



UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

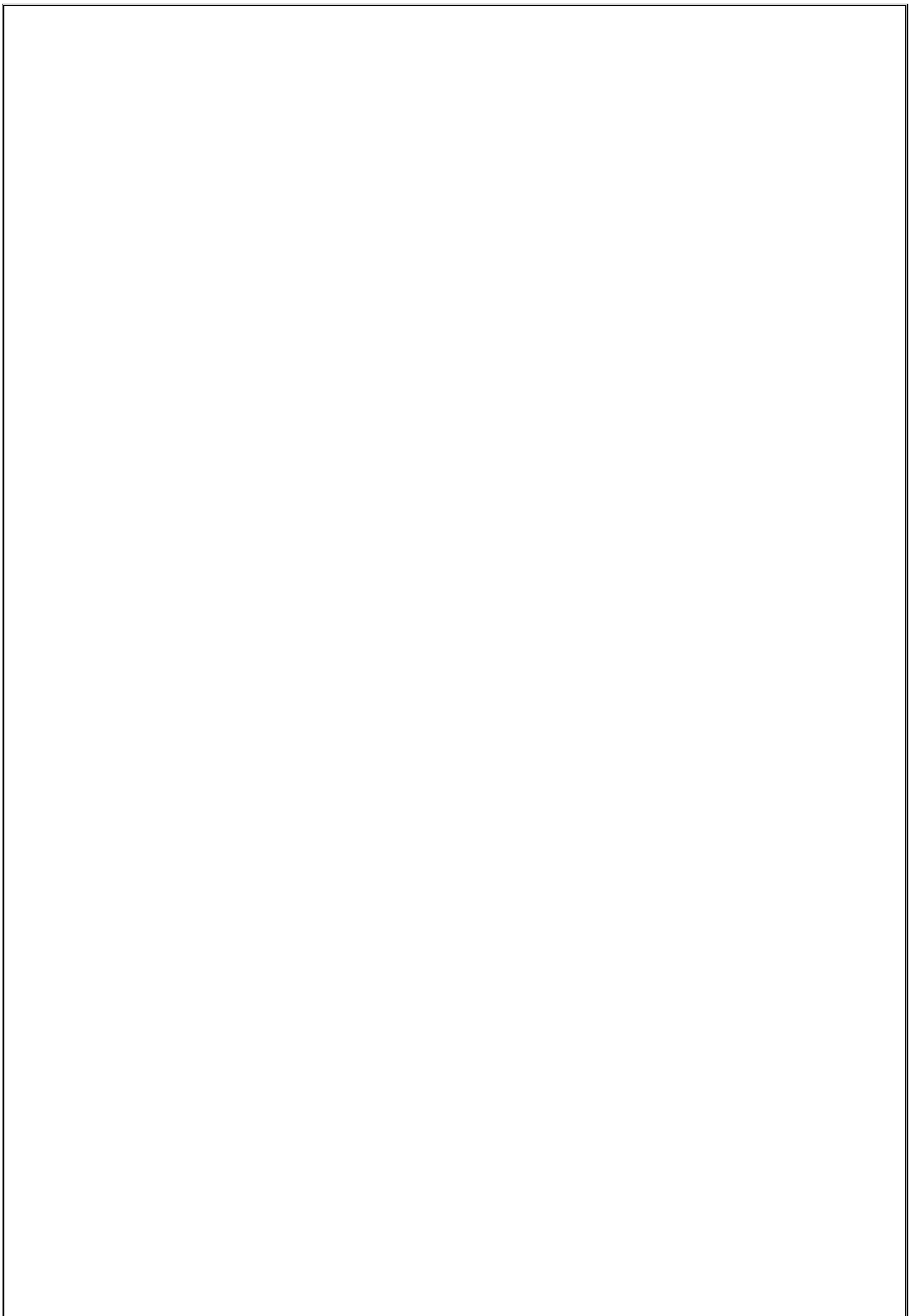
HERRAMIENTA DE REUTILIZACIÓN DE PRODUCTOS SOFTWARE BASADA EN PATRONES DE PRODUCTO PARA ENTORNOS COLABORATIVOS

Ingeniería Técnica en Informática de Gestión

Directora: Maribel Sánchez Segura

Tutor: José Arturo Mora Soto

Autor: David Gómez Sanabria



Quisiera agradecer a todas las personas que han contribuido en este proyecto, ya no sólo aportando sus conocimientos, profesionalidad y paciencia (María Isabel, Fuensanta y Arturo) sino dándome fuerzas y apoyo en los momentos difíciles...

...GRACIAS a mis padres por el esfuerzo realizado todos estos años, reforzando siempre mi educación y mostrándome "el camino" para ser una buena persona, a mi hermana por ser siempre tan "cariñosa" (tú serás la siguiente) y...

... GRACIAS a mi futura esposa Elisa por el apoyo y la implicación realizada durante todo el proyecto, sin ella no hubiera podido emplear tantas horas y tanta dedicación...

... Espero que no sea la última vez que tengo la oportunidad de demostrar mis conocimientos, por lo que espero seguir evolucionando como profesional en esta rama de la Ingeniería que me ha aportado tantas experiencias y buenos momentos...

Fdo.: David Gómez Sanabria



ÍNDICE DE CONTENIDOS

TABLA DE CONTENIDO.....	1
ÍNDICE DE CONTENIDOS	7
CAPITULO 1. INTRODUCCIÓN	12
1.1 CONTEXTO.....	12
1.2 DESCRIPCIÓN DEL PROYECTO	14
1.3 OBJETIVOS.....	16
1.4 HERRAMIENTAS DEL SISTEMA.....	18
1.5 ESTRUCTURA DE LA MEMORIA.....	19
1.6 ACRÓNIMOS Y DEFINICIONES	20
CAPITULO 2. GESTIÓN DEL PROYECTO.....	21
2.1 CICLO DE VIDA	21
2.2 ORGANIZACIÓN DEL PROYECTO	23
2.3 PLANIFICACIÓN DEL PROYECTO	35
2.4 ESTIMACIÓN DE COSTES	43
CAPITULO 3. ESTADO DE LA CUESTIÓN.....	46
3.1 INTRODUCCIÓN	46
3.2 ANÁLISIS DE LA IMPLEMENTACIÓN	47
3.3 TECNOLOGÍA UTILIZADA.....	50
CAPITULO 4. ANÁLISIS DEL SISTEMA	59
4.1 CATÁLOGO DE REQUISITOS	59
4.2 ESPECIFICACIÓN DE CASOS DE USO	81
4.3 DIAGRAMA DE CLASES	115
CAPITULO 5. DISEÑO DEL SISTEMA	117
5.1 CATÁLOGO DE REQUISITOS DE DISEÑO Y CONSTRUCCIÓN.....	117
5.2 DEFINICIÓN DE LA INTERFAZ DE USUARIO	120
5.3 DISEÑO FÍSICO DE DATOS.....	150
5.4 ARQUITECTURA TRES CAPAS.....	152
5.5 GESTIÓN DE TRANSACCIONES	155
5.6 PRINCIPIOS DE DISEÑO DE LA INTERFAZ DE USUARIO.....	158
CAPITULO 6. PRUEBAS.....	160
6.1 ESPECIFICACIÓN DEL PLAN DE PRUEBAS.....	160
6.2 CATÁLOGO DE PRUEBAS.....	161
CAPITULO 7. CONCLUSIONES Y LÍNEAS FUTURAS.....	165
7.1 CONCLUSIONES	165
7.2 FUTURAS LÍNEAS DE TRABAJO	167
CAPITULO 8. REFERENCIAS	168
CAPITULO 9. ANEXOS.....	172



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 0: Índice de contenidos

ANEXO 1. EJEMPLOS DE CÓDIGO DE LA APLICACIÓN..... 173
ANEXO 2. INSTALACIÓN Y CONFIGURACIÓN DEL ENTORNO 181



Tabla de Ilustraciones

Ilustración 1. Diagrama del Modelo de Ciclo de Vida en Cascada	21
Ilustración 2. WBS	23
Ilustración 3. RBS	28
Ilustración 4. PBS	34
Ilustración 5. Diagrama GANTT I	35
Ilustración 6. Diagrama GANTT II	36
Ilustración 7. Diagrama GANTT III	37
Ilustración 8. Diagrama GANTT IV	37
Ilustración 9. Diagrama GANTT V	38
Ilustración 10. Calendario Laboral	42
Ilustración 11. Marco de Trabajo de .NET	51
Ilustración 12. Módulos Funcionales	59
Ilustración 13. Entorno tecnológico del sistema	80
Ilustración 14. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar PIBOK-PB"	82
Ilustración 15. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para gestionar proyectos"	83
Ilustración 16. Diagrama de casos de uso: "Interacción Usuario Experto con el sistema para gestionar una situación"	86
Ilustración 17. Diagrama de casos de uso: "Desglose caso de uso genérico referido a modelos de procesos"	90
Ilustración 18. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar un modelo de proceso personalizado"	91
Ilustración 19. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar un modelo de proceso mediante el sistema experto"	96
Ilustración 20. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para gestionar patrones de producto"	102
Ilustración 21. Diagrama de casos de uso: "Interacción Jefe de Proyecto/Administrador con el sistema para gestionar un plan de proyecto"	105
Ilustración 22. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para validarse en la aplicación."	108
Ilustración 23. Diagrama de casos de uso: "Interacción Administrador con el sistema para gestionar la documentación."	109
Ilustración 24. Diagrama de casos de uso: "Interacción Administrador con el sistema para gestionar permisos."	112
Ilustración 25. Diagrama de clases	115
Ilustración 51. Interfaz de la página principal	120
Ilustración 52. Interfaz Crear nuevo proyecto	121



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

Ilustración 53. Interfaz Editar proyecto existente	122
Ilustración 54. Seleccionar Proyecto existente	122
Ilustración 55. Modificar parámetros principales.....	123
Ilustración 56. Eliminar Proyecto.....	124
Ilustración 57. Editar Proyecto I	125
Ilustración 58. Editar Proyecto II	125
Ilustración 59. Crear Proyecto – Organización I	126
Ilustración 60. Crear Proyecto – Organización II	126
Ilustración 61. Crear Proyecto – Parámetros I.....	127
Ilustración 62. Crear Proyecto – Parámetros II.....	127
Ilustración 63. Crear Proyecto – Parámetros III.....	128
Ilustración 64. Crear Proyecto – Parámetros IV.....	128
Ilustración 65. Crear Proyecto – Modelo de Proceso I.....	129
Ilustración 66. Crear Proyecto – Modelo de Proceso II.....	129
Ilustración 67. Crear Proyecto – Modelo de Proceso III	130
Ilustración 68. Crear Proyecto – Modelo de Proceso IV	131
Ilustración 69. Crear Proyecto – Modelo de Proceso V	131
Ilustración 70. Crear Proyecto – Modelo de Proceso Personalizado I.....	132
Ilustración 71. Crear Proyecto – Modelo de Proceso Personalizado II.....	133
Ilustración 72. Crear Proyecto – Modelo de Proceso Personalizado III.....	134
Ilustración 73. Crear Proyecto – Modelo de Proceso Personalizado IV	134
Ilustración 74. Crear Proyecto – Modelo de Proceso Personalizado V	135
Ilustración 75. Crear Proyecto – Modelo de Proceso Personalizado VI	135
Ilustración 76. Crear Proyecto – Modelo de Proceso Personalizado VII	136
Ilustración 77. Crear Proyecto – Modelo de Proceso Personalizado VIII	136
Ilustración 78. Crear Proyecto – Modelo de Proceso Personalizado IX	137
Ilustración 79. Crear Proyecto – Patrones de Producto I.....	138
Ilustración 80. Crear Proyecto – Patrones de Producto II.....	139
Ilustración 81. Crear Proyecto – Patrones de Producto III.....	140
Ilustración 82. Crear Proyecto – Patrones de Producto IV	141
Ilustración 83. Crear Proyecto – Patrones de Producto V	142
Ilustración 84. Crear Proyecto – Patrones de Producto VI	143
Ilustración 85. Crear Proyecto – Patrones de Producto VII	144
Ilustración 86. Crear Proyecto – Plan de Proyecto I	145
Ilustración 87. Crear Proyecto – Plan de Proyecto II	146
Ilustración 88. Crear Proyecto – Plan de Proyecto III	147
Ilustración 89. Zona de Administración	149



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

Ilustración 90. Modelo de Datos – PIBOK_PB_ProcessModel.....	150
Ilustración 91. Modelo de Datos – PIBOK_PB_CurrentProject	151
Ilustración 92. Paquete Instalación - Estructura de carpetas.....	181
Ilustración 93. Listado de características	183
Ilustración 94. Estructura Característica – DGSContentTypes.....	184
Ilustración 95. Estructura Característica – DGSPageLayoutContentTypeActivator	184
Ilustración 96. Estructura Característica - DGSProvisioning	185
Ilustración 97. Creación Aplicación Web I.....	190
Ilustración 98. Creación Aplicación Web II.....	191
Ilustración 99. Creación de la colección de sitios	192



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

CAPITULO 1. INTRODUCCIÓN

1.1 CONTEXTO

Este proyecto se enmarca en el contexto general de la Ingeniería del Software, concretamente en la aplicación de sus *buenas prácticas* durante el desarrollo de proyectos software. La solución propuesta en este proyecto podemos dividirla en dos componentes principales; por un lado tenemos un Sistema Basado en Conocimiento y por el otro un Marco de Trabajo Colaborativo para que todos los productos software desarrollados a lo largo del ciclo de vida de un proyecto software sean fácilmente localizables por todo el personal involucrado en el desarrollo del proyecto. Los dos componentes principales de la solución desarrollada en este proyecto de fin de carrera, pueden describirse brevemente de la siguiente manera:

1. Un Sistema Basado en Conocimiento es una aplicación que utiliza técnicas heurísticas de Inteligencia Artificial, por tanto ese sistema basado en conocimiento puede convertirse en Sistema Experto cuando representa la experiencia de una persona cualificada en un ámbito de conocimiento.

El presente proyecto se conforma de varios módulos funcionales, donde el primero de ellos pudiera ser un sistema experto, donde a partir de una serie de datos de entrada, en este caso parámetros de la organización y el proyecto software a ejecutar, el sistema muestra al usuario los modelos de proceso que más se ajustan a sus necesidades, para así poder obtener el modelo de proceso ideal y mejorar los procesos del proyecto que vaya a desempeñar. Por tanto, podemos concluir que este primer módulo funcional representa una mejor calidad y rapidez en las respuestas dando así lugar a una mejora de la productividad de la organización.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

2. Un Marco de Trabajo Colaborativo que permitirá la reutilización de activos de procesos y proyectos en el desarrollo software, a través de Patrones de Producto que proporcionarán al equipo de desarrollo del proyecto el conocimiento de las mejores prácticas de la Ingeniería del software, así como el “*know-how*” de la organización para ser utilizado en el desarrollo de sus proyectos. En cuanto a los Patrones de Producto se menciona un artículo muy interesante donde se definen cada uno de ellos: "Antonio Amescua, Javier García, Maria-Isabel Sánchez-Segura, Fuensanta Medina-Domínguez. A pattern-based solution to bridge the gap between theory and practice in using process models. PROSIM 2006, software process simulation and modelling conference. Shanghai, China. May 2006." Este marco de trabajo proporcionará las siguientes funcionalidades:

- a. Modelos y técnicas de procesos software de las mejores prácticas de la Ingeniería del Software.
- b. Selección y personalización de los modelos de procesos y metodologías existentes que más se adecuen a las características de la organización y del proyecto a desarrollar.
- c. Educción de los procesos de la organización y creación de modelos de procesos específicos de la organización.
- d. Educción y almacenamiento del conocimiento necesario para realizar las actividades de modelos de procesos software en artefactos denominados patrones de producto.
- e. Motor de búsqueda para facilitar la localización y reutilización de los activos de proceso en los proyectos.
- f. Entorno colaborativo para la ejecución del proyecto donde se le proporciona al equipo de desarrollo información sobre la planificación de dicho proyecto, además de facilitarle la documentación necesaria. En definitiva tendrán a su disposición un gestor documental muy potente con infinidad de posibilidades.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

1.2 DESCRIPCIÓN DEL PROYECTO

El proyecto que será argumentado en este documento forma parte de la arquitectura "PIBOK-PB", cuya definición viene referenciada del siguiente artículo " Antonio Amescua, Javier García, Maria-Isabel Sánchez-Segura, Fuensanta Medina-Domínguez. A pattern-based solution to bridge the gap between theory and practice in using process models. PROSIM 2006, software process simulation and modelling conference. Shanghai, China. May 2006".

El nombre en concreto de la aplicación es UC3M.PIBOK.Web donde cada una de las partes de dicho nombre tiene un significado propio, ya que es la estructura genérica que se aplica a todos los proyectos el cuál se detalla a continuación:

“[ORGANIZACIÓN]. [NOMBRE-PROYECTO]. [TIPO-PROYECTO]”

Los requisitos que se abordan en este desarrollo podemos dividirlos en dos grandes grupos, por un lado tenemos los requisitos funcionales de la propia aplicación que se argumentarán más adelante y por otro tenemos la instalación, configuración y diseño de un entorno colaborativo muy potente y el cual se implementa bajo el producto **Microsoft Office SharePoint Server 2007 (MOSS)** que también se comentará en este documento. Por tanto, de forma muy genérica se enumerarán los módulos funcionales existentes en este proyecto y que pertenecen a los grupos citados anteriormente:

- Módulo gestor de proyectos:** Capaz de gestionar todos los proyectos creados hasta la fecha de una forma muy rápida y aumentando la productividad del jefe proyecto (Alta/Baja/Modificación/Consulta), todo esto limitándolo al perfil de un usuario en concreto, es decir, este jefe de proyecto sólo podrá gestionar proyectos creados por su perfil, parte de estas limitaciones de seguridad es facilitada por el producto anteriormente citado MOSS.
- Módulo de asignación Organización/Modelo de proceso:** Primer paso de la aplicación, cuya funcionalidad principal es la de vincular un proyecto a una organización en concreto seleccionando posteriormente el “*modus operandi*” del jefe de proyecto para elaborar su modelo de procesos.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

- **Módulo gestor de modelos de procesos:** Como se comentó anteriormente, existen dos formas de gestionar la creación de un modelo de proceso por parte del jefe de proyecto, se puede facilitar un sistema experto para ayudarlo y seleccionar el modelo en su lugar mediante unos parámetros de entrada o también tiene la alternativa de elaborar su propio modelo tomando como punto de partida uno existente, además si lo prefiere puede formar de inicio un modelo de proceso totalmente personalizado.
- **Módulo gestor de patrones de producto:** Este módulo es el encargado de exponer al jefe de proyecto todas las tareas/actividades que tienen asociadas una serie de patrones de producto, facilitando toda la información necesaria de dichos patrones con el fin de que la organización satisfaga todas sus necesidades elaborando un proyecto lo más ajustado posible a su infraestructura y medios obtenidos para implementarlo.
- **Módulo gestor de seguimiento de proyectos:** Mediante el seguimiento de proyectos dentro de un marco colaborativo, se obtiene un mayor rendimiento y productividad por parte del equipo de desarrollo, además de mostrar la planificación de cada una de las tareas a desarrollar se puede modificar el patrón de producto que se seleccionó previamente, esta es una forma de poder depurar más las necesidades del proyecto, ya que muchas veces se deben hacer reconsideraciones con la planificación por errores de estimación u otros motivos.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

1.3 OBJETIVOS

El objetivo principal de este proyecto fin de carrera es adecuar y remodelar a las nuevas tecnologías la aplicación software que permite al Ingeniero del Software seleccionar un proyecto lo más completo y transparente posible para la organización; también se pretende fomentar las buenas prácticas de la Ingeniería del Software mediante parámetros, actividades y patrones de producto que deben emplearse para poder realizar posteriormente un seguimiento robusto del proyecto en todas sus fases.

Cuando se habla de nuevas tecnologías podemos hablar de nuevos entornos o de una nueva visión del desarrollo software, donde un entorno colaborativo para proyectos de estas características es imprescindible, ya que está comprobado que se obtienen mejores resultados cuando compartimos la visión del proyecto con todo el equipo de desarrollo, para así tener claro los objetivos en cada hito del proyecto, conservando toda la base del conocimiento expuesto por cada uno de los integrantes del grupo durante la duración del proyecto, con una única finalidad: *Evitar errores pasados y solucionar problemas fácilmente.*

Para finalizar, se dispone de un último componente que refuerza todo lo dicho anteriormente, dicho componente es un gestor documental donde se puede almacenar toda la documentación necesaria para implementar todos los procesos que el proyecto conlleva, aplicando los patrones de producto en su debida forma. Por ejemplo, actualmente reside en dicho gestor documental toda la documentación referenciada por cada uno de los modelos de proceso, completando así toda la información que el jefe de proyecto pueda necesitar para realizar su trabajo de una forma eficiente, además de contribuir a estar seguro de sí mismo a la hora de tomar decisiones.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

Pero antes de llegar a implementar la aplicación Web citada, será preciso elaborar un documento de análisis que se compondrá de los siguientes apartados:

- Una especificación de requisitos de usuario.
- Un diagrama de casos de usos con la descripción textual de los mismos, siguiendo el estándar UML.
- Un modelo de clases.
- Un diagrama E/R con su correspondiente modelo relacional.
- El modelo físico de la BBDD

Una vez realizados todos estos pasos, se podrá pasar a diseñar cada uno de los módulos que van a formar parte de la aplicación y que posteriormente serán implementados con la ayuda de las herramientas software necesarias, además de contar con una plataforma MOSS totalmente personalizada.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

1.4 HERRAMIENTAS DEL SISTEMA

Las herramientas utilizadas para desarrollar el presente proyecto fin de carrera han sido las siguientes:

- **Sistema Operativo:** Microsoft Windows Server 2003
- **Generador de modelos conceptuales:** Microsoft Office Visio 2007.
- **Gestor de Bases de Datos:** Microsoft SQL Server 2005
- **Herramienta de programación:** Microsoft Visual Studio 2008 Professional
- **Generador de documentos/tablas/figuras:** Microsoft Office System 2007
- **Entorno colaborativo:** Microsoft Office SharePoint Server 2007
- **Herramientas de diseño:** Microsoft SharePoint Designer - Adobe Photoshop

Como se puede comprobar la tecnología utilizada es sumamente compleja, por lo que el proyecto ha sido implementado lo más cuidadosamente posible para que fuese totalmente independiente en cuanto posteriores desarrollos, ya que mantener una aplicación de estas dimensiones es bastante complicado. Por ese mismo motivo, se ha reservado un apartado en el Anexo para la instalación y configuración del entorno, ya que no es trivial y se necesitan ciertos conocimientos.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

1.5 ESTRUCTURA DE LA MEMORIA

Este documento se estructura en ocho capítulos detallados a continuación:

- **Capítulo 1. Introducción:** En este capítulo se realiza una breve introducción del documento indicando cuál es el contexto y objetivos que motivan la creación del proyecto, el marco de trabajo del mismo, las herramientas utilizadas para su consecución, así como la realización del mismo.
- **Capítulo 2. Gestión del proyecto:** En este capítulo se detallan todos los aspectos relacionados con la gestión del proyecto, entre los que se destacan el ciclo de vida del proyecto, los aspectos organizativos en cuanto a tareas, entregables de estas tareas y recursos a utilizar, aspectos de planificación y aspectos de estimación de costes.
- **Capítulo 3. Estado de la cuestión:** Este capítulo se encarga de reflejar el punto de partida del cual comienza el presente proyecto, conocimientos necesarios para poder replicar la aplicación en ASP.NET 2.0, estudiar en profundidad el sistema basado en conocimiento junto con la base de conocimiento conformada por reglas; por último, se explica al lector cuales fueron las decisiones en cuanto a tecnología utilizada y qué beneficios obtenemos con ella.
- **Capítulo 4. Análisis del sistema:** En este capítulo se indican los requisitos de usuario exigidos por parte del cliente, los modelos físicos y conceptuales de la base de datos (BBDD), así como la interfaz de comunicación usuario-BBDD.
- **Capítulo 5. Diseño del sistema:** En este capítulo se enumeran todas las decisiones tomadas a la hora de elaborar la interfaz de usuario, así como un manual de usuario de la aplicación.
- **Capítulo 6. Plan de pruebas:** En este capítulo se enumeran todas las pruebas que se ha realizado al producto software final, para asegurarnos que cumple con las especificaciones de requisitos de usuario.
- **Capítulo 7. Conclusiones y líneas futuras:** En este capítulo se indican las conclusiones obtenidas tras la realización del proyecto así como líneas futuras de trabajo con el fin de ampliar la funcionalidad.
- **Capítulo 8. Referencias:** En este capítulo se enumeran las diferentes fuentes de información utilizadas para la realización del proyecto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

1.6 ACRÓNIMOS Y DEFINICIONES

1.6.1 ACRÓNIMOS

- **BBDD:** Base de Datos
- **CSS:** Cascading Style Sheets.
- **E/R:** Entidad-Relación.
- **HTML:** Hypertext Markup Language.
- **RBS:** Resource Breakdown Structure.
- **MOSS:** Microsoft Office SharePoint Server 2007
- **PBS:** Products Breakdown Structure

1.6.2 DEFINICIONES

- **ASP.NET 2.0:** Es un framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML.
- **GANTT:** El diagrama de Gantt consiste en una representación gráfica sobre dos ejes; en el vertical se disponen las tareas del proyecto y en el horizontal se representa el tiempo.
- **PBS:** Siglas de Products Breakdown Structure que es una representación gráfica de los diferentes productos a obtener a lo largo del desarrollo de un proyecto software.
- **RBS:** Siglas de Resource Breakdown Structure que es una representación gráfica de los diferentes recursos necesarios para la realización de un proyecto software.
- **WBS:** Siglas de Work Breakdown Structure que es una representación gráfica de las diferentes actividades que forman un proyecto software.
- **MOSS:** Siglas del acrónimo definido anteriormente, el cual es un producto de Microsoft dedicado principalmente a la gestión de contenidos y gestión documental.



CAPITULO 2. GESTIÓN DEL PROYECTO

En este capítulo se presentan las principales áreas en la gestión de este proyecto software, como son la elección del ciclo de vida, la organización, la planificación y la estimación de costes referidos al mismo.

2.1 CICLO DE VIDA

En primer lugar, se definirá el concepto de “Ciclo de vida” con el fin de explicar posteriormente todas las actividades pertenecientes a cada una de las fases que establece el modelo elegido para finalmente justificar la elección del ciclo de vida.

Se entiende por ciclo de vida software como el conjunto de fases, procesos y actividades requeridas para ofertar, desarrollar, probar, integrar, explotar y mantener un producto software indicando como funciones principales:

1. Determinar el orden de las fases y procesos
2. Establecer los criterios de transición para pasar de una fase a otra.

El ciclo de vida elegido ha sido el modelo “*Cascada*” que define seis pasos que se muestran en la siguiente figura:

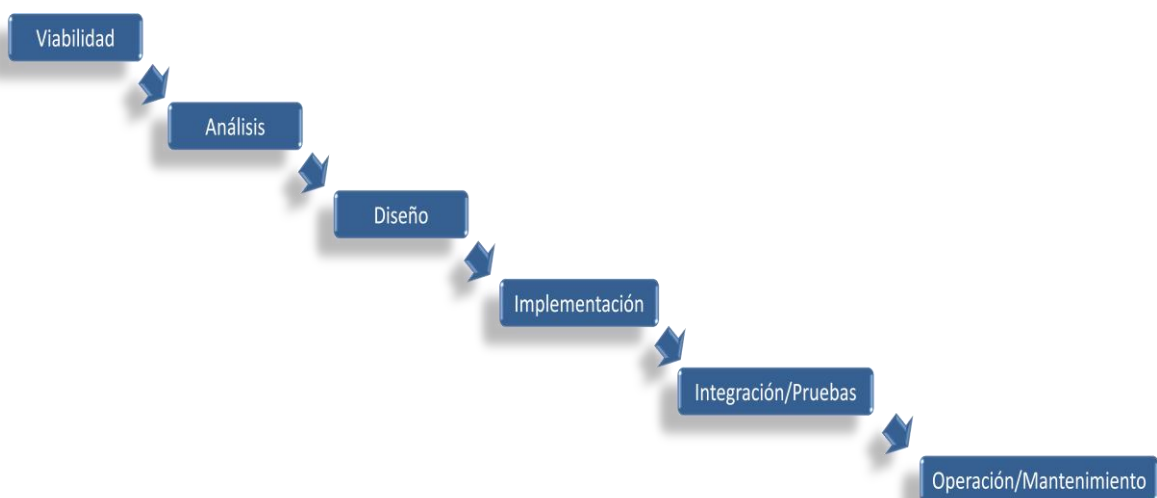


Ilustración 1. Diagrama del Modelo de Ciclo de Vida en Cascada



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

A continuación se explica brevemente el significado de cada fase:

- **Viabilidad:** En esta primera fase se determinan los requisitos de usuario y se realiza el Plan de Proyecto indicando recursos, calendario y responsabilidades.
- **Análisis:** En esta segunda fase se determinan los requisitos software del sistema y el análisis del sistema especificando los casos de uso y la arquitectura a un alto nivel que se propone como solución.
- **Diseño:** En esta tercera fase se diseñan los interfaces del sistema, la base de datos, la arquitectura a más bajo nivel que el análisis y se definen detalladamente los diferentes procesos del sistema.
- **Desarrollo:** En esta etapa se codifican los interfaces y la base de datos del sistema y se documenta el manual de usuario.
- **Integración y Pruebas:** En esta etapa se realizan el conjunto de pruebas unitarias del módulo para posteriormente realizar la integración con otros módulos del sistema para finalmente realizar las pruebas de integración de todo el sistema.
- **Operación y Mantenimiento:** En esta etapa se realiza la instalación y configuración de la aplicación en las infraestructuras del cliente para así poder realizar las pruebas globales de funcionamiento del sistema y posteriormente arreglar los posibles fallos que puedan aparecer, realizando un servicio de mantenimiento.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

2.2 ORGANIZACIÓN DEL PROYECTO

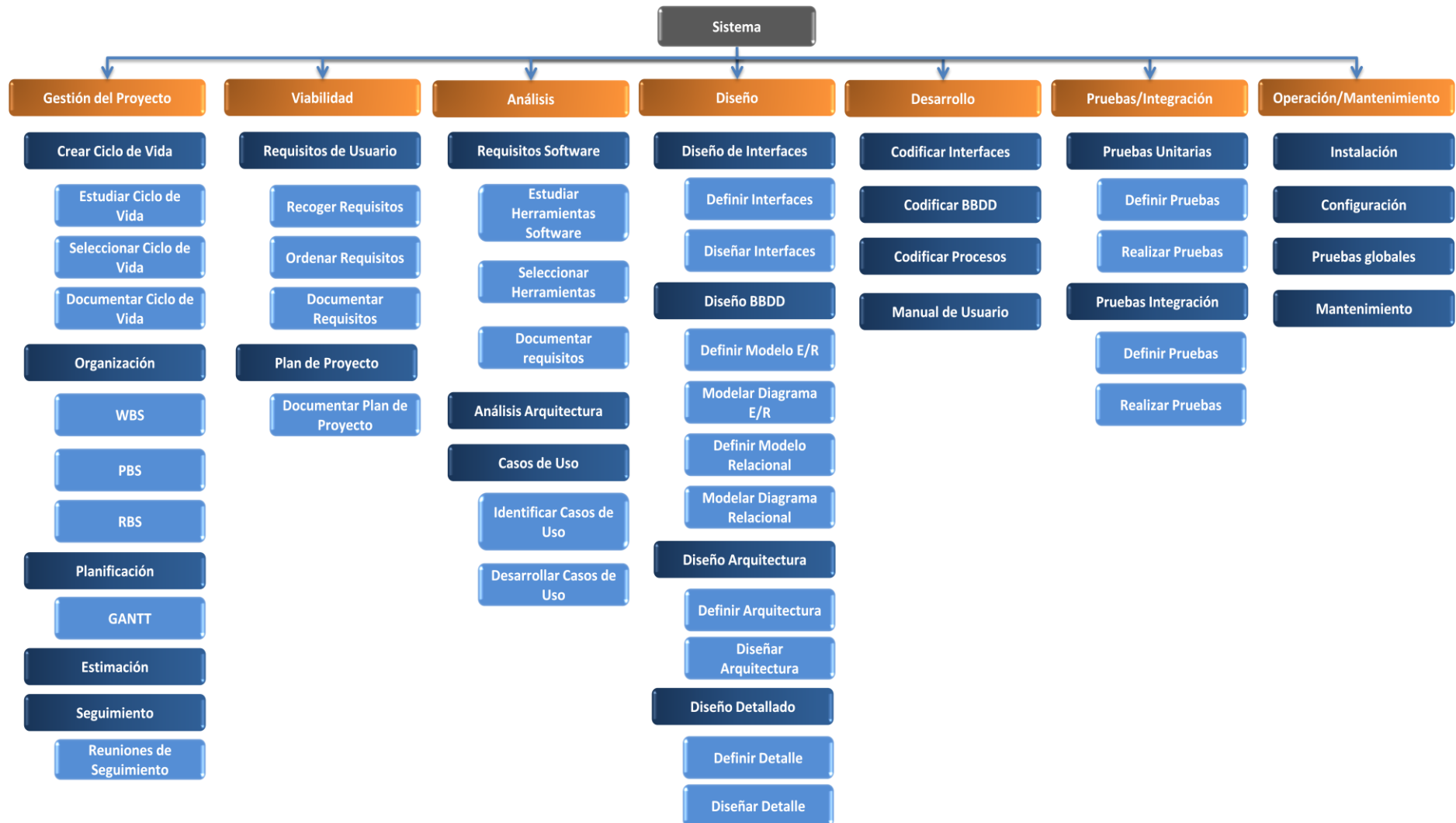
En este apartado se indica cual es la estructura del proyecto descomponiéndolo en tres subgrupos de modo que queden fijadas las tareas que se deben llevar a cabo, los productos que resultan de realizar cada una de las tareas y los recursos necesarios para la realización del proyecto.

