma en que se aborde la solución?

- El tercer y último grupo se centra en la *efectividad* de la reunión. Se denominan metacuestiones:

- ¿Es usted la persona apropiada para responder a estas preguntas? ¿Son oficiales sus respuestas?
- ¿Son relevantes mis preguntas para su problema?
- ¿Estoy realizando muchas preguntas?
- ¿Hay alguien más que pueda proporcionar información adicional?
- ¿Hay algo más que debiera preguntarle?

Estas preguntas y otras similares ayudan a romper el hielo y a iniciar la comunicación esencial para establecer el ámbito del proyecto.



Este paso será realizado por el **Jefe de Proyecto** y el **Planificador** y revisado por el **EQ**. Estos datos se ubican en el Plan de Proyecto.

2.5.3 Creación de la Estructura de Desglose de Trabajo (EDT)

Según [Proyecto, 2004] un EDT es una "agrupación orientada a entregables de los elementos del proyecto que organiza y define el alcance total del trabajo del proyecto. Cada nivel descendente representa una definición cada vez mas detallada del trabajo de proyecto."

Según la definición puede entenderse que esta orientada solamente a nivel de entregable lo que realmente sería un error. Si esto fuera así, no podrían estimarse el costo, duración o planificación temporal, por lo que se hace necesario que un EDT este orientado a nivel de trabajo, para poder obtener la mayor cantidad posible de resultados. Una definición alternativa sería "una agrupación de los elementos del proyecto que organiza y define el alcance total del trabajo del proyecto en una manera estructurada que desglosa el proyecto en los paquetes del trabajo".

Los EDT progresan desde los niveles más altos, a través de los intermedios, hasta los más bajos, lo que una vez terminado se obtiene una lista deshilachada de los trabajos o actividades a realizar. Ayudan además a planificar y controlar el desarrollo del proyecto, evita perder tiempo y recursos en desarrollos obsoletos, clarifica las responsabilidades y proporciona una estructura para medir el éxito.

Según [CMMI, 2006] el EDT evoluciona con el proyecto. Inicialmente habrá un nivel alto del EDT que puede servir para estructurar la estimación inicial. El desarrollo del EDT divide el proyecto global interconectado en un conjunto de componentes manejables. El EDT proporciona una referencia de organización y mecanismo para la asignación de esfuerzo, calendario y la responsabilidad, y se utiliza como marco para planificar, organizar y controlar el trabajo realizado en el proyecto.

Según lo que plantea [CMMI, 2006] el EDT debe permitir la identificación de los siguientes temas:

- Riesgos y sus tareas de mitigación.
- Tareas de las prestaciones y actividades de apoyo.
- Tareas de las aptitudes y la adquisición de conocimientos.
- Tareas para el desarrollo de planes de apoyo necesarios, tales como configuración gestión, control de calidad, verificación y planes.

- Tareas para la integración y la gestión de temas no desarrollados.

2.5.3.1 Estilos para realizar un EDT

Existen varias formas para realizar un EDT, todas con sus ventajas en específico. Las formas más comunes son el indentado y en gráfico.

Con el *formato indentado* (ver Figura 16) se pueden incluir más fácil los detalles del proyecto, llevarlos a las herramientas de gestión de proyecto, de las cuales se hará referencia más adelante, es fácil de corregir, imprimir y supervisar de forma controlada.

ID		
1		1 Proyecto Geografía
2		1.1 Equipo
3		1.1.1 Definir tema
4		1.1.2 Acordar
5		1.1.2.1 Integrantes
6		1.1.2.2 Anteproyecto
7		1.1.3 Aprobación de profesor
8		1.2 Investigación Chiapas
9		1.2.1 Buscar en Internet
10		1.3 Reporte
11		1.3.1 Solución
12		1.3.2 Redacción
13		1.3.2.1 Portada
14		1.3.2.2 Índice
15		1.3.2.3 Justificación
16		1.3.2.4 Hipótesis
17		1.3.2.5 Teoría
18		1.3.2.6 Conclusión
19		1.3.2.7 Materiales
20		1.3.3 Revisión
21		1.3.4 Impresión
22		1.3.5 Engargolar

Figura 16: Formato indentado.

Con el *formato gráfico (organigrama)* (ver Figura 17) se pueden demostrar los niveles de trabajo y ver la relación entre ellos, desde los más altos hasta los más bajos.

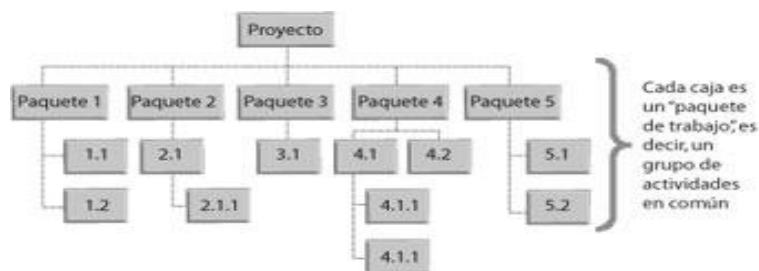


Figura 17: Formato gráfico.

Se recomienda realizar ambos estilos, el indentado para trabajar directamente con los miembros del proyecto y el gráfico para mostrarlos a los clientes o todas aquellas personas que no necesitan tantos detalles del desarrollo del trabajo.

2.5.3.2 ¿Como confeccionar un EDT?

La confección del EDT se debe realizar una vez se haya definido el alcance del proyecto, y se deben tener en cuenta los siguientes pasos:

1. Entender el propósito del proyecto.
2. Establecer los segmentos principales en que se divide el proyecto.
3. Desglosar estos segmentos grandes en otros más pequeños tantas veces como sea necesario, hasta que los mismos se puedan asignar y supervisar de forma rápida y fácil.
4. Llevar a cabo una revisión con el EQ
5. Realizar el diccionario del EDT.

Una vez terminado, se obtendrán como mínimo tres niveles bajo el título global, aunque pueden ser menos, pero la experiencia adquirida por los conocedores del tema indican que dichos niveles tal vez no serían lo suficientemente bien estructurados como para estimarlos y controlarlos correctamente. Por el contrario, si es demasiado detallado sería entonces muy tedioso trabajar con ellos.

Existen algunos términos claves que no se deben pasar por alto durante la confección del EDT. Los mismos son el paquete de trabajo y la cuenta de control.

El *paquete de trabajo* es el nivel más bajo del EDT. En este se asigna y supervisa el trabajo, y el principal para tratar duración, coste y requisitos de recursos. Pero ¿cómo de grande o pequeño debe ser un paquete para no llegar a tanto detalle? Para ello no hay respuesta, pero se pueden seguir determinadas pautas que ayudan en este proceso. Según [Proyecto, 2004] las mismas son:

- Un paquete del trabajo debe ser bastante pequeño para asignarse a un equipo pequeño o persona.
- Una pauta es "la regla 80" según la que ningún paquete del trabajo debe exceder 80 horas de trabajo.
- Tener en cuenta que el paquete no debe ser tan pequeño que sea solo realizable en un día, pero a la vez que no sea tan general pues lo hace parecer muy grande.
- Si es necesario analizar el paquete del trabajo "parte aquí y parte allí", se trata probablemente de más de un paquete de trabajo.

El *coste de control* es el nivel que se utiliza para la presentación a los interesados y la gerencia, pues con él se pueden seguir los costes de manera significativa para una supervisión más general sin demasiados detalles. Se encuentran generalmente un nivel por encima del paquete de trabajo.

2.5.3.3 Diccionario del EDT

El diccionario no es un libro de términos y definiciones. Es una manera de capturar la información esencial de cada paquete de trabajo como las premisas, asignación de recursos, predecesores, sucesores. Su contenido variara dependiendo da la información especifica del proyecto.

El diccionario se asegura de que toda la información sobre un paquete esté capturada y no queden dudas por parte de ningún miembro del proyecto. Esto sirve principalmente para aquellas personas que se incorporan nuevas en el proyecto, es una forma de darles un punto de partida.



Este paso deberá ser realizado por el **Planificador** y revisado por el **EQ**. Para recoger estos datos se debe generar un documento que contenga el EDT en ambos formatos.

2.5.4 Estimaciones iniciales

Estimar es un paso importante en la planificación de proyectos software, y un paso siguiente a la creación de la EDT. Según [Proyecto, 2004] la definición básica de una estimación es "una apreciación del probable resultado cuantitativo. Aplicado generalmente a costes y a la duración del proyecto y debe incluir siempre una cierta indicación de la exactitud." Son el puente para la planificación temporal, costes y recursos.

La validez de las estimaciones influencia mucho en el éxito del proyecto y no en caer en desviaciones, ya sean por irse demasiado por encima o por quedarse demasiado por debajo.

2.5.4.1 Precisiones en la estimación

Según [CMMI, 2006] las estimaciones del esfuerzo y los costos de un proyecto se basan, generalmente, en la utilización de métodos, modelos o datos históricos. Cuanto más se trata de estimar de forma cuidadosa, más posibilidades de acertar se tienen, por lo que no es difícil suponer

que seguido a esto la calidad será más visible y palpable. Sin embargo, ¿cómo estimar lo más certero posible? Existen algunas buenas prácticas para esto, principalmente basadas en la experiencia adquirida durante la realización de otros proyectos, o con la información que pueda existir sobre los mismos.

La UCI no cuenta con una trayectoria muy grande respecto a la experiencias adquiridas de la realización de estimaciones, lo que no sería factible guiarse por estas prácticas, aunque no debe descartarse; por lo que se propone para realizar estas estimaciones trabajar en conjunto con el EQ que deberá estar conformado para este entonces, por poseer mayor experiencia y conocimiento.

Estimar esfuerzos sin precedentes confiables, es un riesgo, por lo que se debe tener especial cuidado a la hora de seleccionar un método de estimación.

Para la estimación además existen varios métodos, a los que se hicieron referencia en el **Capítulo I** del presente trabajo.

Según [CMMI, 2006] los métodos de estimación que propone para software son líneas de código o puntos de función.

La correcta estimación temprana de un proyecto de software es una tarea difícil o casi imposible pues en la misma influyen varios factores:

- No existe un modelo de estimación universal.
- Varía con la etapa de desarrollo en que se encuentra el proyecto.
- Son difíciles de formular, sobre todo al inicio del proyecto.
- Suele hacerse superficialmente.
- Falta de experiencia a la hora de realizar la estimación.
- El estimador tiende a reducir en alguna medida sus estimaciones para hacer más aceptable su oferta.

Por lo que se propone que sea el EQ y el planificador quién decida qué método seleccionar.

La selección del método debe estar bien documentada para garantizar una interpretación común sobre cualquier decisión sobre cualquier hipótesis de partida. [CMMI, 2006].

Por lo que en este trabajo se propone agregar un acápite en la plantilla que se creó durante la creación de la EDT, donde quede reflejado la justificación del método utilizado, las estimaciones de esfuerzo del proyecto y las estimaciones de costo, además se deben recoger los modelos o datos históricos que se utilizaron, y las necesidades de infraestructura de apoyo que no son más que los recursos necesarios de desarrollo y perspectiva de logística para el producto. Dichas necesidades se refiere al entorno de desarrollo, la prueba de medio ambiente, el entorno de producción, el ambiente objetivo, o cualquier combinación adecuada de estos valores al estimar el esfuerzo y los costes.

Una estimación de manera general no garantizaría el éxito esperado, es por esto que se propone realizar las estimaciones de abajo hacia arriba, guiándose por el EDT creado con anterioridad. Una de las ventajas del EDT se pone de manifiesto en este momento del proceso. No significaría lo mismo realizar una estimación de una actividad más general que de una más específica.

El término *paquete de trabajo* se utiliza en este punto del desarrollo. Cada uno es un componente que debe conseguirse. La exactitud es algo que debe tenerse en cuenta aunque no es siempre posible o deseada. Es por esto que se propone llevarlo a cabo de la siguiente forma, *ver tabla 1*:

	<i>Tiempo</i>	<i>Recursos</i>	<i>Coste</i>
Nivel 1			
Nivel 1.1			
Nivel 1.1.1			
Nivel 1.1.1.1 (paquete de trabajo)	20 horas	1hombre, 1PC	\$ 300.00
Nivel 1.1.1.2 (paquete de trabajo)	15 horas	1hombre, 1PC	\$ 250.00

Tabla 1: Forma para realizar las estimaciones. (Estos datos son hipotéticos).



