UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

Sistema para la gestión de proyectos y sus recursos vía Web

TRABAJO ESPECIAL DE GRADO

Presentado ante la

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

Como parte de los requisitos para optar al título de

INGENIERO EN INFORMÁTICA

REALIZADO POR Eliana Onsalo

Juan Pablo Regueira

PROFESOR GUIA Carlo Magurno

FECHA Octubre de 2007

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

Sistema para la gestión de proyectos y sus recursos vía Web

Este Jurado; una vez realizado el examen del presente trabajo ha evaluado su contenido con el resultado: Dieciocho (18) puntos.

J U R A D O E X A M I N A D O R

Firma:	Firma:	Firma:
Nombre: Evelenir Barreto	Nombre: Héctor Gutiérrez	Nombre: Carlo Magurno

REALIZADO POR	Eliana Onsalo
	Juan Pablo Regueira
PROFESOR GUIA	Carlo Magurno
FECHA	Octubre de 2007

DEDICATORIA

Eliana Onsalo Mendiri

A mi familia, en especial a mis padres y a yaya, quienes me han apoyado en todo momento y me han guiado para lograr mis metas.

A mi hermano, con quien he compartido tanto y me ha enseñado las cosas sencillas de la vida.

A mis tíos, por brindarme sus consejos y apoyarme siempre.

Juan Pablo Regueira Vargas

Este trabajo va dedicado a mi familia en especial a mis padres, quienes siempre han sido y serán mi ejemplo a seguir, sin su esfuerzo y apoyo todo esto no hubiera sido posible. Esta y todas la metas que logre de aquí en adelante serán consecuencia de su cariño y educación, siéntanse orgullosos de lo que han logrado.

También va dedicado a la familia Toledo Ochoa, Daniel, Marlene, Gabriela y Jacinto, gracias por brindarme su amistad, apoyo y consejos en todo momento y por hacerme sentir que su casa siempre será un hogar para mí.

A mi gran familia: mi abuela, mis tíos y primos, en especial a mi padrino Romano y a mi primo Hugo con quienes he compartido tantos momentos y me han enseñado tanto, sepan que ustedes son para mi un ejemplo de perseverancia y superación.

A mi abuelo Alfonso aunque no estas presente para compartir este logro, siempre te recuerdo con admiración y cariño.

AGRADECIMIENTOS

Eliana Onsalo Mendiri:

A nuestro tutor Carlo Magurno, muchas gracias por su tiempo y por orientarnos en lograr el éxito.

Al profesor Alfredo García, muchas gracias por haberse tomado el tiempo para revisar el presente trabajo especial de grado, sus observaciones y correcciones fueron de gran ayuda.

A todos los profesores que a lo largo de la carrera fueron dejando su granito de arena para ayudar a lograr este éxito.

A todos mis amigos, en especial a los que estuvieron conmigo a lo largo de mi carrera: Sara, Lelian, Sarita, Juanpi, Gabriel, Oscar, El Negro, Robi; y a todos los que de una u otra forma compartieron conmigo.

A Victor, por apoyarme y entenderme siempre, gracias Chiki te quiero.

A Piru, mivi, gracias por todo, por estar siempre allí y por haber ayudado en el diseño gráfico de la herramienta.

A Carolina, por ser una gran amiga y recibirme siempre en su casa.

A Juanpi en especial, por ser mi compañerito y haberme soportado tantas cosas.

Juan Pablo Regueira Vargas:

Gracias a todos mis amigos Daniel, Mariale, Leo, Sandra, Alejandra, Nacho, Tosta, Gabriela, Sara, Sarita, Oscar, Hugo, Carlos, Eliana, Henry, Lelian, Guillermo y todos aquellos con los que compartí durante mi carrera tantos momentos gratos.

Gracias a nuestro Tutor Carlo por servirme de guía y apoyo durante esta etapa.

A todos aquellos profesores que dejaron enseñanzas valiosas e hicieron de mí un profesional.

Y por ultimo quiero darle las gracias a mi compañera Eliana por haber compartido y trabajado conmigo.

ÍNDICE GENERAL

DEDICATORIA I
AGRADECIMIENTOS II
ÍNDICE GENERAL IV
ÍNDICE FIGURAS VII
INDICE DE TABLAS VIII
ÍNDICE APÉNDICES IX
RESUMEN X
INTRODUCCIÓN 1
CAPÍTULO I. PROBLEMA 3
I.1 Planteamiento del problema 3
I.2 Objetivo General 4
I.3 Objetivos Específicos 4
I.4 Justificación 5
I.5 Limitaciones 5
I.6 Alcances 6
CAPÍTULO II. MARCO REFERENCIAL 7
II.1 Sistemas 7
II.2 Gestión 7
II.3 Tiempo 8
II.4 Proyecto 8
II.5 Control de proyectos 9
II.6 Sistema de gestión 9
II.7 Sistema de información 10
II.8 Aplicación Web 10
II.9 *Internet Information Services* (IIS) 11
II.10 XML 11
CAPÍTULO III. METODOLOGÍA 13
III.1 La Metodología 13
III.2 El Proceso Unificado es guiado por los Casos de Uso 13
III.3 El Proceso Unificado es centrado en la Arquitectura 14
III.4 El Proceso Unificado es Iterativo e Incremental 14
III.5 Descripción del proceso 16

III.6 Descripción de las fases 18
Fase de Inicio 18
Fase de Elaboración 19
Fase de Construcción 19
Fase de Transición 20
III.7 Descripción de las disciplinas 20
Requerimientos 20
Análisis 21
Diseño 21
Implementación 21
Prueba 21
III.8 Elementos adicionales 21
Roles 22
Productos 22
Actividades 22
CAPÍTULO IV. DESARROLLO 23
IV.1 Fase de Inicio 23
Requerimientos del sistema 24
Requerimientos funcionales del sistema 24
Requerimientos no funcionales 25
Modelo de Casos de Uso 27
Actores 27
Casos de Uso del Sistema 29
IV.2 Fase de Elaboración 31
Iteración 1 31
Diagrama de Casos de Uso 31
Diagramas de Actividades 35
Modelo de Domino 36
Diagramas de Clase 36
Diagrama de Secuencia 39
Modelo de datos 40
Diagrama de Navegación 41
Iteración 2 42
Diseño del Archivo de Entrada 43

Diseño del Archivo de Salida 44
Diseño del flujo de aprobación de horas 45
Iteración 3 47
Arquitectura del Software 47
Capas 48
Arquitectura de despliegue 49
Iteración 4 50
Diseño de los componentes del sistema 50
Interfaz de Usuario 53
IV.3 Fase de Construcción 54
Iteración 1 54
Herramientas de Software Utilizadas 55
Iteración 2 56
Desarrollo de los Componentes de Autentificación, Usuarios y Área de Negocio 56
Iteración 3 58
Desarrollo de los Componentes de Tareas y Proyectos 58
Iteración 4 60
Desarrollo del Componente de Reportes 60
Iteración 5 61
Integración y Pruebas 61
IV.4 Fase de Transición 61
Manual de Usuarios 62
CAPÍTULO V. RESULTADOS 63
CONCLUSIONES 66
RECOMENDACIONES 68
BIBLIOGRAFÍA 70

ÍNDICE FIGURAS

Figura 1: Fases e iteraciones de la Metodología RUP 17
Figura 2: Diagrama de Casos de Uso del Sistema *Flex Project Management* 32
Figura 3: Diagrama de Actividad del Caso de Uso: Consultar Proyecto 35
Figura 4: Modelo de Dominio del Sistema *Flex Project Management* 36
Figura 5: Diagrama de Clases del Sistema *Flex Project Management* 38
Figura 6: Diagrama de Secuencia del Caso de Uso: Consultar Proyecto 39
Figura 7: Modelo de Datos del Sistema *Flex Project Management* 41
Figura 8: Diagrama de Navegación del Caso de Uso: Consultar Proyecto ... 42
Figura 9: Formato del Archivo XML de Exportación de *Flex Project Management* 44
Figura 10: Flujo de aprobación de horas 47
Figura 11: Arquitectura del Sistema *Flex Project Management* 48
Figura 12: Diagrama de Despliegue del Sistema *Flex Project Management* 50
Figura 13: Componentes del Sistema *Flex Project Management* 51
Figura 14: Pantalla principal del Sistema *Flex Project Management* 54
Figura 15: Pantalla principal del componente de Autentificación 57
Figura 16: Pantalla principal del componente de Usuarios 58
Figura 17: Pantalla principal del componente de Área de Negocio 58
Figura 18: Pantalla principal del componente de Tareas 59
Figura 19: Pantalla principal del componente de Proyectos 60
Figura 20: Pantalla de reporte de tiempo total de un proyecto 61

INDICE DE TABLAS

Tabla #1: Características de algunas herramientas de gestión de proyectos24
Tabla #2: Características del usuario Administrador 28
Tabla #3: Características del usuario Director de proyectos 28
Tabla #4: Características del usuario Líder de proyectos 29
Tabla #5: Características del usuario Asegurador de Calidad..... 29
Tabla #6: Plantilla de especificaciones del caso de uso Consultar Proyecto 34

ÍNDICE APÉNDICES

Apéndice A: Plan de Desarrollo del Software 80
Apéndice B: Visión 98
Apéndice C: Glosario 122
Apéndice D: Especificación de Requerimientos Funcionales 130
Apéndice E: Especificaciones Adicionales 139
Apéndice F: Modelo de Casos de Uso 147
Apéndice G: Especificación de Casos de Uso 157
Apéndice H: Arquitectura del Software 199
Apéndice I: Diagramas de Actividad 226
Apéndice J: Diagramas de Secuencia 250
Apéndice K: Diagramas de Navegación 274
Apéndice L: Archivo de Entrada 298
Apéndice M: Manual de Usuarios 302

RESUMEN

En la actualidad, el proceso de mantener controlados los recursos y el tiempo que se invierte en la ejecución de un proyecto constituye una de las preocupaciones fundamentales dentro de cualquier empresa o institución académica.

En general, los elementos que conforman un proyecto son los recursos que en él trabajan, las tareas que esos recursos realizan dentro del proyecto y el tiempo que invierten en esas tareas. Teniendo el control de estos tres elementos se pueden culminar los proyectos de una manera más rápida y beneficiosa para la empresa.

Con base en estos tres elementos y con el fin de mejorar el proceso de gestión de proyectos, se identificó y planteó una importante oportunidad para el diseño e implementación de un sistema que pudiese controlar dichos elementos.

Flex Project Management es una herramienta que nace para servir de apoyo en la gestión de proyectos y cubrir la necesidad de pequeñas y medianas empresas e instituciones académicas que desean mantener el control del tiempo que invierten sus recursos en un proyecto.

Con *Flex Project Management* se podrán gestionar proyectos nuevos cargados en línea y/o proyectos manejados anteriormente con *Microsoft Project*.

Para el desarrollo de este sistema se utilizó la metodología *Rational Unified Process* (RUP) y algunos de los productos que esta metodología proporciona. Adicionalmente, se cuenta con un repositorio de datos central donde se almacenará toda la información importante acerca del proyecto en gestión. La herramienta está soportada por una arquitectura de tres capas, utilizando un servidor de aplicaciones y un manejador de base de datos.

INTRODUCCIÓN

Las empresas hoy en día para lograr el éxito, tienen que aprender a distribuir su tiempo de manera que se puedan llevar a cabo la mayor cantidad de tareas en un menor tiempo y con una mejor calidad. Lograr esta sinergia no es fácil, y más difícil aún cuando la empresa no cuenta con sistemas automatizados para respaldar sus procesos primordiales.

Cuando se planteó este trabajo especial de grado, el mercado de herramientas para el control de proyectos y tiempos estaba compuesto por software generalmente licenciados o sencillamente muy complicados. Por éste motivo se formuló desarrollar una herramienta Web, completamente libre, que permitiera gestionar proyectos y sus recursos.

Flex Project Management es una herramienta de control de tiempo y recursos que además de planificar y gestionar proyectos, automatiza el proceso de desarrollo logrando llevar el control de todas las tareas y recursos que intervengan de una u otra forma en los proyectos de la empresa. Cuenta con una interfaz Web que permite cargar proyectos en línea y continuar la gestión de un proyecto generado en *Microsoft Project* a través de la carga del documento XML generado por esta herramienta.

A continuación se describe por capítulos todo el proceso de desarrollo de esta investigación.

En el capítulo I, titulado Problema, se define el planteamiento del problema, el objetivo general, los objetivos específicos, la justificación de esta herramienta, sus limitaciones y el alcance que tendrá.

En el capítulo II, titulado Marco Referencial, se explican todos aquellos conceptos importantes dentro del marco del desarrollo del proyecto.

En el capítulo III, titulado Metodología, se describe la metodología utilizada para el desarrollo del proyecto, sus fases y productos o entregables desarrollados en cada fase.

En el capítulo IV, titulado Desarrollo, se describe cada una de las fases de la metodología utilizada, las iteraciones que se realizaron en cada una de las fases y los productos generados al finalizar cada una de las iteraciones. En general, es en este capítulo donde se explica la funcionalidad completa de la herramienta.

En el capítulo V, titulado Resultados, se describen todos los resultados obtenidos una vez culminado el desarrollo del proyecto. A continuación, se describe una sección con las conclusiones y una sección con las recomendaciones al sistema. Seguido a esto se presenta la bibliografía y para finalizar se incluyen los anexos y apéndices.

CAPÍTULO I. PROBLEMA

I.1 Planteamiento del problema

Las condiciones en las que se compite en la actualidad por acceder a los recursos necesarios, por reducir gastos y costos, por aumentar la calidad de los productos y servicios, han modificado la forma de actuar e interactuar de las organizaciones.

Si existe algo que se debe apreciar en el trabajo es el tiempo, apreciarlo por su importancia y porque su incorrecta gestión puede influir negativamente en la toma de decisiones, en el trabajo realizado, en abordar nuevos asuntos, en definitiva, en la marcha global de la empresa.