2.2.1 Work Breakdown Structure

En este apartado se descompone el proyecto en las tareas que hay que realizar a lo largo del desarrollo. Se representa de forma gráfica tal y como muestra la siguiente ilustración:

Ilustración 2. WBS

(Ver en página siguiente)





Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

Tal y como muestra la figura anterior, este proyecto queda dividido en las diferentes fases que el ciclo de vida indica, obteniéndose las tareas a realizar, las cuales se explican a continuación (estándar IEEE 1074, 1997):

• **Fase Gestión del Proyecto:** Fase que se encarga, como su propio nombre indica, de la gestión del proyecto. Este grupo de actividades inician y controlan el proyecto software durante todo su ciclo de vida.

- **Actividad Crear Ciclo de Vida:** Actividad en la que se estudian las características del proyecto que se va a realizar y los diferentes ciclos de vida existentes con el fin de seleccionar aquel ciclo de vida que mejor se adecue a dichas características. Esta actividad es muy importante de cara a seguir un método de trabajo en el proyecto
- **Actividad Organización:** Organización del proyecto en actividades y tareas obteniendo el presente WBS; en cuanto a los productos resultantes de la realización de las diferentes actividades y tareas obteniendo el PBS; y en cuanto a recursos humanos y materiales se refiere obteniéndose el RBS.
- **Actividad Planificación:** Planificación del tiempo de duración de cada una de las diferentes tareas en que se descompone el proyecto obtenidas en el WBS, reflejándolo de forma gráfica en un diagrama GANTT.
- **Actividad Estimación:** Estimación de costes del proyecto la cual refleja el esfuerzo de las personas por mes y el tiempo necesario a emplear para llevar a cabo el proyecto (medido en meses de calendario).
- **Actividad Seguimiento:** Actividad de seguimiento del proyecto, en el cuál se refleja la labor de control realizada sobre el proyecto en lo que se refiere al cumplimiento de la planificación y en la realización de las diferentes tareas que componen el proyecto.

• **Fase Determinar Viabilidad:**

- **Actividad Requisitos de Usuario:** Recopilación de los requisitos de usuario, los cuales tendrán que cumplirse obligatoriamente a la hora de la entrega del proyecto, ordenándolos y agrupándolos de la manera más clara y eficiente posible, así como documentándolos.
- **Actividad Plan de Proyecto:** Establecimiento de un Plan de Proyecto inicial a partir de los requisitos de usuario indicando hitos a alcanzar en la realización del proyecto, el cual es refinado a través del seguimiento del proyecto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

• **Fase Análisis:**

- **Actividad Requisitos Software:** Realización de un estudio de las herramientas disponibles en el mercado que sean necesarias y/o convenientes para la realización del proyecto. Una vez estudiadas se seleccionarán aquellas que mejor se adecuen a las características del proyecto y finalmente esta actividad será documentada.
- **Actividad Análisis Arquitectura:** Definición a alto nivel de la arquitectura elegida a partir de la cual se dará comienzo el desarrollo.
- **Actividad Casos de Uso:** Identificación de los diferentes casos de uso a partir de los requisitos de usuario para a continuación desarrollarlos y tratar de abarcar toda la casuística posible de precondiciones, actores, y poscondiciones para cada uno de ellos.

• **Fase Diseño:**

- **Actividad Diseño de Interfaces:** Definición de los diferentes interfaces de interacción entre el sistema y el usuario para finalmente diseñarlas.
- **Actividad Diseño BBDD:** Definición y modelado del modelo entidad relación y del modelo relacional referente a la base de datos del sistema.
- **Actividad Diseño Arquitectura:** Definición y diseño de cómo se conectarán los diferentes componentes de la arquitectura en cuanto a qué protocolos de comunicaciones y que primitivas se refiere.
- **Actividad Diseño Detallado:** Definición y desarrollo detallado de los diferentes procesos del sistema obtenido de los casos de uso.

• **Fase Desarrollo:**

- **Actividad Construcción de Interfaces:** Codificación de los diferentes interfaces del sistema.
- **Actividad Construcción de la BBDD:** Codificación de la base de datos del sistema.
- **Actividad Construcción de Procesos:** Codificación de los diferentes procesos del sistema.
- **Actividad Manual de Usuario:** Redacción del Manual de Usuario.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

• **Fase Pruebas/Integración:**

- **Actividad Pruebas Unitarias:** Definición y realización de las pruebas unitarias para cada proceso, actividad y tarea con el fin de verificar el cumplimiento de los diferentes requisitos de usuario.
- **Actividad Pruebas Integración:** Definición y realización de las pruebas de integración de todo el sistema.

• **Fase Operación/Mantenimiento:**

- **Actividad Instalación:** Instalación de la aplicación en el entorno del cliente, paso muy importante y en principio algo complejo, ya que se necesita la utilización de algunos comandos especiales.
- **Configuración:** Configuración de la aplicación para que funcione correctamente.
- **Pruebas globales:** Realizar las mismas pruebas de integración que se formularon anteriormente, pero ahora en el entorno preparado en el cliente.
- **Mantenimiento:** Mantenimiento de la aplicación por si surgen incidencias detectadas por el cliente.



2.2.2 Resource Breakdown Structure

En este apartado se muestra el RBS que es una representación gráfica en forma jerárquica de los recursos, tanto humanos como materiales, que son necesarios para la realización de un proyecto software.

Estos recursos están divididos entre los humanos y los materiales. Los materiales a su vez se dividen en recursos hardware y software.

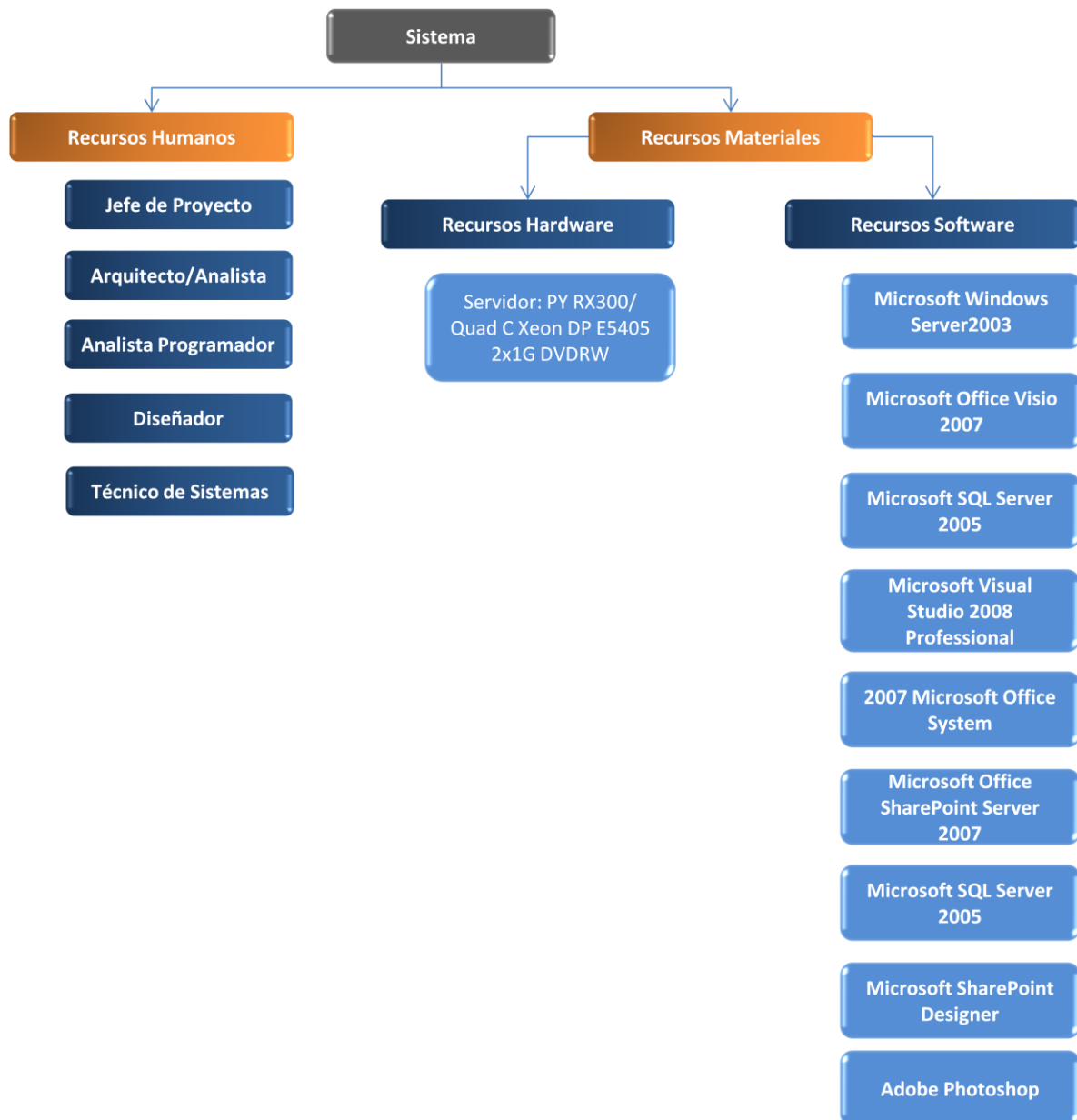


Ilustración 3. RBS



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

Tal y como muestra la figura anterior, los recursos humanos indican cuales son las personas, el rol y la responsabilidad que desempeñan en la realización del proyecto, encargándose de las siguientes labores:

- **Jefe de Proyecto:** Se encarga de la gestión del proyecto, es decir, organizarlo, planificarlo y estimar su coste.
- **Arquitecto/Analista:** Se encarga de obtener y redactar la especificación de requisitos de usuario, establecer los casos de uso del sistema y modelar los procesos y tareas a codificar. Además será el encargado de decidir cuál será la tecnología y tomará todas las decisiones en cuanto a Arquitectura o Análisis se refieran.
- **Analista-Programador:** Se encarga de la codificación del sistema a partir del modelo de procesos y tareas, así como la realización de las pruebas unitarias y de integración.
- **Diseñador:** Se encarga de modelar las interfaces de usuario y de implementar las hojas de estilo CSS, además de maquetar las páginas en HTML.
- **Técnico de Sistemas:** Se encarga de la instalación de los diferentes elementos software y hardware que requiera la aplicación para su correcto funcionamiento.

Al tratarse de un Proyecto de Fin de Carrera los jefes de proyecto de la aplicación resultan ser Maribel Sánchez y José Arturo Mora (Directora y Tutor respectivamente), en cuanto a las tareas de diseño, desarrollo, pruebas... han sido implementadas por un servidor (David Gómez Sanabria).



2.2.3 Products Breakdown Structure

En este apartado se muestra el PBS que es una representación gráfica en forma jerárquica de los productos resultados en cada una de las etapas definidas en el WBS.

Ver ilustración 2. Pág.18.

Tal y como muestra en la ilustración, se obtienen diferentes productos de las diferentes fases y tareas que se indican en el WBS incluyendo la fase propia de la gestión del proyecto, los cuales se desarrollan a continuación:

- **Fase de Gestión del Proyecto**
 - **Actividad Crear Ciclo de Vida**
 - **Ciclo de Vida:** Documento que indica las diferentes etapas del modelo de ciclo de vida elegido para la realización del proyecto.
 - ⇒ **Actividad Organización**
 - **WBS:** Documento que muestra las diferentes fases y actividades del proyecto.
 - **RBS:** Documento que muestra los diferentes recursos necesarios para la realización del proyecto.
 - **PBS:** Documento que muestra los productos que se obtienen de las diferentes tareas del proyecto.
 - **Actividad Planificación**
 - **Gantt:** Documento de planificación de cada una de las diferentes tareas en que se descompone el proyecto, reflejándolo de forma gráfica en un diagrama GANTT.
 - **Actividad Estimación**
 - **Estimación:** Documento de estimación de costes del proyecto el cual refleja el esfuerzo de las personas por mes y el tiempo necesario a emplear para llevar a cabo el proyecto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

- **Actividad Seguimiento**
 - **Seguimiento:** Documentación de las actas de las diferentes tutorías celebradas con los tutores del proyecto en las que se detallan tanto las personas que estaban presentes en la tutoría así como los temas tratados en la misma de forma que refleja la labor de control realizada sobre el proyecto.
- **Fase Determinar Viabilidad**
 - **Actividad Requisitos de Usuario**
 - **Requisitos de Usuario:** Documento que establece los requisitos de usuario, los cuales tendrán un carácter obligatorio en cuanto al cumplimiento a la hora de la entrega del proyecto.
 - **Actividad Plan de Proyecto**
 - **Plan de Proyecto:** Documento que establece a partir de los requisitos de usuario indicando hitos a alcanzar en la realización del proyecto, el cual es refinado a través del seguimiento del proyecto.
- **Fase Análisis**
 - **Actividad Requisitos Software**
 - **Requisitos Software:** Documento de requisitos software que indica las herramientas, incluida su versión, que van a ser necesarias para la realización del proyecto.
 - **Actividad Análisis Arquitectura**
 - **Análisis Arquitectura:** Documento que indica la arquitectura de alto nivel elegida a partir de la cual abordar el proyecto.
 - **Actividad Casos de Uso**
 - **Casos de Uso:** Documento que define los casos de uso a los que será sometido el sistema.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

- **Fase Diseño**
 - **Actividad Diseño de Interfaces**
 - **Diseño de Interfaces:** Documento de detalle del Diseño de Interfaces.
 - **Actividad Diseño BBDD**
 - **Diagrama E/R:** Documento de detalle del diagrama entidad-relación de la base de datos.
 - **Diagrama Relacional:** Documento de detalle del diagrama relacional de la base de datos.
 - **Actividad Diseño Arquitectura**
 - **Diseño Arquitectura:** Documento de detalle de conexión entre las diferentes capas de la arquitectura elegida.
 - **Actividad Diseño Detallado**
 - **Diseño Detallado:** Documento de detalle de los procesos que forman el sistema.
- **Fase Desarrollo:**
 - **Actividad Codificar Interfaces**
 - **Codificar Interfaces:** Documento de construcción del sistema en el que se indica el código de las diferentes interfaces.
 - **Actividad Codificar BD**
 - **Codificar BD:** Documento de construcción del sistema en el que se indica el código de la base de datos.
 - **Actividad Codificar Procesos**
 - **Codificar Procesos:** Documento de construcción del sistema en el que se indica el código de los diferentes procesos.
 - **Actividad Manual de Usuario**
 - **Manual de Usuario:** Documento de construcción del sistema en el que se detalla el manual de uso del sistema para los diferentes usuarios que interactuarán con el sistema.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

- **Fase Pruebas/Integración:**

- **Actividad Pruebas Unitarias**

- **Pruebas Unitarias:** Documento en el que se detallan las pruebas unitarias a realizar para cada proceso, actividad y tarea con el fin de verificar el cumplimiento de los diferentes requisitos de usuario.

- **Actividad Pruebas Integración**

- **Pruebas Integración:** Documento en el que se detallan las pruebas de integración a realizar.

- **Fase Operación/Mantenimiento:**

- **Actividad Instalación**

- **Instalación:** Documento con el proceso de instalación que debe realizarse en el entorno del cliente.
- **Configuración:** Documento con el proceso de configuración que debe realizarse en el entorno previamente instalado.
- **Pruebas globales:** Documento con el conjunto de todas las pruebas que la aplicación web deberá cumplimentar con un resultado satisfactorio.
- **Mantenimiento:** Documento en el cuál se acepta la entrega de la aplicación en perfecto estado, además se detallarán los términos en cuanto a la garantía y mantenimiento del mismo por un período de 1 mes.

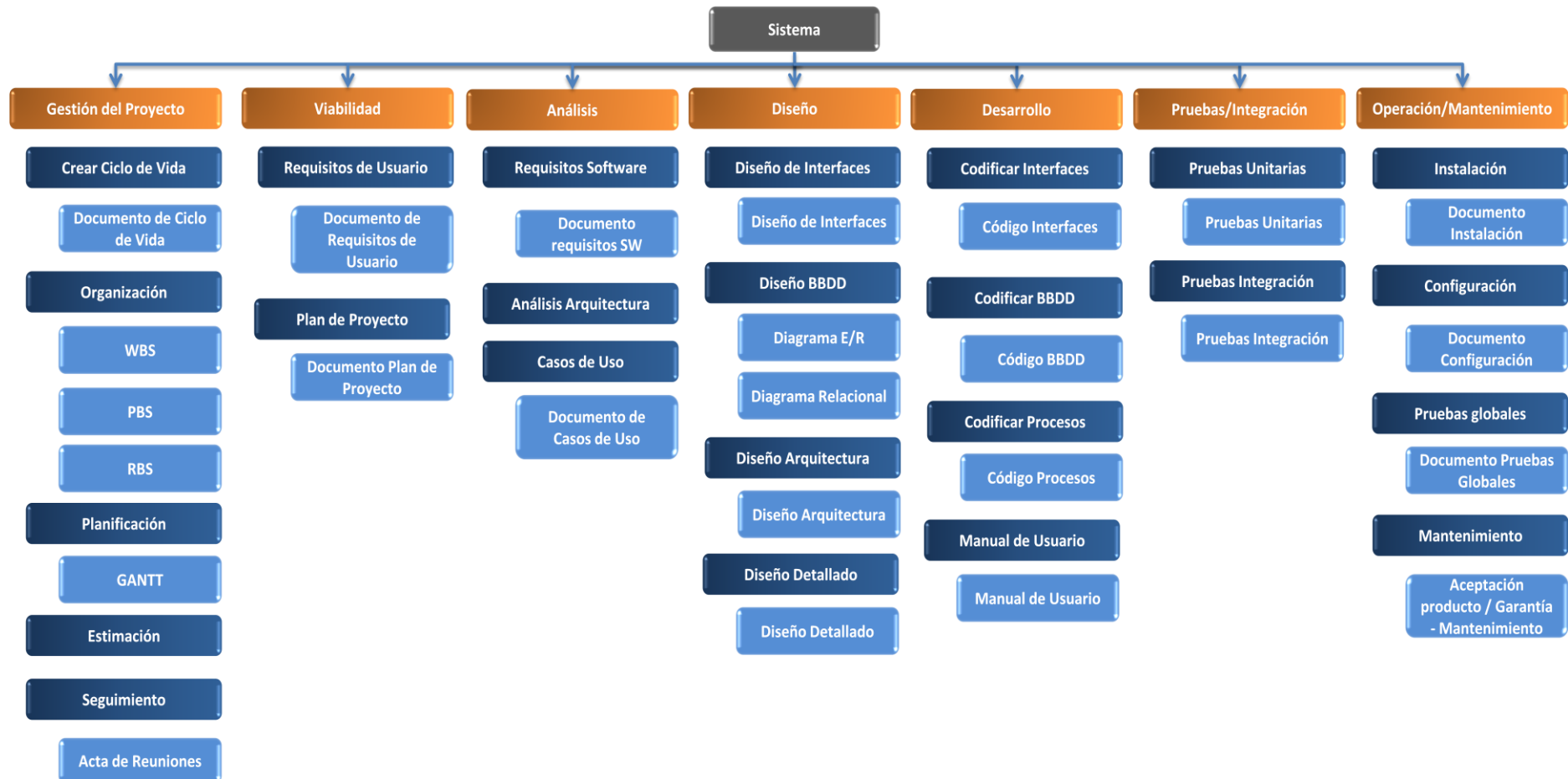


Ilustración 4. PBS



2.3 PLANIFICACIÓN DEL PROYECTO

Todo proyecto, ya sea de software o no, necesita de una estimación del tiempo que va a durar la realización de cada una de las tareas, con el fin de que el proyecto pueda ser acotado en tiempo y costes de recursos, tanto humanos como materiales.

En este epígrafe se muestra la planificación seguida desde el comienzo del proyecto hasta su finalización, detallando el tiempo estimado de realización de las diferentes actividades y tareas así como los recursos asignados a cada una de las citadas tareas reflejándolo en los siguientes diagramas Gantt.