Este paso deberá ser realizado por el **Planificador** y revisado por el **EQ**. Estas estimaciones deben ser agregadas al documento que se generó durante la creación del EDT.

2.5.5 Identificación de riesgos

La identificación de los riesgos es un intento sistemático para especificar las amenazas al Plan del Proyecto (estimaciones, planificación temporal, carga de recursos, entre otros). Identificando los

riesgos conocidos y predecibles, el planificador da un paso adelante para evitarlos cuando sea posible y controlarlos cuando sea necesario.

CMMI plantea que el identificar y analizar los riesgos del proyecto es una tarea que se debe extender a todos los planes que los mismos puedan afectar en caso de materializarse.

Durante la realización de esta actividad se busca como objetivo identificar, analizar y priorizar los riesgos del proyecto.

Basadas en la investigación realizada y lo que plantea CMMI se concluye que deberán tenerse en cuenta dos fases:

- Identificar los riesgos
- Analizar y priorizar los riesgos

2.5.5.1 Identificar los riesgos

Identificar los riesgos es un proceso que debe hacerse en conjunto pues se necesitan todas las ideas que puedan aparecer en este sentido. Con esto se persigue realizar un inventario de los posibles riesgos que puedan aparecer.

Cuando se identifican los riesgos debe considerarse que puede salir mal durante el desarrollo del proyecto, aunque no debe dársele un enfoque pesimista, porque si no todo parecería que irá mal. Se buscan los sucesos que atentarían contra el éxito en una entrega adecuada y con todos los requisitos deseados por el cliente.

La utilización de Tormentas de Ideas es una técnica que se considera una de las formas más factibles para realizar la identificación. Una vez reunido el Equipo Núcleo cada miembro deberá proponer lo que considera un riesgo sin importar cuan superficiales parezcan o si están repetidas y, aclarando siempre que el proponer no significará que le dará solución, pues se plantearían riesgos triviales o en el peor de los casos, no se propondrían.

Según [NAVARRO, 2000] en un artículo que publicó sobre la Gestión de Riesgo, planteó un lista que incluye Boehm en su libro Análisis y Gestión de Riesgo, de los 10 riesgos que para él son significativos y a su lado las técnica que se puede llevar a cabo para cada uno ellos.

Elementos de riesgo	Técnica de gestión de riesgo
1. Deficiencia del personal	Contratar personal con talento, asignación de trabajos, construcción de equipos, acuerdos entre el personal clave, formación cruzada.
2. Planificaciones y presupuestos poco realistas	Estimación multifuente detallada de costes y planificación, diseñar en función del coste, desarrollo incremental, reutilización del software, fregado de requisito
3. Desarrollo de las funciones y propiedades erróneos	Análisis de organización, análisis de la misión, revisiones del usuario y participación del usuario, prototipado, manuales de usuarios preliminares, formulación de operaciones – conceptos, análisis de rendimientos sin nombre, análisis de calidad – factor.
4. Desarrollo erróneo del interfaz de usuario.	Prototipado, escenarios, análisis de tareas, participación del usuario.
5. Chapado	Fregado de requisitos, prototipado, análisis de costes – beneficios, diseñar en función del coste.
6. Continúa corriente de los cambios en los requisitos.	Umbral de alto cambio, ocultación de información, desarrollo incremental.
7. Deficiencias en componentes proporcionados externamente.	Análisis técnicos, análisis coste – beneficio, prototipado, comprobaciones por referencias
8. Deficiencias en tareas desarrolladas externamente	Benchamarking, inspecciones, comprobaciones por referencia, análisis de la competitividad.
9. Deficiencias en rendimiento en tiempo real.	Comprobaciones por referencias, auditorias ante de los incentivos, contratos con incentivos, diseño o prototipado competitivo, construcción de equipos.
10. Exprimir las capacidades informáticas	Simulación, Benchamarking, modelado, prototipazo, instrumentación, ajuste.

Figura 18: Lista de los 10 riesgos más frecuentes.

Antes de continuar con la explicación de lo que sería una segunda fase deben tenerse en cuenta los tipos de riesgos para hacer más fácil la identificación.

2.5.5.2 Tipos de riesgos

Según la versión extendida de RUP los riesgos se clasifican en directos e indirectos. Es importante distinguir entre riesgos directos e indirectos. Sencillamente, sobre un riesgo directo tenemos un cierto grado de control; los riesgos indirectos son los que no podemos controlar.

El RUP en su versión extendida propone algunas preguntas según los tipos de riesgos que existen para tenerlas en cuenta una vez que se decida identificarlos, ver tabla 2.

Categoría	Tipo	Interrogante
Riesgos de Recursos	<i>Organización</i>	<ul style="list-style-type: none"> ✓ ¿Hay suficiente compromiso para este proyecto (incluyendo: la Administración, los proveedores y otros partidos externos implicados)? ✓ ¿Es este el proyecto más grande que ha procurado siempre la organización? ✓ ¿Se tiene un proceso bien definido para la Ingeniería de Software? ✓ ¿Se realiza una buena Gestión y captura de los requisitos?
	<i>Financiero</i>	<ul style="list-style-type: none"> ✓ ¿Está definido de financiamiento completo para el proyecto? ✓ ¿Se ha asignado el financiamiento, para el aprendizaje y la tutoría? ✓ ¿Hay limitaciones con el presupuesto, por lo cual el sistema deba ser entregado con costo fijo o estar de acuerdo con la cancelación?
	<i>Tiempo</i>	<ul style="list-style-type: none"> ✓ ¿Es real el tiempo? ✓ ¿Es crítica la fecha de liberación?
	<i>Persona</i>	<ul style="list-style-type: none"> ✓ ¿El personal disponible es suficiente? ✓ ¿Tienen las habilidades y experiencia apropiada? ✓ ¿Es algún momento han trabajado junto? ✓ ¿Creen que el proyecto pueda tener éxito? ✓ ¿Los representantes por parte de los usuarios están disponibles para las revisiones? ✓ ¿Son expertos en el dominio disponible?
	<i>Riesgos de Negocio</i>	<ul style="list-style-type: none"> ✓ ¿Es mayor el valor del sistema proyectado, que el costo que se estimó? ✓ ¿Qué si los contratos no se pueden hacer con los surtidores principales? ✓ ¿Qué pasa si un competidor puede llegar alcanzar rápidamente el mercado?
Riesgos Técnicos	<i>Alcance del Riesgo</i>	<ul style="list-style-type: none"> ✓ ¿Existen acuerdos para medir el éxito? ✓ ¿Las escalas del tiempo de desarrollo del proyecto son cortas o inflexibles?
	<i>Riesgos Tecnológicos</i>	<ul style="list-style-type: none"> ✓ ¿Se ha probado la Tecnología? ✓ ¿Son razonables los objetivos de la

		reutilización? ✓ ¿Son razonables los volúmenes de transacción? ✓ ¿Son creíbles las estimaciones de las tarifas de la transacción? ✓ ¿Son demasiado optimista? ✓ ¿Existen dependencias externas se interfaces a otros sistemas, incluyendo alguno fuera de la empresa? ¿Las interfaces existen o deben ser creadas? ✓ ¿Existen requisitos disponibles y seguros, los cuales son extremadamente inflexibles (por ejemplo: El sistema nunca debe fallar)?
--	--	---

Tabla 2: Tipos de riesgos según RUP.

Riesgos de planificación

La experiencia muestra que el 85% de los riesgos tiene un impacto directo o indirecto en la planificación y, por lo tanto, implícitamente, en el coste. Quizás un 5% tiene sólo impacto en el coste. El resto no tiene impacto directo en el coste o en la planificación, sino en la calidad, por ejemplo.

Si la hora límite es desfavorable, enfóquela calmadamente con entregas incrementales. Evite una entrega masiva en un intento de cumplir la planificación.

Algunos proyectos tienen horas límite realmente extremas. En general, realice el compromiso de planificación igual que la mejor estimación, con una contingencia razonable.

2.5.5.3 Analizar y priorizar los riesgos

Una vez que se ha obtenido una lista de los riesgos identificados se deberán combinar los riesgos parecidos con el objetivo de reducir la lista y crear un rango de riesgo según su impacto en el proyecto.

Estas actividades se llevarán a cabo de una forma simple. Es muy posible que en una revisión de la lista obtenida aparezcan riesgos que pueden agruparse de forma natural, dígame con esto que entran en el mismo rango, lo que podría significar que es un riesgo fundamental, y deberán combinarse para eliminar duplicados en caso necesario u organizarlos debajo del riesgo más básico muy parecido al formato indentado.

Para priorizar los riesgos se propone organizarlos en forma decreciente de acuerdo a la *exposición* que representa el riesgo para el proyecto.

Para determinar la exposición del riesgo se deberán tener en cuenta:

- *Impacto del riesgo*: Las desviaciones en cuanto a planificación, esfuerzo o costes si se produce el riesgo (expresada en porcentaje).
- *Probabilidad de aparición*: La probabilidad de que el riesgo ocurra realmente (normalmente expresada en forma de porcentaje).
- *Exposición del riesgo*: El tiempo que va a estar el riesgo en el proyecto. Impacto * probabilidad de aparición

Según [Lamas, 2007] el impacto y la probabilidad pueden clasificarse de la siguiente forma, ver *tabla 3* y *4*, respectivamente:

Valor del impacto	Descripción
Catastrófico	Pérdida del sistema. Costo mayor del 50%
Crítico	Recuperación de la capacidad operativa. Costo mayor del 10% y menor del 20%
Tolerable	Capacidad operativa mermada. Costo mayor del 10% y menor del 20%

Tabla 3: Clasificación de los riesgos según su impacto.

Valor de la probabilidad	Descripción
Muy baja	Menor del 10 %
Baja	Del 10 % al 22 %
Moderada	Del 25 % al 75 %
Alta	Del 75 % al 90 %
Muy alta	Mayor del 90 %

Tabla 4: Clasificación de los riesgos según su probabilidad.

Aunque la *exposición* puede determinarla una sola persona lo mejor sería que lo hiciese el Equipo Núcleo para llegar a consensos generales.

De forma general la lista no debe exceder de 15 riesgos pues es muy posible que los riesgos no sean realmente importantes.

Una vez obtenida la *exposición* se debe pasar a organizarlos de forma decreciente para crear una lista de los 10 primeros riesgos. Aunque se recomienda agruparlos en categorías de acuerdo al tiempo que pueden mantenerse en el proyecto. Generalmente con las siguientes categorías deberá ser suficiente:

- Improbable (menor de 30 días)
- Corto plazo (30 días)
- Mediano plazo (De 1 a 4 meses)
- Largo plazo (Más de 4 meses)



Este paso deberá ser realizado por el **EQ** y dirigido por el **Planificador**. Terminado el proceso se generará la plantilla **Lista de riesgos** contenida en el Expediente del Proyecto que deberá ser realizada por el **Planificador**.

2.5.6 Elaborar Plan de Proyecto

El Plan de Proyecto es un artefacto global y compuesto que recopila toda la información necesaria para gestionar el proyecto. Incluye una serie de artefactos desarrollados durante la fase inicial y se mantiene a lo largo del proyecto.

En la UCI este plan es conocido como Plan de Desarrollo de Software.

Según RUP en su versión extendida el objetivo del plan es recopilar toda la información necesaria para controlar el proyecto. Describe el enfoque del desarrollo del software, y es el plan de nivel superior generado y utilizado por los gestores para dirigir el esfuerzo de desarrollo. El *planificador* lo utiliza para realizar la planificación de proyectos y las necesidades de recursos, y para supervisarlo contra la planificación y el resto de los *miembros del equipo del proyecto*, para comprender qué deben hacer, cuándo deben hacerlo y de qué otras actividades dependen.

Si el plan no evoluciona no se considera un buen plan. Un plan de desarrollo de software útil se actualiza periódicamente (no es un software inservible ni estancado), y lo comprenden y aceptan los directivos y los usuarios.

El plan de desarrollo de software es el documento definitorio del proceso del proyecto. Se debe preparar un único plan de desarrollo de software que:

- cumpla con los estándares organizativos sobre contenido
- cumpla con el contrato (si existe)
- proporcione rastreabilidad, o cláusulas de renuncia, de los requisitos de la organización y el contrato
- se actualice en cada objetivo importante
- evolucione junto con el diseño y los requisitos

El plan hace referencia a las siguientes plantillas:

- Plan de Capacitación.
- Documento Visión.
- Roles y responsabilidades.
- Gestión de la Configuración.
- Aseguramiento de la calidad.
- Plan de aceptación del proyecto.
- Plan de Gestión de requisitos.
- Plan de Mediciones.
- Plan de Mitigación.