El tiempo es inflexible: pasa y no se detiene; aunque a veces se tenga la sensación de todo lo contrario, el tiempo no corre. Es, en apariencia, una variable que no se puede modificar. No se puede alargar, estirar, comprar o detener. Sin embargo, se puede controlar.

Un proceso de control de una gestión de tiempo está basado, por tanto, en mecanismos de control relacionados con aspectos cuantificables, derivados de un presupuesto o de un plan, basados en objetivos planteados y en sistemas de controles específicos.

El sistema de control de gestión tiempo se plantea como una solución capaz de ayudar a los distintos niveles de decisión a coordinar las acciones, a fin de alcanzar los objetivos de mantenimiento, desempeño y evolución, fijados a distintos plazos, así como también a proporcionar un instrumento de medición, control de las tareas y objetivos de un proyecto.

I.2 Objetivo General

Diseñar e implementar un sistema Web para gestionar la inversión de tiempo del recurso humano en el desarrollo de proyectos.

I.3 Objetivos Específicos

1. Diseñar e implementar los flujos de trabajo para la gestión de proyectos y la aprobación de horas invertidas en un proyecto por los recursos humanos.
2. Definir los distintos roles que interactuarán en el flujo de trabajo y sus responsabilidades.
3. Definir un estándar para la generación de archivos de planificación en formato XML.
4. Desarrollar un modelo de datos para soportar la gestión de proyectos y el control del tiempo.
5. Desarrollar un módulo de administración que permita cargar las tareas del proyecto a través de la lectura de un archivo XML y que también permita guardar las modificaciones realizadas al proyecto en un archivo de formato XML estándar.
6. Desarrollar un módulo de reportes para la comparación periódica de las horas invertidas versus las horas estimadas, además debe generar las hojas de tiempo por proyecto por recurso.
7. Desarrollar un sistema Web que permita implementar el módulo de administración y el módulo de reportes para llevar a cabo la gestión del tiempo de los recursos humanos invertidos en un proyecto.

I.4 Justificación

El Control de Gestión como una de las herramientas fundamentales para que las empresas e instituciones académicas puedan obtener productos y servicios con una eficiencia relevante, que le permitan el acceso al mundo competitivo de hoy, se ha convertido en un tema de gran actualidad.

El control horario sobre los proyectos es una preocupación que se remonta al siglo pasado. Desde la aparición de los primeros métodos para controlar la hora de llegada de los trabajadores y el tiempo invertido en la realización de proyectos, se han ido perfeccionando los sistemas de medición, de manera que actualmente se pueden detectar problemas que se derivan de estos sistemas. De esta forma, se ha pasado del control del tiempo a la gestión del mismo consiguiendo que éstos aporten mayor información para las empresas.

Los sistemas de medición se plantean como la solución de las empresas a los problemas que se derivan del uso del tiempo que realizan sus empleados. Las nuevas tecnologías aportan novedades y ventajas sobre los diferentes sistemas que hasta ahora han existido y convivido. A partir de éstos se pueden conocer el uso y mejorar el rendimiento de minutos perdidos.

Por ello se plantea el control de gestión de tiempo como una herramienta de retroalimentación para el uso eficiente del los recursos disponibles dentro de una empresa o institución académica, que permita lograr los objetivos planteados.

I.5 Limitaciones

* Este sistema sólo puede ser ejecutado sobre servidores Web IIS.

* La carga de proyectos se realizará mediante la lectura a un archivo XML de formato estándar.

* En caso de cambios en la estructura del proyecto se deberá cargar nuevamente el archivo XML.

* No se eliminará ninguna tarea del proyecto, en caso de que una o mas tareas no sean necesarias pasaran a un estado de deshabilitada o inactiva.

I.6 Alcances

* El sistema brindará la opción de cargar proyectos en línea o a través de un archivo de formato XML.

* El archivo XML que va a ser usado será definido por los desarrolladores haciendo adaptaciones de estándares existentes ajustándolo a las necesidades del sistema.

* El sistema de gestión de tiempo será capaz de administrar las operaciones en cuanto a los proyectos los cuales se podrán cargar, modificar, habilitar o deshabilitar y guardas; en cuanto a las tareas se podrán agregar nuevas tareas, habilitar o deshabilitar y guardar; y en cuanto al archivo XML se podrá cargar, modificar y guardar.

* Por proyecto el sistema podrá gestionar varios recursos.

* Generará reportes generales y específicos sobre los tiempos de realización tanto de las tareas de cada proyecto como de los proyectos en general.

CAPÍTULO II. MARCO REFERENCIAL

II.1 Sistemas

El término sistema designa un conjunto de elementos en interrelación dinámica organizados en función de un objetivo, con vistas a lograr los resultados del trabajo de una organización. [1]

Los sistemas incluyen un conjunto de elementos que guardan relación estrecha entre sí y que trabajan juntos para lograr un objetivo común. Los sistemas se caracterizan por recibir un conjunto de entradas y con las que generan un conjunto de salidas. Gracias a este proceso el sistema adquiere una propiedad muy importante que lo identifica: un conjunto de partes relacionadas entre sí son denominadas un sistema cuando la totalidad de sus propiedades no son comparables a la totalidad de la suma de sus partes. Los sistemas por definición son estables, autónomos, autosuficientes y adaptables. [1]

Otra propiedad que define a un sistema es el tipo: abierto o cerrado; los sistemas abiertos se caracterizan por la necesidad de interacción con su medio ambiente para poder continuar existiendo; los cerrados son simplemente los no abiertos. [1]

II.2 Gestión

Por gestión se entiende, la dirección de las acciones que contribuyan a tomar decisiones orientadas a alcanzar los objetivos trazados, medir los resultados obtenidos para finalmente, orientar la acción hacia la mejora permanente del sistema. [1]

La gestión está caracterizada por una visión más amplia de las posibilidades reales de una organización para resolver determinada situación o arribar a un fin determinado. Puede asumirse, como la disposición y organización de los recursos de un individuo o grupo para obtener los

resultados esperados. Pudiera generalizarse como una forma de alinear los esfuerzos y recursos para alcanzar un fin determinado. [1]

II.3 Tiempo

El tiempo es el factor más conocido por las personas participantes en un proyecto, se pueden desconocer muchos factores pero si existe algo que siempre está claramente definido es para cuándo el proyecto debe estar concluido. Todo proyecto está sujeto a un tiempo, a una duración. [2]

En términos cuantitativos, son las semanas días, horas y/o minutos necesarios para llevar a cabo las tareas de un proyecto. [2]

II.4 Proyecto

Se puede definir proyecto como un conjunto de actividades interdependientes orientadas a un fin específico, con una duración predeterminada. Completar con éxito el proyecto significa cumplir con los objetivos dentro de las especificaciones técnicas, de costo y de plazo de terminación. [3]

Existen muchas otras definiciones de lo qué es un proyecto, pero todas coinciden en que para que se lleve a cabo el mismo, deben existir un conjunto de tareas programadas y planificadas con un solo fin: lograr un producto único.

Un proyecto es exitoso cuando el cliente queda satisfecho con el producto y cuando ha habido un aprendizaje provechoso al final de la ejecución del mismo. La segunda forma de medir el éxito de un proyecto es si se han satisfecho las necesidades de la organización que ejecuta el proyecto. [4]

II.5 Control de proyectos

La gerencia de proyectos es el proceso de planear, organizar y administrar tareas, dinero y recursos para alcanzar un objetivo concreto, generalmente con delimitaciones de tiempo, recursos y/o costos. Un plan de proyecto puede ser sencillo; por ejemplo una lista de tareas y sus fechas de comienzo y fin escritas en un block de notas. Pero también puede ser complejo; por ejemplo, miles de tareas y recursos, y un presupuesto del proyecto de millones de dólares. [5]

La mayoría de los proyectos comparten actividades comunes, como la división del proyecto en tareas de fácil manejo, la programación de las tareas, la comunicación entre los miembros del equipo y el seguimiento de las tareas a medida que progresa el trabajo. Además, todos los proyectos constan de tres fases principales:

1. Crear el plan de actividades.
2. Administrar recursos (financieros y humanos).
3. Realizar un seguimiento del proyecto.
4. Cerrar el proyecto.

Cuanto más satisfactorias sean estas fases, mayor será la posibilidad de éxito del proyecto. [5]

II.6 Sistema de gestión

Un sistema de gestión es una herramienta para que los gerentes descentralizados tomen sus decisiones a su debido tiempo, lo que permite alcanzar los objetivos estratégicos perseguidos por cada organización. Para mantener la eficiencia organizativa y facilitar la toma de decisiones es fundamental controlar la evolución del entorno y las variables de la propia organización que pueden afectar su propia supervivencia. [1]

El sistema de gestión como un conjunto de procedimientos que representa un modelo organizativo concreto para realizar la planificación de las actividades que se llevan a cabo en la empresa, queda determinado por un conjunto de actividades y sus interrelaciones. [1]

Puede hablarse, entonces, de un sistema de gestión, como un conjunto de acciones, funciones, medios y responsables que garanticen, mediante su interacción, conocer la situación de un aspecto o función de la organización en un momento determinado para tomar decisiones y reaccionar ante ella. [1]

II.7 Sistema de información

Son sistemas que se caracterizan por recibir datos o información del medio ambiente donde existen, para procesarlos y generar nueva información en forma de reportes o insumos para otro sistema. Los sistemas de información suelen apoyar a otros sistemas en las organizaciones, de estos extraen los insumos que necesitan. [6]

Un sistema de información es un conjunto de personas, datos, procesos e interfaces que interactúan para apoyar y mejorar las operaciones del día a día en el negocio, así como soportar la solución de problemas y la toma de decisiones de gerentes y usuarios. [6]

Uno de los objetivos más importantes que se persigue con los sistemas de información se basa en buscar la manera de integrar procesos y sistemas en uno sólo para mejorar el desempeño, la seguridad, la funcionalidad, usabilidad y disponibilidad de los procesos y sistemas.

II.8 Aplicación Web

Una aplicación Web es aquella que puede ser utilizada a través de un servidor Web (*Browser*) a través de Internet o de una Intranet. Las aplicaciones Web son populares debido a la utilización del navegador como cliente. Una de las razones de su popularidad es que no es necesaria la instalación de la

herramienta y que toda la información es almacenada en una base de datos que se encuentra en un sitio remoto. [7]

II.9 *Internet Information Services* (IIS)

Internet Information Services (*o Server*), IIS, es una serie de servicios para los ordenadores que funcionan con *Windows*. Originalmente era parte del *Option Pack* para *Windows NT*. Luego fue integrado en otros sistemas operativos de *Microsoft* destinados a ofrecer servicios, como *Windows 2000* o *Windows Server 2003*. *Windows XP Profesional* incluye una versión limitada de IIS. [8]

Este servicio convierte a un computador en un servidor de Internet o Intranet, es decir, en la computadora que se tenga el servicio instalado se pueden publicar páginas Web tanto local como remotamente (servidor Web). [8]

II.10 XML

XML es la sigla del inglés *eXtensible Markup Language* (Lenguaje de mercado ampliable o extensible) desarrollado por el *World Wide Web Consortium* (W3C). [9]

Su objetivo principal es conseguir una página Web más semántica. Aunque una de las principales funciones con las que nace sería suceder al HTML separando la estructura del contenido y permitiendo el desarrollo de vocabularios modulares, compatibles con cierta unidad y simplicidad del lenguaje (objetivo que se viene desarrollando a través de la especificación XHTML). Este lenguaje tiene otras aplicaciones entre las cuales destaca su uso como estándar para el intercambio de datos entre diversas aplicaciones o software con lenguajes privados. [9]

Al igual que HTML, se basa en documentos de texto plano en los que se utilizan etiquetas para delimitar los elementos del documento. Sin embargo,

XML define estas etiquetas en función del tipo de datos que está describiendo y no de la apariencia final que tendrán en pantalla o en la copia impresa, además de permitir definir nuevas etiquetas y ampliar las existentes. [9]

CAPÍTULO III. METODOLOGÍA

III.1 La Metodología

La metodología seleccionada para el desarrollo de este proyecto es *Rational Unified Process* (RUP). RUP ofrece un acercamiento disciplinado a la asignación de tareas y responsabilidades dentro de una organización de desarrollo.