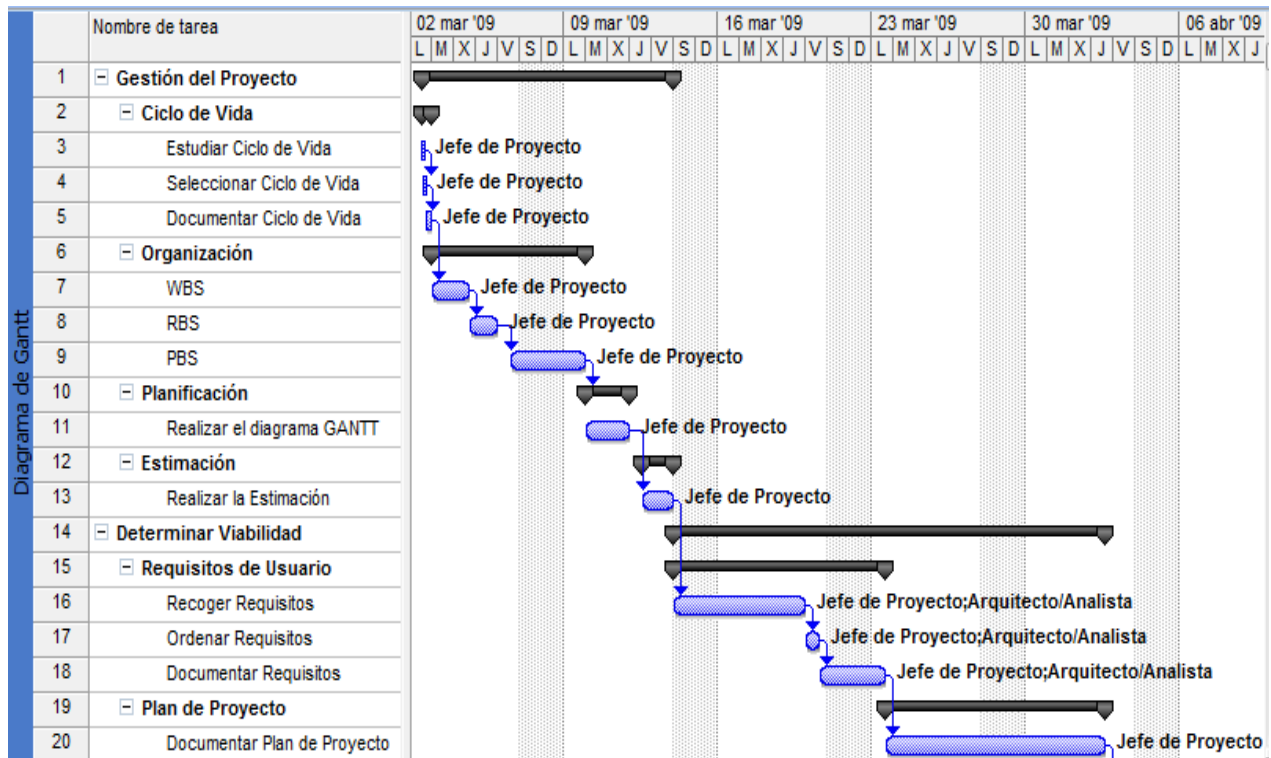


Ilustración 5. Diagrama GANTT I

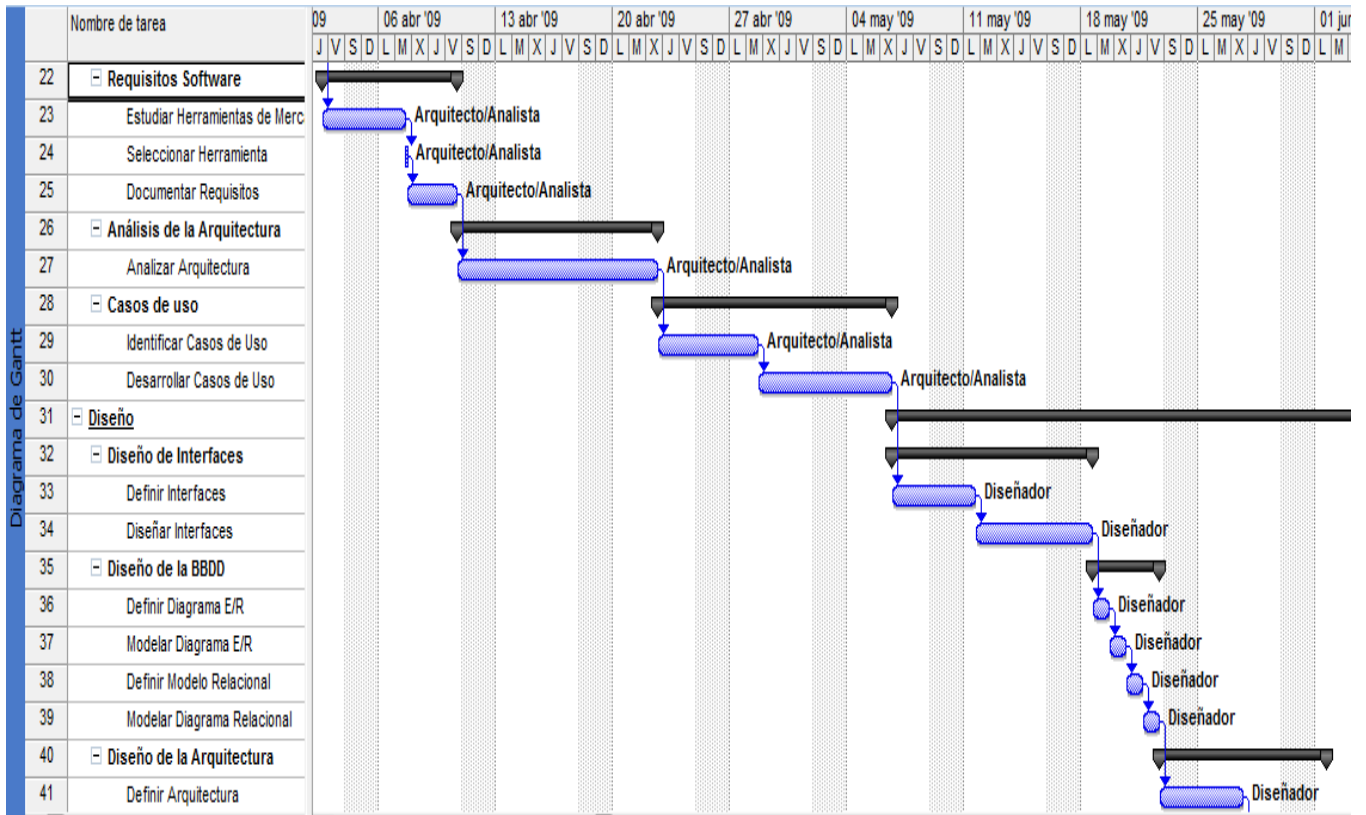


Ilustración 6. Diagrama GANTT II

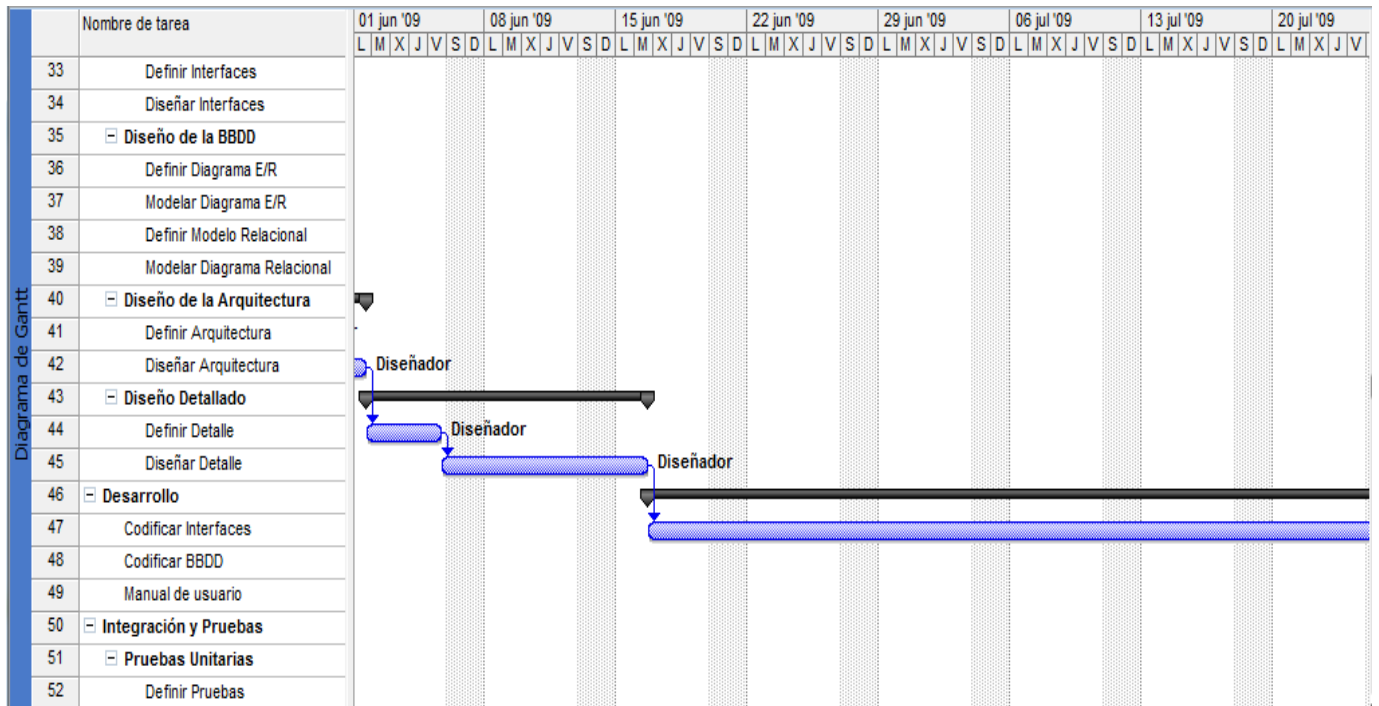


Ilustración 7. Diagrama GANTT III

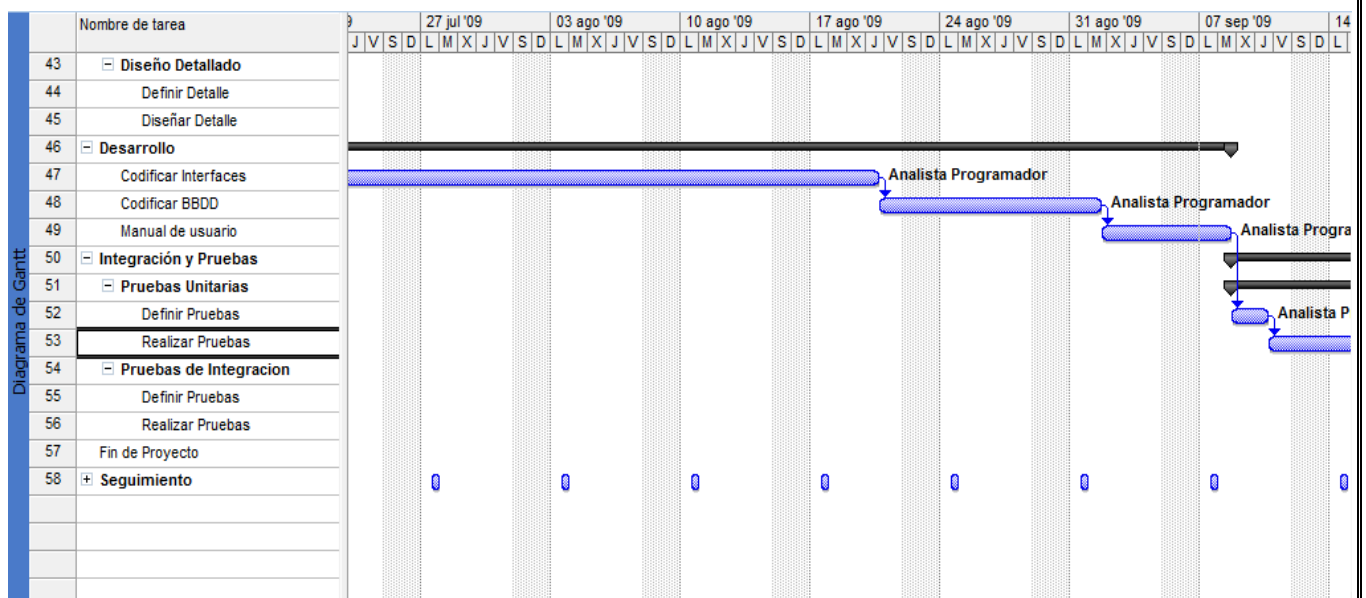


Ilustración 8. Diagrama GANTT IV

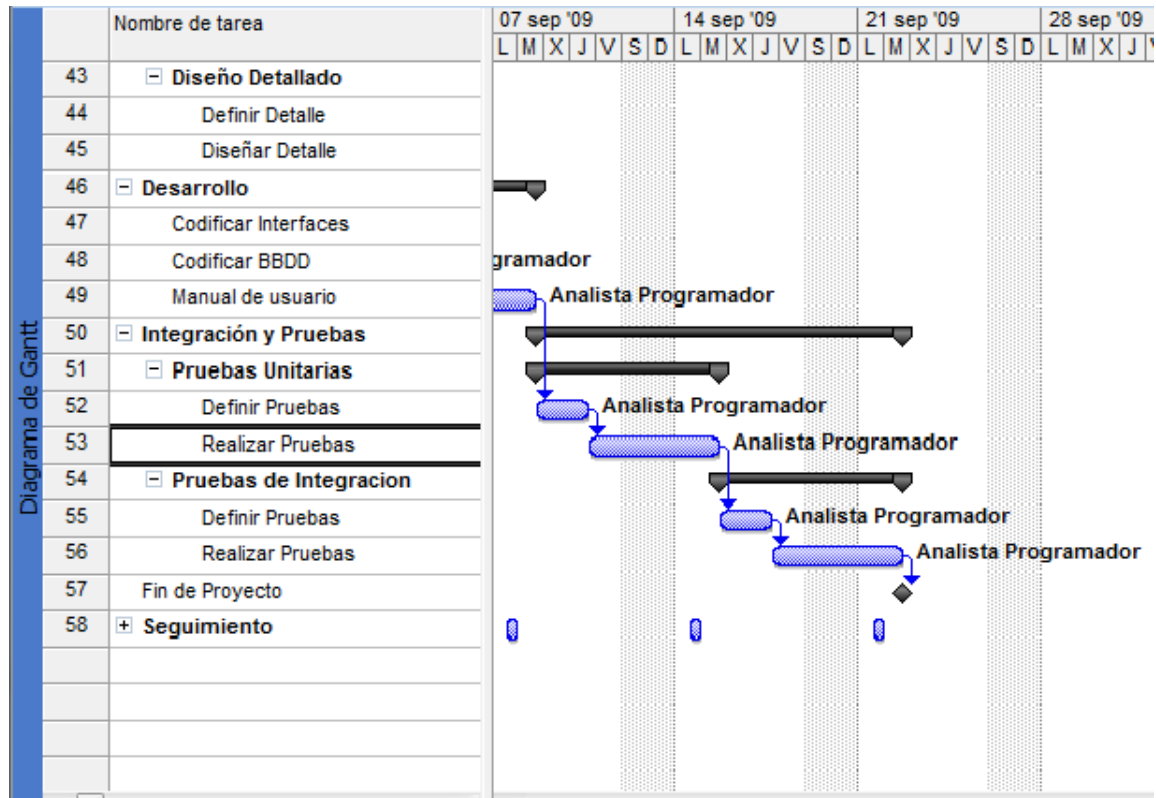


Ilustración 9. Diagrama GANTT V



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

A continuación se muestra una tabla en la cuál se detalla, tanto la duración como la fecha de comienzo y de fin de cada una de las tareas ilustradas en los diagramas anteriores.

Id	Nombre	Duración	Comienzo	Fin
1	Gestión del Proyecto	9,6 días	02/03/2009	13/03/2009
2	Ciclo de Vida	0,8 días	02/03/2009	02/03/2009
3	Estudiar Ciclo de Vida	1,6 horas	02/03/2009	02/03/2009
4	Seleccionar Ciclo de Vida	1,6 horas	02/03/2009	02/03/2009
5	Documentar Ciclo de Vida	2,4 horas	02/03/2009	02/03/2009
6	Organización	4,8 días	02/03/2009	09/03/2009
7	Crear WBS	1,6 días	02/03/2009	04/03/2009
8	Crear RBS	1,6 días	05/03/2009	06/03/2009
9	Crear PBS	1,6 días	06/03/2009	09/03/2009
10	Planificación	2,4 días	09/03/2009	11/03/2009
11	Realizar Gantt	2,4 días	09/03/2009	11/03/2009
12	Estimación	1,6 días	12/03/2009	13/03/2009
13	Realizar Estimación	1,6 días	12/03/2009	13/03/2009
16	Determinar Viabilidad	13,5 días	13/03/2009	02/04/2009
17	Requisitos de Usuario	5,5 días	13/03/2009	23/03/2009
18	Recoger Requisitos	4 días	13/03/2009	19/03/2009
19	Ordenar Requisitos	0,5 días	19/03/2009	20/03/2009
20	Documentar Requisitos	1 día	20/03/2009	23/03/2009



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

21	Plan de Proyecto	8 días	23/03/2009	02/04/2009
22	Documentar Plan de Proyecto	8 días	23/04/2009	02/04/2009
23	Análisis	24,29 días	02/04/2009	06/04/2009
24	Requisitos Software	6,29 días	02/04/2009	10/04/2009
25	Estudiar Herramientas de Mercado	3 días	02/04/2009	07/04/2009
26	Seleccionar Herramientas	2 horas	07/04/2009	07/04/2009
27	Documentar Requisitos	3 días	07/04/2009	10/04/2009
28	Análisis Arquitectura	8 días	10/04/2009	22/04/2009
29	Analizar Arquitectura	8 días	10/04/2009	22/04/2009
30	Casos de Uso	10 días	22/04/2009	06/05/2009
31	Identificar Casos de Uso	4 días	22/04/2009	28/04/2009
32	Desarrollar Casos de Uso	6 días	28/04/2009	06/05/2009
33	Diseño	29 días	06/05/2009	16/06/2009
34	Diseño Interfaces	8 días	06/05/2009	18/05/2009
35	Definir Interfaces	3 días	06/05/2009	11/05/2009
36	Diseñar Interfaces	5 días	11/05/2009	18/05/2009
37	Diseño BBDD	4 días	18/05/2009	22/05/2009
38	Definir Modelo E/R	1 día	18/05/2009	19/05/2009
39	Modelar Diagrama E/R	1 día	19/05/2009	20/05/2009
40	Definir Modelo Relacional	1 día	20/05/2009	21/05/2009
41	Modelar Diagrama Relacional	1 día	21/05/2009	22/05/2009



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

42	Diseño Arquitectura	6 días	22/05/2009	01/06/2009
43	Definir Arquitectura	3 días	22/05/2009	27/05/2009
44	Diseñar Arquitectura	3 días	27/05/2009	01/06/2009
45	Diseño Detallado	11 días	01/06/2009	16/06/2009
46	Definir Detalle	4 días	01/06/2009	05/06/2009
47	Diseñar Detalle	7 días	05/06/2009	16/06/2009
48	Desarrollo	60 días	16/06/2009	08/09/2009
49	Codificar Interfaces	47 días	16/06/2009	20/08/2009
51	Codificar BD	8 días	20/08/2009	01/09/2009
55	Manual de Usuario	5 días	01/09/2009	08/09/2009
57	Integración y Pruebas	10 días	08/09/2009	22/09/2009
58	Pruebas Unitarias	5 días	08/09/2009	15/09/2009
59	Definir Pruebas	2 días	08/09/2009	10/09/2009
60	Realizar Pruebas	3 días	10/09/2009	15/09/2009
63	Pruebas Integración	5 días	15/09/2009	22/09/2009
64	Definir Pruebas	2 días	15/09/2009	17/09/2009
65	Realizar Pruebas	3 días	17/09/2009	22/09/2009
66	Final del Proyecto	0 días	22/09/2009	22/09/2009

Tabla 1. Tabla de Planificación de Actividades y Tareas



Una vez indicada la tabla de planificación de las actividades y tareas, se muestra a continuación el calendario laboral utilizado para llevar a cabo esta planificación.

Vista previa del período laborable

Leyenda:

- Período laborable
- Período no laborable

	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
8 ⁰⁰							
9 ⁰⁰							
10 ⁰⁰							
11 ⁰⁰							
12 ⁰⁰							
13 ⁰⁰							
14 ⁰⁰							
15 ⁰⁰							
16 ⁰⁰							
17 ⁰⁰							
18 ⁰⁰							
19 ⁰⁰							
20 ⁰⁰							
21 ⁰⁰							

Ilustración 10. Calendario Laboral

Tal y como muestra la anterior ilustración del calendario laboral, el horario se fija a razón de 7 horas diarias que hacen un total de 35 horas semanales y de 21 días laborables por mes.

Es importante destacar que todas las tareas descritas en el diagrama las ha llevado a cabo la misma persona desempeñando diferentes roles, por lo cual todas las tareas tienen vínculos Fin a Comienzo con una dedicación del 100%, con el fin de evitar una sobre-asignación de la persona (en este el Autor de este proyecto).



2.4 ESTIMACIÓN DE COSTES

Este cálculo se ha realizado en función de las horas dedicadas a cada tarea y los recursos consumidos en cada una de ellas.

En este caso únicamente se ha utilizado un ingeniero técnico informático con un coste de 50€ / hora y no se han tenido en cuenta recursos materiales para realizar la estimación del coste del proyecto. Como se puede ver el coste es bastante alto para tratarse de un perfil junior en un Proyecto Fin de Carrera, pero la complejidad y el esfuerzo invertido tiene que verse reflejado en el coste total, para así poder ver realmente cuál ha sido el trabajo desempeñado.

La siguiente tabla muestra el coste de las actividades de primer nivel cuyo coste es la suma de los costes desglosados de las tareas que componen la actividad.

Id	Nombre	Duración	Comienzo	Fin	Coste (en €)
1	Gestión del Proyecto	9,6 días	02/03/2009	13/03/2009	480
2	Ciclo de Vida	0,8 días	02/03/2009	02/03/2009	40
6	Organización	4,8 días	02/03/2009	09/03/2009	240
10	Planificación	2,4 días	09/03/2009	11/03/2009	120
12	Estimación	1,6 días	12/03/2009	13/03/2009	50
16	Determinar Viabilidad	13,5 días	13/03/2009	02/04/2009	675
17	Requisitos de Usuario	5,5 días	13/03/2009	23/03/2009	275
21	Plan de Proyecto	8 días	23/03/2009	02/04/2009	400
23	Análisis	24,5 días	02/04/2009	06/04/2009	1225
24	Requisitos Software	6,5 días	02/04/2009	10/04/2009	325
28	Análisis Arquitectura	8 días	10/04/2009	22/04/2009	400



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

30	Casos de Uso	10 días	22/04/2009	06/05/2009	500
33	Diseño	29 días	06/05/2009	16/06/2009	1450
34	Diseño Interfaces	8 días	06/05/2009	18/05/2009	400
37	Diseño BBDD	4 días	18/05/2009	22/05/2009	200
42	Diseño Arquitectura	6 días	22/05/2009	01/06/2009	300
45	Diseño Detallado	11 días	01/06/2009	16/06/2009	550
48	Desarrollo	60 días	16/06/2009	08/09/2009	3000
49	Codificar Interfaces	47 días	16/06/2009	20/08/2009	2350
51	Codificar BD	8 días	20/08/2009	01/09/2009	400
55	Manual de Usuario	5 días	01/09/2009	08/09/2009	250
57	Integración y Pruebas	10 días	08/09/2009	22/09/2009	500
58	Pruebas Unitarias	5 días	08/09/2009	15/09/2009	250
63	Pruebas Integración	5 días	15/09/2009	22/09/2009	250
66	Final del Proyecto	0 días	22/09/2009	22/09/2009	0

Tabla 2. Tabla de Estimación de Costes

Los costes que se obtienen son resultado de la multiplicación del número de días de duración de cada actividad o tarea por el número de horas dedicadas cada día, que ya se ha comentado que son siete, por los 50€ estimados por hora.

Hay que destacar que esta no es una factura real del coste de un proyecto, ya que faltaría contabilizar el IVA y los recursos materiales, formación, soporte... En esta cantidad tampoco están incluidas las retribuciones asociadas a toda factura entre los que se destacan las cotizaciones a la Seguridad Social y las retenciones a cuenta del I.R.P.F. Por tanto, el coste del proyecto no es del todo real.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 2: Gestión del Proyecto

En resumen, el coste total del proyecto es la suma de los costes de cada fase que a continuación se muestra desglosado en la siguiente tabla.

Id	Nombre	Duración	Comienzo	Fin	Coste (en €)
1	Gestión del Proyecto	9,6 días	02/03/2009	13/03/2009	480
16	Determinar Viabilidad	13,5 días	13/03/2009	02/04/2009	675
23	Análisis	24,5 días	02/04/2009	06/04/2009	1225
33	Diseño	29 días	06/05/2009	16/06/2009	1450
48	Desarrollo	60 días	16/06/2009	08/09/2009	3000
57	Integración y Pruebas	10 días	08/09/2009	22/09/2009	500
66	Final del Proyecto	0 días	22/09/2009	22/09/2009	0
TOTAL:					7330 €

Tabla 3. Tabla Resumen de Costes



CAPITULO 3. ESTADO DE LA CUESTIÓN

3.1 INTRODUCCIÓN

El proyecto llevado a cabo, consiste en implementar una herramienta para reutilizar productos software en un entorno colaborativo, facilitando así la aportación de conocimiento por parte del conjunto de integrantes que componen el equipo de trabajo.

Este proyecto da continuidad a una herramienta que fue creada con anterioridad, utilizando otro tipo de tecnología, de la cual solamente se ha reutilizado el módulo administrativo para poder gestionar todos los componentes del sistema.

La idea del cambio de plataforma se planteó cuando varias personas querían interaccionar sobre el mismo proyecto, además de plantear la posibilidad de ser accesible desde cualquier punto, por ello se sugirió un sistema Web para gestionar cada proyecto de forma optimizada.

A medida que se fue profundizando en el trabajo, se plantearon varias posibilidades donde se podría implementar este sistema que resolvería gran parte de los requisitos planteados por el cliente, la elección final fue un producto que se ajustaba al 100% a nuestro sistema llamado **Microsoft Office SharePoint Server 2007**.

Finalmente, después de plasmar el sistema antiguo en esta plataforma se debía ampliar dicha plataforma para poder cerrar el ciclo en cuanto a la idea de la arquitectura PIBOK-PB se refiere. Esta ampliación consistía por un lado, en el desarrollo de pequeñas funcionalidades que faciliten la gestión de proyectos a través de esta herramienta. Por otro lado, en la implementación del módulo de patrones de producto que se asignan a cada una de las tareas integradas en el proyecto, así como la consolidación del sistema en un documento de planificación de proyectos para que posteriormente se puedan asignar recursos y tiempos a cada actividad.

En los apartados que vienen a continuación, se explica detalladamente cada uno de los pasos a seguir en el proyecto software.



3.2 ANÁLISIS DE LA IMPLEMENTACIÓN

A continuación, se van a detallar cada uno de los pasos necesarios en la implementación del sistema:

1. OBTENER CONOCIMIENTO DE LOS CONCEPTOS PLANTEADOS EN EL ANTIGUO SISTEMA

En primer lugar, debemos realizar es un estudio de todo el sistema que queremos replicar, adquiriendo el conocimiento planteado en el antiguo sistema así como la exposición de nuevas ideas con el fin de mejorar la interacción con el usuario.

El conocimiento del sistema experto (este módulo lo podemos identificar como tal, ya que cumple todos los principios necesarios para ser un sistema basado en conocimiento) requiere de un tiempo para adquirir todo su contenido, ya que se utilizan una serie de términos y conceptos que pueden resultar desconocidos para el desarrollador del sistema. Para ello se precisó de la ayuda de la documentación facilitada de la antigua aplicación, así como de la ayuda de los jefes de proyecto que diseñaron el sistema.

Una vez revisada toda la documentación y manuales de referencia existentes, podíamos identificar algunos problemas que resultaban ambiguos para el usuario y los cuales debían pulirse para solucionar ese tipo de ambigüedades.

2. ANALIZAR BBDD RELACIONADA CON EL ANTIGUO SISTEMA

En primer lugar, se debe restaurar la antigua BBDD junto con su estructura y contenidos asociados a la aplicación, esto es muy importante ya que en los manuales de referencia utilizó dichos datos y así podremos simular el mismo proceso con el fin de entender un poco más la BBDD al completo.

Mediante la aplicación de administración que actualmente sigue usándose podemos interactuar con la BBDD, asociando cada uno de los campos de la interfaz con sus correspondientes en el modelo de datos.

Más adelante se estudiará la posibilidad de añadir nuevos campos a la BBDD para mejorar la interacción con el usuario, como bien se comentaba anteriormente.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

3. IMPLEMENTAR DESARROLLO RELACIONADO EN NUEVA PLATAFORMA WEB

Una vez hemos adquirido el conocimiento y analizado el sistema al completo, podemos empezar con el desarrollo de la aplicación, plasmando toda la interfaz en la nueva plataforma comentada anteriormente, donde debido a la especialidad del producto (gestión documental) el desarrollo de dicho sistema en una tecnología basada en Web lo hace más difícil de completar satisfactoriamente.

Para el desarrollo de dicho sistema se emplearon una serie de herramientas, tecnología y lenguaje de programación específicos para la correcta implementación de la aplicación, dichos conceptos se tratarán en puntos posteriores de este documento.

4. ANALIZAR NUEVOS DESARROLLOS (BBDD – Interfaz)

A partir de este punto, la funcionalidad a desarrollar resulta ser bastante complicada ya que los requisitos funcionales eran un poco confusos por la complejidad y duración de los mismos, pero poco a poco estudiando en profundidad la tecnología en la que se iba a implementar dichos desarrollos se fueron abordando junto con todos los problemas que una plataforma como la de SharePoint nos pueda plantear.

En el Anexo 1 (Pág. xxx) se encuentra especificado con todo detalle el análisis de la BBDD – Interfaz junto con todas sus modificaciones posteriores, además de exponer la documentación complementaria que se facilitó y mediante el cual fue viable el desarrollo de este proyecto.

5. DISEÑAR INTERFAZ DE USUARIO

Este apartado resulta ser uno de los más complicados de elaborar, mientras en otros proyectos el diseño de la interfaz no pasa de ser un pequeño desarrollo con poco esfuerzo y diseño preestablecido, hay que destacar la complejidad del mismo aplicado al diseño tanto de las páginas maestras como de las páginas base, ya que se están diseñando la interfaz sobre un gestor documental totalmente estandarizado capaz de ofrecernos multitud de utilidades.

En el Anexo 2 (Pág. xxx) se explica primero el contenido de los distintos componentes que son necesarios para el correcto funcionamiento de la aplicación, y posteriormente se facilita un manual de instalación con cada paso explicado al detalle ya que su instalación resulta ser bastante compleja.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

6. DESARROLLAR INTERFAZ EN EL SISTEMA

El último de los pasos necesarios para completar el proyecto es el propio desarrollo del mismo, el cual otro de los puntos más complicados de ejecutar ya que se utilizaron herramientas y lenguajes de programación muy novedosos.

Todo lo relacionado con la tecnología y herramientas aplicadas en el proyecto se comentará en el siguiente punto, por tanto solamente queda resaltar el esfuerzo invertido desarrollando los componentes no solo para que funcionaran y aportaran lo necesario para salir adelante, sino que se estructuraron de tal manera que más adelante esta aplicación podrá ser mantenida por cualquier otra persona con conocimientos básicos de la tecnología.



3.3 TECNOLOGÍA UTILIZADA

3.3.1 INTRODUCCIÓN

Para llevar a cabo la interfaz de usuario que va a servir al jefe de proyecto como utilidad de planificación en futuros proyectos se necesita una tecnología bastante cualificada, al tratarse de una aplicación Web existen multitud de factores que hacen que el sistema funcione y se muestre tal y como en un principio habíamos imaginado.

Por tanto, se necesita una plataforma que sea compatible con aplicaciones Web y además que pueda integrarse con otro tipo de herramientas, en este caso es necesaria la integración con un gestor documental que para mayor facilidad debe haber sido desarrollado en la misma plataforma, así podrá personalizarse todo lo que queramos.

A continuación, se tratará de explicar cómo está estructurada la plataforma .NET que ha sido la base de todo el trabajo realizado, así como qué librerías nos aporta para desarrollar la interfaz de usuario con la mayor brevedad posible. Además se expondrá los motivos de porqué se desarrolló con el lenguaje C#, junto con un breve argumento del producto de Microsoft utilizado (MOSS).

3.3.2 PLATAFORMA .NET

La plataforma .NET es una plataforma de desarrollo de software con especial énfasis en el desarrollo rápido de aplicaciones, la independencia de lenguaje y la transparencia a través de redes.

La plataforma está compuesta por los siguientes componentes:

- Un conjunto de lenguajes de programación (C#, J#, JScript, C++ gestionado, Visual Básic.NET, y otros proyectos independientes).
- Un conjunto de herramientas de desarrollo (entre ellos Visual Studio.NET de Microsoft o Monodevelop)
- Una librería de clases amplia y común para todos los lenguajes.
- Un sistema de ejecución de Lenguaje Común. (CLR).
- Un conjunto de servidores .NET
- Un conjunto de servicios .NET
- Dispositivos electrónicos con soporte .NET



Ilustración 11. Marco de Trabajo de .NET

Los principales pilares sobre los que se sostiene dicha plataforma son:

1. Independencia de lenguaje

Todos los lenguajes que conformen con los estándares .NET, sin importar cual, podrán inter operar entre sí de forma totalmente transparente, las clases podrán ser heredadas entre unos lenguajes y otros, y se podrá disfrutar de polimorfismo entre lenguajes. Por ejemplo, si yo tengo una clase en C#, esta clase podrá ser heredada y utilizada en Visual Basic o JScript o cualquier lenguaje .NET. Todo esto es posible por medio de una de las características de .NET llamado Common Type System (CTS). También tiene la cualidad de que se pueden incluir más lenguajes a la plataforma. En la actualidad existen proyectos independientes de incluir PHP, Python, Ada y otros lenguajes en la plataforma.

2. Librería de clases común

Más de 4000 clases, objetos y métodos incluidos en la plataforma .NET están disponibles para todos los lenguajes.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

3. Multiplataforma

Cuando un programa es compilado, no es compilado en un archivo ejecutable sino en un lenguaje intermedio llamado “Lenguaje Intermedio” (IL) el cual podrá ser ejecutado por el CLR (Common Language Runtime) en la plataforma en que el CLR esté disponible. Hasta el día de hoy Microsoft solamente tiene un CLR para los sistemas operativos Windows.

El CLR compilará estos archivos IL nuevamente en código de máquina en un proceso que se conoce como JIT (“just in time”) el cual se ejecutará cuando se requiera. Este proceso producirá código de máquina que se reutilizará si es que hubiera código que se repitiera, haciendo que los programas sean ejecutados muy eficientemente.

4. Windows Forms, Web Forms, Web Services

La plataforma .NET incluye un conjunto de clases especial para datos y XML que son la base de 3 tecnologías claves: Servicios Web (Web Services), Web Forms, y Windows Forms los cuales son poderosas herramientas para la creación de aplicaciones tanto para la plataforma como para Web.

5. Estandarización

Además de los méritos técnicos, una de las razones del éxito de la plataforma .NET ha sido por el proceso de estandarización que Microsoft ha seguido (y que ha sorprendido a más de uno). Microsoft, en lugar de reservarse todos los derechos sobre el lenguaje y la plataforma, ha publicado las especificaciones del lenguaje y de la plataforma, que han sido posteriormente revisadas y ratificadas por la Asociación Europea de Fabricantes de Computadoras (ECMA). Esta especificación (que se puede descargar libremente de Internet) permite la implementación del lenguaje C# y de la plataforma .NET por terceros, incluso en entornos distintos de Windows.



3.3.3 ¿POR QUÉ USAR EL LENGUAJE C#?

Principales propiedades del lenguaje C#:

- El lenguaje es muy sencillo, sigue el mismo patrón de los lenguajes de programación modernos. Incluye un amplio soporte de estructuras, componentes, programación orientada a objetos, manipulación de errores, recolección de basura, etc., que es construido sobre los principios de C++ y Java.
- Como ya sabréis, las clases son la base de los lenguajes de programación orientados a objetos, lo cual permite extender el lenguaje a un mejor modelo para solucionar problemas. C# contiene las herramientas para definir nuevas clases, sus métodos y propiedades, al igual que la sencilla habilidad para implementar encapsulación, herencia y polimorfismo, que son los tres pilares de la programación orientada a objetos.
- C# tiene un nuevo estilo de documentación XML que se incorpora a lo largo de la aplicación, lo que simplifica la documentación en línea de clases y métodos.
- C# soporta también interfaces, una forma de estipular los servicios requeridos de una clase. Las clases en C# pueden heredar de un padre pero puede implementar varias interfaces.
- C# también provee soporte para estructuras, un concepto el cual ha cambiado significativamente desde C++. Una estructura es un tipo restringido que no exige tanto del sistema operativo como una clase. Una estructura no puede heredar ni dar herencias de clases pero puede implementar una interfaz.
- C# provee características de componentes orientados, como propiedades, eventos y construcciones declaradas (también llamados atributos). La programación orientada a componentes es soportada por el CLR. C# provee soporte para acceder directamente a la memoria usando el estilo de punteros de C++ y mucho más.



C# frente a Java

C# y Java son lenguajes similares, de sintaxis basada en C/C++, orientados a objetos, y ambos incluyen las características más importantes de los lenguajes modernos, como son la gestión automática de memoria y la compilación a código intermedio. Pero por supuesto, también hay diferencias.

Una de las diferencias más importantes es que C# es mucho más cercano a C++ en cuanto a diseño se refiere. C# toma casi todos sus operadores, palabras reservadas y expresiones directamente de C++. También se han mantenido algunas características que en Java se han desestimado. Por ejemplo las enumeraciones. No hay enumeraciones en Java y sin embargo era un concepto muy usado en C/C++. En C# se han mantenido las enumeraciones, y se han adaptado al nuevo lenguaje, de forma que ahora las enumeraciones no son simplemente enteros, sino que son clases de tipado seguro que derivan de System.Enum en la librería de clases base. Una enumeración de tipo "ej1" no se puede cambiar con una enumeración de tipo "ej2" sin una conversión.

Otra característica que no está presente en Java es la posibilidad de trabajar directamente con direcciones de memoria. Si bien tanto Java como .NET proporcionan gestión automática de memoria, en C# es posible usar lo que se denomina "código no seguro". Cuando se usa código no seguro en C# es posible operar con punteros de forma muy similar a como se haría en C/C++, pero el código que utiliza punteros se queda marcado como no seguro y no se ejecuta en entornos en los que no tiene permisos.

C# frente a C++

Puesto que C# se ejecuta en una máquina virtual, ésta se hace cargo de la gestión de memoria y por lo tanto el uso de punteros es mucho menos importante en C# que en C++.

C# también es mucho más orientado a objetos, hasta el punto de que todos los tipos usados derivan en última instancia el tipo 'object'. Además, muchos tipos se usan de forma distinta. Por ejemplo, en C# se comprueban los límites de los arrays antes de usarlos, evitando así que se pueda escribir pasado el final del vector.