Dentro de este paso del procedimiento se ha situado la planificación de los costos, recursos, tiempo y riesgos, porque, de una forma u otra, todo el análisis y los resultados obtenidos una vez desarrollada, tributarían al Plan.

Según [CMMI, 2006] el Plan de Proyecto es creado y mantenido como la base para la gestión de un proyecto. El mismo se basa en los requisitos y las estimaciones establecidas del proyecto, incluyendo todas las fases del ciclo de vida del mismo.

Durante la planificación se debe velar que todos los planes que afectan el plan estén en equilibrio con el Plan de Proyecto.

Las prácticas que propone CMMI para la elaboración del Plan son:

1. Elaborar el presupuesto y el calendario del proyecto.
2. Identificar los riesgos del proyecto.

Esta tarea según CMMI debe realizarse durante la elaboración del Plan de Proyecto. Se tomó la identificación de riesgo como un paso anterior a la elaboración de este plan, por la importancia que se le confieren a los riesgos a la hora de decidir si el proyecto es factible o no.

3. Elaborar el Plan de Gestión de Datos.
4. Elaborar el Plan de Conocimientos y Habilidades.

Esta tarea se abordará dentro de la explicación de la planificación de recursos humanos, por la vinculación que tiene el mismo con este paso.

5. Elaborar el Plan de Participación de los Interesados.
6. Establecer el Plan de Proyecto.

Estas prácticas son explicadas en los distintos acápites del procedimiento que se refieren a ellas.



Se generará la plantilla **Plan de desarrollo de software** contenido en el Expediente del Proyecto que deberá ser realizada por el **Planificador**.

2.5.6.1 Planificación de Recursos Humanos (RR.HH.)

CMMI propone que la selección del personal para un equipo de trabajo depende fundamentalmente de la descomposición del proyecto y las necesidades de realización de las tareas, funciones y responsabilidades, para esto pueden basarse en los paquetes de trabajo del EDT.

El planificador es quién deberá coordinar todo lo necesario para encontrar las personas más adecuadas para realizar los trabajos y en qué momento serán utilizadas. Para esto se necesitan identificar las habilidades que posee cada uno. El objetivo en este punto es definir los roles y responsabilidades.

En proyectos de gran escala y que nada tienen que ver con las características de la UCI, el personal, en la mayoría de las ocasiones, se contrata por iteraciones, lo que a veces se convierte en un riesgo.

En el caso de la Universidad, los proyectos una vez concebidos, se comienza a captar personal para realizar el trabajo durante todo el ciclo de vida del proyecto. Es válido aclarar que el personal que será captado está compuesto en su mayoría por estudiantes y pocos profesores. Desde un inicio y luego de analizado de forma general, qué se necesitará en el proyecto, datos que se deben recoger en el Plan de Conocimientos y Habilidades que se especificará más adelante, comienza la selección, margen que da lugar y espacio para que se implementen y pongan en práctica Planes de Capacitación en caso que se necesite. Con esto no se quiere decir, que tal vez no sea necesario en un momento determinado seleccionar más personal, pero generalmente se busca que estos estén insertados desde un inicio, aunque no trabajen sino hasta un poco después, con el objetivo de que se sientan identificados con el progreso del proyecto.

De forma general, ya para estos momentos se debe tener un número aproximado de las personas que se necesitan, el cual debe estar en concordancia con el tamaño y alcance del proyecto. Para mayor organización se pueden crear equipos o grupos, que se recomiendan estén organizados por roles en caso de que el proyecto no posea módulos y por intereses en caso de que si los posea. Con esta sugerencia se busca lograr una afinidad que contribuya con la confianza y la realización del trabajo cómodamente.

Basadas en la investigación realizada se define la organización del personal del proyecto de la siguiente forma:

- *Jefe de proyecto*: se destaca como la figura clave en la ejecución y control del proyecto y es el motor que ha de impulsar el avance del mismo mediante la toma de decisiones tendentes a la consecución de los objetivos.
- *Planificador*: es el encargado de planear, gestionar y asignar recursos, establece prioridades, coordina interacciones con clientes y usuarios y mantiene centrado al equipo del proyecto.
- *Equipo Núcleo*: es responsable de las actividades de dirección de proyectos, tales como la planificación, el control y el cierre. Su selección está definida en el epígrafe 2.51.
- *Equipo de trabajo*: Será seleccionado en este punto del procedimiento, y a sus miembros se les asignará roles y responsabilidades.

Además se propone tener en cuenta las siguientes fases para la planificación de RR.HH:

- Elaboración del Plan de Conocimientos y Habilidades.
- Técnicas para la selección del equipo de trabajo
- Roles que se definirán

Elaboración del Plan de Conocimientos y Habilidades

Este Plan contiene los conocimientos teóricos y prácticos necesarios para llevar a cabo un proyecto. Implica tanto la formación del personal como la adquisición de conocimientos de fuentes externas.

Según [CMMI, 2006] se proponen las siguientes prácticas:

1. Identificar los conocimientos teóricos y prácticos.
2. Evaluar los conocimientos y aptitudes disponibles.
3. Seleccionar los mecanismos necesarios para proporcionar los conocimientos identificados.
4. Incorporar mecanismos seleccionados en el Plan de Proyecto.



Adicionar el **Plan de Conocimientos y Habilidades** al acápite de Planes Adicionales contenido en el Plan de Proyecto. Y tendrá como responsable el **Planificador**.

Técnicas para la selección del equipo de trabajo

Dentro la Gestión de RR.HH que plantea [Proyecto, 2004] está el reclutamiento y la selección del personal que tiene como objetivo abastecer el proceso de selección de los candidatos más capacitados.

El *reclutamiento* no es más que la búsqueda de los candidatos para los puestos disponibles que existen, con el fin de escoger entre estos los más idóneos.

La *selección del personal* ya es un proceso que debe hacerse con un poco más de cuidado, pero que no significa que sea difícil. Esto tiene como objetivo según [Baudes, 1996] descubrir, escoger y colocar en cada momento al hombre adecuado en el lugar de trabajo adecuado. La selección es un proceso complejo que consta de distintos pasos, en cada uno de los cuales los candidatos, originalmente

reclutados, son sometidos a distintas pruebas. Los que no superan las pruebas de una fase son rechazados y los que las superan pasan a la fase siguiente, y así sucesivamente hasta llegar a la elección final de un número reducido de candidatos que han pasado satisfactoriamente todas las pruebas.

Las técnicas utilizadas para los procesos de selección de personal más conocidas son: la técnica de la entrevista de incidentes críticos; la entrevista, que es la más utilizada y conocida; el panel de expertos y la tormenta de ideas.

En el caso de la UCI la generalidad es que inicialmente se lance una convocatoria solicitando personal para un proyecto determinado y se citan los interesados. Una vez reunidos se les aplican pruebas de conocimientos, test de habilidades, encuestas y/o entrevistas. Dada esta situación se concluye que las técnicas que más se adaptan a la condiciones de la universidad es una combinación de la técnica del panel de experto y la entrevista. La técnica del panel de experto posibilita determinar cuales son las competencias que realmente permiten a los individuos un desempeño superior, y luego seleccionar a los estudiantes de acuerdo a las competencias previamente definidas. La entrevista es la técnica más utilizada en Cuba y el mundo, el tipo de entrevista que se realizará será la entrevista mixta de una forma breve y dinámica, con preguntas estructurales y con preguntas no estructurales, que permite obtener información sobre las características personales y disposición del estudiante a ingresar al proyecto.

¿En que consisten estas técnicas?

Entrevista

La entrevista de selección consiste en una plática formal y en profundidad, conducida para evaluar la idoneidad para el puesto que tenga el solicitante. El entrevistador se fija como objeto responder a dos preguntas generales: ¿Puede el candidato desempeñar el puesto?, ¿Cómo se compara con respecto a otras personas que han solicitado el puesto?

Las entrevistas se llevan a cabo entre un solo representante del proyecto y un solicitante (entrevistado). Las preguntas que formule el entrevistador pueden ser estructuradas, no estructuradas, mixtas, de solución de problemas o de provocación de tensión.

En la práctica la estructura mixta es la más empleada, aunque cada una de las otras desempeña una función importante.

- Entrevistas mixtas

En la práctica, los entrevistadores despliegan una estrategia mixta, con preguntas estructurales y con preguntas no estructurales. La parte estructural proporciona una base informativa que permite las comparaciones entre candidatos. La parte no estructurada añade interés al proceso y permite un conocimiento inicial de las características específicas del solicitante.

Panel de expertos

En esta técnica participan un grupo de individuos, quienes deben ser buenos conocedores de las funciones y de las actividades que en general deben acometerse en un determinado puesto. Estos individuos tienen la tarea de determinar, atendiendo a estos factores, cuáles son las competencias que realmente permiten a los individuos un desempeño superior.

El grupo de expertos se selecciona utilizando los algunos criterios como: años de experiencia, calificación profesional en la actividad que desempeña, conocimiento profundo del puesto de trabajo a analizar, entre otros.

En el caso específico de la universidad deberá tenerse en cuenta para la selección del personal el nivel de capacitación que ha obtenido el aspirante durante el tiempo que lleva como estudiante en caso de serlo o como profesor o especialista de una empresa vinculado a la UCI. Dígase como nivel de capacitación, en el caso de los estudiantes, los cursos optativos que posee en su currículum y su nivel de preparación.

Roles que se definirán

No se puede definir un estándar de cuántos y qué roles utilizarse en un proyecto específico, pues eso depende de la magnitud, las características propias del mismo y la metodología utilizada sin embargo se puede puntualizar que un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas. Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el dueño de un conjunto de

artefactos.

Para definirlos deben tenerse en cuenta los siguientes aspectos:

Rol: Nombre bien definido del rol a ejecutar.

Nivel: Nivel al que pertenece el rol de acuerdo a la complejidad de sus tareas.

Descripción: Descripción detallada del rol, definición, tareas específicas y generales, así como consideraciones a tener en cuenta.

Conocimientos mínimos: Conocimientos imprescindibles que se deben dominar para realizar el trabajo, cursos optativos y asignaturas que no deben faltar para ejecutar el rol.

Conocimientos óptimos: Conocimientos para tener un mejor rendimiento en la ejecución de un rol determinado.

Habilidades: Habilidades que deben desarrollar para realizar un trabajo con calidad, tanto en lo individual como en el trabajo en equipo.

Estas son algunas consideraciones que no pueden pasarse por alto:

- ✓ De cada rol se definen las competencias que debe tener cada estudiante, así como el nivel que obtendrá como resultado de su trabajo satisfactorio.
- ✓ No todos los roles son necesarios durante todo el proyecto ni en todos los proyectos. En función del ciclo de vida empleado y de las actividades a realizar, se pueden determinar los roles requeridos.
- ✓ En dependencia de la disponibilidad de personal y de las dimensiones del proyecto a desarrollar, una misma persona puede jugar varios de los roles que se proponen.



Terminado el proceso se generará la plantilla **Roles y Responsabilidades** y de ser necesaria una capacitación para el personal se realizara la plantilla **Plan de Capacitación**, ambas contenidas en el Expediente del Proyecto y deberán ser realizadas por el **Planificador**.

2.5.6.2 Planificación de recursos de software reutilizables

Los componentes de software reutilizables, permiten reducir el coste del desarrollo y agilizar la entrega del producto.

Gran parte de la reutilización se debe a la tecnología de componentes. De los mismos existen cuatro tipos:

- *Componentes ya desarrollados*: se refiere a los componentes adquiridos o existentes, los validados y los listos para utilizarse.
- *Componentes ya experimentados*: son las Especificaciones, diseños, códigos o datos de prueba ya existentes desarrollados anteriormente y similares a los requeridos. La utilización de estos garantiza un riesgo bajo ante las modificaciones por parte de los miembros del equipo con experiencia completa en el dominio de aplicación de los componentes.
- *Componentes con experiencia parcial*: son las especificaciones, diseños, códigos o datos de prueba ya existentes desarrollados anteriormente y relacionados con los requeridos. Los mismos requieren una modificación sustancial y una experiencia limitada del equipo en el dominio de aplicación de los componentes, por lo conlleva a un alto riesgo ante las modificaciones.
- *Componentes nuevos*: son los componentes a construir.