El Proceso Unificado de Desarrollo de Software (*The Unified Software Development Process*) es un conjunto de actividades que se utilizan para transformar los requerimientos del usuario en un sistema de software. Sin embargo, el proceso unificado es más que un proceso sencillo; es un marco de procedimiento genérico que puede ser más especializado para una clase muy grande de sistemas de software, para aplicaciones en diferentes áreas, diferentes tipos de organizaciones, diferentes niveles competitivos y en distintos tamaños de proyectos. [10]

El Proceso Unificado usa el Lenguaje Unificado de Modelado UML (*Unified Modeling Language*) para preparar todos los diseños del sistema de software. De hecho, UML es una parte integral del Proceso Unificado. [10]

Sin embargo, los aspectos realmente distintivos del Proceso Unificado y que lo hacen único, son vistos en las tres frases claves: casos de uso, centrado en la arquitectura e, iterativo e incremental. [10]

III.2 El Proceso Unificado es guiado por los Casos de Uso

Un caso de uso es una pieza de funcionalidad en el sistema que le da al usuario un resultado de valor. Un caso de uso obtiene requerimientos funcionales. El conjunto de casos de uso forma el modelo de casos de uso que define la funcionalidad total del sistema. Este modelo reemplaza al tradicional de especificación funcional del sistema. [10]

Sin embargo, los casos de uso no son solamente una herramienta para especificar los requerimientos de un sistema, también guían el diseño del sistema y asisten en su implementación y prueba. [10]

Los casos de uso son desarrollados en conjunto con la arquitectura del sistema. Esto implica que los casos de uso guían la arquitectura del sistema, y la arquitectura del sistema influencia la selección del caso de uso. Por lo tanto, ambos, la arquitectura del sistema y los casos de uso maduran mientras el ciclo de vida continua. [10]

III.3 El Proceso Unificado es centrado en la Arquitectura

Según Kruchten, la arquitectura de un sistema es la organización o estructura de sus partes más relevantes, lo que permite tener una visión común entre todos los involucrados (desarrolladores y usuarios) y una perspectiva clara del sistema completo, necesaria para controlar el desarrollo. [11]

La arquitectura nace de las necesidades de la empresa. Sin embargo, también es influenciado por muchos otros factores tales como la plataforma en la que va a ejecutarse el software, los bloques de construcción reutilizables disponibles, las consideraciones de instalación, los sistemas de legado y los requerimientos no funcionales. [10]

III.4 El Proceso Unificado es Iterativo e Incremental

El desarrollo de un sistema de software es un largo trabajo que puede durar meses y hasta años. Por esta razón es práctico dividir el trabajo en pequeños proyectos o mini proyectos. [10]

Cada mini proyecto es una iteración que resulta en un incremento. Iteraciones se refiere a pasos en el caudal de trabajo, e incrementos, para avanzar en el desarrollo del producto. Para ser más efectivos, las iteraciones deben ser controladas, es decir, deben ser seleccionadas y llevadas a cabo de una manera planificada. Esta es la razón del por qué son mini proyectos. Por

supuesto un incremento no es necesariamente aditivo. Especialmente en las primeras fases del ciclo de vida, donde los analistas pueden estar reemplazando el diseño superficial por uno más detallado o sofisticado. En fases posteriores los incrementos son típicamente aditivos. [10]

Si una iteración alcanza su meta (como usualmente lo hace) el desarrollo procede con la siguiente iteración. Cuando una iteración no alcanza su meta, el analista debe revisar sus decisiones previas y tratar con un nuevo enfoque. [10]

Cuando surjan problemas no previstos que añadan iteraciones o alteren la secuencia de ellas, el proceso de desarrollo tomará más esfuerzo y tiempo. [10]

Según la Universidad de Chile, RUP utiliza la técnica de las seis mejores prácticas para optimizar el desarrollo de un proyecto de software, las cuales son:

1. Desarrollo iterativo: El software moderno es complejo y novedoso. No es realista usar un modelo lineal de desarrollo como el de cascada. Un proceso iterativo permite una comprensión creciente de los requerimientos a la vez que se va haciendo crecer el sistema. RUP sigue un modelo iterativo que aborda las tareas más riesgosas primero. Con esto se logran reducir los riesgos del proyecto y tener un subsistema ejecutable tempranamente. [12]

2. Manejo de requerimientos: RUP describe como: obtener los requerimientos, organizarlos, documentar requerimientos de funcionalidad y restricciones, rastrear y documentar decisiones, captar y comunicar requerimientos del negocio. Los casos de uso y los escenarios indicados por el proceso han probado ser una buena forma de captar requerimientos y guiar el diseño, la implementación y las pruebas. [12]

3. Arquitectura basada en componentes: el proceso se basa en diseñar tempranamente una arquitectura base ejecutable. La arquitectura debe ser: flexible, fácil de modificar, intuitivamente comprensible, que promueva la reutilización de componentes. RUP apoya el desarrollo basado en componentes, tanto nuevos como existentes. [12]

4. Modelado visual: modelamiento visual de la estructura y el comportamiento de la arquitectura y los componentes. UML es la base del modelamiento visual de RUP. [12]

5. Verificación de cualidades: No sólo la funcionalidad es esencial, también el rendimiento y la confiabilidad. RUP ayuda a planificar, diseñar, implementar, ejecutar y evaluar pruebas que verifiquen estas cualidades. El aseguramiento de la calidad es parte del proceso de desarrollo y no la responsabilidad de un grupo independiente. [12]

6. Control de cambios: los cambios son inevitables, pero es necesario evaluar si éstos son necesarios y rastrear su impacto. RUP indica como controlar, rastrear y monitorear los cambios dentro del proceso iterativo de desarrollo. [12]

III.5 Descripción del proceso

RUP divide el ciclo de vida del desarrollo de un proyecto en cuatro fases iterativas: Inicio, Elaboración, Construcción y Transición. Cada una de ellas se concluye con un hito principal. Este hito es muy importante dentro de la metodología, ya que, se revisan los requerimientos establecidos en cada fase, basándose en los controles de calidad. Si en un punto, existe algún proceso que no cumpla con la calidad, se rediseña o se cancela. De esta manera se evita un producto de mala calidad, costos extras y trabajo adicional. [13]

De manera gráfica, el proceso puede ser descrito en dos dimensiones distribuidas en dos ejes:

* El eje horizontal: Representa el tiempo y es considerado el eje de los aspectos dinámicos del proceso. Indica las características del ciclo de vida del proceso expresado en términos de fases, iteraciones e hitos.

* El eje vertical: Representa los aspectos estáticos del proceso. Describe el proceso en términos de componentes de proceso, disciplinas, flujos de trabajo, actividades, productos y roles.

En la siguiente figura se pueden observar las fases e iteraciones de RUP:

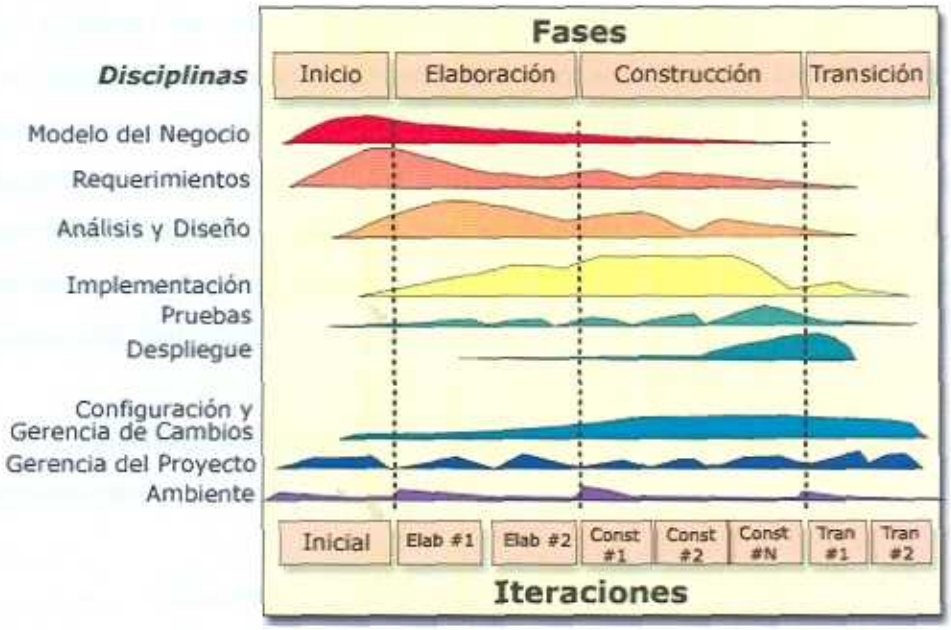

Figura 1: Fases e iteraciones de la Metodología RUP

Fuente: IBM Corp.2005

III.6 Descripción de las fases

Cada fase de RUP puede ser dividida en varias iteraciones. Una iteración consiste en un ciclo completo de desarrollo que resulta en un entregable de un producto ejecutable. Esto hace que los riesgos sean identificados y solucionados más temprano, los cambios sean más manejables, haya un alto nivel de reutilización y una mejor calidad general. [14]

A continuación se describirá con más detalle en qué consiste y cuáles son los entregables en cada fase:

Fase de Inicio

En esta fase se establece la visión del negocio y se delimita el alcance del sistema. Se enfoca principalmente al entendimiento de los requerimientos del negocio. Los objetivos de esta fase son: establecer el ámbito del proyecto y sus límites, encontrar los casos de uso críticos del sistema, estimar tiempos de desarrollo, estimar riesgos. Para la culminación exitosa de esta fase se deben identificar todos los entes externos con los cuales va a interactuar el sistema, es decir, los actores. Adicionalmente, debe definirse correctamente el tipo de interacción del actor con el sistema. [14]

Entre las actividades y entregables más importantes de esta fase se encuentran:

* Documento con la visión del proyecto.
* Plan de desarrollo del sistema.
* Lista de todos los actores del sistema.
* Glosario inicial del proyecto.
* Lista de requisitos funcionales y no funcionales.
* Definición de los casos de uso del proyecto, una visión inicial.

Fase de Elaboración

Esta fase comienza por establecer el dominio del problema, planificando las actividades necesarias, diagrama de flujos y los recursos requeridos. Luego se desarrolla un plan de proyecto donde se eliminan los elementos de alto riesgo y se establece la arquitectura del negocio. Se recomienda la construcción de un prototipo inicial en que se reflejen las funcionalidades más importantes del modelo definidas en la fase de inicio. [14]

Entre las actividades y entregables más importantes de esta fase se encuentran:

* Refinar todos los productos generados en la fase de inicio.
* Definición de todos los casos de uso del sistema.
* Documento de especificación de casos de uso.
* Diagramas de actividad.
* Diagrama de clases.
* Diagramas de secuencia.
* Documento de Arquitectura del Software.

Fase de Construcción

En esta fase se desarrolla el producto y se evoluciona la visión. Todos los componentes, así como las funcionalidades del modelo son desarrollados para obtener un producto final en una primera versión, realizando las pruebas necesarias. Este producto debe estar adecuado a las plataformas correctas y una descripción del entregable. [14]

Entre las actividades y entregables más importantes de esta fase se encuentran:

* Revisar los productos desarrollados en las fases anteriores y refinar si es necesario.

* Optimizar los recursos utilizados en el desarrollo, tales como, herramientas de desarrollo, bases de datos, servidores.
* Desarrollo iterativo.
* Prototipo.

Fase de Transición

Consiste en la entrega del producto a todos los usuarios. Esto incluye la manufactura, envío, entrenamiento, soporte y mantenimiento del producto hasta obtener la satisfacción del cliente. También se esperan comentarios por parte del cliente, para proceder a la elaboración de nuevos entregables, corregir problemas y producir una nueva versión mejorada del producto. [14]

Entre las actividades y entregables más importantes de esta fase se encuentran:

* Realizar los ajustes necesarios.
* Realizar las pruebas finales del producto.
* Manual de usuario.
* Producto completado.

III.7 Descripción de las disciplinas

El ciclo de vida del Proceso Unificado de Desarrollo de Software consta de las siguientes disciplinas:

Requerimientos

El propósito de este flujo es conseguir las especificaciones funcionales del sistema para cumplir con las exigencias y requisitos de los usuarios. [10]

Análisis

En este flujo se analizan los requerimientos tales como fueron descritos en la fase anterior. El propósito de esto es llevar a cabo estudios más profundos y precisos de los requerimientos, lo que va a ayudar a darle más cuerpo al sistema incluyendo su arquitectura. [10]

Diseño

En este flujo de trabajo se le da forma al sistema y se determina la arquitectura basándose en los requerimientos. Una entrada esencial para la fase de diseño es el resultado del modelo de análisis, que provee una comprensión más detallada de los requerimientos. [10]

Implementación

En este flujo de trabajo se comienza con el resultado proveniente de la fase de diseño y se implementa en términos de componentes, como lo son el código fuente, ejecutables, etc. El principal propósito de esta fase es darle mayor detalle a la arquitectura y al sistema como un todo. [10]

Prueba

Este flujo de trabajo se encarga de verificar los resultados de la implementación mediante un conjunto de pruebas elaboradas. Las pruebas se deben planificar para probar el sistema en su totalidad como pruebas de funcionalidad y esfuerzo; para llevar a cabo las mismas se deben preparar previamente los datos que se utilizaran para cada una de las pruebas. [10]

III.8 Elementos adicionales

Además de las fases y disciplinas, RUP define tres elementos adicionales: los roles, los productos y las actividades. La Universidad de Valencia los define así:

Roles

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. [15]

Productos

Un producto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final. [15]

Actividades

Una actividad en concreto es una unidad de trabajo que una persona que desempeñe un rol puede ser solicitado a que realice. Las actividades tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto. [15]

Uniendo todos los elementos antes descritos: fases, disciplinas, roles, productos y actividades, podemos desarrollar un proyecto utilizando la metodología RUP.

Hay que destacar, que no es necesario cumplir con cada uno de los elementos; si estamos desarrollando un proyecto pequeño o mediano, no necesariamente utilizaremos los mismos roles, productos y actividades que se utilizan en la realización de un proyecto grande. Con lo que si se debe cumplir es con las fases de la metodología.

CAPÍTULO IV. DESARROLLO

Este capítulo, denominado Desarrollo, tiene la finalidad de describir en detalle todo el desarrollo del sistema *Flex Project Management*, teniendo como base los principios de la metodología RUP.

El desarrollo está dividido en cuatro fases; en cada una de ellas se describirán todas las actividades que se llevaron a cabo y los productos que se utilizaron para soportar el desarrollo del sistema. A continuación se describen cada una de las fases.

IV.1 Fase de Inicio

La principal característica que ha definido a la fase de inicio ha sido entender realmente cuál es la necesidad existente en el mercado, de contar con una herramienta que ayude a llevar el control eficaz y seguro de los recursos y del tiempo que se invierte en el desarrollo de un proyecto.

El levantamiento de información es un pilar fundamental para entender el mercado y permitir llevar a cabo un desarrollo adaptado a las necesidades del mismo.

Para definir una herramienta que realmente ofreciera un servicio de calidad a sus usuarios y que les ayudara a controlar el tiempo y los recursos de un proyecto, se llevó a cabo un proceso de levantamiento de información el cual incluyó investigar sobre qué herramientas existen en el mercado que se encarguen de gestionar proyectos; cuáles son los procesos fundamentales que componen el control de proyectos; cómo se maneja la información; cuáles son los actores que generalmente intervienen en el desarrollo de un proyecto; cómo se asignan los recursos; cuáles son los niveles de aprobación que se necesitan al momento de aprobar las horas que invierten los recursos; en general, se investigaron los detalles más importantes que componen el proceso de gestión de proyectos.

En la tabla que se muestra a continuación se comparan algunas de las herramientas existentes en el mercado que ofrecen un servicio parecido al que pretende ofrecer el sistema *Flex Project Management*.