Al igual que Java, C# renuncia a la idea de herencia múltiple de clases presente en C++. Sin embargo, referido a clases, C# implementa 'propiedades' del tipo de las que existen en Visual Basic, y los métodos de las clases son accedidos mediante '.' en lugar de '::'.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

¿Porqué C#?

La plataforma .NET acepta varios lenguajes. Por ahora, C#, Visual Basic, C++ gestionado, Nemerle, FORTRAN, Java, Python, etc., y con capacidad para aceptar prácticamente cualquier lenguaje. Entonces la pregunta es, ¿porqué hemos elegido C# en lugar de cualquier otro lenguaje?

La razón fundamental es que C# se diseñó para la plataforma .NET y es capaz de utilizar todo su potencial. También es cierto que es un lenguaje "limpio" en el sentido de que al no tener que proporcionar compatibilidad hacia atrás se ha tenido más libertad en el diseño y se ha puesto especial hincapié en la simplicidad. Por ejemplo, en C# hay un tipo de clase y siempre se le aplica el recolector de basura mientras que en C++ gestionado hay dos tipos de clases, una a la que se aplica el recolector y otra a la que no.

Otro de los motivos fundamentales es el conocimiento amplio de este lenguaje, por lo que resultaría mucho más rápido y eficaz utilizar dicho lenguaje, así se podrá aplicar ese tiempo ganado en realizar un buen análisis y diseño del sistema.

3.3.4 Microsoft Office SharePoint Server

Este producto es la base de la interfaz de usuario, lo cual posiblemente provoque escepticismo en cuanto a la idea de utilizar MOSS como una aplicación Web, ya que sus principales características no son precisamente la publicación de contenidos Web, te da esa posibilidad pero destaca más por la gestión documental tan robusta que contiene aplicando versionado/histórico sobre no sólo sobre una lista de contenidos sino sobre un elemento en concreto, además nos facilita un gestor de alertas y flujos de trabajo muy útiles para cualquier organización donde añadiendo el buscador potente que contiene MOSS hace de esta herramienta algo más que una plataforma para alojamiento Web.

En definitiva, SharePoint te da la posibilidad de ampliar las líneas de trabajo haciendo de este proyecto un mero anticipo de lo que realmente puede llegar a convertirse, por este motivo se expone a continuación una serie de **características principales / ventajas** para que seamos conscientes de lo que realmente tenemos en nuestras manos:



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

1. Proporcionar una experiencia del usuario fácil, familiar y coherente.

Office SharePoint Server 2007 está totalmente integrado con las aplicaciones cliente de escritorio tan conocidas, el correo electrónico y los exploradores Web para proporcionar una experiencia del usuario coherente que simplifique la interacción con el contenido, los procesos y los datos empresariales. Esta interacción total, asociada a una funcionalidad avanzada y sólida, le ayuda a implementar los propios servicios y a facilitar la adopción de productos.

2. Potenciar la productividad de los empleados simplificando las actividades empresariales diarias.

Aproveche los flujos de trabajo listos para usar con el fin de iniciar, realizar el seguimiento y elaborar los informes de las actividades empresariales normales, como la revisión y aprobación de documentos, el seguimiento de problemas y la recopilación de firmas. Puede terminar estas actividades sin necesidad de usar código. La completa integración con las aplicaciones familiares cliente, el correo electrónico y los exploradores Web le proporcionan una experiencia sencilla y coherente. La modificación y ampliación de estos procesos listos para usar se facilita mediante el uso de herramientas como Microsoft Office SharePoint Designer 2007 (la siguiente versión de Microsoft Office FrontPage).

3. Ayudar a cumplir las normativas legales a través del control total del contenido.

Mediante la especificación de la configuración de seguridad, las directivas de almacenamiento, las directivas de auditoría y las acciones de caducidad de los registros empresariales según las normativas legales, puede ayudar a garantizar el control y la administración eficaces de la información empresarial confidencial. Asimismo puede reducir el riesgo de litigios de la organización. La completa integración de Office SharePoint Server 2007 con las aplicaciones de escritorio familiares indica que se muestra la configuración de directivas en las aplicaciones cliente de Microsoft Office System, facilitando así que los empleados conozcan las normativas legales y las cumplan.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

4. Administrar y volver a usar eficazmente el contenido para incrementar el valor empresarial.

Los usuarios de la empresa y los escritores de contenido pueden crear y enviar el contenido para su aprobación e implementación programada a la intranet o a sitios de Internet. La administración de contenido multilingüe se simplifica mediante las nuevas plantillas de la biblioteca de documentos que se han vuelto a diseñar específicamente para mantener una relación entre la versión original y las distintas traducciones de un documento.

5. Simplificar el acceso de toda la organización a la información tanto estructurada como sin estructurar mediante distintos sistemas.

Proporcione a los usuarios acceso a los datos empresariales de los sistemas habituales de línea de negocio como SAP y Siebel a través de Office SharePoint Server 2007. Los usuarios también pueden crear vistas personalizadas e interactuar con sistemas empresariales mediante un explorador arrastrando conexiones de servidor configurables. Los repositorios de documentos administrados de la empresa ayudan a las organizaciones a almacenar y a organizar los documentos empresariales en una ubicación central.

6. Conectar a los usuarios con la información y los conocimientos.

Enterprise Search de Office SharePoint Server 2007 incorpora datos empresariales junto con la información de documentos, personas y páginas Web para proporcionar resultados completos e importantes. Características como la contracción de duplicados, la corrección ortográfica y las alertas mejoran la importancia de los resultados para que pueda encontrar fácilmente lo que necesite.

7. Acelerar los procesos empresariales compartidos en los límites empresariales.

Sin tener que codificar ninguna aplicación personalizada, puede usar soluciones inteligentes basadas en formularios electrónicos para recopilar la información empresarial fundamental de los clientes, asociados y proveedores mediante un explorador Web. Las reglas de validación de datos integradas le ayudan a recopilar datos precisos y coherentes que se pueden integrar directamente en sistemas de servidor para evitar la redundancia y los errores resultantes de la entrada manual de datos.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 3: Estado de la Cuestión

8. Compartir los datos empresariales sin divulgar información confidencial.

Dé a sus empleados acceso en tiempo real a las hojas de cálculo interactivas de Microsoft Office Excel desde un explorador Web a través de Excel Services que se ejecute en Office SharePoint Server 2007. Utilice estas hojas de cálculo para mantener y compartir eficazmente una sola versión centralizada y actualizada, a la vez que ayuda a proteger la información confidencial incrustada de los documentos (como los modelos financieros).

9. Permitir que los usuarios tomen decisiones con más información mostrando la información fundamental empresarial en una ubicación central.

Office SharePoint Server 2007 facilita la creación de portales de inteligencia empresarial (BI) interactivos que reúnan y muestren la información empresarial fundamental de orígenes distintos, mediante el uso de las funcionalidades de la BI como paneles, elementos Web, tarjetas de puntuación, indicadores clave de rendimiento (KPI) y tecnologías de conectividad de los datos empresariales. Los sitios del centro de informes centralizado proporcionan a los usuarios un solo lugar para buscar los informes, las hojas de cálculo o los KPI más recientes.

10. Proporcionar una plataforma única e integrada para administrar las aplicaciones de la intranet, la extranet e Internet en toda la empresa.

Office SharePoint Server 2007 se basa en una arquitectura abierta y escalable, con compatibilidad con servicios Web y estándares de interoperabilidad, incluidos XML y el protocolo SOAP (Simple Object Access Protocol). El servidor tiene interfaces de programación de aplicaciones (API) avanzadas y abiertas, así como controladores de eventos para listas y documentos. Estas características permiten la integración con los sistemas existentes y proporcionan la flexibilidad necesaria para incorporar nuevas inversiones en TI que no sean de Microsoft.



CAPITULO 4. ANÁLISIS DEL SISTEMA

4.1 CATÁLOGO DE REQUISITOS

4.1.1 REQUISITOS FUNCIONALES

A continuación se argumentarán cada uno de los requisitos funcionales, dividiendo toda la estructura por módulos, así podremos estructurar mejor todo su contenido. Los módulos funcionales son los siguientes:

- **Módulo genérico**
- **Módulo gestor de proyectos**
- **Módulo de asignación Organización/Modelo de proceso**
- **Módulo gestor de modelos de procesos mediante sistema experto**
- **Módulo gestor personalizado de modelos de procesos**
- **Módulo gestor de patrones de producto**
- **Módulo gestor de seguimiento de proyectos**

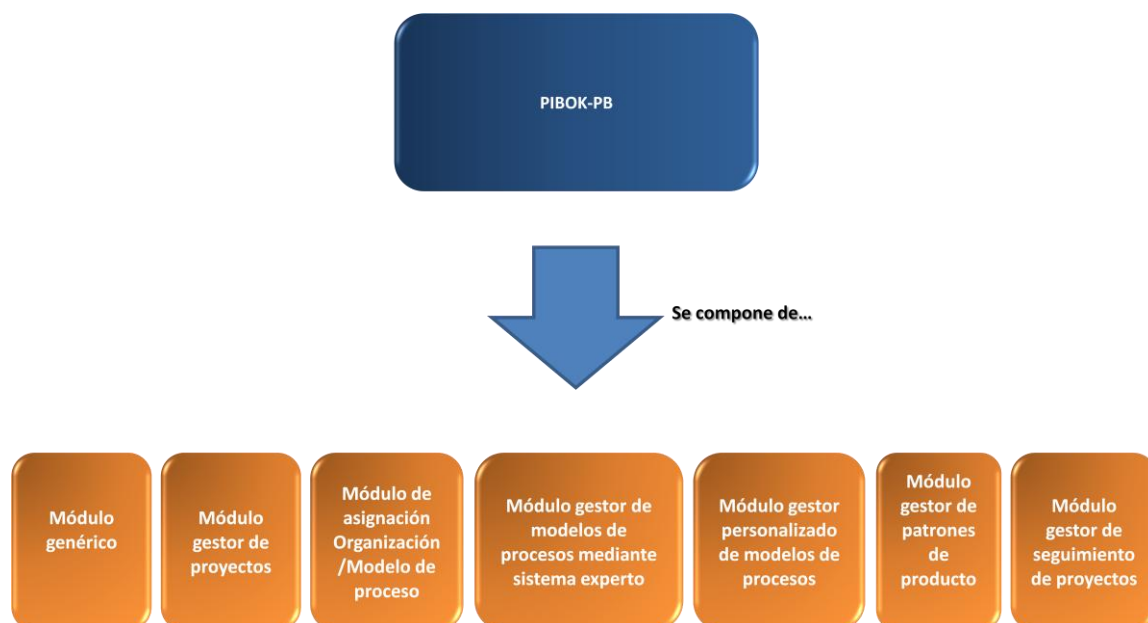


Ilustración 12. Módulos Funcionales



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.1.1 Módulo genérico

<u>ID</u>	REQF.1	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema se desarrollará sobre .NET con el fin de asegurar su escalabilidad y facilitar su diseño, desarrollo y mantenimiento. (Ej. Ampliar funcionalidad a partir del seguimiento de cada uno de los proyectos. Añadir la aplicación administrativa a una plataforma web.)			

<u>ID</u>	REQF.2	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Por el mismo motivo el sistema se desarrollará respetando una arquitectura tres capas (capa de interfaz, capa de lógica de negocio y capa de base de datos), de manera que si en un futuro uno de las tres cambiara, solo se tuviera que modificar la capa correspondiente.			

<u>ID</u>	REQF.3	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe garantizar la privacidad de cada uno de los jefes de proyecto, mostrando solamente los proyectos creados por el perfil especificado en un primer momento, los perfiles serán identificados como usuarios registrados en el dominio de la organización.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.4	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe garantizar que será accesible desde cualquier punto, los únicos requisitos serán obtener acceso a Internet y permisos en dicha aplicación.			

<u>ID</u>	REQF.5	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El administrador de la aplicación podrá gestionar los permisos de una forma rápida y eficaz, mediante el cual podrá identificar distintos perfiles (jefes de proyecto, analistas, programadores...) con el fin de facilitar una herramienta de colaboración con plenas garantías de éxito.			

<u>ID</u>	REQF.6	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema facilitará al jefe de proyecto la posibilidad de crear, modificar, eliminar y consultar todos los proyectos que haya planificado, realizando dichas operaciones de una forma fácil y coherente.			

<u>ID</u>	REQF.7	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> La BBDD que suministrará al sistema podrá estar alojada en cualquier servidor, siempre y cuando se tenga acceso desde el servidor origen, que es donde está alojada la aplicación Web.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.8	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El diseño utilizado para toda la aplicación debe ser coherente con los colores de la organización, así como todos los componentes deben mostrar claramente cuál es su estructura y finalidad, para ello se deben resaltar los títulos, marcar las tablas...			

4.1.1.2 Módulo gestor de proyectos

<u>ID</u>	REQF.9	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema facilitará un hipervínculo para crear nuevos proyectos.			

<u>ID</u>	REQF.10	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema facilitará un hipervínculo para acceder a un componente que gestione los proyectos creados por una persona determinada.			

<u>ID</u>	REQF.11	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El gestor citado anteriormente listará todos los proyectos en un menú.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.12	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Una vez se seleccione uno de los proyectos listados en el menú, el sistema mostrará el nombre y la descripción del proyecto seleccionado, estos campos aparecerán deshabilitados.			

<u>ID</u>	REQF.13	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema da la posibilidad de cambiar el título y la descripción del proyecto seleccionado.			

<u>ID</u>	REQF.14	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema facilitará un mecanismo para acceder al árbol de actividades del proyecto seleccionado en el menú, pudiendo retocar los patrones de producto y las propiedades de las tareas de seguimiento expuestas en una estructura "especial".			

<u>ID</u>	REQF.15	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema dará la posibilidad de eliminar un proyecto de una forma fácil y muy rápida, no solo lo elimina de la BBDD sino que también se eliminará de la lista de seguimiento.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.1.3 Módulo de asignación Organización/Modelo de proceso

<u>ID</u>	REQF.16	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Una vez se creó el proyecto en la BBDD el siguiente paso es asociarlo a una organización que ya pertenezca al sistema, esta carga de organizaciones de momento no está implementada, ya que deberá añadirse en el apartado de administración. Por tanto, se mostrará un listado con todas las organizaciones existentes.			

<u>ID</u>	REQF.17	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Antes de poder generar nuestro modelo de proceso se debe validar la selección de una organización mediante un texto identificativo.			

4.1.1.4 Módulo gestor de modelos de procesos mediante sistema experto

<u>ID</u>	REQF.18	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cuando seleccionamos la opción del sistema experto, dicho sistema debe facilitar en primera instancia una serie de parámetros, para así poder determinar cuál es el modelo de proceso que más se ajusta a las necesidades del jefe de proyecto. Los parámetros facilitados por el sistema experto son el resultado de meses de investigación.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.19	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Los parámetros comentados anteriormente deben dividirse en dos tipos: <ul style="list-style-type: none">• Organización• Proyecto Quando seleccionamos cualquiera de los dos tipos expuestos anteriormente, se deben listar claramente cada uno de los parámetros pertenecientes al tipo seleccionado.			

<u>ID</u>	REQF.20	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cada parámetro mostrará un listado de valores asociados a dicho parámetro, no es obligatorio seleccionar un valor por lo que siempre tendremos la opción de dejarlo vacío.			

<u>ID</u>	REQF.21	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Quando hemos finalizado la selección de valores de los parámetros que consideramos importantes para nuestro proyecto, el sistema verificará que al menos uno de los parámetros fue seleccionado.			

<u>ID</u>	REQF.22	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe facilitar al jefe de proyecto la posibilidad de redefinir los parámetros seleccionados anteriormente, por si el resultado no fue el esperado.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.23	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema recuperará una serie de reglas almacenadas en la BBDD, las cuales serán utilizadas para mostrar los modelos de proceso que más se ajusten a los valores seleccionados por el jefe de proyecto.			

<u>ID</u>	REQF.24	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Si las reglas heurísticas se ejecutaron correctamente, se mostrarán todos los modelos de procesos estándares almacenados en la BBDD, junto con el porcentaje de coincidencia entre las reglas seleccionadas por el usuario y las asociadas con el modelo de proceso. De esta forma se visualizará claramente qué modelo de proceso se ajusta más a nuestras necesidades, pudiendo depurar un poco más ese modelo redefiniendo de nuevo los parámetros de entrada.			

<u>ID</u>	REQF.25	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> En los modelos de proceso resultantes, se mostrarán todas las reglas asociadas al modelo en cuestión, para que así el usuario pueda percibir las incompatibilidades entre “su” modelo de proceso y el estandarizado.			

<u>ID</u>	REQF.26	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Las reglas listadas en cada modelo de proceso se desglosarán en dos tipos: <ul style="list-style-type: none">• Simples			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- Compuestas

<u>ID</u>	REQF.27	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cada regla expuesta en el resultado se compondrá de tres campos: <ul style="list-style-type: none">• Descripción del parámetro• Valor del parámetro en el modelo de proceso estándar• Icono con la correspondencia entre la selección por parte del usuario y el correspondiente del modelo estándar. (Coincidencia → TICK; No coincidencia → ASPA)			

<u>ID</u>	REQF.28	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cada modelo de proceso tiene asociado un árbol de actividades que solamente estará desplegado hasta el segundo nivel, indicando cual es la estructura del modelo de proceso estándar.			

<u>ID</u>	REQF.29	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cada modelo de proceso tiene asociado una lista de documentos que sirven como documentación complementaria, a veces muy útil para jefes de proyecto que desconozcan la existencia de algún modelo estándar.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.30	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Los documentos relacionados con un modelo de proceso en concreto se almacenarán en el gestor documental, pudiendo añadir todos los documentos que queramos en una biblioteca específica. En cuanto a la asociación de la documentación con el modelo de proceso, actualmente debe hacerse desde la propia BBDD ya que no existe ninguna aplicación que añada los documentos automáticamente.			

<u>ID</u>	REQF.31	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> La estructura base de todos los componentes pertenecientes al sistema experto, deberá seguir unas bases de diseño aportadas en un inicio mediante imágenes.			

<u>ID</u>	REQF.32	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Los documentos asociados a un modelo de proceso podrán descargarse en cualquier momento y así obtener físicamente cada uno de los documentos.			



4.1.1.5 Módulo gestor personalizado de modelos de procesos

<u>ID</u>	REQF.33	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> La opción para crear un modelo de proceso personalizado permitirá al jefe de proyecto elegir multitud de posibilidades, en cuanto a la selección de actividades con tal de satisfacer las necesidades del proyecto.			

<u>ID</u>	REQF.34	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe listar todas las actividades creadas hasta el momento, guardando su jerarquía concreta para si fuera necesario añadir todas las tareas a nuestro modelo de proceso personalizado.			

<u>ID</u>	REQF.35	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe listar todos los modelos de procesos existentes en la BBDD junto con todas sus actividades, de esta forma el jefe de proyecto podrá basarse en un modelo estándar pero con algunas ligeras modificaciones y así personalizarlo a su gusto.			

<u>ID</u>	REQF.36	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema listará en primera instancia, un modelo de proceso totalmente limpio sin actividades y sin tipos de actividad, por tanto existe la opción de crear proyectos que no tengan la estructura lógica (análisis, diseño, desarrollo...).			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.37	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cuando se añada una actividad a nuestro árbol de actividades se tendrán en cuenta todos los nodos/subactividades que pertenezcan a la tarea añadida.			

<u>ID</u>	REQF.38	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Si el tipo de actividad de esa tarea añadida no se representa en el modelo personalizado, será añadida automáticamente formando parte del árbol resultado. Por tanto, cuando se añade un elemento de segundo nivel, se añadirán todos sus hijos y el padre del primer nivel además del tipo de actividad asociado.			

<u>ID</u>	REQF.39	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema detectará si la actividad que se quiere añadir ya pertenece al árbol destino, por tanto en ningún momento existirán duplicados.			

<u>ID</u>	REQF.40	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema deberá facilitar una opción al usuario para eliminar actividades del árbol resultante, por tanto al eliminar una actividad tendremos en cuenta a todos sus hijos.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

ID

REQF.41

TIPO:

FUNCIONAL

DESCRIPCIÓN

El sistema deberá facilitar una opción al usuario para poder mover actividades (subir o bajar) dentro de un mismo nivel, haciendo que el modelo de proceso sea aún más personalizable.

ID

REQF.42

TIPO:

FUNCIONAL

DESCRIPCIÓN

Para poder acceder al siguiente paso, el modelo de proceso resultante deberá contener al menos una actividad ya que un modelo de proceso sin procesos es "un sin sentido".

ID

REQF.43

TIPO:

FUNCIONAL

DESCRIPCIÓN

El sistema podrá facilitar la opción de volver al sistema experto para obtener el modelo de proceso recomendado, de gran utilidad para usuarios indecisos.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.1.6 Módulo gestor de patrones de producto

<u>ID</u>	REQF.44	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Una vez escogido el modelo de proceso, se guardará en la BBDD asociándolo al proyecto y organización correspondientes. Posteriormente, se facilitará al usuario un gestor de patrones de producto para continuar con la creación del proyecto.			

<u>ID</u>	REQF.45	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe mostrar el árbol de actividades resultante, con el fin de asignar posteriormente los patrones de producto correspondientes a cada una de las actividades.			

<u>ID</u>	REQF.46	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema debe señalar de alguna forma a qué actividades de todo el árbol le corresponden patrones de producto. Estas asociaciones se realizan directamente en la BBDD, por tanto es otra de las tareas que deberían añadirse a la aplicación de administración.			

<u>ID</u>	REQF.47	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> Cuando se selecciona una de las actividades que contiene patrones de producto asociados, se mostrará en primera instancia el patrón de producto seleccionado. Si no hubiera patrón asociado entonces se mostrará un listado con todos los posibles patrones.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.48	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> La información que se debe presentar para cada uno de los patrones de producto es la siguiente: <ul style="list-style-type: none">• Nombre• Problema• Contexto inicial• Contexto final• Enlace• Estado selección El enlace será una dirección Web válida, probablemente será externa a la aplicación. El campo Estado selección, mostrará si en ese momento ese patrón está seleccionado o no, diferenciándolo del resto en el listado de patrones.			

<u>ID</u>	REQF.49	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema deberá ser capaz de gestionar la forma de seleccionar los patrones de producto para una actividad en concreto, guardando cada vez los patrones asociados a la actividad que se esté tratando además de identificar en todo momento la actividad actualizada.			

<u>ID</u>	REQF.50	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema no debe permitir la asociación de más de un patrón de producto con una actividad, por tanto la relación patrón de producto – actividad es 1->1.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQF.51	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema no debe permitir acceder al siguiente paso de la creación de un proyecto hasta que no se han completado todas las asociaciones, identificar mediante alguna tipografía especial cuando se completó una asociación.			

4.1.1.7 Módulo gestor de seguimiento de proyectos

<u>ID</u>	REQF.52	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema deberá mostrar un listado de actividades representadas en una lista con los patrones de producto asociados.			

<u>ID</u>	REQF.53	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> La vista representada por el sistema no sólo deberá mostrar el listado de actividades sino que dichas actividades serán plasmadas en un "Diagrama de Gantt".			

<u>ID</u>	REQF.54	<u>TIPO:</u>	FUNCIONAL
<u>DESCRIPCIÓN</u> El sistema será capaz de facilitar al usuario la posibilidad de editar cada una de las actividades y cambiarles la asociación de patrón de producto que tenga en ese momento. Por supuesto, este cambio se verá reflejado también en la BBDD, ya que la lista de seguimiento pertenece al gestor documental.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

ID

REQF.55

TIPO:

FUNCIONAL

DESCRIPCIÓN

La lista de seguimiento mostrará solamente las actividades que se están creando en ese momento, ya que realmente el sistema deberá almacenar todas las planificaciones en la misma lista, pero mostrando solamente el proyecto que el usuario esté editando.

ID

REQF.56

TIPO:

FUNCIONAL

DESCRIPCIÓN

Cada actividad en la lista de seguimiento se compondrá de los siguientes campos:

- Título
- Asignado a
- Estado de la tarea
- Prioridad
- Fecha de inicio
- Fecha de vencimiento
- Porcentaje completado
- Patrón seleccionado

Al completar cada una de estas propiedades deberá ser suficiente para representar esta actividad como una tarea de un documento plan de proyecto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.2 REQUISITOS DE RENDIMIENTO

<u>ID</u>	REQR.57	<u>TIPO:</u>	RENDIMIENTO
<u>DESCRIPCIÓN</u> El servidor donde esté alojada la aplicación debe garantizar un rendimiento óptimo de todo el sistema.			

<u>ID</u>	REQR.58	<u>TIPO:</u>	RENDIMIENTO
<u>DESCRIPCIÓN</u> El servidor donde se realice todo el almacenamiento de datos debe tener buena comunicación con el servidor de alojamiento Web, ya que a través de dichas comunicaciones se procesarán todas y cada una de las operaciones de BBDD.			

<u>ID</u>	REQR.59	<u>TIPO:</u>	RENDIMIENTO
<u>DESCRIPCIÓN</u> El sistema garantizará un buen comportamiento de la aplicación en cuanto al tiempo de espera del usuario por cada acción realizada, para todas y cada una de las acciones posibles el rendimiento será óptimo.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.3 REQUISITOS DE SEGURIDAD

<u>ID</u>	REQS.60	<u>TIPO:</u>	SEGURIDAD
<u>DESCRIPCIÓN</u> Los usuarios encargados de almacenar datos para ampliar el conocimiento del sistema experto, tendrán usuario y clave de acceso identificando cada una de sus acciones.			

<u>ID</u>	REQS.61	<u>TIPO:</u>	SEGURIDAD
<u>DESCRIPCIÓN</u> Los usuarios encargados de administrar la herramienta de gestión documental tendrán usuario y clave de acceso, serán los encargados de otorgar permisos a los jefes de proyecto.			

<u>ID</u>	REQS.62	<u>TIPO:</u>	SEGURIDAD
<u>DESCRIPCIÓN</u> Los jefes de proyecto tendrán usuario y clave de acceso exclusivos, ya que a partir del login se muestran todos los proyectos correspondientes.			

<u>ID</u>	REQS.63	<u>TIPO:</u>	SEGURIDAD
<u>DESCRIPCIÓN</u> Se deberá establecer un control de copias de seguridad para cumplir todos los estándares referidos a la seguridad, tareas programadas... de un proyecto software.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

<u>ID</u>	REQS.64	<u>TIPO:</u>	SEGURIDAD
<u>DESCRIPCIÓN</u> Se deberá establecer un control de versiones en el servidor que aloje todo el código fuente de la aplicación.			

4.1.4 REQUISITOS DE IMPLANTACIÓN

<u>ID</u>	REQUI.65	<u>TIPO:</u>	IMPLANTACIÓN
<u>DESCRIPCIÓN</u> La implantación del sistema debe incluir documentación relativa a la instalación/configuración del sistema, como nos encontramos ante un sistema bastante complejo ayudaría desglosar la documentación bloques.			

<u>ID</u>	REQUI.66	<u>TIPO:</u>	IMPLANTACIÓN
<u>DESCRIPCIÓN</u> La implantación debe realizarse respetando la arquitectura tecnológica anteriormente expuesta.			

<u>ID</u>	REQUI.67	<u>TIPO:</u>	IMPLANTACIÓN
<u>DESCRIPCIÓN</u> Respecto a la documentación se aportará un documento tipo FAQ, así se podrán resolver posibles problemas para futuros desarrollos.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.1.5 REQUISITOS TECNOLÓGICOS

<u>ID</u>	REQT.68	<u>TIPO:</u>	TECNOLÓGICOS
<u>DESCRIPCIÓN</u> El sistema debe basarse en una arquitectura flexible y ampliamente extendida para facilitar su distribución. La arquitectura tecnológica seleccionada para los desarrollos será .NET. Las plataformas utilizadas para su ejecución se basarán en entornos Windows que soporten la arquitectura (Windows Server 2003 o superior).			

<u>ID</u>	REQT.69	<u>TIPO:</u>	TECNOLÓGICOS
<u>DESCRIPCIÓN</u> El sistema además deberá tener instalados algunos componentes añadidos, como pueden ser el IIS 6 o superior, SharePoint Designer 2007, SQL Server 2005 o superior... todos con el fin de integrarse con el gestor documental que tantas veces se ha mencionado (MOSS)			

<u>ID</u>	REQT.70	<u>TIPO:</u>	TECNOLÓGICOS
<u>DESCRIPCIÓN</u> El servidor que contenga la BBDD origen tendrá un sistema operativo Windows Server 2003 o superior , con el fin de poder abordar todas las peticiones que le lleguen de la aplicación, además como dato añadido decir que la versión de SQL Server deberá ser 2005 o superior.			



Este es un ejemplo de una posible arquitectura del sistema, ahora mismo el sistema es bastante más reducido pero según pase el tiempo seguramente el sistema global sea muy parecido a esto:

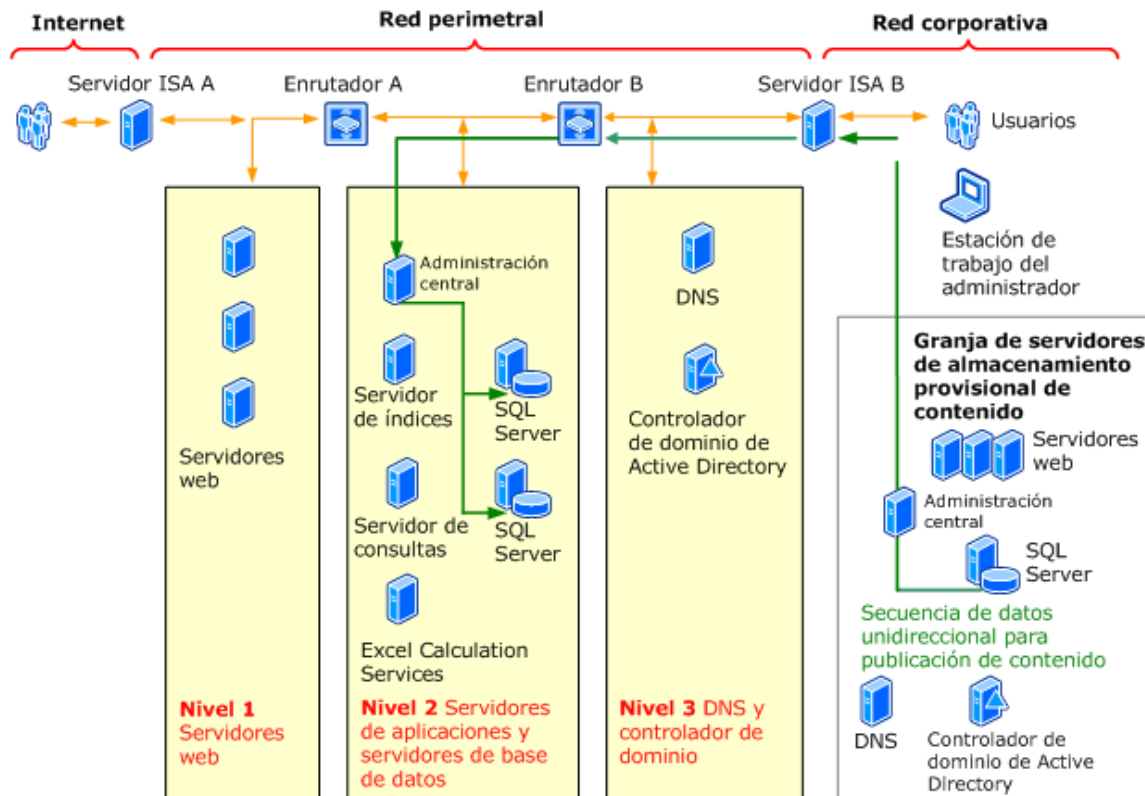


Ilustración 13. Entorno tecnológico del sistema



4.2 ESPECIFICACIÓN DE CASOS DE USO

Existen nueve escenarios diferentes los cuales se recogen en los siguientes diagramas de casos de uso, como punto de referencia se tomará al usuario con jefe de proyecto/administrador ya que la aplicación está orientada para estos perfiles:

- **Interacción Jefe de Proyecto con el sistema para gestionar proyectos.**
- **Interacción Jefe de Proyecto con el sistema para gestionar una organización.**
- **Interacción Jefe de Proyecto con el sistema para gestionar un modelo de proceso personalizado.**
- **Interacción Jefe de Proyecto con el sistema para gestionar un modelo de proceso mediante el sistema experto.**
- **Interacción Jefe de Proyecto con el sistema para gestionar patrones de producto.**
- **Interacción Jefe de Proyecto/Administrador con el sistema para gestionar un plan de proyecto.**
- **Interacción Jefe de Proyecto con el sistema para validarse en la aplicación.**
- **Interacción Administrador con el sistema para gestionar documentación.**
- **Interacción Administrador con el sistema para gestionar permisos.**

La siguiente ilustración muestra todos ellos, y en los sucesivos apartados se explicarán cada uno de estos escenarios en detalle.