No existe un método o pasos para decidir qué componentes serían los más adecuados reutilizar puesto que esto depende del proyecto, pero se recomienda no pasar esto por alto, pues en proyectos compuestos por módulos generalmente un módulo clasificaría como componente reutilizable de otro.

2.5.6.3 Planificación de recursos de entorno

El entorno es donde se apoya el proyecto de software, el cual incorpora hardware y software. La realización de una buena planificación, del hardware y del software, se necesita para el desarrollo del producto y verifica que estos recursos estén disponibles.

En el caso particular de la UCI cuando se inicia un proyecto se le es asignado por parte de la facultad o las partes interesadas uno o varios laboratorios, depende de la magnitud del mismo, para desarrollar el software. Ya esto sería los recursos de entorno respecto al hardware, que incluiría las computadoras u

otros dispositivos necesarios. Se debe tener en cuenta que tal vez se necesiten más computadoras que las asignadas, servidores u otros dispositivos que no califiquen en la asignación. Se recomienda entonces realizar un levantamiento de lo necesario y compararlo con lo que existe en ese momento. Esto servirá una vez que se haga un análisis del costo más a profundidad.

Con el software sucede otra situación. Puede darse el caso de que el cliente sea el que pida trabajar con determinado software, o que pida realizar un estudio para el luego decidir, o que sencillamente, se limite a aceptar la decisión que ha tomado el proyecto para trabajar con una selección muy particular del mismo. Como quiera que ocurra, no se propone un método específico para realizar selecciones, solo se recomienda que una vez se haya decidido realizarla, se justifiquen y valoren todas las opciones posibles.



Terminado el proceso se generará la plantilla **Ambiente de desarrollo** contenida en el Expediente del Proyecto y deberá ser realizada por el **Planificador**.

2.5.6.4 Planificación temporal

Con la planificación temporal se determina la duración del tiempo de terminación del proyecto, clarifica las relaciones entre los paquetes de trabajo y ayuda a manejar el aspecto tiempo.

Se debe tener en cuenta que la planificación del tiempo, costos y la de recursos no necesariamente debe ser en ese orden, pues eso depende de la decisión de la dirección del proyecto, pero si deben ser planificaciones que se lleven a cabo lo bastante cerca posibles porque son interdependientes.

Esta planificación se basa en la complejidad y dependencia de las tareas. Además se debe tener en cuenta los riesgos para prever en caso necesario y de forma flexible las fechas de realización de las tareas, garantizando obtener una comprensión común de los que se espera, una mejor visión del estado del proyecto para lograr mayor precisión en la identificación de tareas.

Para esto [CMMI, 2006] plantea que se debe llevar a cabo lo siguiente:

1. Determinar los principales hitos del proyecto.
2. Identificar un calendario inicial.
3. Señalar las limitaciones.

4. Identificar dependencias entre las tareas.
5. Definir un calendario más elaborado.
6. Establecer criterios de acción correctiva en caso que sea necesario.

Existen muchas técnicas para la planificación temporal. Algunas de las más comunes son: diagramas de red, de Gantt, calendarios del proyecto y diagramas de hitos. Todas sirven para controlar el tiempo de trabajo durante la ejecución del proyecto. No se propone ninguna de estas técnicas como la más adecuada, pues cada una de ellas cumple un propósito en momentos determinados, por lo que se sugiere utilizarlas en combinación según las necesidades existentes.

Técnicas para la planificación temporal

Diagramas de red (ver Figura 19): Su rasgo esencial es la relación lógica entre los paquetes de trabajo. Se pueden demostrar como las relaciones entre las actividades afectan la secuencia y pueden o no atrasar el tiempo de duración de un proyecto. Se puede determinar la demora y la trayectoria crítica. Por su forma gráfica les es más fácil a todos los que lo utilicen llevarse una impresión general de la situación del proyecto. Se utiliza en el manejo del funcionamiento y el orden del trabajo en los paquetes del EDT.

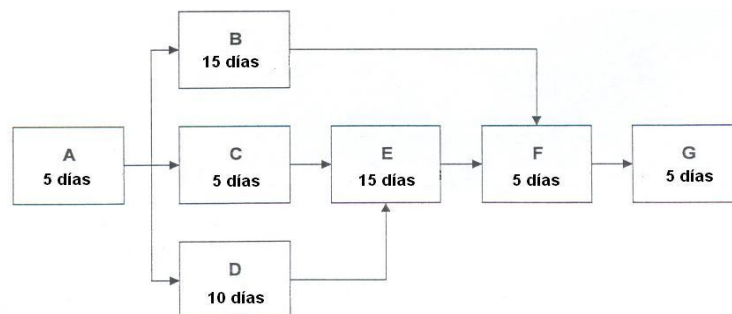


Figura 19: Diagrama de red.

Diagramas de Gantt (ver Figura 20): trazan actividades o grupos de actividades con barras horizontales a través de una escala de tiempo. Muestran las fechas del comienzo de la actividad, las flechas del final y las duraciones. Expresan simplemente cuando se quieren hacer las actividades. Es útil para la divulgación y las decisiones de alto nivel.

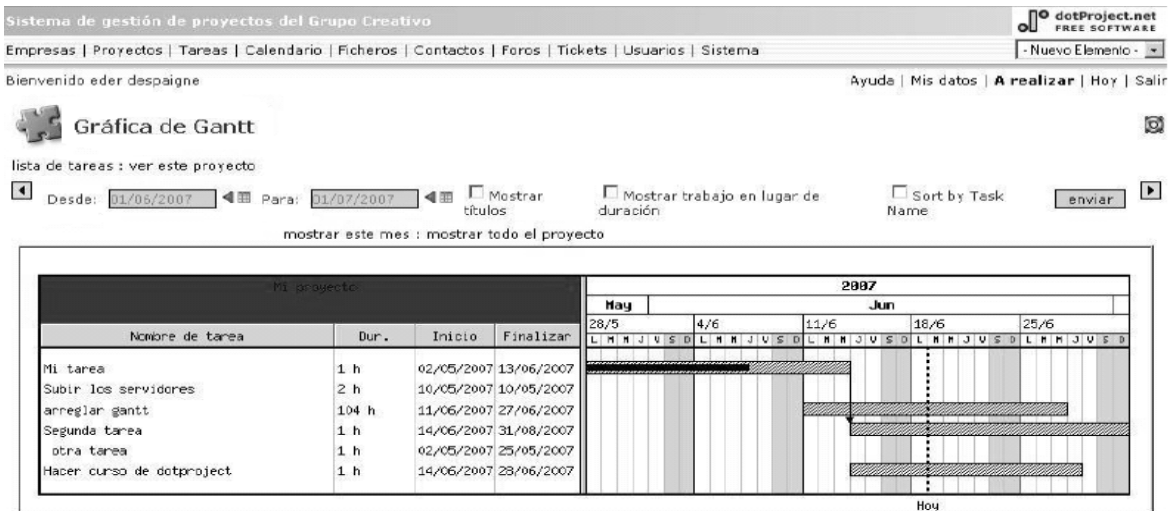


Figura 20: Diagrama de Gantt.

Calendarios de proyecto (ver Figura 21): Se usa para centrarse en la actividad diaria, pues es un calendario que permite ver los aspectos específicos de un día determinado. Proporciona una vista limitada de la situación general del trabajo realizado y a realizar.

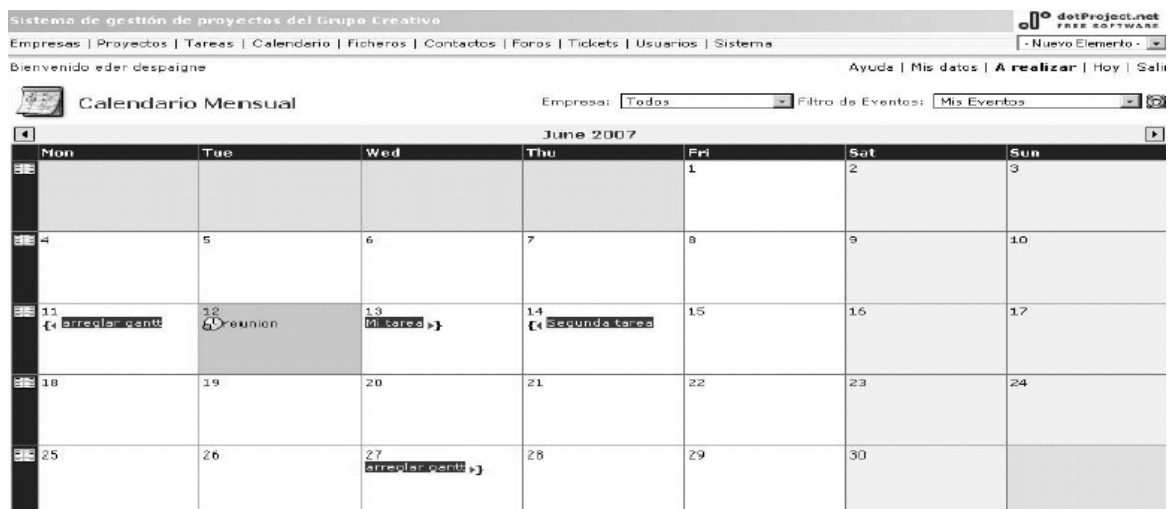


Figura 21: Calendario de proyecto.

Hitos: Son acontecimientos significativos en la terminación de un proyecto, pues están asociados a la terminación de actividades importantes. No tienen duración y no usan recursos. Sirven como recordatorios para comprobar el estado total del proyecto en momentos determinados. Para mayor eficacia deben ponerse durante la realización del diccionario del EDT con expectativas específicas en esos puntos.

2.5.6.5 Planificación de costos

Una vez que se ha planificado la duración de las actividades debe pasarse a la planificación de los costos, pues aunque inicialmente durante el proceso de estimación se realizó un análisis inicial ahora deben considerarse otros aspectos, y mantener el estudio a nivel de paquete de trabajo.

Según [Proyecto, 2004] se deben tener en cuenta lo siguiente para realizar una buena planificación de costos:

- El EDT.
- Requisitos de recursos (los tipos y las cantidades de recursos necesitados para cada paquete del trabajo).
- Tarifas (coste por hora del personal, costo de material, etc.).
- Estimaciones de la duración de la actividad (además del hecho de que los proyectos más largos puedan implicar costes más altos del personal, puede haber cargos derivados de la financiación y de interés bancario a considerar).
- Información histórica (archivos comerciales, de personal, de otras fuentes).
- Plan Contable (asegurarse de que los costes estén asignados a la categoría apropiada según la organización).
- Riesgos (que puedan tener impacto en costes).

Componentes de los costos

Los costos se clasifican de dos formas: directos e indirectos. Este es un aspecto que no debe pasarse por alto por la importancia que tiene una vez que se ha decidido planificar costos tener en cuenta todos los detalles para que no existan pérdidas monetarias o un trabajo mal remunerado.

Los *costos directos* son los costos atribuidos directamente al proyecto. Según [Proyecto, 2004] incluyen típicamente el coste de trabajo, de los materiales, y del equipo que se utilizan en terminar el proyecto. Se incluye el coste de propios empleados de la organización y el coste de trabajo de subcontratados. Otros costes directos incluyen los costos tales como honorarios, viajes, y otros elementos.

Los *costes indirectos* son costes de la organización no directamente atribuibles al proyecto, tal como electricidad y calefacción. Según [Proyecto, 2004] serán incurridos a veces sin importar si un proyecto particular está lanzado o no. Sin embargo, estos costes se deben repartir entre todas las iniciativas de una organización (proyectos incluidos) con una cierta fórmula racional, y deben ser compensados por el suficiente rédito. Típicamente, incluyen los costos generales y administrativos tales como costes de oficinas centrales, beneficios complementarios, y depreciación. La comercialización, las ventas, y la investigación y el desarrollo son otros costos indirectos comunes.

¿Cómo planificar costos?

Existen dos formas de estimar los costos, ambas teniendo en cuenta el EDT. Una es de arriba hacia abajo y la otra es de abajo hacia arriba.

De *arriba hacia abajo* se usa principalmente para dar estimaciones rápidas, es decir calcular un coste general sin tener en cuenta la precisión. Este no es tan factible si se quiere calcular costos exactos o lo más acertados posibles. Tiene como ventaja que no pasa por alto detalles que se pueden olvidar cuando se estima a nivel de paquete de trabajo, de abajo hacia arriba. Este tipo de estimaciones sufren una desviación aproximada de 25 a 50 por ciento.