	Gestión de Proyectos	Manejo de Información con XML	Gestión de Recursos	Gestión de Tareas	Uso de Niveles de Aprobación	Manejo de horas	Facilidad de exportar datos	Herramienta Web	Generación de Reportes
B-Kin	Sí	No	Sí	Sí	No	Sí	No	Sí	Sí
WhoDoes	Sí	No	Sí	Sí	No	Sí	No	Sí	No
Active Collab	Sí	No	Sí	Sí	No	No	No	No	Sí
Web Timesheet	Sí	No	Sí	Sí	No	Sí	No	Sí	Sí
Achievo	Sí	No	Sí	Sí	No	Sí	No	Sí	Sí
DotProject	Sí	No	Sí	Sí	No	No	No	No	No

Tabla #1: Características de algunas herramientas de gestión de proyectos

Fuente: Elaboración Propia, 2007

Tomando en cuenta este análisis, se definieron los requerimientos funcionales del sistema

Requerimientos del sistema

Para obtener una visión global del sistema, es muy importante definir los requerimientos funcionales y no funcionales del sistema; por esta razón, se identificaron tempranamente aportando así una base estable acerca de cuáles eran las necesidades del mercado.

Requerimientos funcionales del sistema

Los requerimientos funcionales con los cuales debe cumplir el sistema *Flex Project Management* son:

1. Definir un estándar para la generación de archivos de planificación en formato XML.
2. Cargar proyectos en línea.
3. Cargar proyectos mediante un archivo XML estándar predefinido.
4. Gestionar los proyectos.
5. Gestionar las tareas de un proyecto.
6. Gestionar los usuarios del sistema.
7. Exportar el archivo de planificación generado por *Flex Project Management*.
8. Diseñar un flujo de trabajo para la aprobación de las horas invertidas en un proyecto por un recurso.
9. Generar reportes generales y específicos sobre los recursos y proyectos.

Para obtener una descripción más detallada de los requerimientos funcionales del sistema *Flex Project Management* se puede consultar el Apéndice D: Especificación de Requerimientos Funcionales.

Requerimientos no funcionales

Para definir los requerimientos no funcionales del sistema *Flex Project Management* se utilizó el estándar internacional para la evaluación de software ISO9126, el cual divide a los requerimientos no funcionales en: requerimientos de funcionalidad, portabilidad, usabilidad, desempeño, confiabilidad y mantenibilidad. A continuación se describen dichos requerimientos enfocados al sistema *Flex Project Management*.

1. Requerimientos de funcionalidad

Las características que se tomaron en cuenta para los requerimientos de funcionalidad del sistema *Flex Project Management* fueron:

* Interoperabilidad: esta característica viene representada por el hecho de que en la herramienta *Flex Project Management* se pueden cargar

archivos de planificación en formato XML generados por *Microsoft Project*.

* Exactitud de la información: toda la información que ingresa a la base de datos está validada previamente, evitando así que información inconsistente o errónea sea almacenada.

* Seguridad de la información: el acceso a la aplicación se lleva a cabo mediante la verificación de un nombre de usuario y una contraseña. Hay que destacar que actualmente no se está encriptando la contraseña.

* Privacidad de la información: Toda la información y los datos cargados en el sistema se mantienen privados y solamente tienen acceso a ellos los usuarios que cuenten con la permisología necesaria para acceder. Cada usuario tendrá un nombre y una clave de acceso que será la llave para acceder a la información.

2. Requerimientos de portabilidad

Flex Project Management es un sistema Web, puede ser accedido desde cualquier explorador, lo único que se necesita es contar con una conexión Web.

3. Requerimientos de desempeño

La característica más importante que se tomó en cuenta en el desarrollo del sistema *Flex Project Management* en cuanto a desempeño, fue la variable tiempo. Minimizar tanto los tiempos de carga del sistema como los tiempos de realización de los procesos dentro de la herramienta.

4. Requerimientos de confiabilidad

Con respecto a la confiabilidad, en caso de que una transacción sea fallida el sistema regresará a su estado inicial, quedando respaldados todos los datos almacenados hasta la última transacción exitosa.

Para obtener una descripción más detallada de los requerimientos no funcionales del sistema *Flex Project Management* se puede consultar el Apéndice E: Especificaciones Adicionales.

Modelo de Casos de Uso

El modelado de los casos de uso es una de las técnicas más simples y efectivas para modelar los requisitos del sistema desde la perspectiva de los usuarios. El modelo de casos de uso está compuesto por actores y casos de uso; los actores para el sistema *Flex Project Management*, están representados por los usuarios del sistema y los casos de uso representan la funcionalidad del mismo. A continuación se definen los actores del sistema *Flex Project Management* y los casos de uso identificados.

Actores

Los actores son los involucrados del sistema, pueden ser usuarios humanos u otros sistemas. Para el sistema *Flex Project Management* los actores están compuestos por usuarios humanos solamente. Un actor utiliza un caso de uso para ejecutar cierto trabajo de valor para el negocio. El conjunto de casos de uso al que un actor tiene acceso define su rol en el sistema y el alcance de su acción.

Los actores que se definieron para el sistema *Flex Project Management* fueron:

* El administrador

Nombre	Descripción	Responsabilidades
Administrador	Es el usuario que tiene todas las permisologías dentro del sistema.	* Es el encargado de inicializar el sistema. * Es el encargado de cargar todos los usuarios del sistema. * Es la única persona que tiene permisologia para ver los datos de todos los usuarios cargados en el sistema, todos los proyectos y las tareas asociadas a los mismos. * Debe tener conocimiento total del sistema. * Puede generar todos los reportes.

Tabla #2: Características del usuario Administrador

Fuente: Elaboración Propia, 2007

* El director de proyectos

Nombre	Descripción	Responsabilidades
Director de Proyectos	Es el usuario encargado de planificar, diseñar, cargar y supervisar los proyectos que se desean gestionar en el sistema.	* Es el encargado de cargar los proyectos que se van a gestionar. * Puede modificar los datos de los proyectos ya cargados en el sistema. * Gestionar los proyectos en general. * Puede generar todos los reportes.

Tabla #3: Características del usuario Director de proyectos

Fuente: Elaboración Propia, 2007

* El líder de proyectos

Nombre	Descripción	Responsabilidades
Líder de Proyectos	Es el usuario encargado de dirigir y supervisar un proyecto específico.	* Gestionar las tareas en general. * Aprueba las horas invertidas en una tarea en específico. * Asigna aseguradores de calidad a un proyecto.

		* Puede generar reportes específicos.

Tabla #4: Características del usuario Líder de proyectos

Fuente: Elaboración Propia, 2007

* El asegurador de calidad

Nombre	Descripción	Responsabilidades
Asegurador de calidad	Es el usuario encargado de supervisar la gestión de proyectos.	* Consulta los proyectos (y sus tareas) que estén bajo su responsabilidad. * Puede generar reportes específicos

Tabla #5: Características del usuario Asegurador de Calidad

Fuente: Elaboración Propia, 2007

* Los usuarios.

Nombre	Descripción	Responsabilidades
Usuarios	Es el usuario más básico del sistema. Puede solamente gestionar las tareas a las cuales está asignado.	* Gestionar las tareas a las que está asignado. * Puede generar reportes específicos

Tabla #6: Características del Usuario en General

Fuente: Elaboración Propia, 2007

Para obtener una descripción más detallada de los actores del sistema *Flex Project Management* se puede consultar el Apéndice B: Visión.

<u>Casos de Uso del Sistema</u>

Un caso de uso representa una función que ofrece el sistema de información desde el punto de vista del usuario. Típicamente son un conjunto de transacciones ejecutadas entre el sistema y los actores.

A continuación se presenta una lista de los casos de uso que conforman al sistema *Flex Project Management*.

1. Iniciar sesión.
2. Cambiar clave de acceso.
3. Olvidar clave de acceso.
4. Cargar proyecto con XML.
5. Cargar proyecto en Línea.
6. Modificar proyecto.
7. Consultar proyecto.
8. Exportar proyecto.
9. Asignar aseguradores de calidad.
10. Cargar tareas.
11. Consultar y actualizar tareas.
12. Modificar tareas.
13. Aprobar horas invertidas.
14. Asignar recursos.
15. Cargar usuarios.
16. Modificar usuarios.
17. Consultar usuarios.
18. Consultar mis datos.
19. Modificar mis datos.
20. Cargar área de negocio.
21. Consultar área de negocio.
22. Modificar área de negocio.
23. Generar reportes.

Para obtener una descripción más detallada de los casos de uso del sistema *Flex Project Management* se puede consultar el Apéndice F: Modelo de Casos de Uso.

En la fase de inicio también se creó un documento denominado Plan de Desarrollo del Software, en el cual se especifica la duración estimada de cada fase de desarrollo del proyecto y las actividades que se realizan en cada fase de la metodología. Se explican las iteraciones que se llevan a cabo en cada fase y los productos de las mismas. Para consultar en detalle el documento de

Plan de Desarrollo del Software dirigirse al Apéndice A: Plan de Desarrollo del Software.

Adicionalmente se generó un documento denominado Glosario, en el cual se definen los términos que pueden resultar confusos al lector. Para consultar este documento dirigirse al Apéndice C: Glosario.

IV.2 Fase de Elaboración

La fase de elaboración para el desarrollo del sistema *Flex Project Management* comienza con revisar cuidadosamente los documentos generados en la fase de inicio, con el fin de comprobar que éstos realmente estén acorde con las necesidades y requerimientos del sistema. Luego se realizan las cuatro iteraciones para lograr un diseño refinado del sistema, que contenga todos los elementos necesarios para servir de apoyo en el proceso de construcción del sistema. A continuación se detallan cada una de las iteraciones realizadas.

Iteración 1

La primera iteración tuvo como objetivo la descripción de todos los casos de uso del sistema. Se generaron los documentos de especificaciones de casos de uso. Además, se generaron todos los diagramas necesarios para comprender y desarrollar mejor el sistema: diagramas de actividades, modelo de dominio, diagrama de clases, diagramas de secuencia, diagramas de navegación. Finalmente, se generó el modelo de datos del sistema.

Diagrama de Casos de Uso

En la fase de inicio se presentó una lista con los casos de uso y actores del sistema.

Para entrar más en detalle en cuanto a los casos de uso se refiere, se realizó una descripción completa de los mismos. Para ello se utilizó el producto de especificaciones de casos de uso que provee RUP.

Luego de tener la descripción de todos los casos de uso, se generó el modelo de casos de uso. Este modelo permite observar mejor la interacción de los actores con los casos de uso del sistema.

A continuación se muestra el diagrama de casos de uso de primer nivel, en él se puede observar la funcionalidad general del sistema y de que forma interactúan los actores con el mismo.

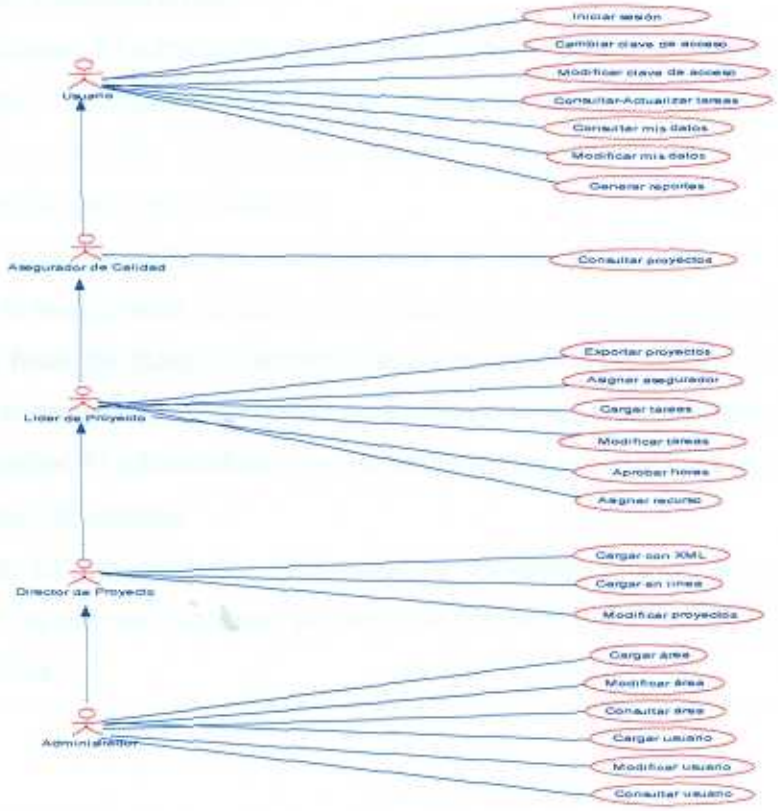

Figura 2: Diagrama de Casos de Uso del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Para cada caso de uso se utilizó el formato estándar propuesto por RUP para la descripción de casos de uso, el cual contiene una definición del caso de

uso, la información característica del mismo, el escenario principal de éxito, las extensiones del caso de uso, los requerimientos especiales y la planificación. A continuación se muestra la descripción del caso de uso Consultar Proyecto:

Caso de Uso # 7: Consultar Proyecto
Definición: El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad son quienes pueden consultar un proyecto completo. El administrador podrá consultar todos los proyectos que están en el sistema y el director de proyecto, el líder de proyecto y el asegurador de calidad podrán consultar solo los proyectos que ellos están gestionando.
Información Característica: **Precondiciones:** El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad deben haber ingresado a la página principal de *Flex Project Management*, además, el proyecto a consultar ya debe estar previamente cargado en el sistema. **Condición final de éxito:** El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad realizan la consulta exitosamente. **Condición final de fallo:** El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad no pudieron realizar la consulta. **Actor primario:** El administrador, el director de proyecto, el líder de proyecto y el asegurador de calidad **Disparador:** El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad ingresan al menú de Proyectos y seleccionan el link de "Ver Proyectos".