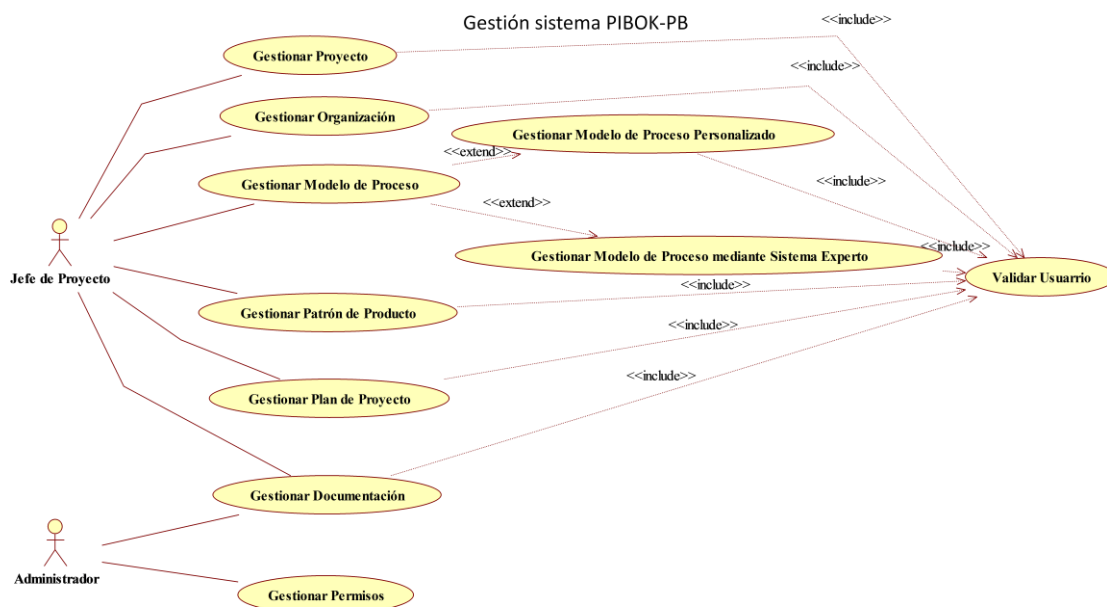


Ilustración 14. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar PIBOK-PB"

A continuación se presentará el formato expandido de cada caso de uso definido en este documento:

- **Caso de Uso:** Validar Usuario.
 - **Actores:** Jefe de Proyecto / Administrador
 - **Objetivo:** Validar al usuario dentro del sistema para que pueda acceder a toda su funcionalidad.
 - **Precondiciones:** No hay.
 - **Postcondiciones:** Usuario registrado en el sistema.
 - **Escenario básico:**
 - Introducir nombre de usuario.
 - Introducir contraseña.
 - Aceptar.



4.2.1 GESTIONAR UN PROYECTO

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un proyecto. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

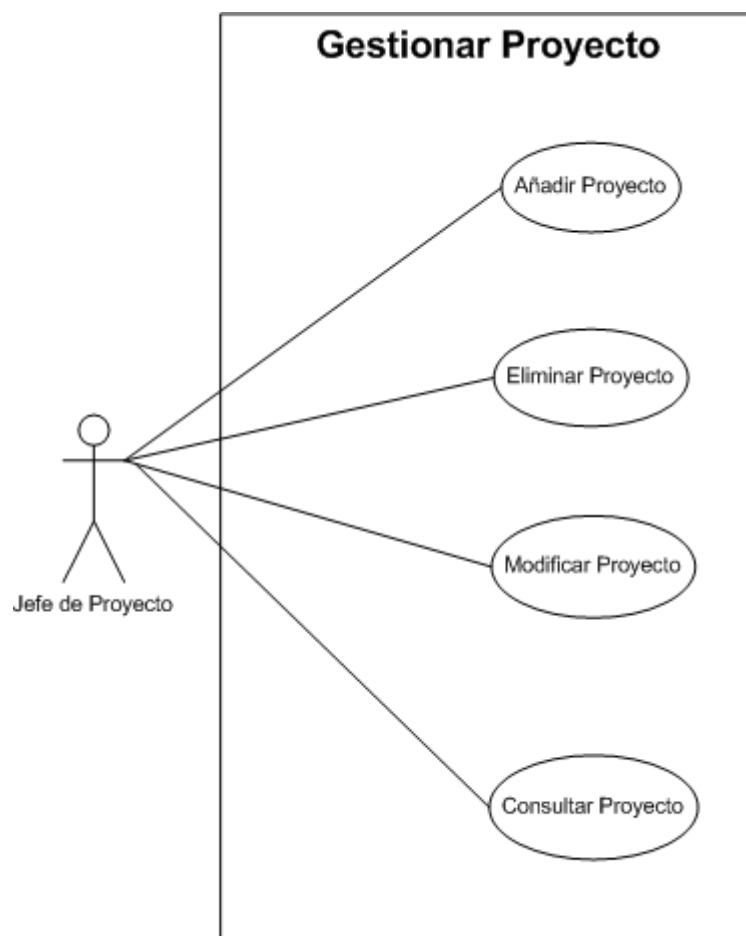


Ilustración 15. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para gestionar proyectos"



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Añadir Proyecto
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un nuevo proyecto a la base de datos del sistema, para poder elaborar su plan de proyecto correctamente.
 - **Precondiciones:** Usuario registrado como Jefe de Proyecto.
 - **Postcondiciones:** Nuevo proyecto insertado en la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Rellenar los campos nombre y descripción necesarios para identificar un proyecto unívocamente.
 - ↗ Aceptar

- **Caso de Uso:** Modificar Proyecto
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Modificar los datos de un proyecto existente en la base de datos del sistema, para redefinir los conceptos que forman parte de dicho proyecto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto asignado a ese proyecto.
 - ↗ Al menos un proyecto insertado en la base de datos.
 - **Postcondiciones:** Información de proyecto modificado.
 - **Escenario básico:**
 - ↗ Seleccionar el proyecto deseado de entre todos los existentes.
 - ↗ Modificar la información necesaria.
 - ↗ Aceptar los cambios.



- **Caso de Uso:** Eliminar Proyecto
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Eliminar un proyecto de la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto con permisos respecto a ese proyecto (debe haber sido creado por él).
 - ↗ Al menos un proyecto insertado.
 - **Postcondiciones:** Proyecto eliminado de la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Seleccionar el proyecto deseado de entre todos los existentes.
 - ↗ Confirmar la eliminación.

- **Caso de Uso:** Consultar Proyecto
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Consultar la información del proyecto seleccionado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de proyecto asignado a este proyecto.
 - ↗ Al menos un proyecto insertado.
 - **Postcondiciones:** Modificación en la fecha de consulta.
 - **Escenario básico:**
 - ↗ Seleccionar el proyecto que deseemos consultar de entre todos los existentes.



4.2.2 GESTIONAR UNA ORGANIZACIÓN

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar una organización. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

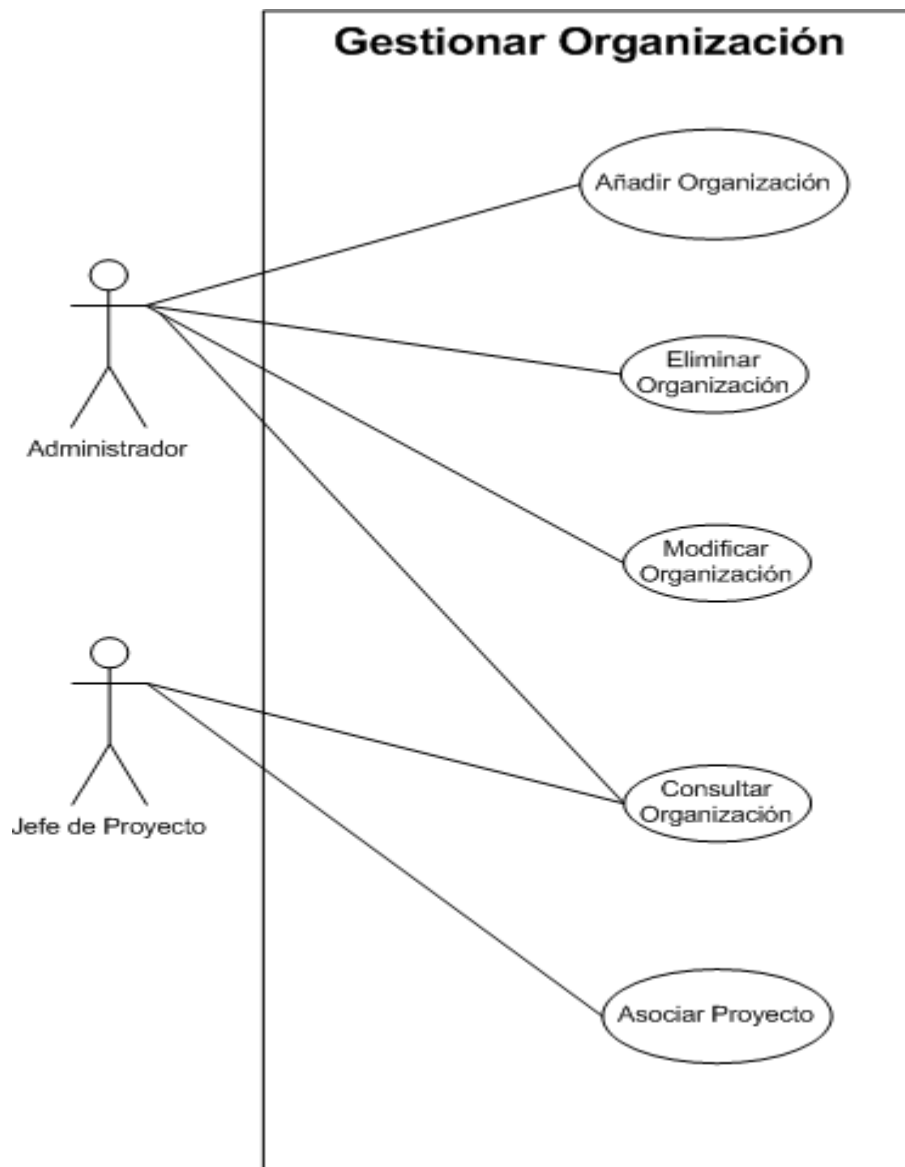


Ilustración 16. Diagrama de casos de uso: "Interacción Usuario Experto con el sistema para gestionar una situación"



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Añadir Organización.
 - **Actores:** Administrador.
 - **Objetivo:** Añadir una nueva organización a la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como administrador.
 - **Postcondiciones:** Nueva organización insertada en la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Abrir la base de datos.
 - ↗ Acceder como administrador.
 - ↗ Insertar la información referida a una organización.
 - ↗ Aceptar.

- **Caso de Uso:** Modificar Organización.
 - **Actores:** Administrador.
 - **Objetivo:** Modificar los datos de una organización existente en la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como administrador.
 - ↗ Al menos una organización insertada.
 - **Postcondiciones:** Información de la organización modificada.
 - **Escenario básico:**
 - ↗ Abrir la base de datos.
 - ↗ Acceder como administrador.
 - ↗ Modificar la información referida a una organización.
 - ↗ Aceptar.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Eliminar Organización
 - **Actores:** Administrador.
 - **Objetivo:** Eliminar una organización de la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como administrador.
 - ↗ Al menos una organización insertada.
 - **Postcondiciones:** Organización eliminada de la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Seleccionar la organización deseada de entre todos los existentes.
 - ↗ Confirmar la eliminación.

- **Caso de Uso:** Consultar Organización.
 - **Actores:** Administrador/Jefe de Proyecto.
 - **Objetivo:** Consultar la información de la organización seleccionada.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador/Jefe de Proyecto.
 - ↗ Al menos una organización insertada.
 - **Postcondiciones:** No hay.
 - **Escenario básico:**
 - ↗ Seleccionar la organización que deseemos consultar de entre todas las existentes.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Asociar Proyecto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Asociar una organización a un proyecto que se esté creando en ese mismo momento.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Al menos una organización insertada.
 - **Postcondiciones:** Asociación entre una organización y un proyecto.
 - **Escenario básico:**
 - ↗ Crear un proyecto asignando su nombre y descripción.
 - ↗ Asociar a dicho proyecto una organización existente en el sistema.
 - ↗ Confirmar asociación.



4.2.3 GESTIONAR MODELO DE PROCESO

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un modelo de proceso. Cada una de estas opciones corresponde con un caso de uso en el diagrama, pero para tener una mejor visión de las partes integrantes de este caso de uso, se va a proceder a desglosar dicho bloque en 2, uno referido al **proceso automatizado** ayudándose de un sistema experto y el otro pertenece al **proceso personalizado** para crear nuevos modelos.

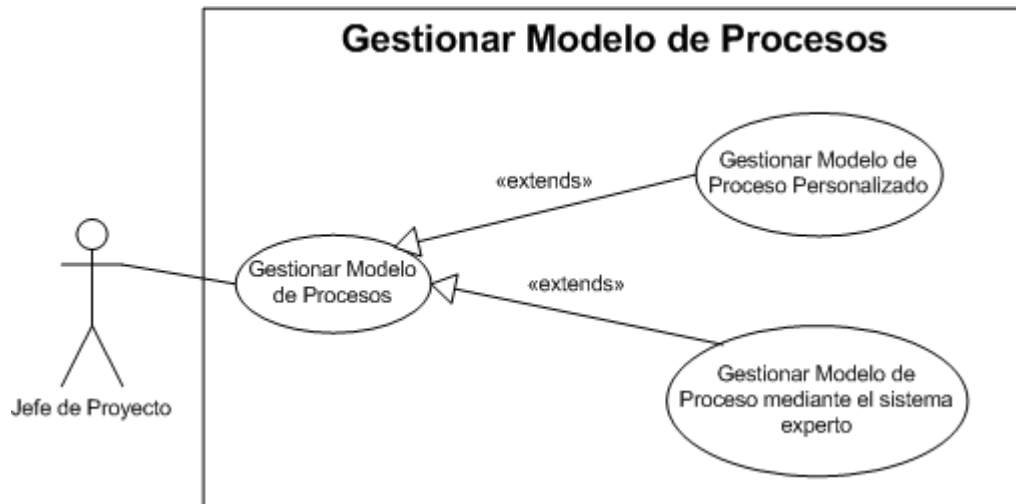


Ilustración 17. Diagrama de casos de uso: "Desglose caso de uso genérico referido a modelos de procesos"



4.2.4 Gestionar Modelo de Proceso Personalizado

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un modelo de proceso personalizado. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

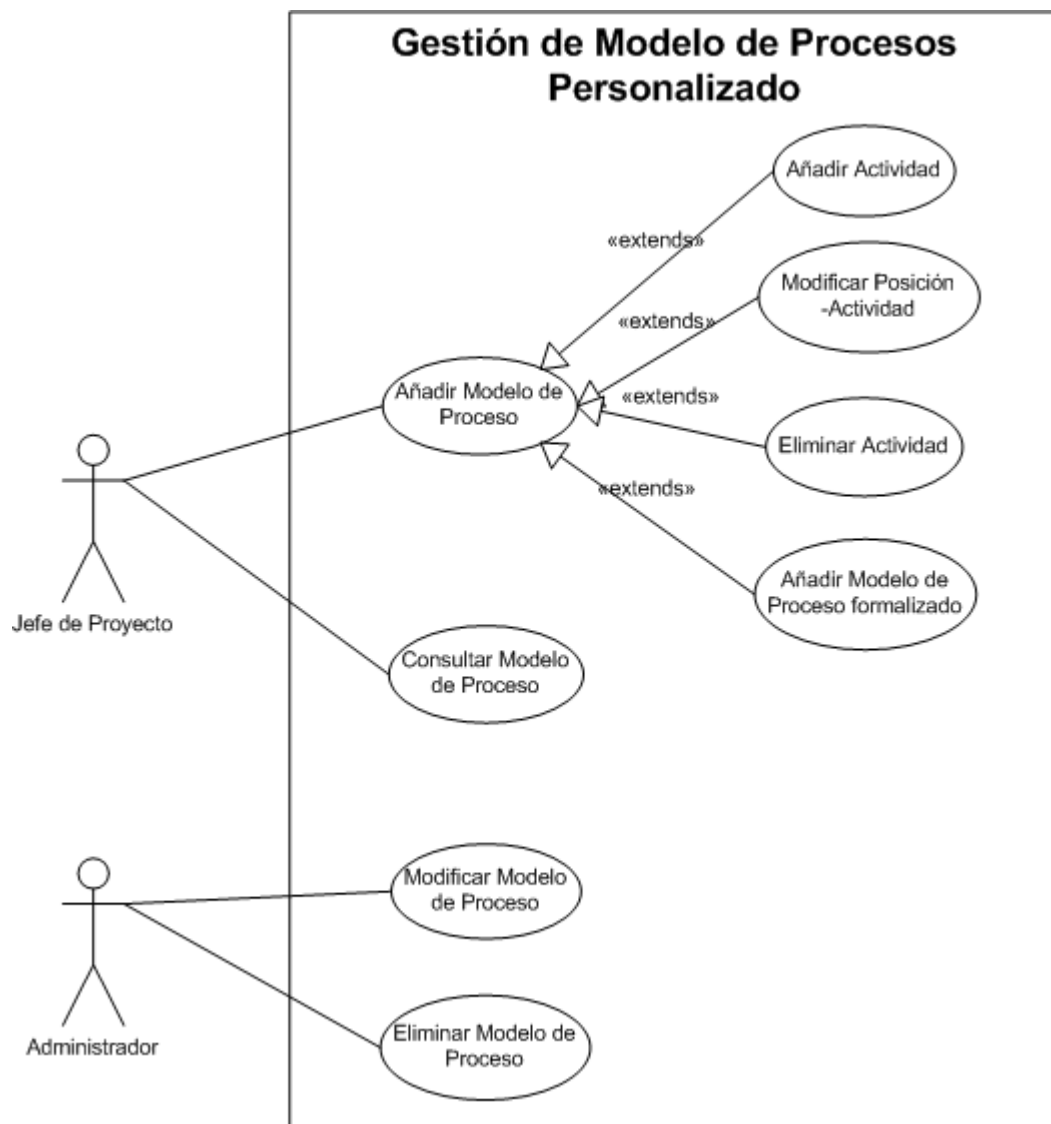


Ilustración 18. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar un modelo de proceso personalizado"



- **Caso de Uso:** Añadir Modelo de Proceso.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un nuevo modelo de procesos a la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos personalizado
 - **Postcondiciones:** Nuevo modelo de proceso insertado en la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Seleccionar el conjunto de actividades que pertenecen a la base de datos del sistema y que formarán parte del modelo de proceso.
 - ↗ Aceptar.

- **Caso de Uso:** Añadir Actividad.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir una nueva actividad al modelo de proceso en cuestión, si fuera necesario se añadirían todas las actividades dependientes de dicha actividad.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos personalizado
 - **Postcondiciones:** Asociar una actividad / conjunto de actividades a un modelo de proceso.
 - **Escenario básico:**
 - ↗ Seleccionar la actividad / conjunto de actividades que pertenecen a la base de datos del sistema.
 - ↗ Asociarla con el modelo de proceso.



- **Caso de Uso:** Modificar posición - actividad.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Modificar posición de una actividad en el modelo de proceso en cuestión.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos personalizado
 - ↗ La actividad debe pertenecer al modelo de proceso en cuestión.
 - **Postcondiciones:** Actividad posicionada en el lugar que el usuario cree conveniente.
 - **Escenario básico:**
 - ↗ Seleccionar la actividad / conjunto de actividades que pertenecen a la base de datos del sistema.
 - ↗ Posicionarla en un nuevo lugar.

- **Caso de Uso:** Eliminar actividad.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Eliminar una actividad del modelo de proceso en cuestión.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos personalizado
 - ↗ La actividad debe pertenecer al modelo de proceso en cuestión.
 - **Postcondiciones:** Actividad eliminada del modelo de proceso en cuestión.
 - **Escenario básico:**
 - ↗ Seleccionar la actividad / conjunto de actividades que pertenecen a la base de datos del sistema.
 - ↗ Eliminar esa actividad / conjunto de actividades.



- **Caso de Uso:** Añadir modelo de proceso formalizado.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un modelo de proceso identificado en la base de datos como formalizado, junto con toda la jerarquía de actividades incluida.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos personalizado
 - ↗ Debe existir al menos un modelo de proceso formalizado.
 - **Postcondiciones:** Modelo de proceso formalizado añadido al modelo de proceso en cuestión.
 - **Escenario básico:**
 - ↗ Seleccionar el modelo de proceso formalizado que pertenece a la base de datos del sistema.
 - ↗ Añadir ese modelo de proceso.

- **Caso de Uso:** Modificar Modelo de Proceso.
 - **Actores:** Administrador.
 - **Objetivo:** Modificar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - ↗ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Modelo de proceso modificado, pudiendo realizar cualquiera de los 3 casos de uso mencionados anteriormente.
 - **Escenario básico:**
 - ↗ Seleccionar el modelo de proceso que se quiere modificar.
 - ↗ Realizar las operaciones necesarias.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Eliminar Modelo de Proceso.
 - **Actores:** Administrador.
 - **Objetivo:** Eliminar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - ↗ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Modelo de proceso eliminado.
 - **Escenario básico:**
 - ↗ Seleccionar el modelo de proceso que se quiere eliminar.

- **Caso de Uso:** Consultar Modelo de Proceso.
 - **Actores:** Administrador / Jefe de Proyecto.
 - **Objetivo:** Consultar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador / Jefe de Proyecto.
 - ↗ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Seleccionar el proyecto que contenga el modelo de proceso que queremos consultar.



4.2.5 Gestionar Modelo de Proceso mediante Sistema Experto

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un modelo de proceso mediante un el sistema experto que tiene integrado el sistema. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

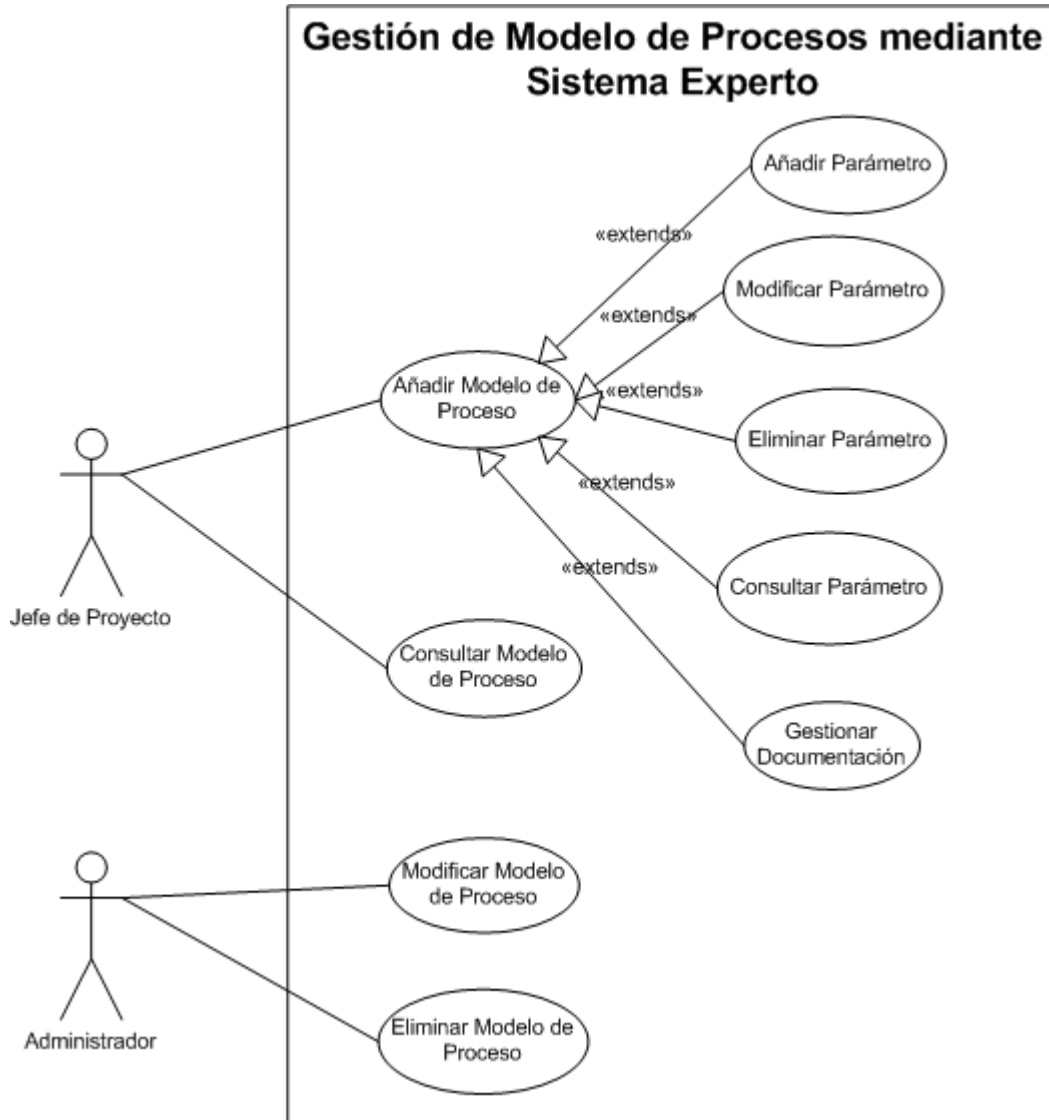


Ilustración 19. Diagrama de casos de uso: "Interacción de usuarios con el sistema para gestionar un modelo de proceso mediante el sistema experto"



- **Caso de Uso:** Añadir Modelo de Proceso.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un nuevo modelo de procesos a la base de datos del sistema.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos formalizados.
 - **Postcondiciones:** Nuevo modelo de proceso insertado en la base de datos del sistema.
 - **Escenario básico:**
 - ↗ Seleccionar el conjunto de parámetros para que el sistema experto pueda recomendar un modelo de proceso que se ajuste a las características identificadas para cada parámetro.
 - ↗ Aceptar.

- **Caso de Uso:** Añadir Parámetro.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un nuevo parámetro a la búsqueda del modelo de proceso formalizado aplicando las reglas heurísticas necesarias, este parámetro puede ser de 2 tipos, organización y proyecto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos formalizado.
 - **Postcondiciones:** Asociar un parámetro a la búsqueda de modelos de proceso formales.
 - **Escenario básico:**
 - ↗ Seleccionar el parámetro correspondiente que queremos introducir en dicha búsqueda.
 - ↗ Seleccionar uno de los posibles valores mostrados por el sistema para un parámetro en concreto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Modificar Parámetro.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Modificar el valor que toma un parámetro.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos formalizado
 - **Postcondiciones:** Valor del parámetro modificado.
 - **Escenario básico:**
 - ↗ Seleccionar el parámetro que se quiere modificar.
 - ↗ Posicionarla en un nuevo lugar.

- **Caso de Uso:** Eliminar parámetro.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Eliminar un parámetro de la búsqueda que realizará el sistema experto posteriormente.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos formalizado.
 - **Postcondiciones:** Parámetro eliminado de la búsqueda realizada por el sistema experto.
 - **Escenario básico:**
 - ↗ Seleccionar el parámetro en cuestión.
 - ↗ Eliminarlo de la búsqueda del sistema experto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Gestionar Documentación.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Observar la documentación referenciada por cada uno de los modelos de proceso estandarizados, además de poder descargar cada uno de esos documentos alojados en el gestor documental.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Elegir la opción de creación de modelo de procesos formalizado.
 - ↗ Seleccionar un modelo de proceso formal.
 - ↗ Documentación asociada existente.
 - **Postcondiciones:** Obtención de toda la documentación asociada a un modelo de proceso.
 - **Escenario básico:**
 - ↗ Seleccionar los parámetros necesarios.
 - ↗ Realizar la búsqueda implementada por el sistema experto.
 - ↗ Seleccionar un modelo de proceso formalizado que pertenece a la base de datos del sistema.
 - ↗ Seleccionar un documento de los asociados a dicho modelo de procesos.
 - ↗ Descargar el documento



- **Caso de Uso:** Modificar Modelo de Proceso.
 - **Actores:** Administrador.
 - **Objetivo:** Modificar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↻ Usuario registrado como Administrador.
 - ↻ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Modelo de proceso modificado.
 - **Escenario básico:**
 - ↻ Seleccionar el modelo de proceso que se quiere modificar.
 - ↻ Realizar las operaciones necesarias.

- **Caso de Uso:** Eliminar Modelo de Proceso.
 - **Actores:** Administrador.
 - **Objetivo:** Eliminar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↻ Usuario registrado como Administrador.
 - ↻ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Modelo de proceso eliminado.
 - **Escenario básico:**
 - ↻ Seleccionar el modelo de proceso que se quiere eliminar.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Consultar Modelo de Proceso.
 - **Actores:** Administrador / Jefe de Proyecto.
 - **Objetivo:** Consultar la jerarquía de actividades que formen parte del modelo de proceso seleccionado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador / Jefe de Proyecto.
 - ↗ El modelo de proceso debe existir en la base de datos del sistema.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Seleccionar el proyecto que contenga el modelo de proceso que queremos consultar.