Una estimación de *abajo hacia arriba* se puede lograr una vez que esté terminado el EDT, pues se hace a nivel de paquete de trabajo. De manera general se refiere a aquellos calcular el coste específico de un paquete y luego ir sumando de abajo hacia arriba. Tiene como ventaja que posee mayor exactitud, y posee una desviación del 5 al 10 por ciento. Si este tipo de estimación no se hace correctamente, por lo general se tendrá que inflar los costes para cada paquete y por ende aumenta el costo total del proyecto, lo que traería consigo, tal vez, que el cliente no esté de acuerdo y no se logre desarrollar el trabajo.

Curvas de costes acumulados

Las curvas de costes acumulados (*ver Figura 22*) son una forma gráfica de mostrar el dinero gastado hasta un momento determinado que se quiera enmarcar. De igual forma que la planificación temporal se puede mostrar gráficamente, la planificación de costos puede mostrarse. Tiene como principal ventaja que son herramientas útiles para informes y que la mayoría de las personas pueden leer fácilmente. Sin embargo debe ponerse especial cuidado a una mala interpretación de los componentes

subyacentes por parte del público a quién se mostrará. Para realizarla se debe tener en cuenta el calendario y los costos estimados con la utilización del EDT.

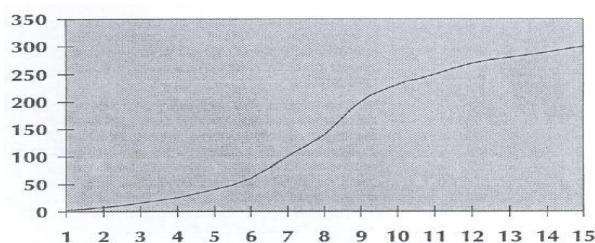


Figura 22: Curvas de costes acumulados.



Terminado el proceso se generará la plantilla **Presupuesto** contenida en el Expediente del Proyecto y deberá ser realizada por el **Planificador**.

2.5.6.6 Planificación de riesgos

Durante la realización de esta actividad se busca como objetivo reorganizar el proyecto para eliminar riesgos, desarrollar planes para reducir el impacto de los riesgos sobre el proyecto y en caso necesario desarrollar planes alternativos. Esta actividad es realizada por el Equipo Núcleo y por el Planificador, quién deberá organizar todo el proceso.

Basadas en la investigación realizada se concluye que se deberán tener en cuenta dos fases:

- Identificar estrategias de mitigación
- Identificar estrategias de contingencia

Identificar estrategias de mitigación

Aunque no siempre es posible, en muchas ocasiones se pueden evitar los riesgos. Por ejemplo, algunas veces los riesgos vienen determinados por un alcance mal definido, lo que en muchas ocasiones con redefinirlo o eliminando requisitos que no son fundamentales se pueden eliminar algunos riesgos. Otro de los riesgos que a veces se pasa por alto es la falta de tiempo para realizar el proyecto, y en muchas ocasiones con mejorar la tecnología es suficiente para igualar o acomodar el tiempo disponible a la fecha de entrega.

Soluciones como las anteriormente expresadas son parte de las estrategias de mitigación que se intentan explicar durante este acápite.

La estrategia a utilizar durante la mitigación depende del riesgo. No existe un plan predeterminado para esto. Es muy probable que riesgos iguales en proyectos diferentes lleven a soluciones diferentes, lo que no se puede encasillar una solución X a un riesgo Y, sin analizar las características propias del proyecto.

Para los riesgos directos se deben identificar que estrategia es la más conveniente seguir para su mitigación. En la mayoría de las ocasiones los riesgos se deben a falta de información por lo que con estudiar más a fondo el tema sería la estrategia ideal para reducir la incertidumbre.

Existen riesgos en los que se puede realizar alguna acción para materializarlos o mitigarlos.

En la tabla 5 se muestra un ejemplo de cómo se pueden planear los riesgos y qué estrategias se pueden realizar. [Fuente, 2006].

Riesgo	Estrategia para la planificación.
Problemas financieros organizacionales.	Preparación de un documento para la dirección de la empresa en la que se muestre la importante contribución del proyecto a los objetivos del negocio.
Retrasos en las entregas.	Comunicación al cliente de posibles dificultades y de la posibilidad de que se retarse en proyecto. Investigación de los componentes que provocaron el retraso.
Enfermedad de algún componente.	Reorganización del trabajo de manera que no haya solapamientos, ni huecos en los trabajos de manera que todos puedan saber sus responsabilidades y conozcan el trabajo de los demás.
Componentes defectuosos.	Reemplazar los componentes defectuosos con otros comprados (si existen) y que tengas una efectividad.

Tabla 5: Riesgos y estrategias para su planificación.

El resultado de estas acciones debe ser reducir la posibilidad de que se produzcan determinados riesgos en un valor próximo a cero. ¿Pero que sucederá si con las estrategias de mitigación no es suficiente? En este caso será necesario desarrollar estrategias de contingencia.

Identificar estrategias de contingencia

Para cada riesgo, tanto si tiene un plan para mitigarlo como si no, debe decidir qué acciones se deben realizar y cuándo. Esto se denomina “plan B” o plan de contingencia. Este plan es necesario cuando la mitigación ha fallado y debe enfrentarse directamente el riesgo. Este es el caso para los riesgos indirectos, para los que el proyecto no tiene ningún control o cuando las estrategias de mitigación son demasiado costosas.

La elaboración del Plan de Contingencia incluye, la descripción del riesgo, los indicadores que muestran que el mismo se ha hecho realidad y las acciones a llevar a cabo, para eliminar los riesgos que están afectando el proyecto en ese momento, *ver tabla 6*.

Riesgo	Indicador	Acción
¿Cuál es el riesgo?	¿Cómo sabrá que el riesgo se ha convertido en una realidad? ¿Cómo se reconoce el 'evento de pérdidas'?	¿Qué debe hacer para solucionar el 'evento de pérdidas' (¿cómo puede detener la "brecha"?)?

Tabla 6: Indicadores y acciones para elaborar el Plan de Contingencia.

Identificar los indicadores del riesgo

Los indicadores pueden identificarse por:

- Ser monitoreados usando métricas de proyecto.
- Ser monitoreados basados en los requerimientos del proyecto y los resultados de las pruebas.
- Ser asociados a eventos específicos.

Identificar las acciones de “pérdida”

En los casos más simples el plan de contingencia enumera soluciones alternativas. El impacto debe ser de costes y retardos para descartar la solución actual e implementar una nueva. Sin embargo para algunos riesgos el plan no supone de una acción sino de un conjunto de varias acciones que garanticen una solución asequible a la situación.

La aparición de una pérdida tiene un lado positivo pues obliga a tomar acciones. De lo contrario puede que dichas pérdidas aparezcan en un momento límite del proyecto y sea el doble de difícil recuperarse.

Para identificar las acciones de pérdida no existe un método predeterminado por el cual guiarse, cada proyecto posee características específicas que deberán ser utilizadas a favor de este sentido para analizar y decidir qué acción será la correcta usar en cada caso. No obstante algunas de estas acciones pueden ser:

- Evitar el riesgo: Se elige una alternativa de menor riesgo.
- Controlar el riesgo: Se decide mitigar el riesgo.
- Asumir el riesgo: Se acepta que el riesgo ocurra.
- Transferir el riesgo: Se reduce el riesgo compartiéndolo.



Terminado el proceso se generará la plantilla **Mitigación de Riesgos** contenida en el Expediente del Proyecto que deberá ser realizada por el **Planificador**.

2.5.6.7 Elaboración del Plan de Gestión de Datos

Los datos de un proyecto incluyen los datos de estudios desarrollados y utilizados exclusivamente de uno o varios equipos de trabajo de acuerdo a la organización del proyecto.

Estos datos pueden ser, según lo planteado en [CMMI, 2006] documentos necesarios para apoyar las diferentes áreas de un proyecto (ingeniería, gestión de la configuración, calidad, seguridad, entre otras).

Los datos pueden adoptar diferentes formas: informes, manuales, esquemas, descripciones, etcétera, y pueden existir en formato duro y digital.

En caso de que algunos de estos datos sean de carácter confidencial, se propone que el Jefe de Proyecto sea el encargado de determinar los datos del proyecto para identificarlos, recogerlos y distribuirlos. Además debe ser el responsable también de establecer los requisitos y procedimientos destinados a garantizar la privacidad y seguridad de estos datos, y establecer mecanismos de acceso a ellos.



Terminado el proceso el Plan deberá ser agregado en el acápite **Planes Adicionales del Plan de Proyecto** y será realizado por el **Planificador** y el **Jefe de Proyecto**.

2.5.6.8 Elaboración del Plan de Participación de los Interesados

Este plan se refiere a la participación de los interesados durante todo el ciclo de vida del proyecto. En la UCI solo los proyectos de exportación son los que deben realizar este paso.

Los interesados no necesariamente deben ser los mismos pues pueden variar de acuerdo a la fase del ciclo de vida en que se encuentre el proyecto.

La realización de este plan incluye, según [CMMI, 2006]:

1. Lista de todas las partes interesadas.
2. Justificación de la participación de los interesados.
3. Funciones y responsabilidades de las partes interesadas con respecto a al fase del ciclo de vida en que se encuentra el proyecto.
4. Relaciones de las partes interesadas.
5. Importancia relativa a las partes interesadas para el éxito del proyecto.
6. Recursos necesarios para garantizar la interacción de las partes interesadas.
7. Calendario de la retirada progresiva de las partes interesadas.



Terminado el proceso el Plan deberá ser agregado en el acápite **Planes Adicionales del Plan de Proyecto** y será realizado por el **Planificador** y el **Jefe de Proyecto**.

2.5.7 Planificación de comunicaciones

Las reuniones que se realizan en un proyecto tienen como objetivo principal asegurar la generación, recogida, distribución, almacenamiento, recuperación y destino final de la información del proyecto en tiempo y forma. Las mismas proporcionan los enlaces cruciales entre las personas y la información, necesarios para una comunicación exitosa.

Los Jefes de Proyectos pueden invertir una enorme cantidad de tiempo comunicándose con el equipo del proyecto, interesados y clientes. En proyectos que tenga un equipo de desarrollo pequeño, no será necesario realizar una planificación de reuniones pues por las características de la UCI este equipo se encontrará trabajando en un mismo sitio y pueden interactuar constantemente. De lo contrario, para proyectos, donde existe un gran equipo de trabajo será necesario coordinar encuentros que serán planificados en dependencia de las necesidades del proyecto. Se propone que estos encuentros tengan una mayor frecuencia durante las primeras fases del desarrollo del proyecto, debido a que son en estas donde mayor comunicación debe existir entre los integrantes e interesados del proyecto en caso necesario, aunque el Jefe de Proyecto y el Planificador decidirán estas frecuencias.

Se debe realizar un Plan de Comunicaciones que incluye de forma general procesos como los siguientes:

- *Planificación de las Comunicaciones*: determinar las necesidades de información y comunicaciones entre los interesados de un proyecto.
- *Distribución de la Información*: poner la información necesaria a disposición de los interesados en el proyecto cuando corresponda.
- *Informar el Rendimiento*: recopilar y distribuir información sobre el rendimiento. Esto incluye informes de estado, medición del progreso y proyecciones.

En el proceso de Planificación de la Comunicaciones es donde se define quién necesita qué información, cuándo la necesitará, cómo le será suministrada y por quién. Un factor importante para el éxito de un proyecto es la identificación de las necesidades de información de los interesados y determinar una forma adecuada para satisfacer dichas necesidades.

La mayoría de los proyectos realizan este proceso en las primeras fases del desarrollo del proyecto, aunque es válido aclarar que los resultados de este proceso de planificación se deben revisar regularmente a lo largo del proyecto y siempre que sea necesario para asegurar la continuidad del mismo.

Una vez que se realice una reunión no debe olvidarse generar la plantilla Minuta de Reuniones contenida en el Expediente del Proyecto, por ser esta el documento central de los encuentros realizados y planificados con anterioridad. Para llenar esta plantilla se designará algún miembro del proyecto y no necesariamente el Planificador es el responsable de la realización de la misma.

2.5.8 Replanificación y control

El objetivo principal de esta fase es definir la información y los procesos que se utilizarán para supervisar y controlar el progreso, la calidad y los riesgos del proyecto.