Escenario Principal de Éxito: 1. El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad ingresan al menú de Proyectos y seleccionan el link de "Ver Proyectos". 2. El sistema muestra una lista con todos los proyectos cargados en el sistema. 3. El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad presiona el botón de "Aceptar". 4. El sistema regresa a la página principal del actor.
Extensiones: 3a El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad presiona el botón de "Cancelar". 3a 1 El sistema regresa a la pagina principal del actor. 3b. El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad presionan el link de "Ver Tareas". 3b.1 El sistema muestra una lista con todas las tareas asociadas al proyecto seleccionado. 3c. El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad seleccionan el link de "Recursos Asociados". 3c.1 El sistema muestra una lista con todos los recursos asociados al proyecto seleccionado.
Requerimientos Especiales: 1. El administrador, el director de proyecto, el líder de proyecto o el asegurador de calidad deben ser usuarios validos en el sistema y su estado debe ser activo. 2. El proyecto que se desea consultar debe estar previamente cargado en el sistema.
Planificación: * Prioridad: alta. o Fecha de entrega: 23/02/2007

Tabla #6: Plantilla de especificaciones del caso de uso Consultar Proyecto

Fuente: Elaboración Propia, 2007

Para consultar las especificaciones de los casos de uso restantes dirigirse al Apéndice G: Especificación de Casos de Uso.

Diagramas de Actividades

Un diagrama de actividades muestra la serie de tareas que deben ser realizadas en un caso de uso, así como las distintas rutas que pueden irse desencadenando en el caso de uso para auxiliar a los miembros del equipo de desarrollo y ayudar a entender como es utilizado el sistema y como reacciona en determinados eventos.

Para *Flex Project Management* los diagramas de actividades constituyen una herramienta de suma importancia, pues es a través de ellos que se define gráficamente el curso que tendrá el caso de uso. Estos diagramas ayudan mucho en el momento de generar la interfaz del sistema.

Para cada caso de uso del sistema *Flex Project Management* se generó un diagrama de actividad. A continuación de muestra el diagrama de actividad correspondiente al caso de uso de Consultar Proyecto.

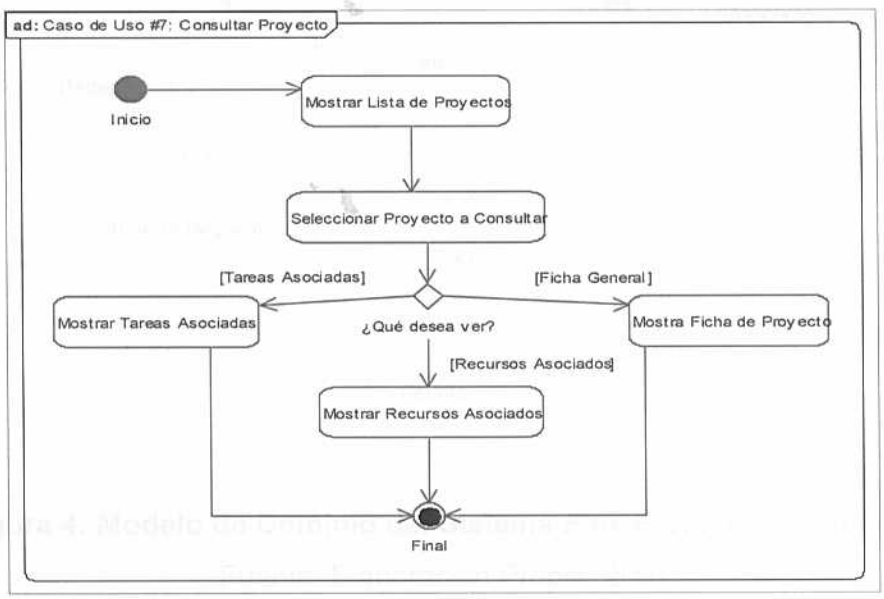

Figura 3: Diagrama de Actividad del Caso de Uso: Consultar Proyecto

Fuente: Elaboración Propia, 2007

Para consultar los diagramas de actividad generados, referirse al Apéndice I: Diagramas de Actividades.

Modelo de Domino

El modelo de dominio pretende delimitar el dominio del sistema, es decir, definir exactamente cuál será el alcance del mismo.

En el desarrollo del sistema *Flex Project Management*, antes de la realización de los diagramas de secuencia y los diagramas de clases se definió el modelo de dominio.

A continuación se muestra el modelo de dominio del sistema *Flex Project Management*.

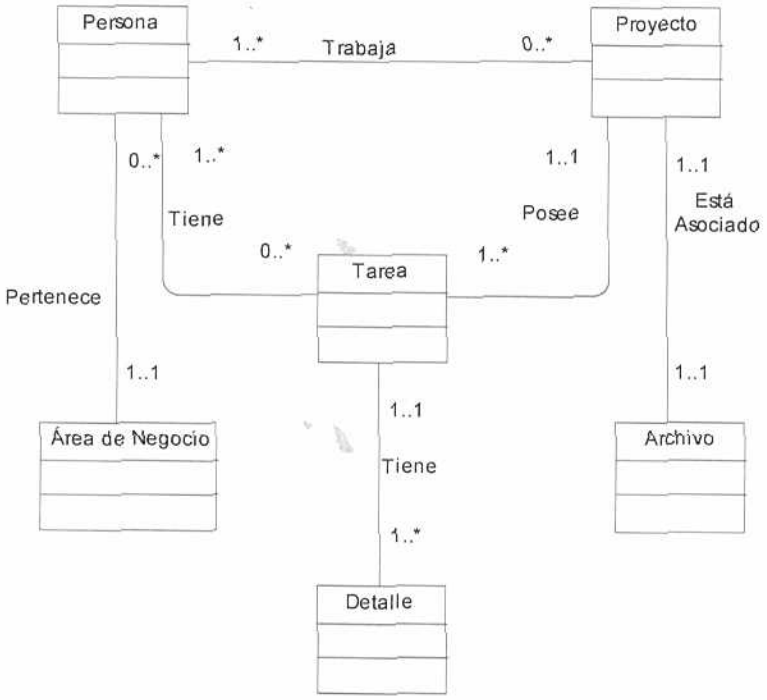

Figura 4: Modelo de Dominio del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Diagramas de Clase

El diagrama de clases recoge las clases de objetos y sus asociaciones. En este diagrama se representa la estructura de cada uno de las clases del sistema y sus relaciones con las demás clases.

Para *Flex Project Management* los diagramas de clases representan la base para iniciar el desarrollo del código del sistema.

A continuación se presenta el diagrama de clases de la herramienta *Flex Project Management*, donde se puede visualizar las relaciones entre las clases del sistema.

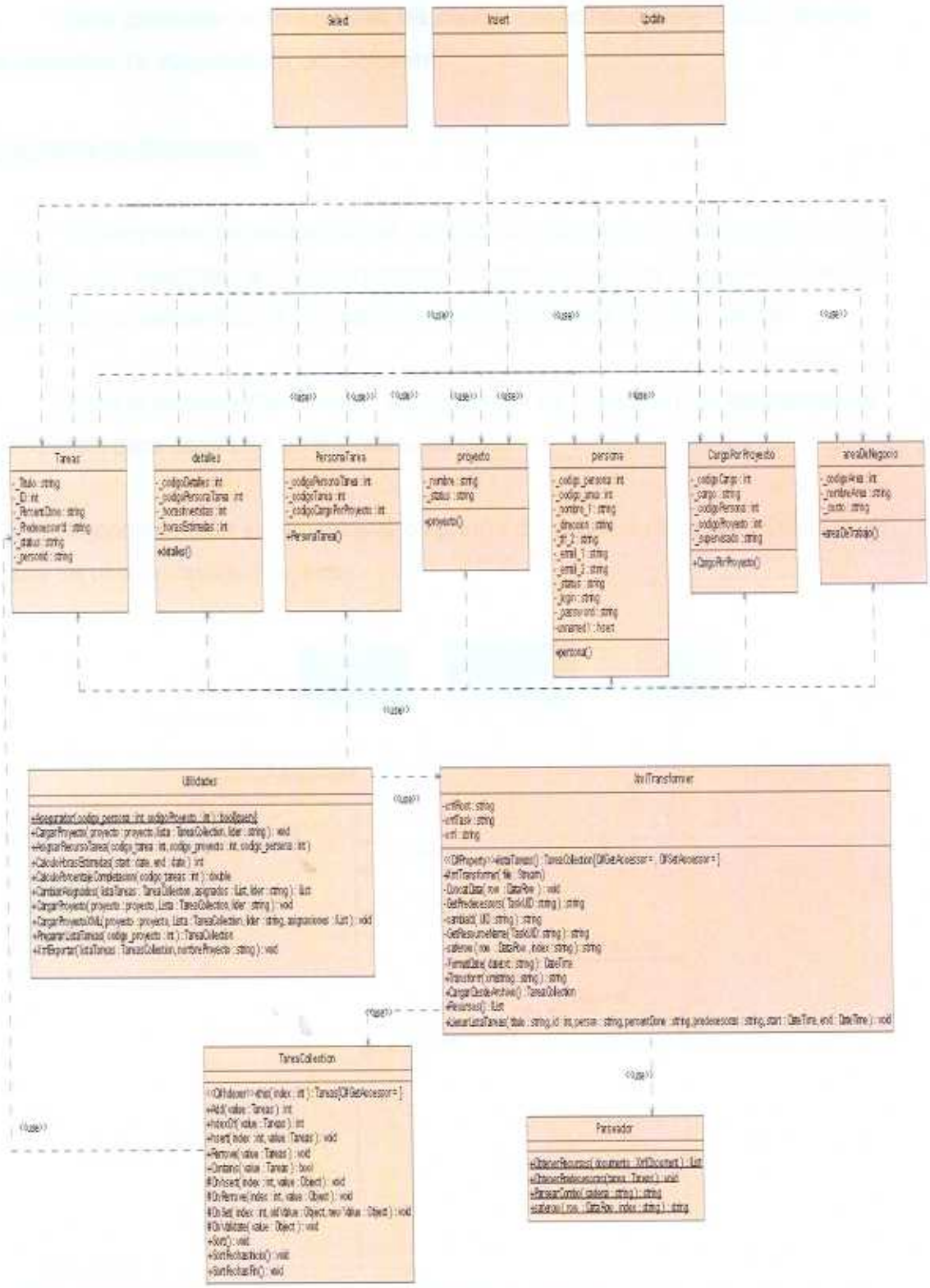

Figura 5: Diagrama de Clases del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Para consultar los diagramas de clases específicos generados, dirigirse al Apéndice H: Arquitectura del Software.

Diagrama de Secuencia

El diagrama de secuencia es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico de los objetos haciendo énfasis en la secuencia de los mensajes intercambiados por los objetos.

Para el sistema *Flex Project Management* se realizaron los diagramas de secuencia para cada uno de los casos de uso.

A continuación se muestra el diagrama de secuencia correspondiente al caso de uso Consultar Proyecto.

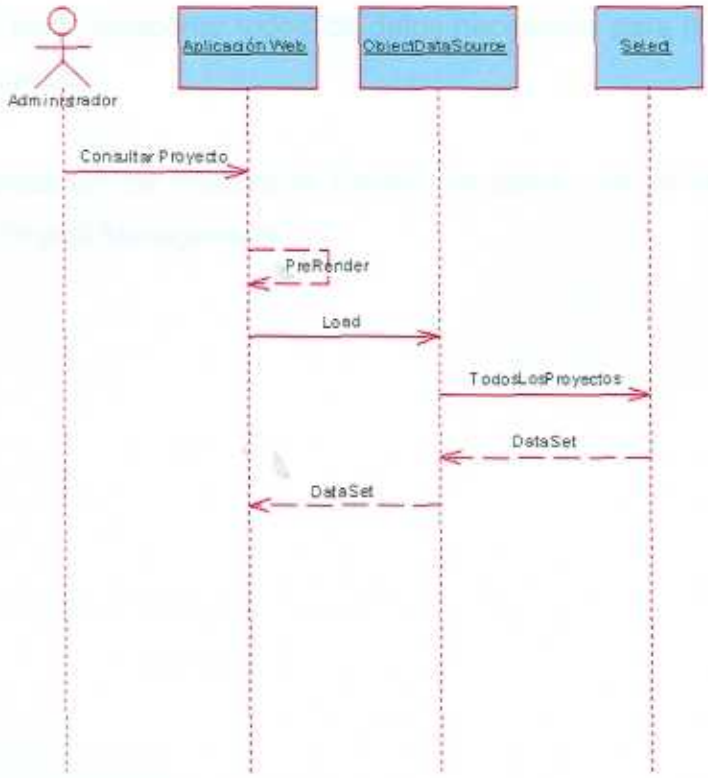

Figura 6: Diagrama de Secuencia del Caso de Uso: Consultar Proyecto

Fuente: Elaboración Propia, 2007

Para consultar los diagramas de secuencia generados para el sistema *Flex Project Management*, dirigirse al Apéndice J: Diagramas de Secuencia.

Modelo de datos

El objetivo principal del modelo de datos es la representación y definición de todos los datos que se introducen, almacenan, transforman y producen dentro del sistema. Es un medio para comunicar el significado de los datos, las relaciones entre ellos y las reglas de negocio del sistema.

Uno de los objetivos específicos del desarrollo del sistema *Flex Project Management* es diseñar y crear una base de datos para el almacenamiento de la información requerida por el sistema.

Para cumplir con este objetivo, se diseñó un modelo de datos que pudiera servir para almacenar todos los datos necesarios para llevar a cabo la gestión de proyectos

A continuación se muestra el modelo de datos que se definió para el sistema *Flex Project Management*.