4.2.6 GESTIONAR PATRÓN DE PRODUCTO

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un patrón de producto. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

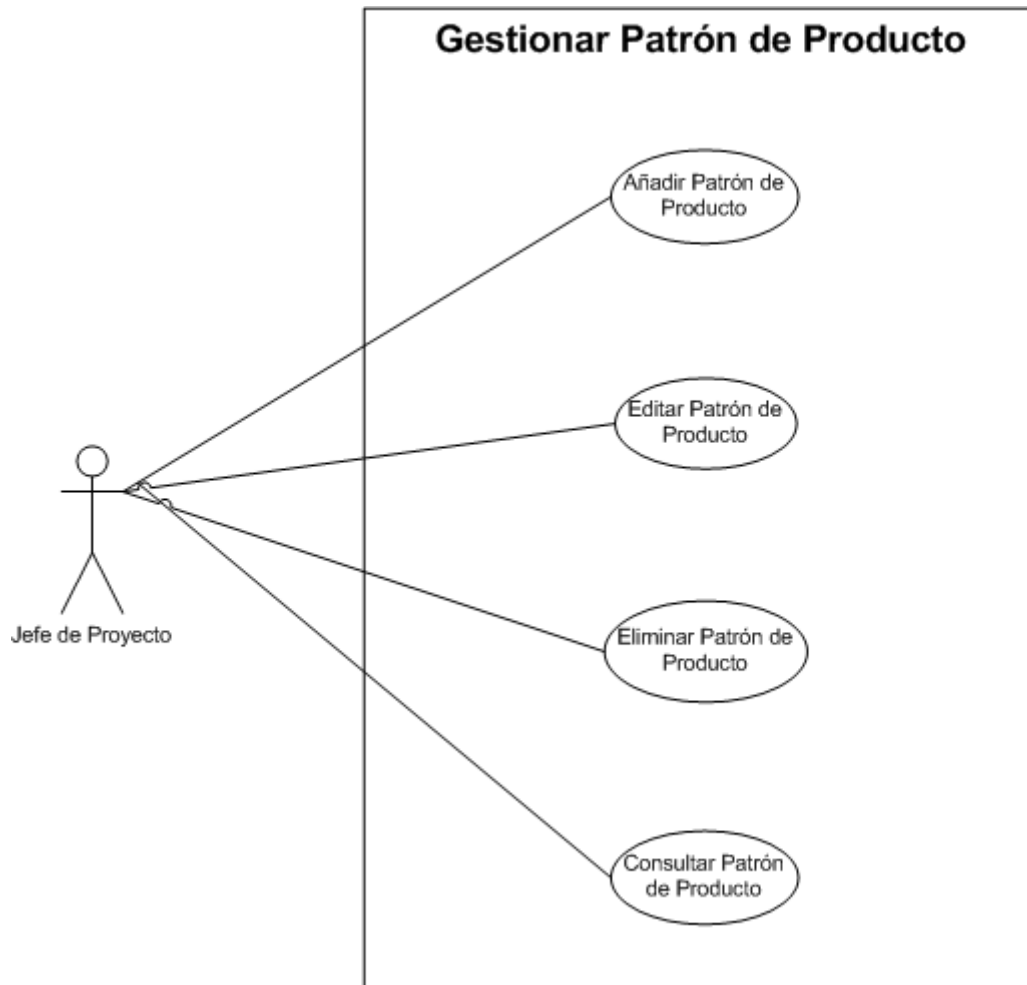


Ilustración 20. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para gestionar patrones de producto"



- **Caso de Uso:** Añadir Patrón de Producto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir un patrón de producto a un proyecto en concreto, realmente se deben asignar dichos patrones a actividades pertenecientes a un proyecto.
 - **Precondiciones:**
 - ↗ Usuario registrado como jefe de proyecto.
 - ↗ Dicho jefe de proyecto debe haber creado un proyecto con por lo menos una actividad que tenga relacionada patrones de producto.
 - **Postcondiciones:** Actividad con nuevo patrón de producto asociado.
 - **Escenario básico:**
 - ↗ Acceder al proyecto que queremos añadirle el patrón de producto.
 - ↗ Seleccionar la actividad que tiene el patrón de producto asociado.
 - ↗ Seleccionar el patrón de producto.
 - ↗ Aceptar.

- **Caso de Uso:** Modificar Patrón de Producto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Modificar la asociación que tenga una actividad con un patrón de producto para asociarlo con otro patrón.
 - **Precondiciones:**
 - ↗ Usuario registrado como jefe de proyecto.
 - ↗ La actividad en cuestión tenga más de un patrón de producto asociado.
 - **Postcondiciones:** Asociación de nuevo patrón de producto efectuado.
 - **Escenario básico:**
 - ↗ Acceder al proyecto que queremos añadirle el patrón de producto.
 - ↗ Seleccionar la actividad que tiene el patrón de producto asociado.
 - ↗ Seleccionar el patrón de producto que se prefiera.
 - ↗ Aceptar.



- **Caso de Uso:** Eliminar Patrón de Producto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Eliminar la asociación que tenga una actividad con un patrón de producto.
 - **Precondiciones:**
 - ↗ Usuario registrado como jefe de proyecto.
 - ↗ La actividad en cuestión tenga un patrón de producto ya asignado.
 - **Postcondiciones:** Asignación de nuevo patrón de producto eliminada.
 - **Escenario básico:**
 - ↗ Acceder al proyecto que queremos añadirle el patrón de producto.
 - ↗ Seleccionar la actividad que tiene el patrón de producto asociado.
 - ↗ Eliminar el patrón de producto asignado a dicha actividad.
 - ↗ Aceptar.

- **Caso de Uso:** Consultar Patrón de Producto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Consultar la asociación que tenga una actividad con un patrón de producto.
 - **Precondiciones:**
 - ↗ Usuario registrado como jefe de proyecto.
 - ↗ La actividad en cuestión tenga un patrón de producto ya asignado.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Acceder al proyecto que queremos añadirle el patrón de producto.
 - ↗ Seleccionar la actividad que tiene el patrón de producto asociado.
 - ↗ Aceptar.



4.2.7 GESTIONAR PLAN DE PROYECTO

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el jefe de proyecto a la hora de gestionar un plan de proyecto. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

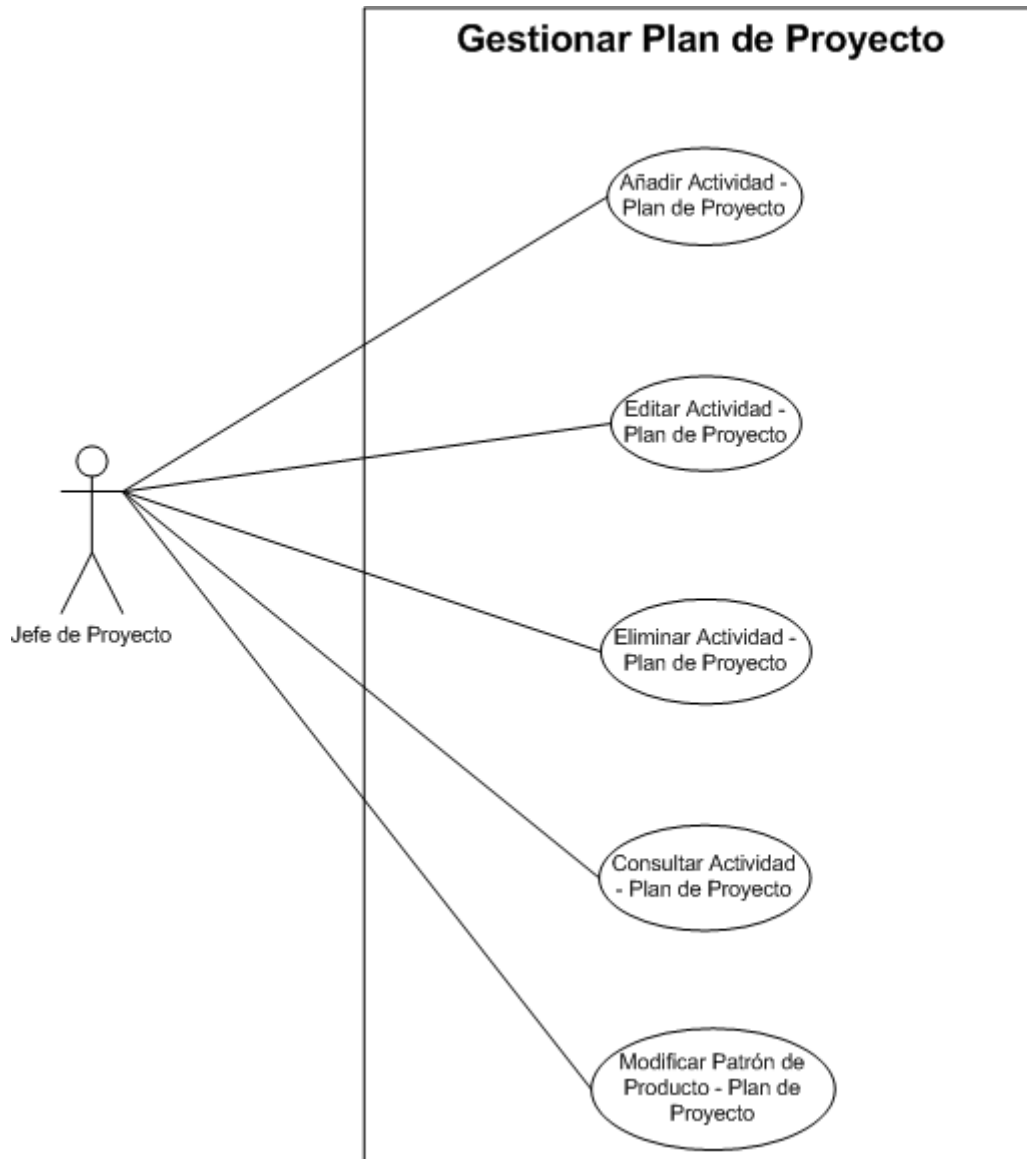


Ilustración 21. Diagrama de casos de uso: "Interacción Jefe de Proyecto/Administrador con el sistema para gestionar un plan de proyecto"



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Añadir Actividad – Plan de Proyecto
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Añadir una nueva actividad al plan de proyecto en cuestión, asociándole un patrón de producto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Paso de creación / actualización de plan de proyecto.
 - **Postcondiciones:** Asociar una actividad a un plan de proyecto en concreto.
 - **Escenario básico:**
 - ↗ Rellenar los campos obligatorios para crear una nueva actividad.
 - ↗ Aceptar

- **Caso de Uso:** Modificar Actividad – Plan de Proyecto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Modificar alguno de los campos pertenecientes a una actividad y que forma parte de un plan de proyecto ya creado.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Paso de creación / actualización de plan de proyecto.
 - ↗ La actividad debe pertenecer al plan de proyecto en cuestión.
 - **Postcondiciones:** Actividad modificada correctamente.
 - **Escenario básico:**
 - ↗ Seleccionar la actividad que forma parte del plan de proyecto.
 - ↗ Modificar las propiedades que crea conveniente el usuario.
 - ↗ Aceptar



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Eliminar actividad – Plan de Proyecto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Eliminar una actividad del plan de proyecto en cuestión.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Paso de creación / actualización de plan de proyecto.
 - ↗ La actividad debe pertenecer al plan de proyecto en cuestión.
 - **Postcondiciones:** Actividad eliminada del plan de proyecto en cuestión.
 - **Escenario básico:**
 - ↗ Seleccionar la actividad que forma parte del plan de proyecto.
 - ↗ Eliminar esa actividad.

- **Caso de Uso:** Consultar Actividad – Plan de Proyecto.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** Consultar la asociación que tenga una actividad con un patrón de producto, también podemos consultar qué actividades forman parte de un plan de proyecto en concreto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Jefe de Proyecto.
 - ↗ Paso de creación / actualización de plan de proyecto.
 - ↗ La actividad debe pertenecer al plan de proyecto en cuestión.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Acceder al plan proyecto que queremos consultar.
 - ↗ Seleccionar la actividad en concreto de dicho plan de proyecto.
 - ↗ Aceptar.



4.2.8 GESTIONAR VALIDACIÓN EN EL SISTEMA

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el usuario cuando tiene que validarse en el sistema para acceder a la aplicación en cuestión. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

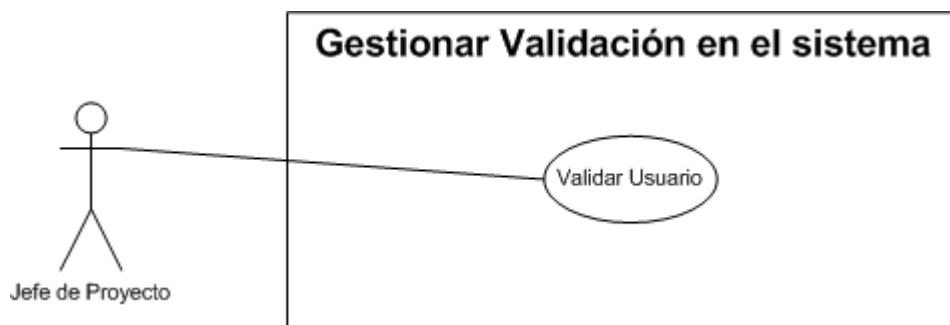


Ilustración 22. Diagrama de casos de uso: "Interacción Jefe de Proyecto con el sistema para validarse en la aplicación."

- **Caso de Uso:** Validar Usuario.
 - **Actores:** Jefe de Proyecto.
 - **Objetivo:** El jefe de proyecto pueda validarse en la aplicación para poder utilizarla y gestionarla correctamente, para poder hacer todas las operaciones necesarias el administrador previamente debe haberle asignado los permisos correspondientes dentro del gestor documental.
 - **Precondiciones:**
 - ↗ El usuario tiene acceso a la aplicación (no quiere decir que tenga permisos).
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Acceder a la aplicación introduciendo la dirección web correspondiente.
 - ↗ Introducir los credenciales correspondientes.
 - ↗ Aceptar.



4.2.9 GESTIONAR DOCUMENTACIÓN

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el administrador a la hora de gestionar la documentación que se almacena en el gestor documental. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

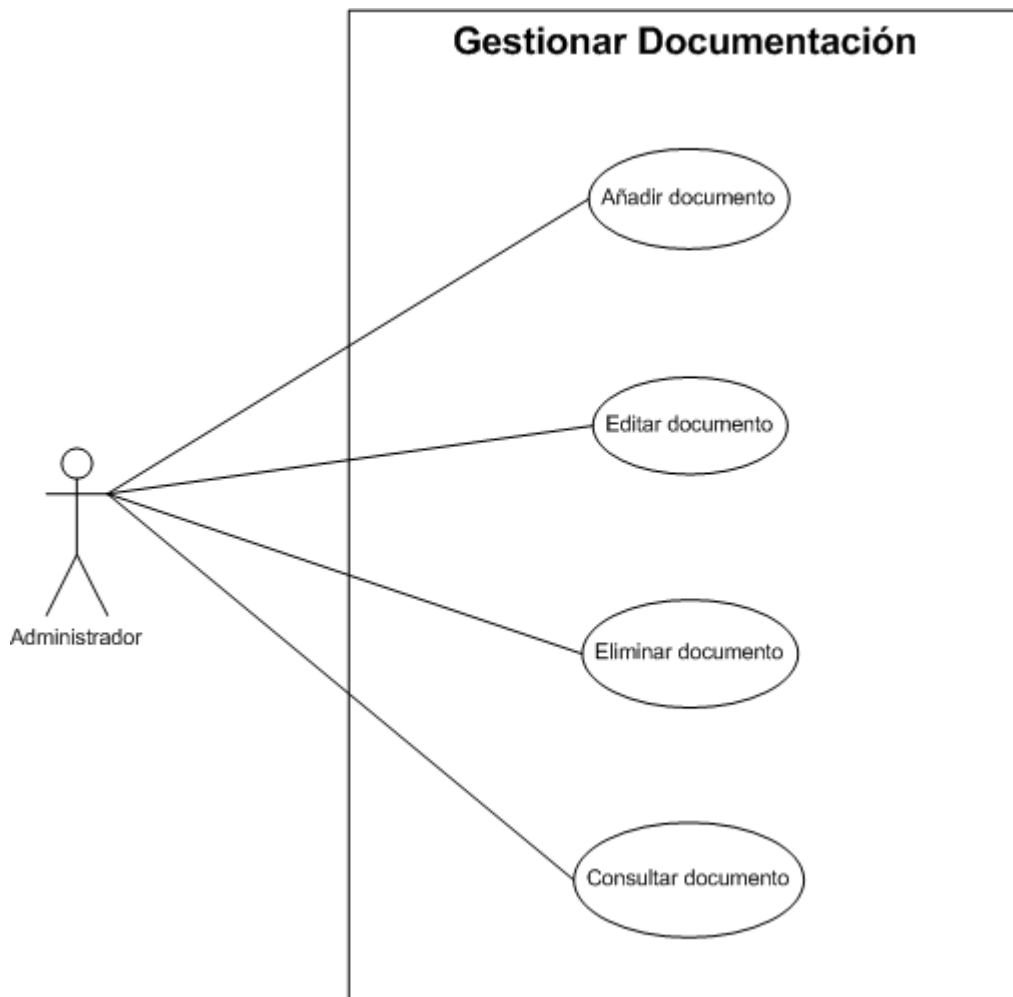


Ilustración 23. Diagrama de casos de uso: "Interacción Administrador con el sistema para gestionar la documentación."



- **Caso de Uso:** Añadir Documento
 - **Actores:** Administrador.
 - **Objetivo:** Añadir un nuevo documento a la biblioteca dedicada para almacenar documentación referida a la aplicación en cuestión.
 - **Precondiciones:**
 - ↻ Usuario registrado como Administrador
 - **Postcondiciones:** Nuevo documento añadido al gestor documental.
 - **Escenario básico:**
 - ↻ Adjuntar el documento correctamente, rellenando cada uno de los campos que forman parte de la biblioteca.
 - ↻ Aceptar

- **Caso de Uso:** Modificar Documento.
 - **Actores:** Administrador.
 - **Objetivo:** Modificar alguno de los documentos que ya están almacenados en el gestor documental.
 - **Precondiciones:**
 - ↻ Usuario registrado como Administrador.
 - ↻ El documento debe pertenecer a la biblioteca de documentación de la aplicación.
 - **Postcondiciones:** Documento modificado correctamente.
 - **Escenario básico:**
 - ↻ Seleccionar el documento que se quiere modificar.
 - ↻ Modificar las propiedades que crea conveniente el usuario o adjuntar un nuevo documento actualizado.
 - ↻ Aceptar



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Eliminar documento.
 - **Actores:** Administrador.
 - **Objetivo:** Eliminar una documento de la biblioteca correspondiente.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - ↗ El documento debe pertenecer a la biblioteca de documentación de la aplicación.
 - **Postcondiciones:** Documento eliminado correctamente.
 - **Escenario básico:**
 - ↗ Seleccionar el documento que se desea eliminar.
 - ↗ Eliminar esa actividad.

- **Caso de Uso:** Consultar Documento.
 - **Actores:** Administrador.
 - **Objetivo:** Consultar la documentación almacenada en el gestor documental.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Acceder a la aplicación, concretamente al sitio web concreto donde esté alojada la documentación.
 - ↗ Consultar el documento que se desee.



4.2.10 GESTIONAR PERMISOS

El escenario de este diagrama de casos de uso, corresponde con las operaciones que puede realizar el administrador a la hora de gestionar los permisos de la aplicación. Cada una de estas opciones corresponde con un caso de uso en el diagrama.

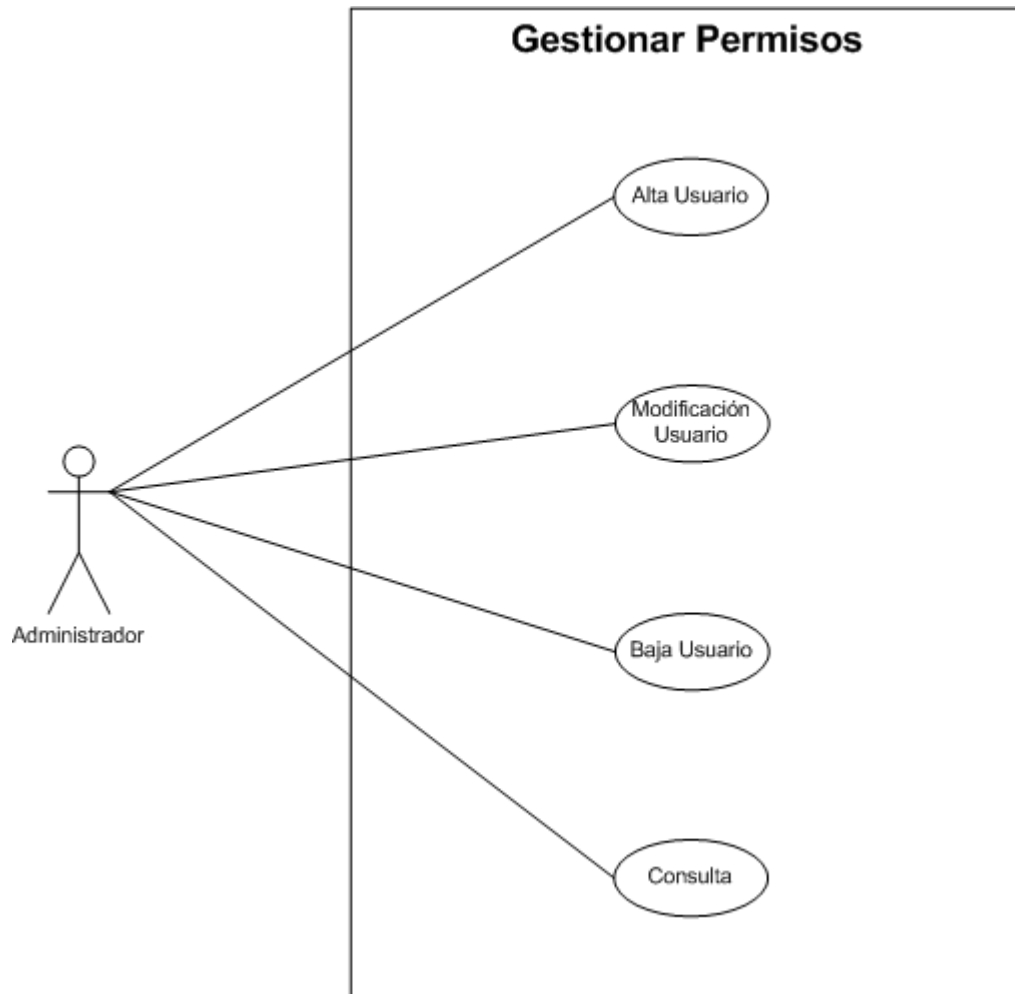


Ilustración 24. Diagrama de casos de uso: "Interacción Administrador con el sistema para gestionar permisos."



- **Caso de Uso:** Alta Usuario
 - **Actores:** Administrador.
 - **Objetivo:** Dar de alta a un nuevo usuario, ya sea un jefe de proyecto o un simple programador.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador
 - **Postcondiciones:** Nuevo usuario dado de alta en el sistema.
 - **Escenario básico:**
 - ↗ Acceder al apartado de seguridad del gestor documental.
 - ↗ Rellenar todos los datos necesarios para dar de alta a un usuario (nombre, login, perfil...)

- **Caso de Uso:** Modificación Usuario.
 - **Actores:** Administrador.
 - **Objetivo:** Modificar los permisos de un usuario en concreto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - **Postcondiciones:** Permisos del usuario modificados correctamente.
 - **Escenario básico:**
 - ↗ Acceder al apartado de seguridad del gestor documental.
 - ↗ Modificar los datos que el administrador crea conveniente.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

- **Caso de Uso:** Baja Usuario.
 - **Actores:** Administrador.
 - **Objetivo:** Dar de baja al usuario que el administrador cree conveniente.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - **Postcondiciones:** Usuario dado de baja correctamente.
 - **Escenario básico:**
 - ↗ Acceder al apartado de seguridad del gestor documental.
 - ↗ Eliminar dicho usuario.

- **Caso de Uso:** Consultar Usuario.
 - **Actores:** Administrador.
 - **Objetivo:** Consultar la información correspondiente a un usuario en concreto.
 - **Precondiciones:**
 - ↗ Usuario registrado como Administrador.
 - **Postcondiciones:** Nada.
 - **Escenario básico:**
 - ↗ Acceder al apartado de seguridad del gestor documental.
 - ↗ Consultar los datos de un usuario en concreto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

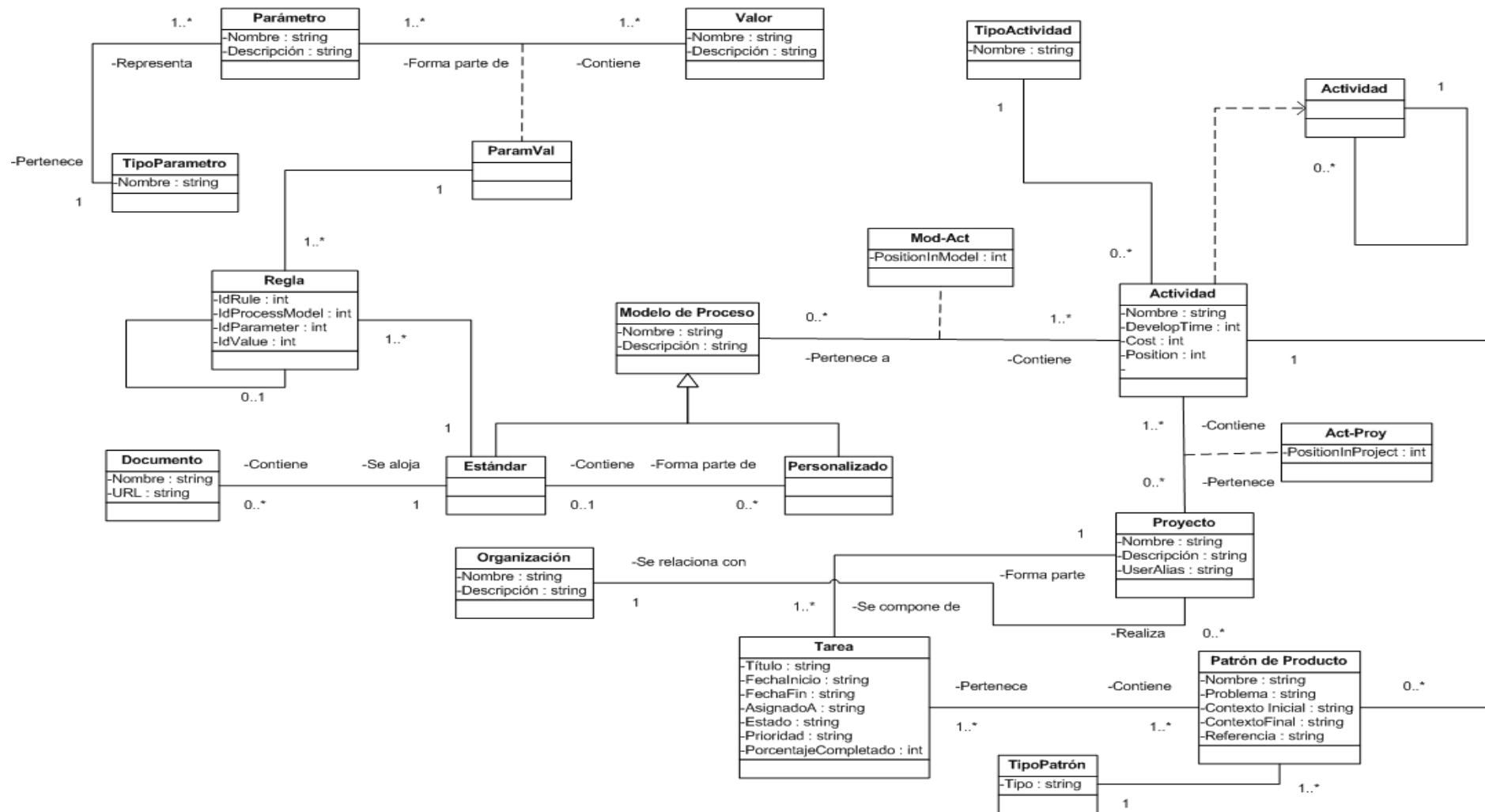
Alumno: David Gómez Sanabria

Capítulo 4: Análisis del Sistema

4.3 DIAGRAMA DE CLASES

A Continuación se muestra el diagrama de clases de la aplicación:

Ilustración 25. Diagrama de clases(Ver página siguiente)





Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

CAPITULO 5. DISEÑO DEL SISTEMA

5.1 CATÁLOGO DE REQUISITOS DE DISEÑO Y CONSTRUCCIÓN

Tomando como referencia el catálogo de requisitos mostrado en el capítulo dedicado al análisis del sistema (4.1), se va a actualizar éste con los requisitos propios del desarrollo del software.

<u>ID</u>	REQD.1	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> El software a desarrollar debe realizarse en capas, diferenciándose la capa de acceso a datos por un lado, la capa de lógica de negocio por otro y por último la capa de presentación.			

<u>ID</u>	REQD.2	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> El diseño permitirá un fácil mantenimiento de código, presentando especial cuidado a hacer el código lo más comprensible posible, unificando el estilo de todos los módulos, empleando comentarios, regiones públicas, privadas, zona exclusiva de propiedades y módulos funcionales independientes.			

<u>ID</u>	REQD.3	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Se deberán emplear las características del gestor de base de datos para aligerar los procesos de la aplicación.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

<u>ID</u>	REQD.4	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> La lógica de la aplicación se desarrollará en C#.NET.			

<u>ID</u>	REQD.5	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> La publicación de la aplicación se realizará mediante ASP.NET con IIS6 mantenimiento las páginas de servidor activas.			

<u>ID</u>	REQD.6	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> La plataforma bajo la cual se montarán todos los componentes, así como el empleo de dicha plataforma como gestor de contenidos y gestor documental.			

<u>ID</u>	REQD.7	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Se seguirán reglas de estilo en todo el código que se implemente.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

<u>ID</u>	REQD.8	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Se aprovechará el estándar de programación orientada a objetos.			

<u>ID</u>	REQD.9	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Las consultas a la base de datos se optimizarán de forma que su ejecución sea lo más rápida posible.			

<u>ID</u>	REQD.10	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Todos los scripts irán encapsulados en ficheros genéricos, donde cada uno de ellos solamente se cargará cuando sea necesario sin exceder la carga de la página.			

<u>ID</u>	REQD.11	<u>TIPO:</u>	DISEÑO
<u>DESCRIPCIÓN</u> Las hojas de estilos (css) seguirán una estructura jerárquica, donde cada fichero representará un módulo genérico y dentro de cada uno existirán apartados totalmente diferenciados.			



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

5.2 DEFINICIÓN DE LA INTERFAZ DE USUARIO

En primer lugar, aclarar que la definición del manual de usuario no se mostrará en ningún caso como un tutorial por lo que en un Anexo se facilitarán todos los pasos necesarios para la instalación y configuración del sistema, además se aportará un recurso adicional donde mediante un video explicativo se realizarán todos los procesos en detalle. De esta manera, cualquier usuario que sea el encargado de mantener la aplicación pueda hacerla con garantías de éxito.