El planificador del proyecto es el responsable de encargarse de los indicadores que afectan al ámbito de trabajo, presupuesto, calidad y riesgos del proyecto. Los indicadores de un proyecto no son más que fragmentos de información sobre el estado del progreso del proyecto respecto al plan de desarrollo de software. El planificador es el encargado de supervisar estos indicadores a medida que avance el proyecto.

La mayoría de estos indicadores son medidas del proyecto calculadas a partir de métricas que el equipo del proyecto notifica de forma periódica.

Después que se definen las métricas y los indicadores del proyecto, el Equipo Núcleo debe definir la frecuencia con qué los miembros del proyecto deben informar su estado (finalización de las tareas, notificar los problemas, etcétera.).

Para mantener un control eficaz del proyecto, el Planificador define valores o condiciones de umbral (o de desencadenante) asociados a cada uno de los indicadores que con anterioridad deben haberse definido. Si estos umbrales son sobrepasados en algún momento del desarrollo del software, el Planificador es el encargado de realizar acciones correctivas con el objetivo de colocar el proyecto nuevamente en la dirección correcta, según la gravedad de la situación, si las acciones que se deben realizar para dar solución al problema, no está dentro de la autoridad del Planificador, el mismo debe informar al Jefe de proyecto y emitir una solicitud de cambio y activar el proceso de control de cambios del proyecto.

2.6 Conclusiones parciales

Con la realización del presente trabajo se espera que el procedimiento propuesto cumpla con las expectativas de la UCI, en cuanto a la producción y organización del software. Facilitando que se pueda llevar a cabo una mejor planificación de los proyectos productivos que la Universidad produce actualmente. Esperando que los pasos explicados en este capítulo, sean entendibles y aplicables.

Capítulo III: Validación del procedimiento

Para la validación y aceptación del la propuesta del proceso de Planificación en los proyectos productivos de la UCI, que se presentó anteriormente en el Capítulo II, se utilizó el Método de Expertos, consiste en la evaluación cuantitativa de criterios que permite determinar si la propuesta analizada es aceptada o no.

Este grupo de expertos se conformó con especialistas que poseen un vasto conocimiento de los temas relacionados con este trabajo, en su mayoría Jefes y planificadores de algunos proyectos. La correcta elección de los expertos propicia obtener resultados con calidad y una opinión grupal con un alto grado de consenso.

En este capítulo se hará la descripción de los pasos utilizados para la selección de los Expertos y los resultados obtenidos de los mismos.

3.1 Proceso de selección de los expertos

Un experto debe ser, como se planteó anteriormente, una persona experimentada, que posee una gran experiencia o habilidad en una actividad, capaz de ofrecer valoraciones conclusivas de un problema en cuestión y hacer recomendaciones al respecto.

Esta selección se realizó atendiendo a las posibilidades reales de participación de los candidatos, pues este grupo de expertos seleccionado son todos profesionales de la UCI que tienen como promedio 3 años de experiencia en el proceso productivo de la universidad. Poseen además, amplios conocimientos en temas relacionados con el proceso a evaluar, estos son:

1. Proceso de Desarrollo de Software.
2. Calidad de Software.
3. Gestión de Proyectos.
4. Planificación de proyectos.

El conocimiento sobre estos temas ha permitido que las opiniones brindadas sean confiables y válidas para el objetivo propuesto.

3.2 Cantidad de Expertos seleccionados

No existe una norma generalizada que determine un número óptimo respecto al número de expertos. Los investigadores de *Rand Corporation*, [Landeta, 1999] indican que es necesario como mínimo de 7 expertos y un máximo de 30.

En este trabajo se decidió contar con un número de 7 expertos, teniendo en cuenta nivel de complejidad y profundidad del contenido.

3.3 Guía para la validación de la propuesta

Para llevar a cabo el desarrollo de la validación se efectuaron un conjunto de pasos, los cuales se detallan a continuación:

- 1) *Elaboración de los criterios de evaluación que fueron utilizado en el desarrollo de la validación y se agruparon por categoría.*

Grupo No 1: Criterios de mérito científico.

1. Valor científico de la propuesta.
2. Calidad de la investigación.
3. Contribución científica de la propuesta.

Grupo No 2: Criterios de implantación.

4. Necesidad de empleo de la propuesta.
5. Posibilidades de aplicación.

Grupo No 3: Criterios de flexibilidad.

6. Adaptabilidad a proyectos productivos.
7. Capacidad de admisión de cambios que impliquen mejoras.

Grupo No 4. Criterios de impacto.

8. Impacto en el área para la cual está destinada la propuesta.

- 9. Organización en el proceso de desarrollo.
- 10. Repercusión en los proyectos productivos.

2) *Determinar el peso relativo de cada grupo de criterios de acuerdo al por ciento que representa cada grupo del total y los intereses a evaluar.*

Grupo No.1..... 30
 Grupo No.2..... 20
 Grupo No.3..... 20
 Grupo No.4..... 30

- 3) *Solicitud a los expertos seleccionados la evaluación de cada uno de los criterios en una escala del 1 al 10, teniendo en cuenta que la suma del valor dado por parte de los expertos a cada criterio de un grupo no exceda del peso relativo asignado a este. Para recoger la información anterior se definieron dos modelos. Ver anexos 7 y 8.*
- 4) *Construcción de una tabla que guarda el resultado del trabajo de los expertos. Ver tabla 7.*

G	C/E	E1	E2	E3	E4	E5	E6	E7	Ep
30	C1	8	9	7	7	10	10	9	8.57
	C2	14	14	15	13	11	13	15	13.57
	C3	8	7	8	10	9	7	6	7.85
20	C4	11	10	8	10	12	9	13	10.42
	C5	9	10	12	10	8	11	7	9.57
20	C6	13	12	10	11	10	9	10	10.71
	C7	7	8	10	9	10	11	10	9.28
30	C8	10	9	8	10	12	10	7	9.42
	C9	10	11	13	10	11	12	10	11
	C10	10	10	9	10	7	8	13	9.57
Total		100	100	100	100	100	100	100	100

Tabla 7. Resultado del trabajo de expertos

- 5) *Verificación de la consistencia en el trabajo de los expertos, para lo que se utiliza el coeficiente de concordancia de Kendall y el estadígrafo Chi cuadrado (X²).*

Para esto se sigue el procedimiento siguiente:

- Sea C el número de criterios que van a evaluarse y (E) el número de expertos que realizan la evaluación.
- Para cada criterio se determina:
 - $\sum E$: Sumatoria del peso dado por cada experto.
 - E_p : Puntuación promedio del peso dado por cada experto.
 - $M\sum E$: media de los $\sum E$.
 - ΔC : Diferencia entre $\sum E$ y $M\sum E$.

- Se determina la desviación de la media, que posteriormente se eleva al cuadrado para obtener la dispersión (S) por la expresión:

$$S = \sum (\sum E - \sum \sum E / C)^2$$

- Conociendo la dispersión se puede calcular el coeficiente de concordancia de Kendall (W):

$$W = S / E^2 (C^3 - C) / 12$$

- El coeficiente de concordancia de Kendall permite calcular el Chi cuadrado real:

$$\chi^2 = E (C-1) W$$

- Los valores obtenidos se muestran en la siguiente tabla 8.

Expertos/Criterios	E ₁	E ₂	E ₃	E ₄	E ₅	E ₆	E ₇	$\sum E$	E_p	ΔC	ΔC^2
C ₁	8	9	7	7	10	10	9	60	8.57	-7	49
C ₂	14	14	15	13	11	13	15	95	13.57	28	784
C ₃	8	7	8	10	9	7	6	55	7.85	-12	144
C ₄	11	10	8	10	12	9	13	73	10.42	6	36
C ₅	9	10	12	10	8	11	7	67	9.57	0	0
C ₆	13	12	10	11	10	9	10	75	10.71	8	64
C ₇	7	8	10	9	10	11	10	65	9.28	-2	4
C ₈	10	9	8	10	12	10	7	66	9.42	-1	1
C ₉	10	11	13	10	11	12	10	77	11	10	100
C ₁₀	10	10	9	10	7	8	13	67	9.57	0	0
DC	100	100	100	100	100	100	100	700	100	30	1182

M Σ E	67
S	1092
W	0.27
X²	17.01
X²_(α, c-1)	21.6660

Tabla 8. Tabla para el cálculo de concordancia de Kendall.

- El Chi cuadrado calculado se compara con el obtenido del las tablas estadísticas y de esta forma se obtiene la siguiente conclusión.

$$X^2_{\text{real}} < X^2_{(\alpha, c-1)}$$

17.01 < 21.6660 Lo que demuestra que existe concordancia en el trabajo de expertos.

- 6) Si no existe concordancia se hace necesario repetir el trabajo de expertos.
- 7) Después de comprobar la consistencia del trabajo de expertos se puede definir el peso relativo de cada criterio (P).
- 8) Conociendo el peso de cada criterio y la calificación dada por los evaluadores en una escala de 1-5 se puede construir la Tabla 9, para obtener el valor de de $P \times c$, donde (c), es el criterio promedio concebido por los expertos.

Criterios	Calificación (c)					P	P x c
	1	2	3	4	5		
C₁				X		0.0875	0.35
C₂					X	0.1357	0.6785
C₃			X			0.0785	0.2355
C₄					X	0.1042	0.521
C₅				X		0.0957	0.3828
C₆				X		0.1071	0.4284
C₇				X		0.0928	0.3712
C₈				X		0.0942	0.3768
C₉					X	0.11	0.55
C₁₀					X	0.0957	0.4785

Tabla 9. Tabla de calificación de cada criterio

- 9) Se calcula el Índice de aceptación del proyecto (IA).

$$IA = \sum (P \times c) / 5$$

$$IA = 1.6971$$

10) Por último se determina la probabilidad de éxito de la propuesta

Rangos predefinidos de Índice de Aceptación

IA > 0,7 Existe alta probabilidad de éxito

0,7 > IA > 0,5 Existe probabilidad media de éxito

0,5 > IA > 0,3 Probabilidad de éxito baja

0,3 > IA Fracaso seguro

Por lo que se puede concluir que la propuesta tiene una alta probabilidad de éxito.

3.4 Conclusiones parciales

El presente capítulo, junto con los resultados de la aplicación del método de Experto para la obtención de un pronóstico confiable de la utilidad práctica y real del procedimiento propuesto, constituyen el punto de partida para una aplicación de la propuesta y aportan, además, elementos de interés para la evolución y mejora del procedimiento.

Con los resultados arrojados de los modelos que los expertos respondieron se puede concluir de forma general que la propuesta tiene una probabilidad alta de éxito.

La propuesta de forma general obtuvo una categoría final, por parte de todos los expertos, como bueno, con resultados destacados, adaptable y con amplias posibilidades de aplicación a los diferentes proyectos productivos en la UCI.

Conclusiones

Se realizó un análisis sobre el proceso de planificación de software y sobre la situación actual por la que atraviesan los proyectos productivos de la Universidad en cuanto a este tema.

A partir de la investigación realizada se logró seleccionar una herramienta libre para dar apoyo a la Gestión de Proyectos.

Se obtuvo una propuesta de un procedimiento para la planificación de software que cumple con el objetivo principal de brindar una serie de pasos que guíen y ayuden a realizar buenas planificaciones en los proyectos productivos en la UCI.

Finalmente se hizo la validación de la propuesta por el Método del Experto arrojando que la probabilidad de éxito es alta, lo que resulta un punto de partida importante para la aplicación del procedimiento en casos reales.

Recomendaciones

Partiendo de los resultados de la investigación efectuada y de la experiencia adquirida durante la realización de este trabajo, y con el propósito de asegurar la posterior ampliación, modificación y mejora del procedimiento propuesto, se exponen a continuación algunas líneas de investigación que se considera importante desarrollar; así como las recomendaciones.

- ✓ Mantener actualizado el procedimiento propuesto, incorporándole las recomendaciones realizadas por los expertos y extender su aplicación a un número más amplio de casos reales.
- ✓ Fomentar un espacio para la asimilación del procedimiento por parte de los líderes de proyecto con el objetivo de entenderlo y luego poder aplicarlo.
- ✓ Aplicar el procedimiento a los proyectos productivos de la UCI para obtener mejoras y modificarlo en caso necesario.