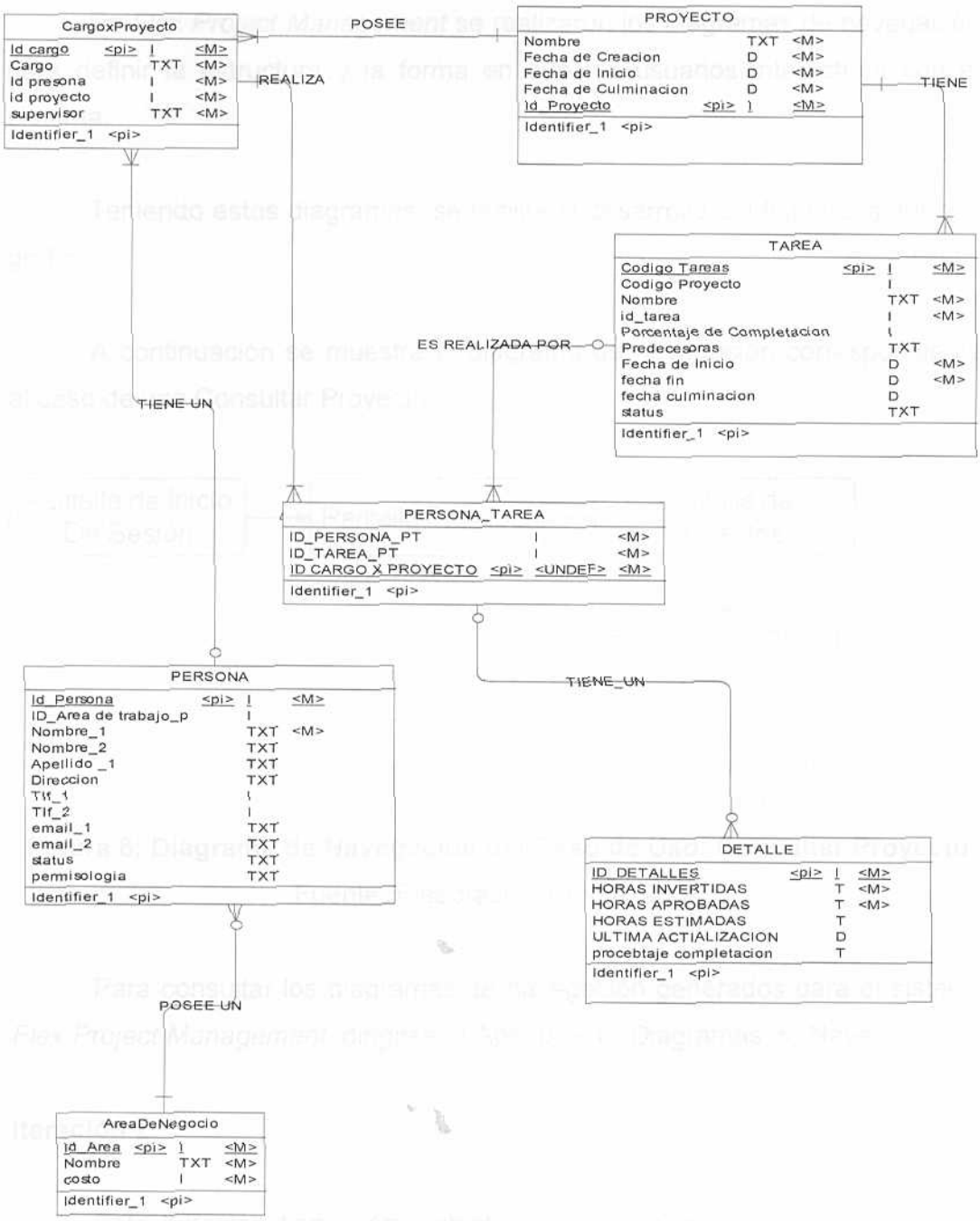

Figura 7: Modelo de Datos del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Diagrama de Navegación

Los diagramas de navegación se utilizan para mostrar la interacción de la interfaz gráfica con los usuarios.

En *Flex Project Management* se realizaron los diagramas de navegación para definir la estructura y la forma en que los usuarios interactúan con el sistema.

Teniendo estos diagramas, se facilita el desarrollo del flujo de la interfaz gráfica.

A continuación se muestra el diagrama de navegación correspondiente al caso de uso Consultar Proyecto.

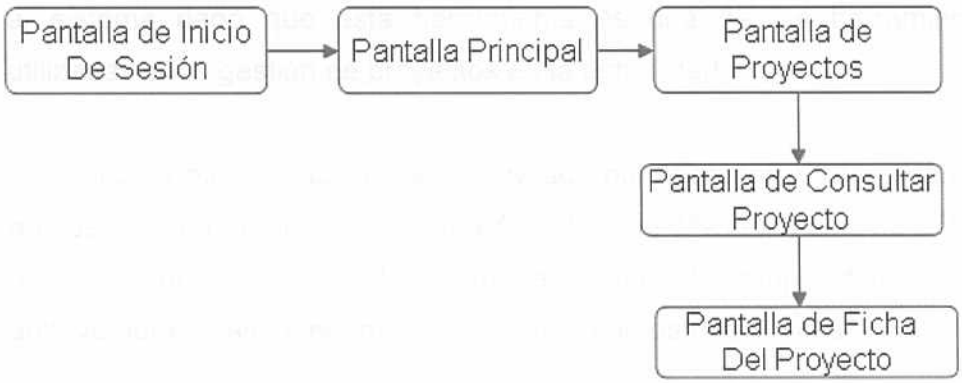

Figura 8: Diagrama de Navegación del Caso de Uso: Consultar Proyecto
Fuente: Elaboración Propia, 2007

Para consultar los diagramas de navegación generados para el sistema *Flex Project Management*, dirigirse al Apéndice K: Diagramas de Navegación.

Iteración 2

Esta iteración tiene como objetivo explicar el análisis y diseño de los archivos de entrada y salida del sistema *Flex Project Management*. Además, se explican las consideraciones que se tomaron en cuenta para la realización del flujo de aprobación de las horas invertidas por los recursos.

Diseño del Archivo de Entrada

Existen dos formas de cargar un proyecto, la primera, cargando el archivo XML predefinido con los datos del proyecto que se desea gestionar; y la segunda es creando el proyecto manualmente.

Los únicos archivos que pueden ser cargados en el sistema *Flex Project Management* son el archivo XML generado por *Microsoft Project* y el archivo generado por el mismo *Flex Project Management*. Se tomó el archivo XML generado por *Microsoft Project* como base para generar el archivo de entrada al sistema dado que esta herramienta es una de las herramientas más utilizadas en la gestión de proyectos en la actualidad.

La carga del archivo XML generado por *Microsoft Project* representa un proceso crítico dentro del sistema *Flex Project Management*, no sólo por ser uno de los objetivos específicos, sino también por la complejidad del parseo del archivo para su almacenamiento en la base de datos del sistema.

El archivo XML de *Microsoft Project* es un archivo complejo, compuesto por una serie de etiquetas y atributos. Para que este archivo XML pudiera ingresar al sistema *Flex Project Management* se identificaron las etiquetas y atributos más significativos y luego de tenerlos identificados se llevó a cabo el proceso de parseo del archivo.

Este proceso de parseo consiste en leer el árbol XML y almacenar los atributos en la base de datos para ser utilizados posteriormente cuando se necesite. A continuación se muestra el algoritmo que resume el parseo del archivo:

1.- Leer archivo

2.- Cargar documento XML

3.- Por cada tarea

    3.1.- Obtener información de la tarea

    3.2.- Por cada recurso

3.2.1.- Obtener información de recurso

3.3.- Obtener tareas predecesoras

3.4.- Por cada predecesora

3.4.1.- Obtener información de las tareas predecesoras

4.- Retornar lista de las tareas

Para mayor información sobre el archivo de entrada consultar el Apéndice L: Archivo de Entrada.

Diseño del Archivo de Salida

El sistema *Flex Project Management* ofrece la capacidad de guardar los proyectos cargados en el sistema en un archivo XML con un formato predefinido, utilizando un estándar abierto como lo es XML.

La generación del archivo XML de exportación o archivo de salida está comprendida en tres etapas fundamentales: la extracción de la información asociada al proyecto que se desea exportar, el procesamiento de la misma para la construcción del archivo XML y la escritura del archivo de exportación.

El archivo XML de exportación generado por el sistema *Flex Project Management* tiene el siguiente formato:

```
- <TODOLIST PROJECTNAME="Nombre del Proyecto">
    <TASK TITLE="Nombre Tarea 1" ID="0" PERSON="Recurso 1,Recurso 2" PID="123,132"
      PERCENTDONE="45" PREDECESSORSID="" STARTDATESTRING="20/02/2007"
      FINISHDATESTRING="20/02/2007" />
    . . .
    <TASK TITLE="Nombre Tarea n" ID="0" PERSON="Recurso n,Recurso m" PID="###,###"
      PERCENTDONE="%" PREDECESSORSID="n-1" STARTDATESTRING="dd/mm/YYYY"
      FINISHDATESTRING="dd/mm/YYYY" />
  </TODOLIST>
```

Figura 9: Formato del Archivo XML de Exportación de *Flex Project Management*

Fuente: Elaboración Propia, 2007

El archivo de exportación está compuesto de las siguientes etiquetas:

TodoList: es la etiqueta padre del archivo, está compuesta por un atributo denominado ***ProjectName***, el cual contiene el nombre del proyecto exportado.

Task: son las etiquetas hijas de ***TodoList***, cada una representa una tarea del proyecto a exportar. Está compuesto por los siguientes atributos:

Title: contiene el nombre de la tarea.

Id: es el identificador de la tarea.

Person: contiene el nombre del recurso o recursos asociados a la tarea.

Pid: contiene los identificadores del recurso o recursos asociados a la Tarea. La relación entre ***Person*** y ***Pid*** es uno a uno respectivamente.

PercentDone: contiene el porcentaje de completación de la tarea. En caso de que la tarea sea realizada por dos o más recursos, este porcentaje será el promedio de completación de los recursos que intervengan en la realización de la misma.

PredecessorId: contiene el identificador de la tarea que la precede.

StartDateString: contiene la fecha de inicio de la tarea.

FinishDateString: contiene la fecha de culminación de la tarea.

Diseño del flujo de aprobación de horas

Cada recurso involucrado en el sistema tiene la capacidad de registrar las horas que ha trabajado en una tarea específica. El administrador, el director de proyectos y el líder de proyectos son los encargados de verificar y aprobar si las horas que se invirtieron en esa tarea son en realidad las horas que se habían estimado al comienzo de la misma.

El director de proyectos y el líder de proyectos forman parte directa del proceso de aprobación, mientras que el administrador tiene el privilegio de llevar a cabo el proceso aunque no es necesaria su intervención directa, dada su condición de administrador del sistema completo.

Antes de describir el flujo que se llevará a cabo para el proceso de aprobación es importante definir ciertos términos y consideraciones en cuanto a las variables que intervienen en el mismo.

- Horas estimadas: se refiere a las horas en las que se pretende culminar la tarea, en función de los días de desarrollo de la misma. Cuando se hace la carga de la tarea se le asigna una fecha de inicio y una fecha de culminación estimada para su completación; a partir de esto, el sistema calcula las horas estimadas de completación de la tarea en base a ocho horas de trabajo diarias.

- Horas invertidas: los recursos que tienen asignadas tareas dentro de un proyecto tienen la responsabilidad de llevar la bitácora de desarrollo de las mismas. Para registrar las horas de desarrollo de la tarea se utiliza el caso de uso Consultar/Actualizar tareas, en él se registran las horas de desarrollo de esa tarea, denominadas para efectos del sistema horas invertidas.

- Horas aprobadas: se refiere a las horas de desarrollo consideradas válidas para efectos de calcular su costo.

En un escenario de desarrollo perfecto, la cantidad de horas estimadas, horas invertidas y horas aprobadas son las mismas. En el caso de que la cantidad de horas invertidas en una tarea superen a la cantidad de horas estimadas, el administrador, el director de proyectos o el líder de proyectos podrán aprobar las horas extras invertidas en esa tarea para efectos de calcular su costo. Si se presentara el caso contrario, es decir, la cantidad de horas invertidas en el desarrollo de la tarea es menor a la cantidad de horas estimadas, entonces se tomará como horas aprobadas a las horas invertidas en dicho desarrollo.

El sistema *Flex Project Management* considera flujo de trabajo de aprobación a la capacidad que tienen el director de proyectos y el líder de proyectos de modificar las horas aprobadas por un recurso en el desarrollo de una tarea específica dentro de un proyecto.

A continuación se muestra gráficamente como se lleva a cabo el flujo de aprobación de horas:

Figura 10: Flujo de aprobación de horas

Fuente: Elaboración Propia, 2007

Los recursos asignados a un proyecto llevan a cabo las tareas que tienen asignadas, luego, registran las horas invertidas en la realización de esas tareas en la bitácora del proyecto. El director o el líder de proyecto verifican esas horas y de estar de acuerdo las aprueban.

Iteración 3

El objetivo de esta iteración es definir la arquitectura que soportará al sistema. A continuación se explica detalladamente esta arquitectura.

Arquitectura del Software

El desarrollo del sistema *Flex Project Management* está basado en una arquitectura de 3 capas: capa de presentación, capa de negocio y capa de acceso a datos. Esta arquitectura ofrece grandes beneficios cuando se trata de desarrollo de aplicaciones Web.