A continuación se mostrará la interfaz de usuario para ver realmente cuál es el diseño de cada una de las pantallas argumentadas anteriormente en los casos de uso extendido:

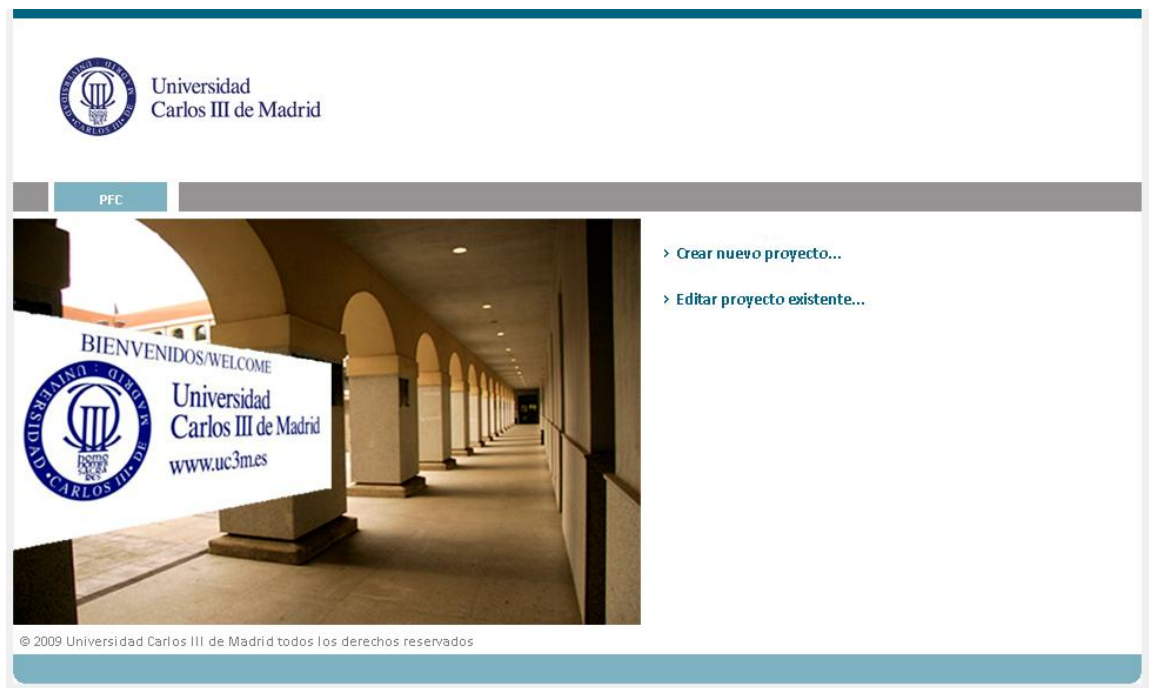


Ilustración 26. Interfaz de la página principal

Como podemos ver en la Ilustración 51 el diseño de la interfaz está muy elaborada, cuidando hasta el último detalle, utilizando una imagen de la propia Universidad Carlos III de Leganés.

Si nos fijamos en los elementos funcionales podemos ver dos hipervínculos, para poder crear o editar proyectos ya existentes. En estos momentos, estamos viendo la plataforma de Microsoft Office SharePoint Server 2007 (aunque no lo parezca) como un sitio Web creado "PFC", este sitio web se empleará para exponer toda la funcionalidad del sistema.



Ilustración 27. Interfaz Crear nuevo proyecto

En cuanto pinchamos sobre el hipervínculo “Crear nuevo proyecto”, nos aparece un formulario para rellenar los datos de un proyecto, en este caso el nombre y la descripción.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

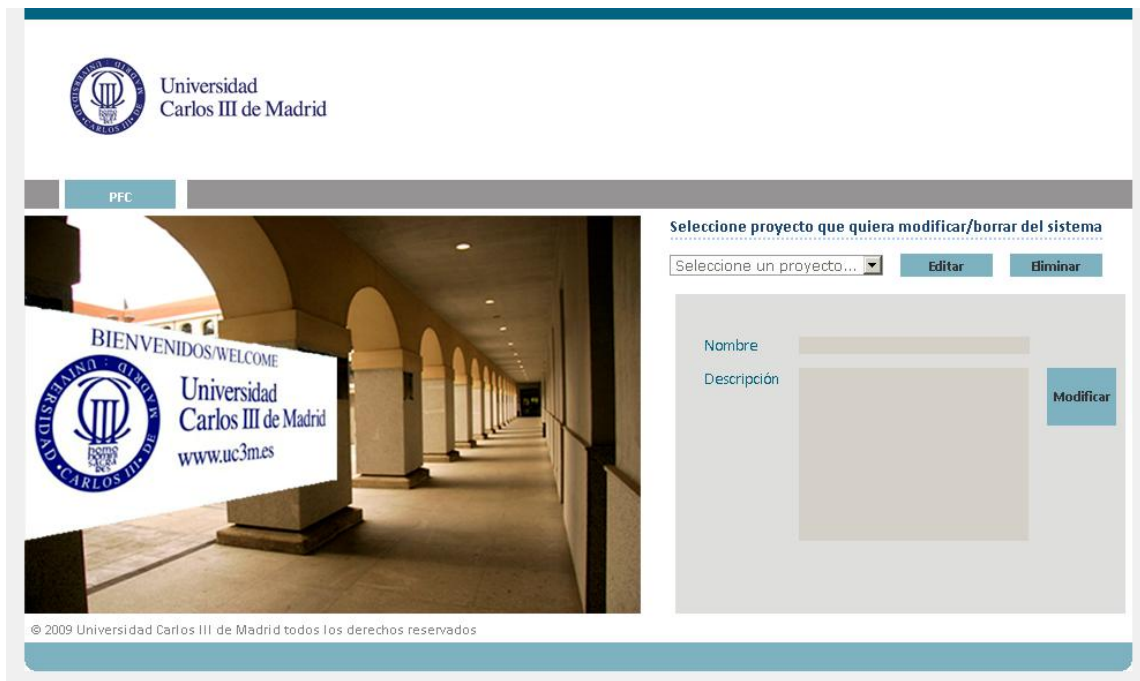


Ilustración 28. Interfaz Editar proyecto existente

Por otro lado, si seleccionamos el link “Editar proyecto existente” la aplicación nos presenta multitud de posibilidades a la hora de poder gestionar proyectos (Modificar parámetros principales, editar todo el proyecto, eliminarlo...).

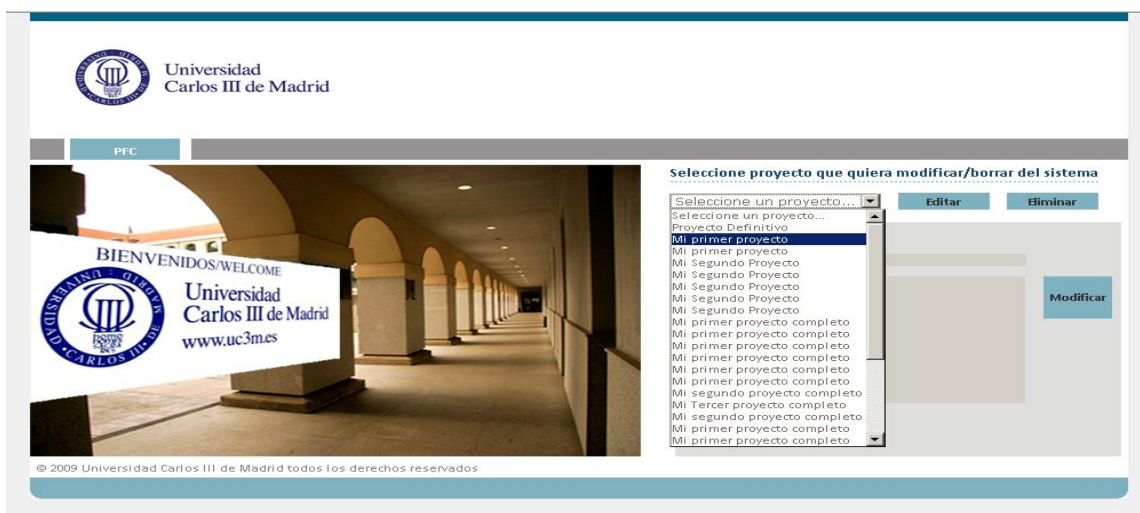


Ilustración 29. Seleccionar Proyecto existente

Aquí se puede ver el listado de proyectos creados por un usuario en concreto (según el usuario que se haya autenticado en ese momento).



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

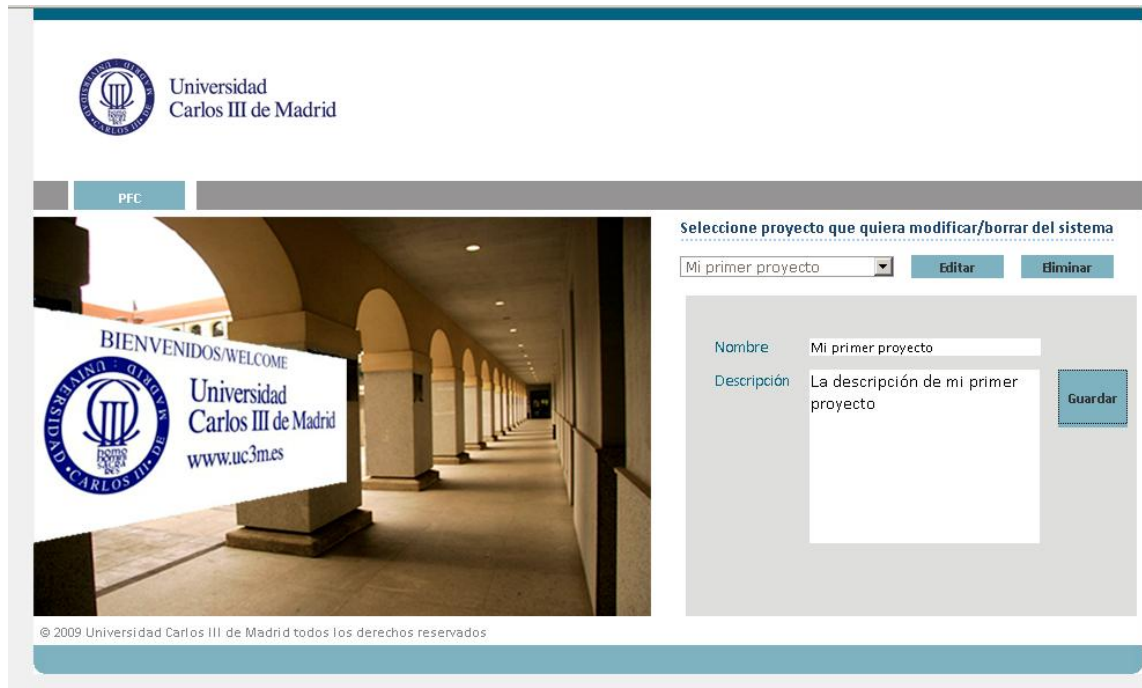


Ilustración 30. Modificar parámetros principales

Si se selecciona un proyecto existente y presionamos el botón “Modificar” que vimos anteriormente, podremos modificar tanto el nombre como la descripción del proyecto seleccionado.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

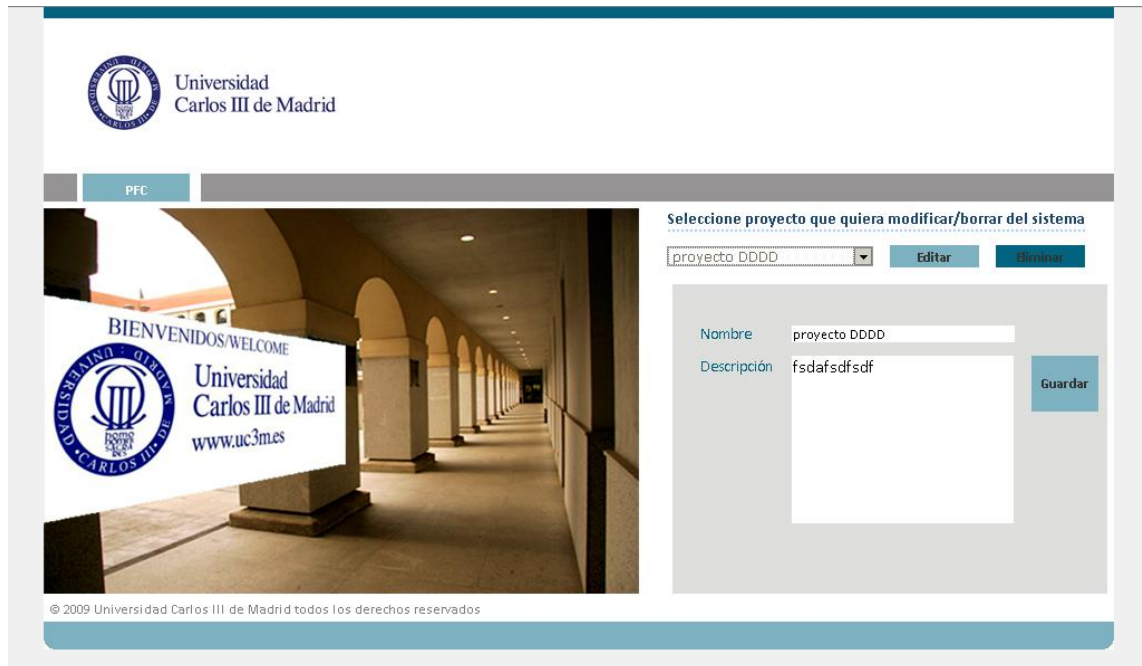


Ilustración 31. Eliminar Proyecto

Quando hemos seleccionado un proyecto tenemos la opción de eliminarlo, teniendo la opción de borrar cualquier referencia que exista en la BBDD.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

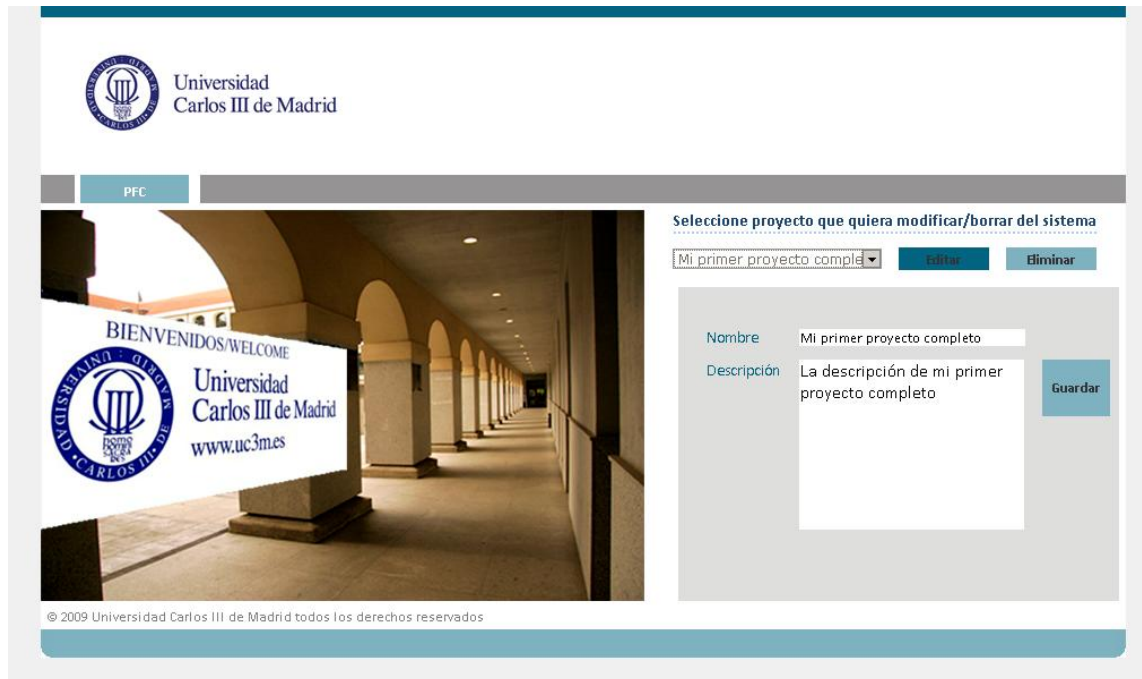



Ilustración 32. Editar Proyecto I



Ilustración 33. Editar Proyecto II

	Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos	Fecha: 12/10/09
	Alumno: David Gómez Sanabria	Capítulo 5: Diseño del Sistema

En las 2 imágenes anteriores podemos ver como cuando seleccionamos un proyecto podemos pulsar el botón editar, para ver a continuación la estructura de actividades del proyecto que queremos modificar.

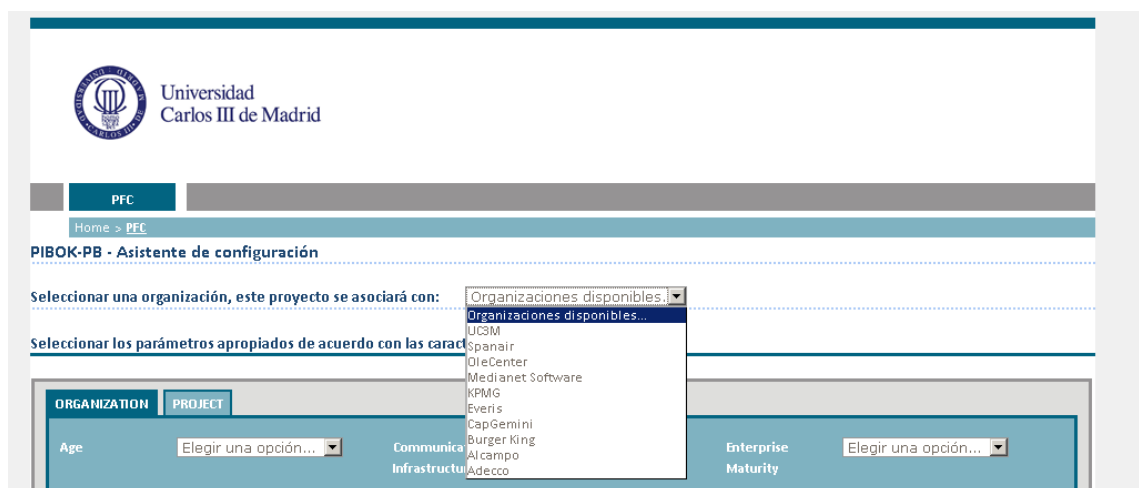


Ilustración 34. Crear Proyecto – Organización I

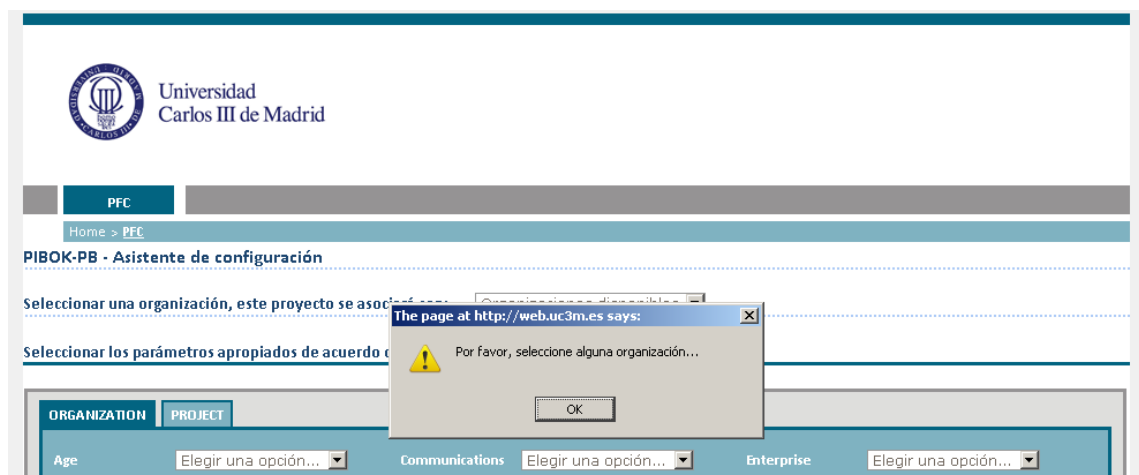


Ilustración 35. Crear Proyecto – Organización II

Una vez hemos creado el proyecto, se debe asociar a una organización determinada (este paso es obligatorio, ya que si no seleccionamos ninguna existe un validador encargado de garantizar el cumplimiento de dicho requisito).



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

Universidad Carlos III de Madrid

PFC

Home > PFC

PIBOK-PB - Asistente de configuración

Seleccionar una organización, este proyecto se asociará con: UC3M

Seleccionar los parámetros apropiados de acuerdo con las características de tu compañía...

ORGANIZATION	PROJECT
Age	Elegir una opción...
Geographic Distribution	Elegir una opción...
Organization Size	Elegir una opción...
Communications Infrastructure	Elegir una opción...
Invoice	Elegir una opción...
Organizational Management Layers	Elegir una opción...
Enterprise Maturity	Elegir una opción...
Number of Employees	Elegir una opción...
Team/Project Based	Elegir una opción...

Ilustración 36. Crear Proyecto – Parámetros I

Universidad Carlos III de Madrid

PFC

Home > PFC

PIBOK-PB - Asistente de configuración

Seleccionar una organización, este proyecto se asociará con: UC3M

Seleccionar los parámetros apropiados de acuerdo con las características de tu compañía...

ORGANIZATION	PROJECT
Age	Elegir una opción... Elegir una opción... First Value Second Value Third Value
Geographic Distribution	Elegir una opción...
Organization Size	Elegir una opción...
Communications Infrastructure	Elegir una opción...
Invoice	Elegir una opción...
Organizational Management Layers	Elegir una opción...
Enterprise Maturity	Elegir una opción...
Number of Employees	Elegir una opción...
Team/Project Based	Elegir una opción...

Ilustración 37. Crear Proyecto – Parámetros II



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

Universidad Carlos III de Madrid

PFC

Home > PFC

PIBOK-PB - Asistente de configuración

Seleccionar una organización, este proyecto se asociará con: UC3M

Seleccionar los parámetros apropiados de acuerdo con las características de tu compañía...

ORGANIZATION PROJECT

Application Experience: Elegir una opción... Execution Time Constraint: Elegir una opción... Language & Tools Experience: Elegir una opción...

Paradigm: Elegir una opción... Personnel Continuity: Elegir una opción... Platform Experience: Elegir una opción...
low
medium
high
very high

Platform Volatility: Elegir una opción... Required Fiability: Elegir una opción... Required Reusability: Elegir una opción...

Ilustración 38. Crear Proyecto – Parámetros III

Universidad Carlos III de Madrid

PFC

Home > PFC

PIBOK-PB - Asistente de configuración

Seleccionar una organización, este proyecto se asociará con: UC3M

Seleccionar los parámetros apropiados de acuerdo con las características de tu compañía...

ORGANIZATION PROJECT


Application Experience: Elegir una opción... Execution Time Constraint: Elegir una opción... Language & Tools Experience: Elegir una opción...

Paradigm: Elegir una opción... Personnel Continuity: Elegir una opción... Platform Experience: Elegir una opción...

Platform Volatility: Elegir una opción... Required Fiability: Elegir una opción... Required Reusability: Elegir una opción...

The page at http://web.uc3m.es says:
Por favor, seleccione algún parámetro...
OK

Ilustración 39. Crear Proyecto – Parámetros IV

	Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos	Fecha: 12/10/09
	Alumno: David Gómez Sanabria	Capítulo 5: Diseño del Sistema

En las 3 imágenes expuestas anteriormente se puede ver claramente como se gestionan los parámetros, necesarios para poder elaborar mediante el sistema experto las reglas heurísticas correspondientes. Además se debe señalar que los parámetros se dividen en 2 tipos (organización y proyecto) y dónde como mínimo se debe elegir un parámetro, ya que no tendría sentido que un sistema experto recomendara un modelo de proceso formal sin parámetros de entrada por los que partir.

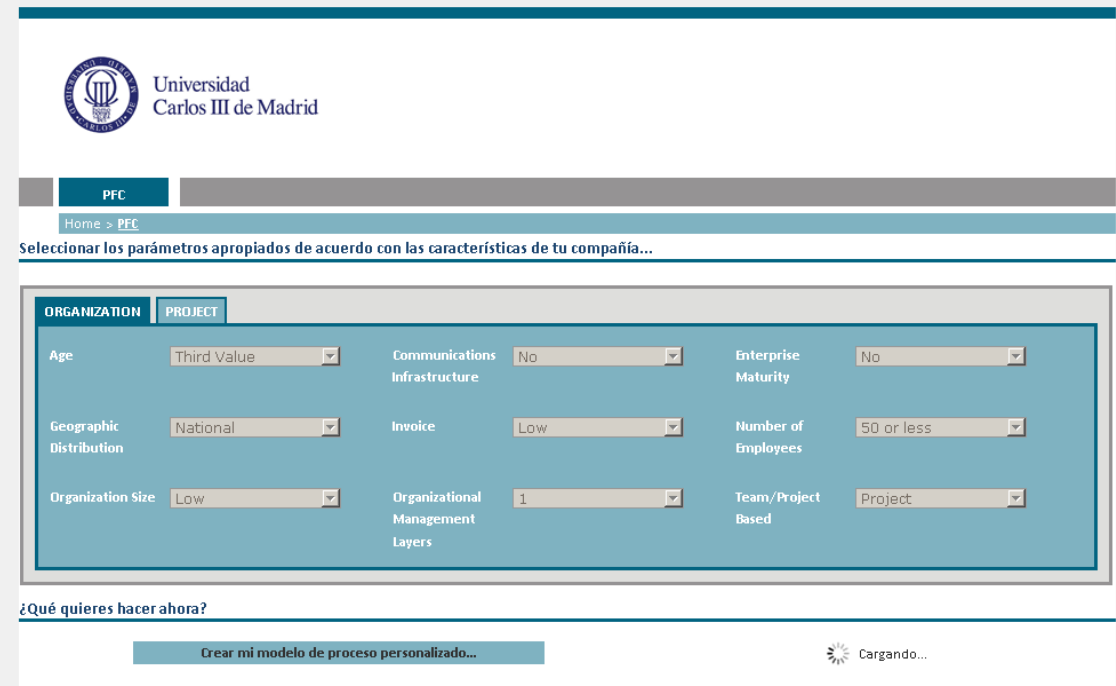


Ilustración 40. Crear Proyecto – Modelo de Proceso I

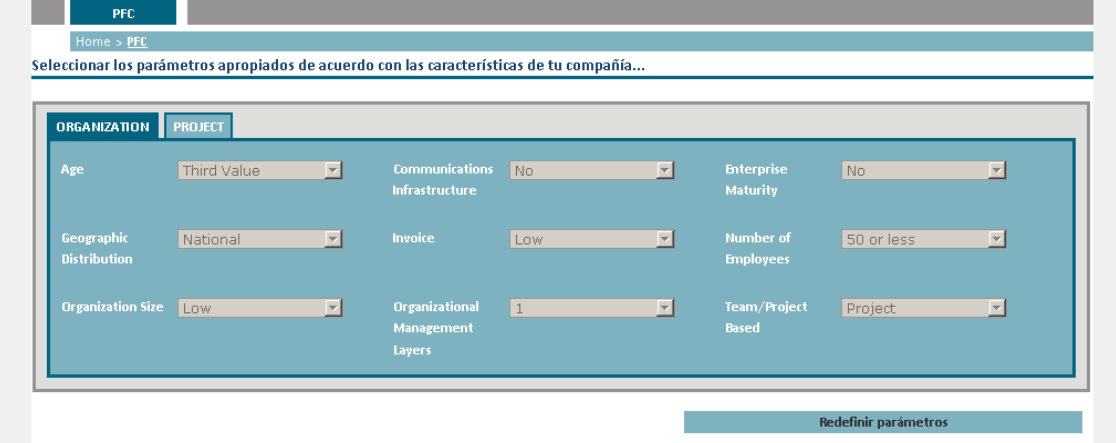


Ilustración 41. Crear Proyecto – Modelo de Proceso II



Como se puede ver, una vez seleccionamos la opción donde el sistema experto nos recomienda que modelos de proceso nos conviene más, el sistema empieza a cargar todos los modelos (I), una vez el sistema experto ha concluido su tarea, el formulario que vimos anteriormente aparece deshabilitado, donde mediante un botón podemos volver a habilitar para efectuar de nuevo el proceso (II).

Modelos recomendados

CRAIG LARMAN - (0%) PERSONALIZADO - (66%) **RUP - (50%)**

Reglas Compuestas - (0/2)		
Application Experience	Low	✓
Geographic Distribution	commarcal	✗

Execution Time Constraint	No	✓
Application Experience	medium	✗

Reglas Simples - (3/4)		
Number of Employees	50 or less	✓
Enterprise Maturity	No	✓
Team/Project Based	Project	✓
Required Fiability	No	✗

- RUP
 - ANALYSIS
 - USE CASES DEFINITION
 - HIJO USECASES DEFINITION
 - EXTENDED USE CASES DEFINITION
 - REQUIREMENTS DEFINITION
 - DESIGN
 - DESIGN 1
 - DEVELOPMENT
 - TESTING
 - TEST 3
 - TESTTTT
 - TEST 1
 - TEST 2
 - DEPLOYMENT
 - DEP 2

Documentación/Referencias... Download

¿Qué quieres hacer ahora?

Crear mi modelo de proceso personalizado... Finalizar

© 2009 Universidad Carlos III de Madrid todos los derechos reservados

Ilustración 42. Crear Proyecto – Modelo de Proceso III



Modelos recomendados

CRAIG LARMAN - (0%) PERSONALIZADO - (66%) RUP - (50%)

Reglas Compuestas - (0/2)		
Application Experience	Low	✓
Geographic Distribution	commarcal	✗

Execution Time Constraint	No	✓
Application Experience	medium	✗
Reglas Simples - (3/4)		
Number of Employees	50 or less	✓
Enterprise Maturity	No	✓
Team/Project Based	Project	✓
Required Fiability	No	✗

¿Qué quieres hacer ahora?

Crear mi modelo de proceso personalizado... Finalizar

© 2009 Universidad Carlos III de Madrid todos los derechos reservados

Ilustración 43. Crear Proyecto – Modelo de Proceso IV

Modelos recomendados

CRAIG LARMAN - (0%) PERSONALIZADO - (66%) RUP - (50%)

Reglas Compuestas - (0/2)		
Application Experience	Low	✓
Geographic Distribution	commarcal	✗
Execution Time Constraint	No	✓
Application Experience	medium	✗
Reglas Simples - (3/4)		
Number of Employees	50 or less	✓
Enterprise Maturity	No	✓
Team/Project Based	Project	✓
Required Fiability	No	✗

¿Qué quieres hacer ahora?

Crear mi modelo de proceso personalizado... Finalizar

© 2009 Universidad Carlos III de Madrid todos los derechos reservados

Ilustración 44. Crear Proyecto – Modelo de Proceso V



Por otro lado, en cuanto al componente que nos muestra los modelos de proceso formales alojados en nuestra BBDD, podemos ver en primer lugar (III) el porcentaje de las reglas que son acordes con lo que buscaba el usuario. Se pueden ver claramente la división de reglas simples y compuestas que lo constituyen, marcando a su vez cuáles se cumplen y cuáles no.

Si estudiamos el formulario al detalle (IV), se muestra todo el árbol de actividades creado para un modelo de proceso asociado, donde podemos interactuar él desplegando los nodos hasta llegar a los nodos finales.