Referencias bibliográficas

- [Baudes, 1996] BUADES, C. R.-L. ADMINISTRACION DE RECURSOS HUMANOS. 3a Edición revisada). ETEA, 1996, [Abril, 2008] 409 p.
- [Canós, 2006] CANÓS, J.; LETELIER, P., Metodologías Ágiles en el Desarrollo de Software. [Portal de la Universidad Politécnica de Valencia]. Valencia. España: [Enero, 2008]. Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>
- [Carlos, 2008] CARLOS, J. *Gestiona tus proyectos con OpenProj*, enero, 2008. [Febrero, 2008]. Disponible en: <http://www.nativos2020.com/2008/01/27/gestiona-tus-proyectos-con-openproj/>
- [CMMI, 2006] CMMI, P. T. CMMI® for Development, Version 1.2. Agosto, 2006. 573 p.
- [Fernández, 2001] FERNÁNDEZ, L. Calidad del Software, mayo, 2001. [Febrero, 2008]. Disponible en: <http://www.ati.es/gt/calidad-software/presentacion.htm>
- [Fernández, 2008] FERNÁNDEZ, L. *OpenProj, planeando proyectos con software libre*, enero, 2008. [Febrero, 2008]. Disponible en: <http://softwarelibre.deusto.es/openproj-planeando-proyectos-con-software-libre/>
- [Freitag, 2008] FREITAG, C. S. Y. K. *Sitio oficial de la herramienta TaskJuggler*, 2008. [Marzo, 2008]. Disponible en: <http://www.taskjuggler.org/>
- [Fuente, 2006] FUENTE, A. A. J. y LOVELLE, J. M. C. Gestión de Riesgo. 2006 [Abril, 2008]. Disponible en: <http://www.di.uniovi.es/~aquilino/Asignaturas/ProyectosInformatica/Documentos/Proyectos.v2006.C7.V2.pdf>.
- [Gracia, 2005] GRACIA, J. *CMM - CMMI Nivel 2*, noviembre, 2005. [Febrero, 2008]. Disponible en: <http://www.ingenierosoftware.com/calidad/cmm-cmmi-nivel-2.php>
- [Jones C, 2004] JONES, C., *Estimating Software Costs*, McGraw Hill, 1998.
- [Lamas, 2007] LAMAS, Y. L. C. Y. T. A. Propuesta para la Gestión de Riesgo en los proyectos productivos de la UCI [Trabajo de diploma]. Ciudad de La Habana, UCI, 2007, [marzo, 2008] 119. p.
- [Landeta, 1999] LANDETA, J. El método Delphi: Una técnica de previsión para la incertidumbre. 1999, [2008].
- [Navarro, 2000] NAVARRO, A. Gestión de Riesgo, 2000 [Abril, 2007]. Disponible en: http://209.85.165.104/search!=cache:9k0ediu_2b0j:www.fld.ucm.es/profesor/jlsierra/lsl/.
- [Navarro, 2005] NAVARRO, A. *Planificación de proyectos de software*, 2005. [2008]. Disponible en: http://www.fdi.ucm.es/profesor/anavarro/5._Planificacion_de_proyectos_de_software.pdf

- [Nova, 2005] NOVA, P. C. *Planificación de Proyectos de Software*, 2005. [2008]. Disponible en:
<http://www.getec.etsit.upm.es/articulos/gproyectos/art4.htm>
- [Pressman, 1998] PRESSMAN, R. S. *Ingeniería del Software. Un Enfoque Práctico*. 1998. p.
- [Proyecto, 2004] PROYECTO, E. D. L. D. *Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK®)*. Tercera edición. 2004. 392 p.
- [Smarciani, 2004] SMARCIANI, *Financiamiento Institucional*, 2004. [2007]. Disponible en:
<http://www.inta.gov.ar/bariloche/desarrollo/gesrural/trabajos/planificacion/Archivos/>
- [Sommerville, 2002] SOMMERVILLE, I.; *Ingeniería de Software*, Addison Wesley, 6ta Edición, 2002
- [Team, 2007] TEAM, A. *Gestión de proyectos con dotProject*, 2007. [Marzo, 2008]. Disponible en:
<http://www.abartiateam.com/dotproject.html>
- [Teruel, 2001] TERUEL, A. *La Planificación de un Proyecto de Desarrollo de Software*, 30 de abril de 2001: 8.

Bibliografía consultada

- ANÓNIMO. *Estimación de proyectos de software*, 2007. [Febrero, 2008]. Disponible en: <http://members.fortunecity.com/patriciob/estimacion.htm>
- . *Sitio oficial de la herramienta GanttProject*, 2007. [Marzo, 2008]. Disponible en: <http://ganttproject.biz/learn.php&prev=/search%3Fq%3DGanttProject%26hl%3Des%26sa%3DX>
- . *Sitio oficial de la herramienta OpenProj*, 2008. [Febrero, 2008]. Disponible en: <http://openproj.org/>
- . *CMMI*, enero, 2002. [Febrero, 2008]. Disponible en: <http://www.proginternet.com.ar/cmml.php>
- . *Modelos de Gestión de la Calidad del Software*, enero, 2008. [Febrero, 2008]. Disponible en: http://modelosdegestiondelcalidad.blogspot.com/2008/01/modelo-cmml_12.html
- . *CMMi y software libre*, Mayo, 2006. [Febrero, 2008]. Disponible en: http://people.warp.es/~xtor/blog/?page_id=127
- . *Que es SCRUM*, noviembre 2006. [Febrero, 2008]. Disponible en: http://www.baufest.com/spanish/scrum/scrumconference2006/Que_es_scrum.pdf
- . *Documentación para las aplicaciones OpenProj, Project -ON-Demand y salesforce POD.*, noviembre, 2007. [Febrero, 2008]. Disponible en: <http://openproj.org/wiki/index.php?title=Spanish>
- BUADES, C. R.-L. *ADMINISTRACION DE RECURSOS HUMANOS*. 3a Edición revisada). ETEA, 1996, [Abril, 2008] 409 p.
- CANÓS, J.; LETELIER, P., *Metodologías Ágiles en el Desarrollo de Software*. [Portal de la Universidad Politécnica de Valencia]. Valencia. España: [Enero, 2008]. Disponible en: <http://www.willydev.net/descargas/prev/TodoAgil.Pdf>
- CARABIAS, J. *OpenProj, alternativa a MS Project*, febrero, 2008. [Febrero, 2008]. Disponible en: <http://tecnobetas.com/openproj-alternativa-a-ms-project/>
- CARLOS, J. *Gestiona tus proyectos con OpenProj*, enero, 2008. [Febrero, 2008]. Disponible en: <http://www.nativos2020.com/2008/01/27/gestiona-tus-proyectos-con-openproj/>
- CISNERO, Y. B. L. Y. Y. V. *Propuesta de un Proceso de Selección de Roles y Personal con sus Niveles de Competencia para Proyectos Multimedia*. [Trabajo de diploma]. Ciudad de La Habana, UCI, Junio, 2007, [Mayo, 2008]116. p.
- CMMI, P. T. *CMMI® for Development, Version 1.2*. Agosto, 2006. 573 p.
- ETEROVIC, Y. *El Proceso de Desarrollo Unificado (RUP): Técnicas Modernas para Desarrollo de Aplicaciones*, octubre, 2001. [Febrero, 2008]. Disponible en:

<http://www2.ing.puc.cl/~iic3194/rupppt.pdf>

FERNÁNDEZ, L. *Calidad del Software*, mayo, 2001. [Febrero, 2008]. Disponible en:
<http://www.ati.es/gt/calidad-software/presentacion.htm>

---. *OpenProj, planeando proyectos con software libre*, enero, 2008. [Febrero, 2008]. Disponible en:
<http://softwarelibre.deusto.es/openproj-planeando-proyectos-con-software-libre/>

FREITAG, C. S. Y. K. *Sitio oficial de la herramienta TaskJuggler*, 2008. [Marzo, 2008]. Disponible en:
<http://www.taskjuggler.org/>

FUENTE, A. A. J. y LOVELLE, J. M. C. *Gestión de Riesgo*. 2006 [Abril, 2008]. Disponible en:
<http://www.di.uniovi.es/~aquilino/Asignaturas/ProyectosInformatica/Documentos/Proyectos.v2006.C7.V2.pdf>.

GELVIS SEQUERA, J. B., KENNY VIVAS. *CMMI. Planificación del Proyecto*, junio, 2005. [Febrero, 2008]. Disponible en:
http://carolina.terna.net/ingsw3/datos/CMMI_PlanificacionProyecto.ppt#258,3,CMMI_y_la_Planificacion_del_Proyecto

GIRALDO, O. P. *Métricas, Estimación y Planificación en Proyectos de Software*, 2004. [2008].
Disponible en: http://www.willydev.net/descargas/willydev_planeasoftware.pdf

GRACIA, J. *CMM - CMMI Nivel 2*, noviembre, 2005. [Febrero, 2008]. Disponible en:
<http://www.ingenierosoftware.com/calidad/cmm-cmmi-nivel-2.php>

---. *Gestión de proyectos con SCRUM*, Septiembre 2006. [Febrero, 2008]. Disponible en:
<http://www.ingenierosoftware.com/equipos/scrum.php>

GRAMAJO, E., GARCÍA-MARTÍNEZ, R., ROSSI, B., CLAVERIE, E. Y BRITOS, P. *Combinación de alternativas para la estimación de proyectos software*, septiembre, 2002. [Febrero, 2008].
Disponible en: <http://www.itba.edu.ar/capis/webcapis/RGMITBA/articulosrgm/R-ITBA-22-estimacion.pdf>

HIGHSMITH, J. A. *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems*. 1999. P

HURTADO, L. L. *Modelos de Estimación*, febrero, 2006. [Febrero, 2008]. Disponible en:
http://www.wikilearning.com/articulo/estimacion_de_proyectos-modelos_de_estimacion/9637-3

JOSÉ H. CANÓS, P. L. Y. M. C. P. *Metodologías Ágiles en el Desarrollo de Software*, abril, 2007.
[Febrero, 2008]. Disponible en:
<http://www.willydev.net/descargas/prev/TodoAgil.Pdf>

LAMAS, Y. L. C. Y. T. A. *Propuesta para la Gestión de Riesgo en los proyectos productivos de la UCI* [Trabajo de diploma]. Ciudad de La Habana, UCI, 2007, [marzo, 2008] 119. p.

- LANDETA, J. El método Delphi: Una técnica de previsión para la incertidumbre. 1999, [2008].
- MARCOS, C. A. *Desarrollo de Software con UML*, marzo, 2007. [Febrero, 2008]. Disponible en: <http://www.exa.unicen.edu.ar/catedras/metodol2/estimacion.pdf>
- MORA, J. I. Z. *RUP en la Gestión de Proyectos de Software*, diciembre, 2007. [Febrero, 2008]. Disponible en: <http://juazammo.blogspot.com/2007/12/rup-en-proyectos.html>
- NAVARRO, A. *Gestión de Riesgo*, 2000 [Abril, 2007]. Disponible en: http://209.85.165.104/search!=cache:9k0ediu_2b0j:www.fld.ucm.es/profesor/jlsierra/lsl/.
- . *Planificación de proyectos de software*, 2005. [Febrero, 2008]. Disponible en: http://www.fdi.ucm.es/profesor/anavarro/5._Planificacion_de_proyectos_de_software.pdf
- NOVA, P. C. *Planificación de Proyectos de Software*, 2005. [2008]. Disponible en: <http://www.getec.etsit.upm.es/articulos/gproyectos/art4.htm>
- PERALTA, M. *Estimación del esfuerzo basada en Casos de Usos*, mayo, 2004. [Febrero, 2008]. Disponible en: <http://www.itba.edu.ar/capis/rtis/rtis-6-1/estimacion-del-esfuerzo-basada-en-casos-de-usos.pdf>
- POW-SANG, J. *Estudio de las técnicas basadas en puntos de función para la estimación del esfuerzo en proyectos software*, agosto, 2005. [Febrero, 2008]. Disponible en: http://sisbib.unmsm.edu.pe/BibVirtualData/publicaciones/risi/N1_2004/a10.pdf
- PENADÉS, M. C. *Metodologías Ágiles para sus Proyectos*, noviembre, 2007. [Febrero, 2008]. Disponible en: <http://www.trabesoluciones.com/es/development/methodology>
- PRESSMAN, R. S. *Ingeniería del Software. Un Enfoque Práctico*. 1998. p.
- PROYECTO, E. D. L. D. *Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK®)*. Tercera edición. 2004. 392 p.
- SANCHEZ, M. A. M. *Metodologías De Desarrollo De Software*, Junio 7 del 2004. [Febrero, 2008]. Disponible en: http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html
- TEAM, A. *Gestión de proyectos con dotProject*, 2007. [Marzo, 2008]. Disponible en: <http://www.abartiateam.com/dotproject.html>
- TERUEL, A. *La Planificación de un Proyecto de Desarrollo de Software*, 30 de abril de 2001: 8.