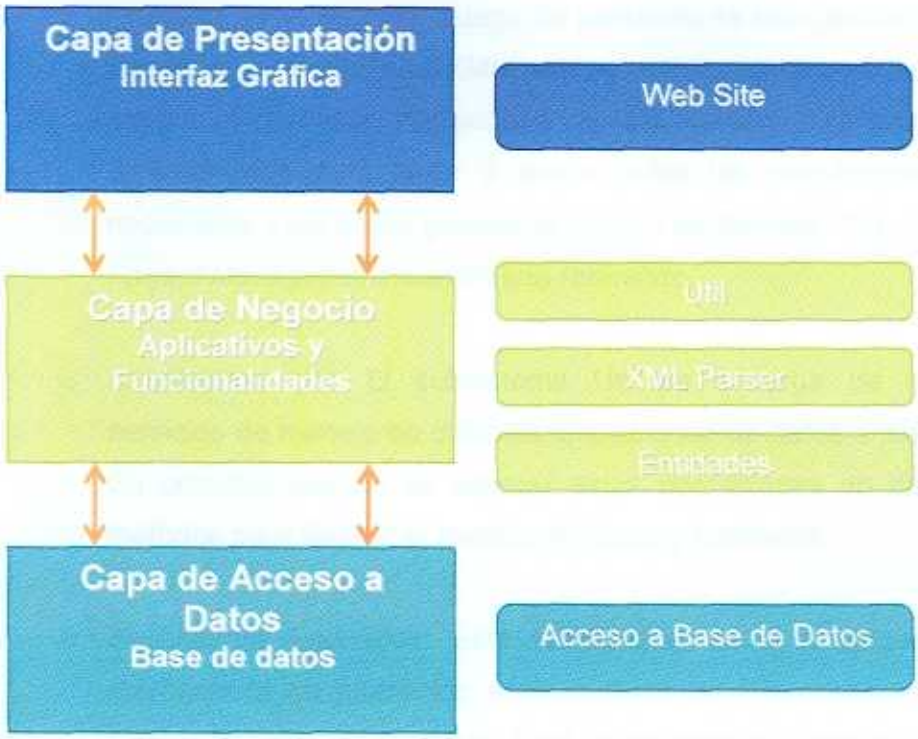

Figura 11: Arquitectura del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Capas

* Capa de Presentación: La capa de presentación es la capa que ve el usuario. Su principal objetivo es capturar y comunicar información. Esta capa se comunica únicamente con la capa de negocio. Cuando se hace referencia al Web Site estamos englobando todas la paginas Web (interfaz gráfica) que interactúan directamente con los usuarios.

* Capa de Negocio: Se denomina capa de negocio ya que es aquí donde se establecen todas las reglas que deben cumplirse. Es la capa donde residen los programas que se ejecutan, recibiendo las peticiones del usuario y enviando las respuestas tras el proceso. Está compuesta por tres subsistemas:

    - *Subsistema XML Parser*: Este subsistema es el encargado de llevar a cabo el proceso de parsear el archivo XML que se

introduce en el sistema. Luego de parsearlo se reorganizan todos los datos del archivo XML original y se almacena en una estructura de datos manejable y de fácil acceso, esta estructura será utilizada para llevar a acabo todas las transformaciones necesarias y así poder generar el archivo de formato XML de *Flex Project Management* cuando sea requerido.

- *Subsistema Util:* El subsistema Util se encarga de ofrecer servicios de manejo de distintas operaciones de datos al sistema. Su principal función es separar estas operaciones en distintos métodos para facilitar el manejo de datos y funciones.

- *Subsistema Entidades:* Este subsistema es el encargado de mantener la comunicación entre la Capa de Presentación (Web Site) y la Capa de Negocio. Está compuesto por estructuras de datos que facilitan el flujo de información entre estas dos capas.

* Capa de Acceso a Datos: Es la capa donde residen los datos. Está formada por uno gestor de base de dato que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio. El subsistema de Acceso a Base de datos es el encargado de llevar a cabo todas las operaciones de lectura y escritura en la base de datos.

Para obtener mayor información acerca de la arquitectura del sistema consultar el Apéndice H: Arquitectura del Software.

Arquitectura de despliegue

El despliegue del sistema *Flex Project Management* se ejecutará sobre un servidor, el cual ofrece sus servicios a los usuarios que lo soliciten. En este servidor se encontrarán ubicadas tanto las páginas Web como la base de datos en la que se guarda la información del sistema. Cuando un usuario necesite acceder al sistema solamente tendrá que contar con un explorador Web y

dirigirse a la página principal del sistema *Flex Project Management* para completar el proceso de inicio de sesión y comenzar a utilizar la herramienta.

Figura 12: Diagrama de Despliegue del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

Iteración 4

Esta iteración tiene como objetivo explicar los componentes de la aplicación y las consideraciones de la interfaz de usuario del sistema.

Diseño de los componentes del sistema

Para facilitar el desarrollo del sistema *Flex Project Management* se dividió en componentes. Cada componente está compuesto por una serie de casos de usos; esta división ayuda a entender el diseño de cada uno de los componentes y de este modo se logra una implementación acorde y centrada en la funcionalidad específica de cada caso de uso.

Esta división no significa que los componentes estén aislados uno de los otros, es simplemente una división que se hace con el fin de organizar de cierto modo el desarrollo del sistema.

Figura 13: Componentes del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

A continuación se explican cada uno de los componentes:

* Componente de Autentificación

Este componente está formado por todos los casos de uso que se encargan de una u otra forma de autentificar a los usuarios que ingresan al sistema: iniciar sesión, cambiar clave de acceso y olvidar clave de acceso.

* Componente de usuarios

En el componente de usuarios se lleva a cabo la gestión de usuarios. Está compuesto por los casos de uso de: cargar usuarios, modificar usuarios, consultar usuarios, consultar mis datos y modificar mis datos.

* Componente de Área de Negocio

El componente de área de negocio es uno de los componentes secundarios del sistema. En este componente se desarrollará todos aquellos casos de uso que tengan que ver con agregar una nueva área de negocio al proyecto: cargar área de negocio, modificar área de negocio y consultar área de negocio.

* Componente de Tareas

En el componente de tareas se lleva a cabo la gestión de las tareas de los proyectos. Está compuesto por los casos de uso de: cargar tareas, consultar/actualizar tareas, modificar tareas, aprobar horas invertidas y asignar recursos.

* Componente de Proyectos

En el componente de proyectos se lleva a cabo la gestión de proyectos. Está compuesto por los casos de uso de: cargar proyecto con XML, cargar proyecto en línea, modificar proyecto, consultar proyecto, exportar proyecto y asignar asegurador de calidad.

* Componente de Reportes

En el componente de reportes se desarrollan los casos de uso que soportan la gestión de reportes. Está compuesto por todos los reportes que se pueden generar en el sistema *Flex Project Management*, en los que se destacan:

- Mostrar el porcentaje de completación del proyecto, semanal y mensualmente.
- Mostrar las tareas completadas por un recurso antes del tiempo previsto.
- Mostrar las tareas completadas por proyecto.
- Mostrar las tareas no completadas por proyecto.
- Mostrar el costo de una tarea.
- Mostrar el costo de un proyecto según las horas aprobadas, invertidas y estimadas.
- Mostrar el costo de un recurso en base a un proyecto, dependiendo de las horas aprobadas, invertidas y estimadas.
- Mostrar las horas invertidas en un proyecto.

- Mostrar comparaciones entre tiempos estimados y tiempos reales.

Dependiendo del rol y de la permisología que tengan los usuarios cargados en el sistema podrán realizar ciertos reportes.

Interfaz de Usuario

La interfaz de usuario del sistema *Flex Project Management* se diseñó rigurosamente tratando de conservar siempre un ambiente agradable y amigable para cada uno de los usuarios que interactuaran con el sistema. Además, se tomaron en cuenta algunos de los estándares de interfaces Web propuestas por la W3C.

Los usuarios tienen la libertad de salir del sistema cuando lo deseen y de manera rápida. Se tomó en cuenta el uso de iconos estándares para indicar a los usuarios situaciones o acciones que pudieran realizar dentro del sistema. Se realizó un diseño cuidadoso de la interfaz para que minimice la ocurrencia de errores, sobretodo en cuanto a la completación de formularios se refiere. El diseño minimalista de la interfaz es uno de los puntos claves, se utilizaron colores que no cansasen la vista del usuario y se evitó recargar las páginas con datos innecesarios.

A continuación se muestra la pantalla principal del sistema. Esta pantalla está dividida por zonas para su explicación. La zona amarilla, identificada con la letra A, es donde se ubica el logo del sistema. La zona azul claro, identificada con la letra B, es el área de botones de "Ir a la página principal" y "cerrar sesión". La zona morada, identificada con la letra C, es la zona donde se ubica la fecha. La azul oscuro, identificada con la letra D es donde se ubica el menú con todas las funciones y servicios que ofrece el sistema. La zona verde identificada con la letra E es la zona de trabajo del sistema.

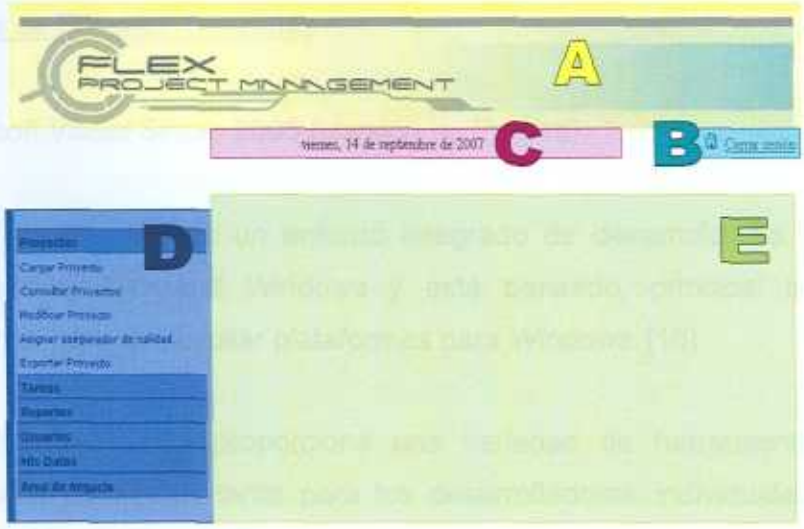

Figura 14: Pantalla principal del Sistema *Flex Project Management*

Fuente: Elaboración Propia, 2007

IV.3 Fase de Construcción

La fase de construcción para el sistema *Flex Project Management* se basa en el desarrollo e implementación de todos los casos de uso diseñados en la fase anterior. La construcción final del sistema es el objetivo de esta fase.

Esta fase fue dividida en cinco iteraciones:

* En la primera iteración se definieron todas las herramientas utilizadas en el desarrollo del sistema.
* En la segunda iteración y hasta la cuarta iteración se desarrollaron los componentes del sistema, en cuanto a la construcción de los mismos se refiere y las pruebas realizadas.
* En la quinta iteración se integraron los componentes y se realizaron las pruebas necesarias.

Iteración 1

El principal objetivo de esta iteración fue definir las herramientas utilizadas en el desarrollo del sistema *Flex Project Management*.

Herramientas de Software Utilizadas

* *Microsoft Visual Studio* 2005 (Versión de Prueba)

Visual Studio .Net es un entorno integrado de desarrollo. Es para el sistema operativo *Microsoft Windows* y está pensado, principal pero no exclusivamente, para desarrollar plataformas para *Windows*. [16]

Visual Studio 2005 proporciona una variedad de herramientas que ofrecen amplios beneficios tanto para los desarrolladores individuales como para equipos de desarrollo. [16]

Para el desarrollo del sistema *Flex Project Management* se escogió este entorno de desarrollo ya que ayuda a crear interfaces amigables, acorta los tiempos de respuesta y se integra fácilmente a otras herramientas aportando así un escenario favorable para la portabilidad del sistema.

* *MySQL*

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño permite soportar una gran carga de forma muy eficiente. Además es software libre. [17]

Este gestor de base de datos es, probablemente, el gestor más usado en el mundo de software libre, debido a su gran rapidez y facilidad de uso. Es de gran aceptación debido, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. [17]

Las principales características de este gestor de bases de datos son las siguientes:

- Aprovecha la potencia de sistemas multiprocesadores, gracias a su implementación multihilo.

- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API`s en gran cantidad de lenguajes (C, C++, Java, PHP, etc.).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y claves, manteniendo un nivel muy bueno de seguridad. [17]

MySQL fue el manejador de base de datos seleccionado para el desarrollo de *Flex Project Management*, gracias a las características que ofrece.

* *AJAX.Net*

Es una magnifica tecnología que proporciona a los desarrolladores una base para crear aplicaciones basadas en la Web que ofrezcan una mejora en la usabilidad. *Ajax* tiene el poder de revolucionar el modo en que se diseñan las aplicaciones basadas en Web. [18]

Ajax permite simular el comportamiento de aplicaciones de escritorio en la Web. Su funcionamiento se basa en realizar la comunicación entre el navegador y el servidor pero de forma asíncrona. [18]

Iteración 2

Desarrollo de los Componentes de Autentificación, Usuarios y Área de Negocio

El objetivo principal de esta iteración es desarrollar los componentes de autentificación, usuarios y área de negocio.

Para realizar un desarrollo acorde a las especificaciones y necesidades de los usuarios, se revisó detalladamente toda la documentación existente sobre estos componentes. Luego de refrescar los requerimientos, se procedió a llevar a cabo el desarrollo del código de dichos componentes.

Durante esta iteración se llevó a cabo el desarrollo del código de los componentes de Autentificación, Usuarios y Área de Negocio del sistema *Flex Project Management.*

Para estos componentes se desarrollaron todas las clases y métodos necesarios para cumplir con las exigencias de cada uno, definidas en capítulos anteriores.

Durante el desarrollo de cada uno de los componentes se llevaron a cabo las pruebas de sistema necesarias para asegurar el perfecto funcionamiento de los mismos.

Como resultado, se obtuvieron los tres componentes: Autentificación, Usuarios y Área de Negocio, altamente estables, consistentes y acorde con las necesidades del sistema *Flex Project Management.*

A continuación se muestran las pantallas principales de cada uno de los componentes desarrollados en esta iteración:

Figura 15: Pantalla principal del componente de Autentificación

Fuente: Elaboración Propia, 2007

Figura 16: Pantalla principal del componente de Usuarios

Fuente: Elaboración Propia, 2007

Figura 17: Pantalla principal del componente de Área de Negocio

Fuente: Elaboración Propia, 2007

Iteración 3

Desarrollo de los Componentes de Tareas y Proyectos

Al igual que la iteración 2, el objetivo de la iteración 3, es desarrollar el componente de tareas y el componente de proyectos. Estos son los componentes más críticos para el sistema *Flex Project Management*.