Anexos

Anexo 1: Encuesta a los Proyectos Productivos.

Planificación de Software.

1. ¿Conoce acerca del proceso de planificación de un software? Si__ No__
2. ¿En su proyecto se lleva una planificación? Si__ No__
En caso de ser negativa su respuesta:
¿Considera necesario realizar la planificación? Si__ No__
3. ¿Utiliza alguna metodología de desarrollo para el desarrollo de su proyecto? Si__ No__
En caso afirmativo ¿Cuál fue?_____
4. ¿Utiliza alguna herramienta para la planificación de su proyecto? Si__ No__
En caso afirmativo ¿Cuál fue?_____
5. ¿Se basa en algún procedimiento o algoritmo para realizar la planificación de su proyecto?
Si__ No__
6. ¿Cuándo comenzó a realizar el proceso de planificación en su proyecto?
__ antes de comenzar __ en cada fase __ en cada iteración __ otra
7. ¿Qué rol dentro del proyecto ocupa el que realizó dicha planificación? _____
8. ¿Cuál es la jerarquía que existe en su proyecto?

9. En caso de haber realizado estimaciones de costo, tiempo y recursos
¿Qué método utilizó? _____
10. ¿Qué pasos o qué orden considera usted que debe seguir para realizar una planificación?
Mencione los primeros 5 y señale con un asterisco (*) los mas importantes.

Gestión de Proyectos.

1. Según su criterio ¿cuál es la metodología que sugiere para llevar a cabo el desarrollo de software en los proyectos productivos que hay actualmente en la Universidad?

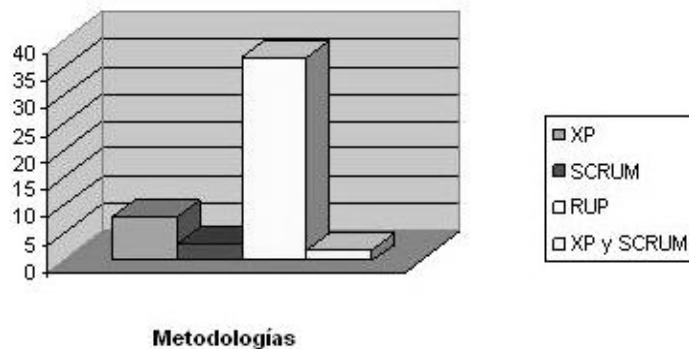
2. En su opinión ¿cómo considera la Gestión de Proyectos en la Universidad?
 Buena _____ Regular _____ Mala _____

3. ¿Como usted considera el proceso de planificación de software de los proyectos productivos en nuestra Universidad?
 Buena _____ Regular _____ Mala _____

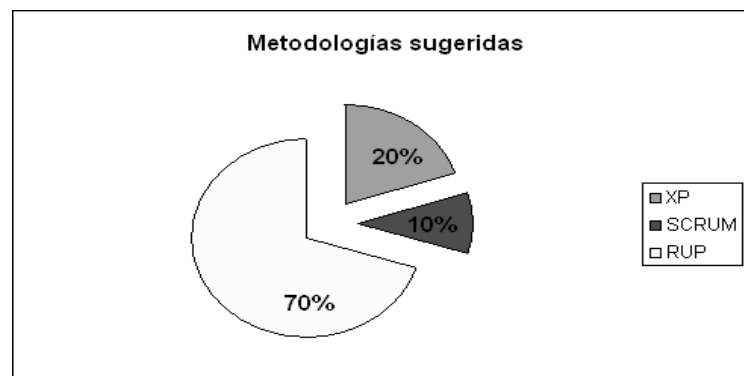
4. ¿Cree necesario la existencia de algún algoritmo o procedimiento para mejorar este proceso en la Universidad? Si _____ No _____

Anexo 2: Metodologías más usadas en el Proceso de Desarrollo del Software.

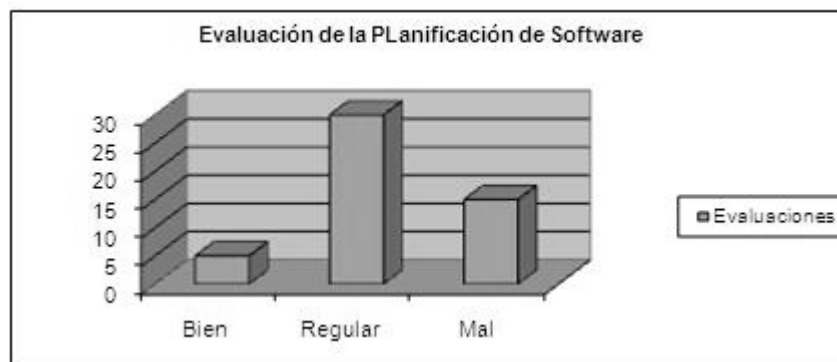
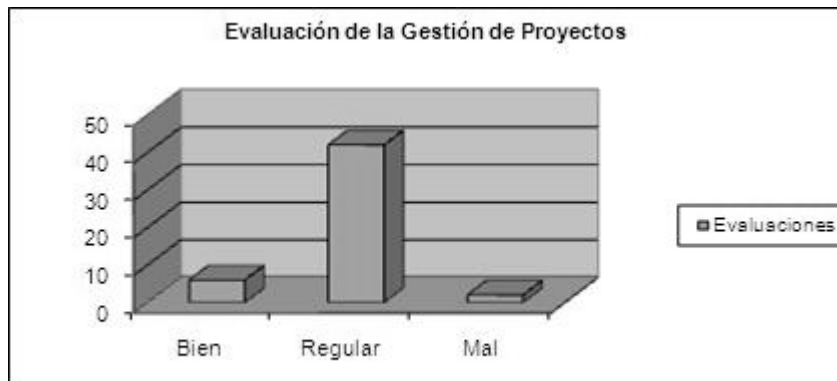
Metodologías más usadas en la UCI



Anexo 3: Metodologías sugeridas para el Proceso de Desarrollo del Software.



Anexo 4: Evaluación de la Gestión de Proyectos y Planificación de los Proyectos Productivos en la UCI.

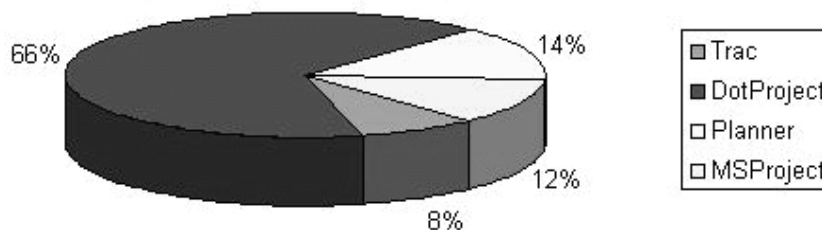


Anexo 5: Momento en que realizan la planificación los Proyectos Productivos de la UCI.



Anexo 6: Herramientas más usadas para la Gestión de Proyectos en la UCI.

Herramientas más usadas



Anexo 7: Modelo para la recogida de información referente al peso de los criterios.

Guía para informar el peso de los criterios.

Fecha de recepción...00/00/00.....

Fecha de entrega....00/00/00.....

Nombre y Apellidos del evaluador.....

Le otorgará un peso a cada criterio de acuerdo a su opinión y el peso total de cada grupo debe sumar:

- Grupo No.1..... 30
- Grupo No.2..... 20
- Grupo No.3.....20
- Grupo No.4.....30

Para que el peso total asignado sea 100.

Grupo No 1: Criterios de mérito científico.

1. Valor científico de la propuesta.

Peso.....

2. Calidad de la investigación.

Peso.....

3. Contribución científica.

Peso.....

Grupo No 2: Criterios de implantación.

4. Necesidad de empleo de la propuesta.

Peso.....

5. Posibilidades de aplicación.

Peso.....

Grupo No 3: Criterios de flexibilidad.

6. Adaptabilidad a proyectos productivos.

Peso.....

7. Capacidad de admisión de cambios que impliquen mejoras.

Peso.....

Grupo No 4. Criterios de impacto.

8. Impacto en el área para la cual está destinada la guía.

Peso.....

9. Organización en el proceso de desarrollo.

Peso.....

10. Repercusión en los proyectos productivos.

Peso.....

Anexo 8: Modelo para la recogida de información referente a la calificación de los criterios.

Fecha de recepción...00/00/00.....

Fecha de entrega.... 00/00/00.....

Nombre y Apellidos del Evaluador.....

Criterios de medida que se evalúan en una escala de 1 - 5

Grupo No 1: Criterios de mérito científico.

1. Valor científico de la propuesta.

Evaluación.....

2. Calidad de la investigación.

Evaluación.....

3. Contribución científica.

Evaluación.....

Grupo No 2: Criterios de implantación.

4. Necesidad de empleo de la propuesta.

Evaluación.....

5. Posibilidades de aplicación.

Evaluación.....

Grupo No 3: Criterios de flexibilidad.

6. Adaptabilidad a proyectos productivos.

Evaluación.....

7. Capacidad de admisión de cambios que impliquen mejoras.

Evaluación.....

Grupo No 4. Criterios de impacto.

8. Impacto en el área para la cual está destinada la guía.

Evaluación.....

9. Organización en el proceso de desarrollo.

Evaluación.....

10. Repercusión en los proyectos productivos.

Evaluación.....

Categoría final.

___ Excelente: Alta novedad científica, con aplicabilidad y resultados relevantes.

___ Bueno: Novedad científica, resultados destacados.

___ Aceptable: Suficientemente bueno con reservas.

___ Cuestionable: No tiene relevancia científica y los resultados son malos.

___ Malo: No aplicable.

Valoración final

Sugerencias del experto para mejorar la calidad del proyecto

Elementos críticos que deben mejorarse.

Glosario de Términos

Proyecto: Es el elemento organizativo a través del cual se gestiona el desarrollo de software y el resultado del mismo es una versión de un producto.

Software: Conjunto de instrucciones que las computadoras emplean para manipular datos.

Producto: Artefactos que se crean durante el transcurso de la vida el proyecto.

Procedimiento: Sucesión cronológica de operaciones concatenadas entre sí, que se constituyen en una unidad de función para la realización de una actividad o tarea específica dentro de un ámbito predeterminado de aplicación.

Proceso: Conjunto de actividades necesarias para transformar los requisitos de los usuarios en un producto.

CMMI: *Capability Maturity Model Integration* es un modelo para la mejora o evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. Además, se desarrolló para facilitar y simplificar la adopción de varios modelos de forma simultánea, y su contenido integra y da relevo a la evolución de sus predecesores. Fue desarrollado por el Instituto de Ingeniería del Software de la Universidad Carnegie Mellon (SEI), y publicado en su primera versión en enero de 2002.

RUP: El *Rational Unified Process (RUP)* es una metodología formal, a veces también llamada proceso. El RUP describe a gran detalle todas las actividades, roles, responsabilidades, productos de trabajo y herramientas para definir quién hace qué y en qué momento en un proyecto de desarrollo de software.

Roger. S. Pressman: Es una autoridad internacionalmente reconocida en la mejora de proceso de software y en tecnología de Ingeniería de Software. Por más de tres décadas, ha trabajado como ingeniero, gerente, profesor, autor y consultor de software en temas de Ingeniería de Software. Actualmente es presidente de R. S. Pressman and Asóciate, Inc., una firma consultora especialista en métodos y entrenamiento en Ingeniería de Software.

MSF: *Microsoft Solution Framework (MSF)* es una metodología flexible que se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

XP: *Extreme Programming (XP)* es una de las metodologías de desarrollo de software se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes.

SCRUM: Es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software. Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle en 1993. Define un marco para la gestión de proyectos. En SCRUM, el equipo se focaliza en construir software de calidad.

ASD: *Adaptive Software Development (ASD)* es una metodología de desarrollo ágil *que se enfoca en la naturaleza adaptable de las nuevas metodologías.*

COCOMO: es un modelo de estimación de costes de software. Fue desarrollado por Barry W. Boehm a finales de los 70 y comienzos de los 80.