Antes de comenzar a desarrollar el código de dichos componentes se realizó una revisión completa de todos los productos y diagramas generados en

las fases anteriores, con el fin de refrescar los requerimientos y especificaciones del sistema. Luego de la revisión de la documentación existente acerca del sistema y en específico de los componentes a desarrollar, se comenzó el desarrollo del código; primero se desarrolló el componente de tareas, ya que era base para posteriormente desarrollar el componente de proyectos.

El componente de tareas contiene uno de los casos de uso críticos del sistema: Aprobar Horas Invertidas en una Tarea por un Recurso. Al finalizar el desarrollo del componente de tareas, se procedió con el desarrollo del componente de proyectos.

El componente de proyectos está compuesto por una serie de casos de uso entre los cuales se encuentran los dos casos de uso más críticos del sistema: Cargar Proyecto con un XML predefinido y Exportar Proyecto. Estos casos de uso conforman la esencia del sistema *Flex Project Management*. Su desarrollo se llevó a cabo cumpliendo con los requerimientos y condiciones definidas en las fases anteriores del desarrollo.

Como resultado de esta iteración se obtuvo el desarrollo completo de los componentes de Tareas y Proyectos del sistema *Flex Project Management*. A continuación se muestran las pantallas principales de estos componentes:

Figura 18: Pantalla principal del componente de Tareas

Fuente: Elaboración Propia, 2007

Figura 19: Pantalla principal del componente de Proyectos

Fuente: Elaboración Propia, 2007

Iteración 4

Desarrollo del Componente de Reportes

El objetivo de esta iteración es desarrollar el componente de reportes del sistema *Flex Project Management*. Una vez completados los componentes anteriores se procedió al desarrollo de los reportes. En fases anteriores del sistema, se definieron los reportes que ofrecería el sistema *Flex Project Management*; por esta razón y para refrescar la información acerca de los reportes, se revisó la documentación existente y se procedió al desarrollo formal del componente.

Para cada uno de los reportes se llevaron a cabo las pruebas de sistema necesarias para garantizar la consistencia de la información presentada.

Como resultado, se obtuvieron todos los reportes necesarios para apoyar el desarrollo de una buena gestión de proyectos. A continuación se muestra la pantalla de uno de los reportes principales del sistema:

Figura 20: Pantalla de reporte de tiempo total de un proyecto

Fuente: Elaboración Propia, 2007

Iteración 5

Integración y Pruebas

La quinta iteración tiene como objetivo revisar cada uno de los componentes desarrollados anteriormente y confirmar el funcionamiento correcto de los mismos.

Durante el desarrollo individual de cada componente, se verificó que cumpliera con los requisitos de funcionamiento; ahora, en esta fase, se llevan a cabo las pruebas de sistema necesarias para asegurar el funcionamiento perfecto del sistema *Flex Project Management*.

Las pruebas realizadas fueron pruebas de sistema, las cuales tienen como objetivo ejercitar profundamente el sistema comprobando la integración del mismo globalmente y verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen.

IV.4 Fase de Transición

La fase de transición se enfoca en ofrecer al usuario la oportunidad de contar con un manual del sistema *Flex Project Management*, que ayudará al

aprendizaje de la herramienta. Además, se empaquetará el sistema y su documentación en un CD para ser distribuido.

Manual de Usuarios

Se redactó un manual de usuarios, en el cual se describen, con ejemplos gráficos, la utilización de las funciones del sistema *Flex Project Management*. Para comprender este manual no hace falta tener conocimientos previos en el área de gestión de proyectos. Se utilizaron ejemplos gráficos para explicar de manera más fácil las funcionalidades críticas del sistema. Para consultar el manual de usuarios referirse al Apéndice M: Manual de Usuarios.

Para finalizar esta fase, se empaquetó el producto final, compuesto por la documentación, el manual de instalación, el manual de usuarios y la aplicación en si, en un CD para ser entregado.

CAPÍTULO V. RESULTADOS

Este capítulo, denominado Resultados, tiene como objetivo presentar, de manera puntual, todos los resultados obtenidos con el desarrollo del sistema *Flex Project Management*.

A continuación se exponen y analizan los resultados.

* Se identificaron las herramientas existentes en el mercado que ofrecen un servicio parecido al que pretende ofrecer *Flex Project Management*, se analizaron y a partir del análisis se pudieron identificar las virtudes y debilidades de estas herramientas, ayudando así a definir un sistema *Flex Project Management* acorde a las necesidades del mercado.

* Se identificó a *Microsoft Project*, como una de las herramientas más utilizadas en el ámbito de la planificación de proyectos. Además, la herramienta ofrece la opción de generar archivos de planificación en formato XML. Es por esta razón que se tomó la decisión de adoptar el archivo XML generado por esa herramienta y utilizarlo como una de las entradas al sistema *Flex Project Management*, logrando así una integración satisfactoria y útil para los usuarios.

* Se diseñó e implementó un sistema Web que permitiera gestionar el tiempo que invierte un recurso humano en el desarrollo de un proyecto.

* Se desarrolló un sistema en el cual la información es almacenada en una base de datos, permitiendo de esta manera acceder a la información de manera rápida y fácil cuando el usuario lo necesite. Para el almacenamiento, se definieron las estructuras de datos necesarias, que pudieran soportar la gestión de proyectos.

* Se diseñó una interfaz Web para el sistema, que permitiera a los usuarios interactuar con la herramienta de una manera amigable y sencilla.

* Se desarrolló un sistema que ofrece distintos roles para acceder al mismo. Para esto, se definieron cinco roles específicos: administrador del sistema, director de proyectos, líder de proyectos, asegurador de calidad y usuarios básicos; cada uno de ellos con una permisologia diferente.

* Se definió un archivo XML, propio del sistema *Flex Project Management*, para almacenar y exportar los datos correspondientes a un proyecto cargado en el sistema.

* Se desarrolló un sistema que pudiera cargar archivos de planificación en formato XML en línea o a través de un archivo predefinido. Los archivos predefinidos que se pueden cargar en el sistema son: los archivos de planificación generados por *Microsoft Project*, o los archivos de exportación generados por *Flex Project Management*; cualquier otro archivo que no pertenezca a esa familia, tendrá que ser gestionado en línea, es decir, cargado manualmente en el sistema.

* Se desarrolló un procedimiento que pudiera identificar si el archivo que se está cargando en el sistema es un archivo proveniente de *Microsoft Project*, un archivo propio del sistema *Flex Project Management* o cualquier otro archivo no compatible con *Flex Project Management*.

* Se desarrolló un sistema que ofrece a sus usuarios la posibilidad de llevar la bitácora de sus tareas. Todo usuario que realice alguna tarea dentro de un proyecto que este siendo gestionado por *Flex Project Management*, tendrá que llevar el control de las horas invertidas en el desarrollo.

* Se desarrolló e implementó un protocolo de flujo de trabajo para la aprobación final de las horas invertidas en la realización de una tarea.

* Se desarrolló e implementó un módulo de reportes del sistema, donde se generan los reportes predefinidos sobre los proyectos cargados en el sistema. Estos reportes son de gran ayuda a la hora de tomar decisiones estratégicas en cuanto a la realización y culminación de un proyecto.

CONCLUSIONES

Flex Project Management se desarrolló con el objetivo de ayudar a las empresas e instituciones académicas en el proceso de gestión de proyectos, minimizando los tiempos y optimizando los recursos que intervienen en el desarrollo de un proyecto.

A continuación se presentan, de forma puntual, las conclusiones obtenidas luego del proceso de desarrollo:

* Se desarrolló un sistema capaz de asistir la gestión de proyectos, ayudando no sólo a los directores de proyectos, sino también a todos los recursos que intervienen de una u otra forma en el proceso de desarrollo.

* La utilización de la metodología RUP, ayudó a desarrollar un sistema de forma incremental, permitiendo corregir errores anticipadamente y colaborando con obtener un producto final de alta calidad.

* Los productos que provee la metodología RUP facilitaron el análisis y el desarrollo del sistema. La utilización de los mismos ayudó a desglosar cada fase e iteración del desarrollo, facilitando así la realización del sistema y garantizando la ejecución incremental del mismo.

* Investigar sobre las distintas herramientas existentes en el mercado para la gestión de proyectos ayudó a definir una herramienta que se enfocara realmente en las necesidades de los usuarios.

* Se definieron cinco roles específicos: administrador del sistema, director de proyectos, líder de proyectos, asegurador de calidad y usuarios; cada uno de ellos con una permisologia diferente.

* Se desarrolló un sistema que permite cargar archivos de planificación en formato XML. Estos archivos son: el archivo XML generado por *Microsoft Project* y el archivo XML de exportación generado por *Flex Project Management*; en caso de no contar con ninguno de los archivos anteriormente nombrados, el sistema ofrece la posibilidad de cargar en línea el proyecto.

* Se diseñó e implementó un sistema Web que permitiera gestionar el control del tiempo invertido por los recursos en el desarrollo de un proyecto. Dada su característica de ser un sistema Web puede ser accedido desde cualquier estación de trabajo que este conectada a Internet o en su defecto a la Intranet respectiva.

RECOMENDACIONES

Una vez culminado el desarrollo del sistema *Flex Project Management* y habiendo analizado los resultados y conclusiones, a continuación se exponen una serie de recomendaciones generales del sistema, que pueden ayudar en un futuro a complementar la herramienta:

* Se recomienda para futuras versiones del sistema generar resultados con herramientas gráficas, en las que se pueda observar de una manera más fácil el comportamiento de los proyectos, tareas y recursos.

* Se sugiere definir para próximas versiones un calendario de trabajo acorde a las necesidades de la empresa, en el que se tomen en cuenta los días laborables y no laborables, ayudando así a generar una aproximación de costos más precisa.

* Se puede desarrollar un módulo de alertas, en el cual, se mantenga informado a los recursos del avance, cambios y aprobaciones relacionadas con el proyecto en el cual están trabajando. Además, se pueden generar alertas de tiempo de culminación de desarrollo de una tarea o de un proyecto en general, ayudando así al buen desempeño de los recursos.

* Se recomienda realizar encuestas periódicas en las que se pueda certificar el buen funcionamiento de la herramienta y poder mejorar así su usabilidad.

* Dado que *Flex Project Management* es una herramienta Web, se podrían tomar medidas en cuanto a la transferencia segura de datos e incluir seguridad en el envío de paquetes. Además se podrían encriptar las claves de acceso al sistema u otra información privada.

* Se recomienda para próximas versiones tomar en cuenta el control de costos, de esta manera se podrá determinar con más exactitud el tanto el precio de venta de un proyecto como la utilidad que este pueda generar.

BIBLIOGRAFÍA

[1] Pérez Marisol, *El sistema de control de gestión. Conceptos básicos para su diseño.* [Publicación en Línea]. Obtenido en: http://www.gestiopolis.com/canales/gerencial/articulos/67/siscontrges.htm. Consultado en: Febrero del 2006.

[2] ADR Infor S.L., Logroño, *Curso de MS Project 2003.* [Publicación en Línea]. Obtenido en: http://www.adrformacion.com/cursos/project/leccion1/tutorial2.html. Consultado en: Diciembre del 2006.

[3] Grupo de Gestión de la Tecnología. Universidad Politécnica de Madrid. [Publicación en Línea]. Obtenido en: http://www.getec.etsit.upm.es/docencia/gproyectos/gproyectos.htm. Consultado en: Febrero del 2006.

[4] Coello, Helkyn. *Más allá de la metodologías de desarrollo.* [Publicación en Línea]. Obtenido en: http://www.informatizate.net/articulos/mas_alla_de_las_metodologias_de_desarrollo_12102004.html. Consultado en: Febrero del 2006.

[5] IBM Corp. [Publicación en Línea]. Obtenido en: http://w303.ibm.com/transform/project/education/program_education.pdf. Consultado en: Diciembre del 2006.

[6] Whitten, Jeffrey L.; Bentley, Lonnie D.; Dittman, Kevin C. 2002, "*Systems Analysis and Design Methods*", Capítulo 2. Quinta Edición, McGraw Hill.

[7] IBM Corp. [Publicación en Línea]. Obtenido en: http://www-306.ibm.com/software/webservers/appserv. Consultado en: Diciembre del 2006.

[8] http://es.wikipedi.org/wiki/IIS Consultado en: Febrero de 2006.

[9] http://es.wiipedia.org/wiki/XML. Consultado en: Febrero del 2006.

[10] Jacobson, I.; Booch, Grady; Rumbaugh, James. 1999. *The Unified Modeling Language User Guide.* Editorial Addison Wesley. U.S.A.

[11] Kruchten, P.2002 *The Rational Unified Process: An Introduction.*

[12] Guerrero, Luis. *Rational Unified Process.* Universidad de Chile. [Publicación en Línea]. Obtenido en:
www.dcc.uchile.cl/~luguerre/cc61j/recursos/clase2.ppt Consultado en: Enero del 2007.

[13] Griman, Ana. 2000. "*Estilos Arquitectónicos para la Calidad de Sistemas de Información Empresariales*".

[14] Tournier, Christopher. *Implementing the RUP an Rational Tools in a Small Nonsoftware Project: Writing a Book.*

[15] Universidad Politécnica de Valencia. *Rational Unified Process (RUP).* Departamento de Sistemas Informáticos y computación. Valencia, España. [Publicación en Línea]. Obtenido en:
https://pid.dsic.upv.es/C1/Material/Documentos%20Disponibles/PFC-ETSIA-UPV%20Ejemplo1%20de%20aplicaci%C3%B3n%20de%20RUP.pdf
Consultado en: Enero del 2007.

[16] Microsoft Corporation. 2007 *Introducción a Visual Studio.* [Publicación en Línea]. Obtenido en:
http://msdn2.microsoft.com/es-es/library/fx6bk1f4(VS.80).aspx

[17] MySQL AB. 2007. *Panorámica del sistema de gestión de base de datos MySQL.* [Publicación en Línea]. Obtenido en:
http://dev.mysql.com/doc/refman/5.0/es/what-is.html

[18] McClure, Wallace B. ; Cate, Scott ; Glavich, Paul. 2007. *Ajax con ASP.Net*. Editorial Anaya Multimedia. Primera Edición.