Por último, podemos estudiar en qué se basa cada uno de los modelos de procesos formales gracias a la documentación asociada (V), ésta se representa en un listado donde mediante un botón podremos descargar todos estos documentos sin ningún tipo de problema.

Modelos recomendados

CRAIG LARMAN - (0%) PERSONALIZADO - (66%) RUP - (50%)

Reglas Compuestas - (0/2)		
Application Experience	Low	✓
Geographic Distribution	commarcal	✗

Execution Time Constraint	No	✓
Application Experience	medium	✗
Reglas Simples - (3/4)		
Number of Employees	50 or less	✓
Enterprise Maturity	No	✓
Team/Project Based	Project	✓
Required Fiability	No	✗

¿Qué quieres hacer ahora?

© 2009 Universidad Carlos III de Madrid todos los derechos reservados

Ilustración 45. Crear Proyecto – Modelo de Proceso Personalizado I

Como se puede ver en la imagen anterior, en cualquier momento si decidimos que nuestro modelo de proceso no se adapta a nuestras necesidades, tendremos en nuestra mano la opción de utilizar el otro módulo funcional relacionado con modelos de proceso, “Creación de un modelo de proceso personalizado”.

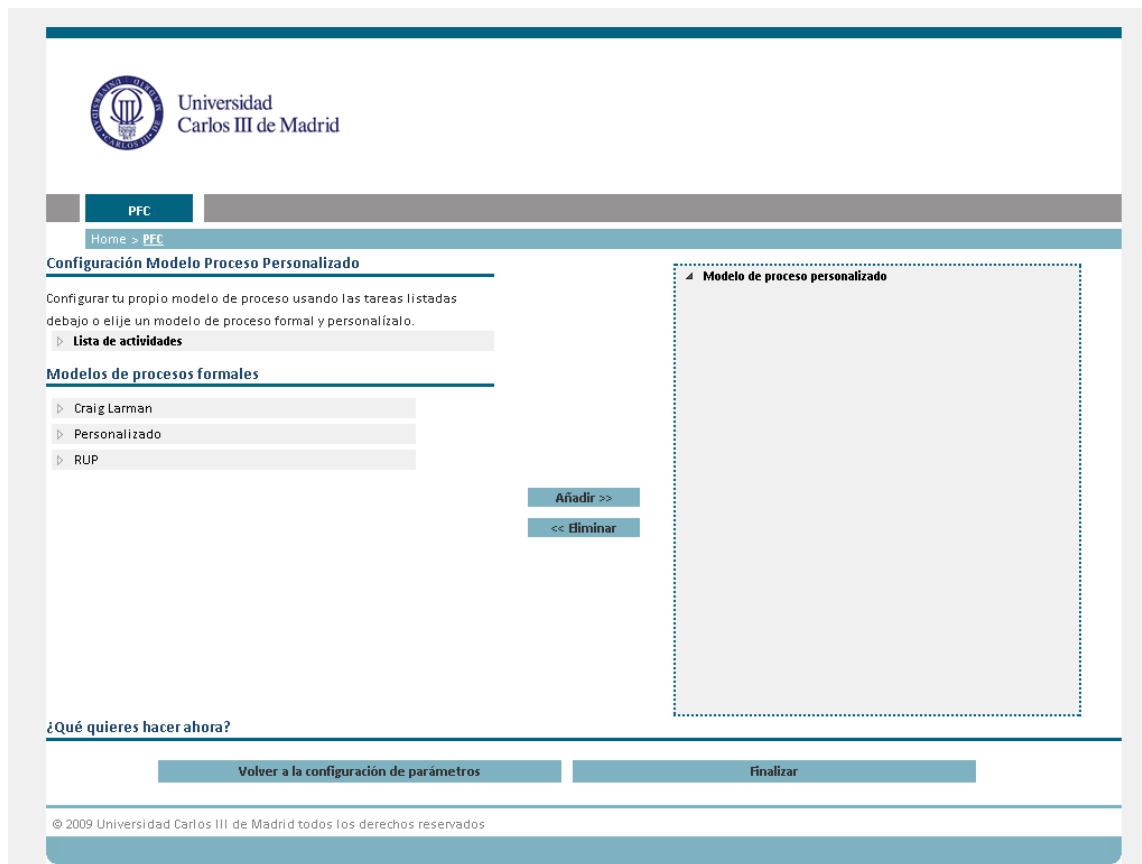


Ilustración 46. Crear Proyecto – Modelo de Proceso Personalizado II

A partir de este momento, se puede ver el módulo funcional de personalización de modelos de procesos. Este módulo se divide en los siguientes apartados:

- En la parte superior izquierda, se listarán todas las actividades que están alojadas en la BBDD, estarán desglosadas por niveles donde se podrán localizar fácilmente a partir de tipos de actividad.
- Debajo del listado de actividades genérico, se muestra cada uno de los modelos de proceso formales que de una forma jerárquica presenta sus actividades, de esta forma el usuario podrá componer su modelo a partir de uno ya creado.
- En la parte central, vemos las acciones que pueden desempeñarse en dicho módulo funcional, por un lado se pueden añadir actividades y por otro eliminarlas, siempre nos guiaremos por la actividad o grupo de actividades que estén marcados en ese momento.
- Por último, vemos a la derecha el árbol de actividades resultante.



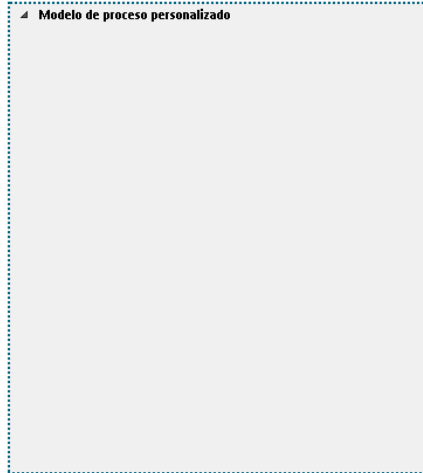
Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalízalo.

Lista de actividades

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
- DEVELOPMENT
- TESTING
- DEPLOYMENT

Añadir >>



Modelos de procesos formales

- ▷ Craig Larman
- ▷ Personalizado
- ▷ RUP

Ilustración 47. Crear Proyecto – Modelo de Proceso Personalizado III

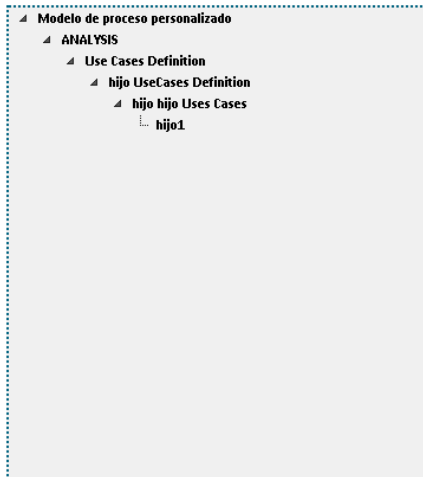
Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalízalo.

Lista de actividades

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
- DEVELOPMENT
- TESTING
- DEPLOYMENT

Añadir >>



Modelos de procesos formales

- ▷ Craig Larman
- ▷ Personalizado
- ▷ RUP

Ilustración 48. Crear Proyecto – Modelo de Proceso Personalizado IV



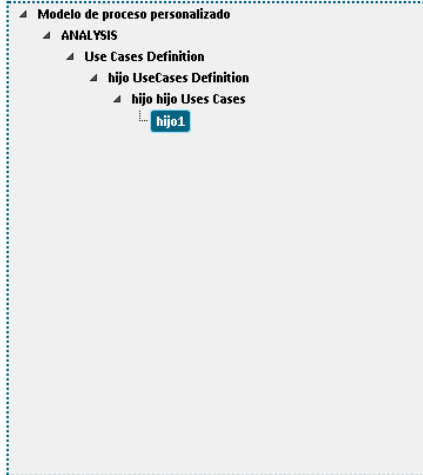
Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalizado.

Lista de actividades

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hajo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
- DEVELOPMENT
- TESTING
- DEPLOYMENT

<< Eliminar



Modelos de procesos formales

- ▷ Craig Larman
- ▷ Personalizado
- ▷ RUP

Ilustración 49. Crear Proyecto – Modelo de Proceso Personalizado V

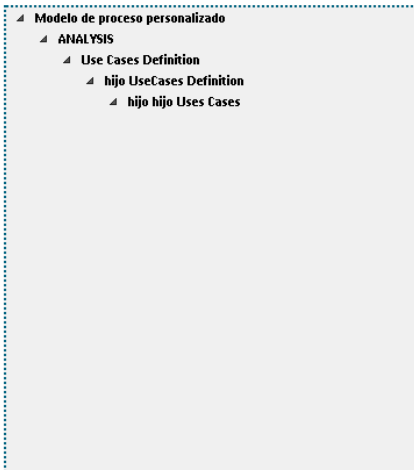
Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalizado.

Lista de actividades

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hajo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
- DEVELOPMENT
- TESTING
- DEPLOYMENT

<< Eliminar



Modelos de procesos formales

- ▷ Craig Larman
- ▷ Personalizado
- ▷ RUP

Ilustración 50. Crear Proyecto – Modelo de Proceso Personalizado VI



Universidad Carlos III de Madrid

PFC

Home > PFC

Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalízalo.

Lista de actividades

- ANALYSIS
- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
- TESTING
- DEPLOYMENT

<< Eliminar

Modelos de procesos formales

- Craig Larman
- Personalizado
- RUP

Modelo de proceso personalizado

- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3

¿Qué quieres hacer ahora?

Volver a la configuración de parámetros Finalizar

© 2009 Universidad Carlos III de Madrid todos los derechos reservados

Ilustración 51. Crear Proyecto – Modelo de Proceso Personalizado VII

Configuración Modelo Proceso Personalizado

Configurar tu propio modelo de proceso usando las tareas listadas debajo o elije un modelo de proceso formal y personalízalo.

Lista de actividades

- ANALYSIS
- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
- TESTING
- DEPLOYMENT

<< Eliminar

Modelos de procesos formales

- Craig Larman
- Personalizado
- RUP

Modelo de proceso personalizado

- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
 - dev 3
 - dev 1
 - dev 2

Ilustración 52. Crear Proyecto – Modelo de Proceso Personalizado VIII



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

Como hemos podido ver, existen multitud de posibilidades a la hora de añadir y eliminar actividades de un modelo de proceso, en las 6 imágenes expuestas anteriormente vemos el resultado de añadir un grupo de actividades y eliminar una actividad, todo esto de una forma fácil y eficaz. Además se pueden reubicar dichas actividades utilizando los botones de desplazamiento de tareas.

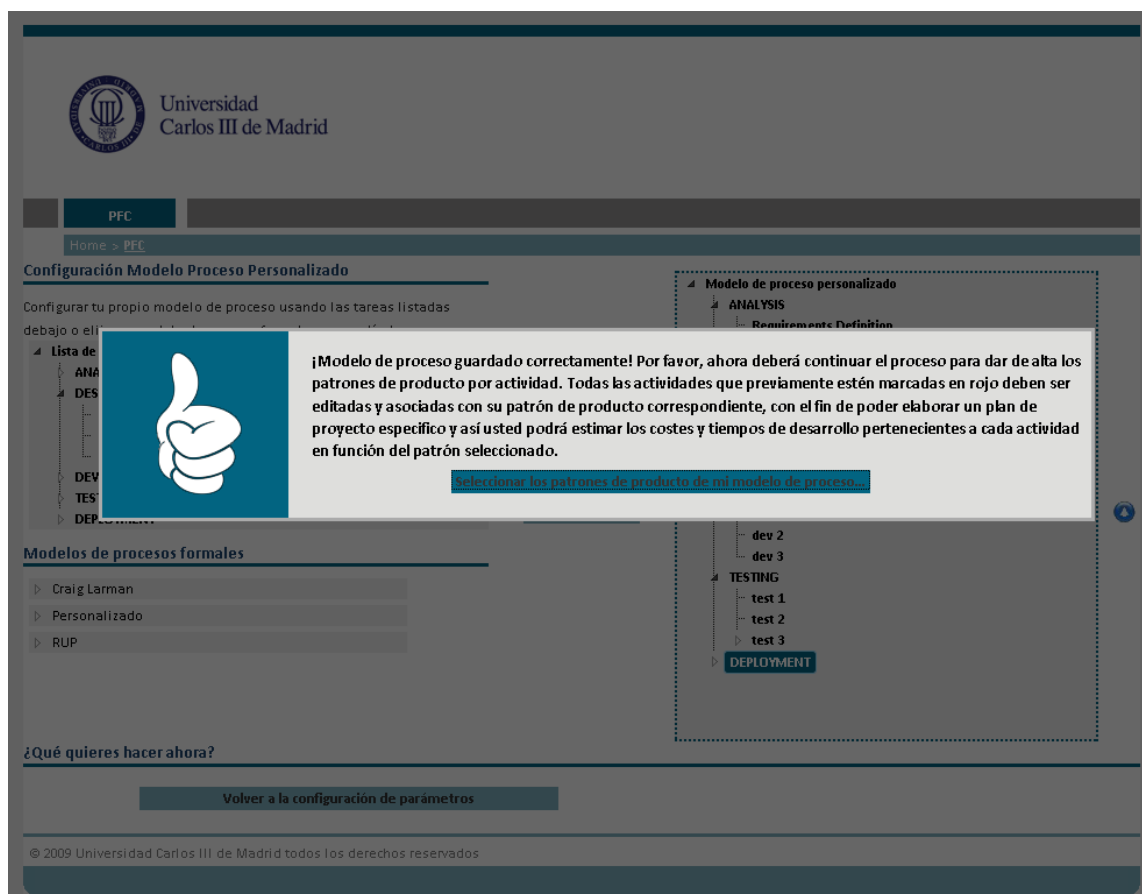


Ilustración 53. Crear Proyecto – Modelo de Proceso Personalizado IX

Como podemos ver en esta imagen, una vez decidimos que la creación de nuestro modelo de proceso está finalizada, pinchamos en el botón “Finalizar”. En este momento, se guardan todos los datos asociados a dicho modelo de procesos (organización, proyecto, actividades, documentación...) y posteriormente vemos la pantalla que precisamente vemos en la imagen.



PFC

Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

- Modelo de proceso de la compañía
 - ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
 - DESIGN
 - design 1
 - design 2
 - design 3
 - DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3
 - TESTING
 - test 1
 - test 2
 - test 3
 - testttt
 - hijo testtttt test
 - DEPLOYMENT
 - dep 1
 - dep 2
 - dep 3

El próximo paso será desarrollar un plan de proyecto con el coste y el tiempo de desarrollo de cada una de las actividades

Siguiente

Ilustración 54. Crear Proyecto – Patrones de Producto I

En este momento, se pueden empezar a gestionar los patrones de producto. Las actividades que están marcadas en rojo son aquellas que tienen asignadas patrones de producto asociados, por tanto hasta que no se asigna un patrón de producto por actividad no podremos continuar con la creación del proyecto.

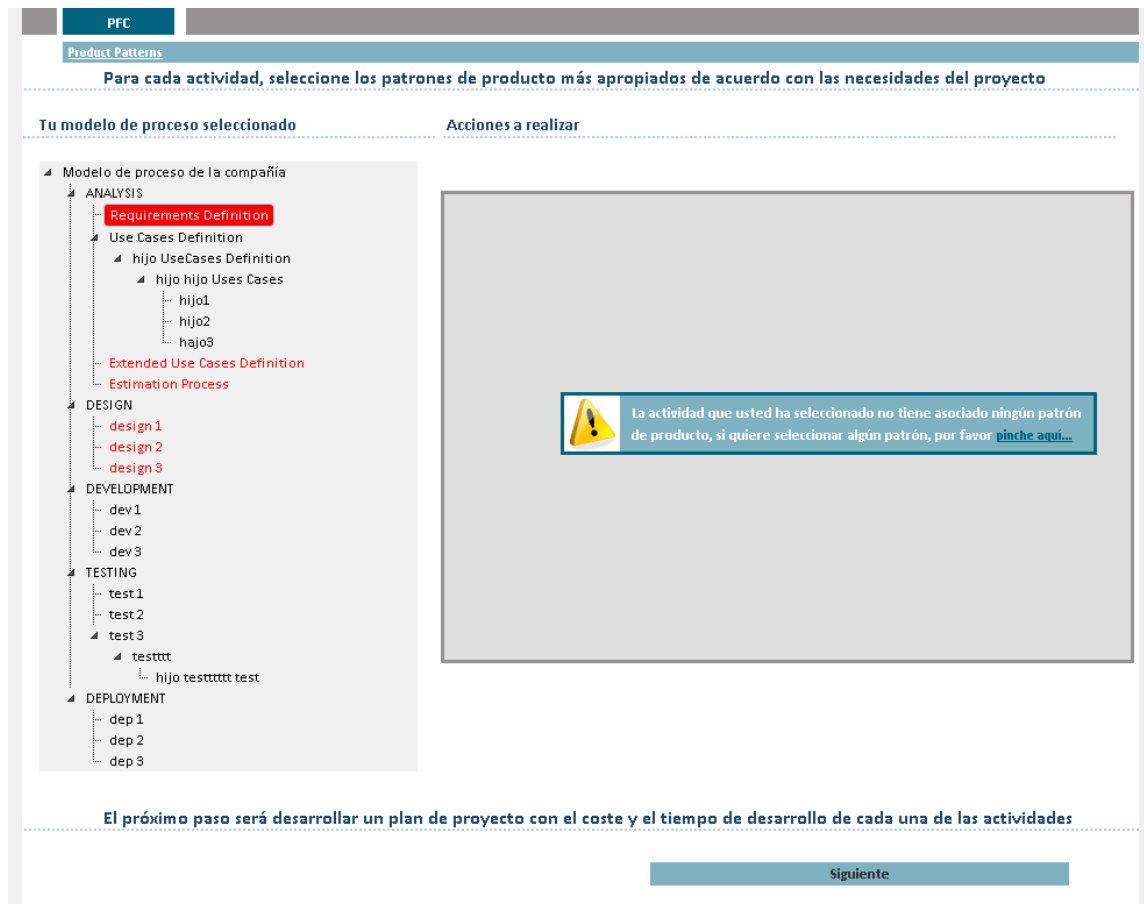


Ilustración 55. Crear Proyecto – Patrones de Producto II

Una vez se selecciona una actividad, el sistema detecta si ya se seleccionó anteriormente un patrón de producto, si no es así veremos la pantalla expuesta en la imagen.



Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

Modelo de proceso de la compañía

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3
- TESTING
 - test 1
 - test 2
 - test 3
 - testttt
 - hijo testtttt test
- DEPLOYMENT
 - dep 1
 - dep 2
 - dep 3

Guardar

Nombre: Requirements Specification Product Pattern

Problema: This document serves as contract with the client. It represents the functionality system.

Contexto Inicial: Realize the Requirements Specification.

Contexto Final: Technology apprehension of potential requirements in a new system or updating software.

Más Información: http://productpatterns.sei.inf.uc3m.es/Product_Patterns/Analysis_Patterns/Requirements_Specification

Patrón Seleccionado:

Nombre: High-Level Use Case Diagram Product Pattern

Problema: Represent the interaction between the system and the functionality in textual description.

Contexto Inicial: There is needed the functionality system specification.

Contexto Final: -The Use Case Diagram will be obtained in High-level. - The Use

Ilustración 56. Crear Proyecto – Patrones de Producto III

A continuación, se puede ver un listado con cada uno de los patrones de producto asociados, dichos patrones tienen una serie de campos asociados (Nombre, problema, contexto inicial...). Por último, existe un campo donde el usuario puede comprobar cuál es el patrón de producto que quiere de todo el listado.



Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

Modelo de proceso de la compañía

- ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
- DESIGN
 - design 1
 - design 2
 - design 3
- DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3
- TESTING
 - test 1
 - test 2
 - test 3
 - testttt
 - hijo testttttt test
- DEPLOYMENT
 - dep 1
 - dep 2
 - dep 3

Guardar

Nombre: Requirements Specification Product Pattern

Problema: This document serves as contract with the client. It represents the functionality system.

Contexto Inicial: Realize the Requirements Specification.

Contexto Final: Technology apprehension of potential requirements in a new system or updating software.

Más Información: http://productpatterns.sel.inf.uc3m.es/Product_Patterns/Analysis_Patterns/Requirements_Specification

Patrón Seleccionado:

Nombre: High-Level Use Case Diagram Product Pattern

Problema: Represent the interaction between the system and the functionality in textual description.

Contexto Inicial: There is needed the functionality system specification.

Contexto Final: -The Use Case Diagram will be obtained in High-level - The Use

Ilustración 57. Crear Proyecto – Patrones de Producto IV



PFC

Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

EDITAR

Nombre:	Class Diagram Product Pattern
Problema:	The system static description is needed, internal properties and relations between them.
Contexto Inicial:	System Functional Description.
Contexto Final:	Classes Diagram concepts. They have to satisfy the application requirements as those who form a part of the application implementation.
Más Información:	http://productpatterns.sel.inf.uc3m.es/Product_Patterns/Design_Patterns/Class_Diagram

El próximo paso será desarrollar un plan de proyecto con el coste y el tiempo de desarrollo de cada una de las actividades

Siguiente

The screenshot shows a web application interface for 'Product Patterns'. On the left, there is a tree view of a 'Modelo de proceso de la compañía' (Company Process Model) with categories like ANALYSIS, DESIGN, DEVELOPMENT, TESTING, and DEPLOYMENT. Under ANALYSIS, 'design 3' is selected. On the right, a detailed view for 'Class Diagram Product Pattern' is shown, including fields for 'Nombre', 'Problema', 'Contexto Inicial', 'Contexto Final', and 'Más Información'. A 'Siguiente' button is at the bottom right.

Ilustración 58. Crear Proyecto – Patrones de Producto V

En las 2 imágenes vistas anteriormente, se ve cual es el estado de la aplicación cuando hemos asociado un patrón de producto a una actividad.



PFC

Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

EDITAR

- Modelo de proceso de la compañía
 - ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hijo3
 - Extended Use Cases Definition
 - Estimation Process
 - DESIGN
 - design 1
 - design 2
 - design 3
 - DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3
 - TESTING
 - test 1
 - test 2
 - test 3
 - testttt
 - hijo testtttt test
 - DEPLOYMENT
 - dep 1
 - dep 2
 - dep 3

Nombre:

Problema:

Contexto Inicial:

Contexto Final:

Más Información:

El próximo paso será desarrollar un plan de proyecto con el coste y el tiempo de desarrollo de cada una de las actividades

Siguiente

Ilustración 59. Crear Proyecto – Patrones de Producto VI

Ya se ha finalizado el proceso de asociación de patrones de producto, por tanto podemos pasar a desarrollar el plan de proyecto, pero antes me gustaría mostraros la validación que existe en cuanto a dicha asociación como se puede ver en la siguiente imagen:



PFC

Product Patterns

Para cada actividad, seleccione los patrones de producto más apropiados de acuerdo con las necesidades del proyecto

Tu modelo de proceso seleccionado Acciones a realizar

- Modelo de proceso de la compañía
 - ANALYSIS
 - Requirements Definition
 - Use Cases Definition
 - hijo UseCases Definition
 - hijo hijo Uses Cases
 - hijo1
 - hijo2
 - hajo3
 - Extended Use Cases Def
 - Estimation Process
 - DESIGN
 - design 1
 - design 2
 - design 3
 - DEVELOPMENT
 - dev 1
 - dev 2
 - dev 3
 - TESTING
 - test 1
 - test 2
 - test 3
 - testttt
 - hijo testtttt test
 - DEPLOYMENT
 - dep 1
 - dep 2
 - dep 3

The page at http://web.uc3m.es says:
Por favor, seleccione un patrón de producto por actividad (completar las actividades marcadas en rojo)...

OK

El próximo paso será desarrollar un plan de proyecto con el coste y el tiempo de desarrollo de cada una de las actividades

Siguiente

Ilustración 60. Crear Proyecto – Patrones de Producto VII



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

Título	09/11/2009	16/11/2009	23/11/2009	30/11/2009	07/12/2009	14/12/2009	21/12/2009	28/12/2009	04/01/2010
Requirements Definition	█								
Use Cases Definition	█								
Extended Use Cases Definition	█								
design 1		█							
design 2		█							
design 3		█							
dev 1			█						
dev 2			█						

Título	Asignado a	Estado de la tarea	Prioridad	Fecha de vencimiento	% completado
Requirements Definition		No iniciada	(2) Normal		
Use Cases Definition		No iniciada	(2) Normal		
Extended Use Cases Definition		No iniciada	(2) Normal		
design 1		No iniciada	(2) Normal		
design 2		No iniciada	(2) Normal		
design 3		No iniciada	(2) Normal		
dev 1		No iniciada	(2) Normal		
dev 2		No iniciada	(2) Normal		
dev 3		No iniciada	(2) Normal		

Ilustración 61. Crear Proyecto – Plan de Proyecto I

En este momento acabamos de llegar al proceso de asignación de recursos y costes, el proyecto ya se ha creado correctamente con todas las actividades y patrones de producto asociados. Puntualizar que la información que se puede ver en la imagen está ya alojada correctamente en el gestor de contenidos y gestor documental de la aplicación.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

The screenshot shows a project management tool with a Gantt chart at the top and a task list below. The Gantt chart has columns for dates from 09/11/2009 to 28/12/2009. The task list has columns for title, assignee, task status, priority, due date, and completion percentage. A context menu is open over the 'Requirements Definition' task.

Titulo	09/11/2009	16/11/2009	23/11/2009	30/11/2009	07/12/2009	14/12/2009	21/12/2009	28/12/2009	
Requirements Definition	◆								
Use Cases Definition	◆								
Extended Use Cases Definition	◆								
design_1	◆								
design_2	◆								
design_3	◆								
dev_1	◆								
dev_2	◆								

Titulo	Asignado a	Estado de la tarea	Prioridad	Fecha de vencimiento	% completa
Requirements Definition <i>Nuevo</i>		No iniciada	(2) Normal		
Use Cases Definition <i>Nuevo</i>		No iniciada	(2) Normal		
Extended Use Cases Definition <i>Nuevo</i>		No iniciada	(2) Normal		
design_1 <i>Nuevo</i>		No iniciada	(2) Normal		
design_2 <i>Nuevo</i>		No iniciada	(2) Normal		
design_3 <i>Nuevo</i>		No iniciada	(2) Normal		
dev_1 <i>Nuevo</i>		No iniciada	(2) Normal		
dev_2 <i>Nuevo</i>		No iniciada	(2) Normal		
dev_3 <i>Nuevo</i>		No iniciada	(2) Normal		

Context menu options:

- Ver elemento
- Editar elemento
- Administrar permisos
- Eliminar elemento
- Enviar alertas

Ilustración 62. Crear Proyecto – Plan de Proyecto II



Home > PFC > Plan de proyecto > Requirements Definition > Editar elemento

Plan de proyecto: Requirements Definition

Aceptar Cancelar

Adjuntar archivo | Eliminar elemento | Ortografía... * indica un campo obligatorio

IdProject	<input type="text" value="51"/>
IdActivity	<input type="text" value="2"/>
Título *	<input type="text" value="Requirements Definition"/>
Patrón de producto seleccionado	<input type="text" value="1"/>
Patrones de producto	Requirements Specification Product Pattern
Prioridad	(2) Normal
Estado de la tarea	No iniciada
% completado	<input type="text" value=""/> %
Asignado a	<input type="text" value=""/>
Descripción	<div style="border: 1px solid #ccc; padding: 5px;"><p>A A1 N K U [List of icons]</p></div>
Fecha de inicio	<input type="text" value="09/11/2009"/>
Fecha de vencimiento	<input type="text" value=""/>

Creado el 09/11/2009 10:18 por Cuenta del sistema
Última modificación realizada el 09/11/2009 10:18 por Cuenta del sistema

Aceptar Cancelar

Ilustración 63. Crear Proyecto – Plan de Proyecto III

Como podemos ver en las dos imágenes anteriores, cada actividad se puede editar y así poder rellenar cada uno de los campos que contiene una actividad en un plan de proyecto. A continuación se procederá a realizar una breve descripción de cada uno de los campos de la actividad:

- **IdProject:** Identificador del proyecto en cuestión, información adicional que puede ayudar al administrador a la hora de gestionar la BBDD.
- **IdActivity:** Identificador de la actividad en cuestión, información adicional que puede ayudar al administrador a la hora de gestionar la BBDD.
- **Título:** Título de la actividad que queremos modificar en el plan de proyecto.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 5: Diseño del Sistema

- **Patrón de producto seleccionado:** Identificador del patrón de producto seleccionado en la gestión de patrones de producto vista anteriormente. Esta información puede resultar muy útil para el administrador, ya que la puede utilizar para gestionar la BBDD así como para solucionar posibles incidencias en el futuro.
- **Patrones de producto:** Listado de todos los patrones de producto asociados con la actividad en cuestión, dicha lista coincide con la expuesta en la gestión de patrones de producto.
- **Prioridad:** Campo que indica cuál es la prioridad de la actividad en cuestión, este campo resulta muy útil para los jefes de proyectos, así puede priorizar todas las actividades del proyecto y obtener mayores beneficios en cuanto a la correcta gestión de un plan de proyecto. Valores { Alta, Normal, Baja }
- **Estado de la tarea:** Campo que indica cuál es el estado actual de cada una de las tareas/actividades. Resulta muy útil para jefes de proyecto y supervisores de sus empleados con la única finalidad de llevar un mayor control del proyecto, así se podrá garantizar al cliente la presentación del producto en el plazo acordado.
- **% completado:** Siempre que se va a gestionar un proyecto, se debe realizar una serie de tareas de seguimiento para poder obtener informes detallados del estado de cada una de las tareas pertenecientes a la planificación, por tanto este campo indica el % completado de la tarea en cuestión.
- **Asignado a:** Indica cual es el recurso asignado a la tarea especificada, información útil para el jefe de proyecto porque así puede gestionar los recursos de una forma más eficiente.
- **Descripción:** Descripción de la tarea en cuestión, esta información sirve para que el recurso asignado sepa exactamente lo que tiene que hacer o a donde tiene que ir para extender la funcionalidad descrita.
- **Fecha de inicio:** Fecha en la que la tarea comenzó.
- **Fecha de vencimiento:** Fecha en la que está prevista finalizar la tarea.



Ilustración 64. Zona de Administración

Como se puede ver en la imagen, esta es la estructura de la zona de administración donde están alojadas las listas utilizadas, documentos, imágenes, páginas... estas últimas como podemos ver corresponden a bibliotecas de documentos. Por tanto, ya que la zona de administración contiene numerosos procesos estándares del propio producto (MOSS 2007), se facilita una dirección Web donde podemos encontrar toda la información relativa a dicho producto: <http://sharepoint.microsoft.com/Pages/SearchResult.aspx/Results.aspx?k=axpertia>



5.3 DISEÑO FÍSICO DE DATOS

A continuación, se presentará el Modelo Lógico de Datos que ya se mostró en el diagrama de clases anteriormente pero con la excepción de que se añaden todos los módulos funcionales que pertenecen a la BBDD y que más adelante se implementarán herramientas para consumir todos los datos pertenecientes a dicho modelo. Por tanto, para desglosar un poco más los datos de la aplicación se expondrán 2 modelos (uno por BBDD):

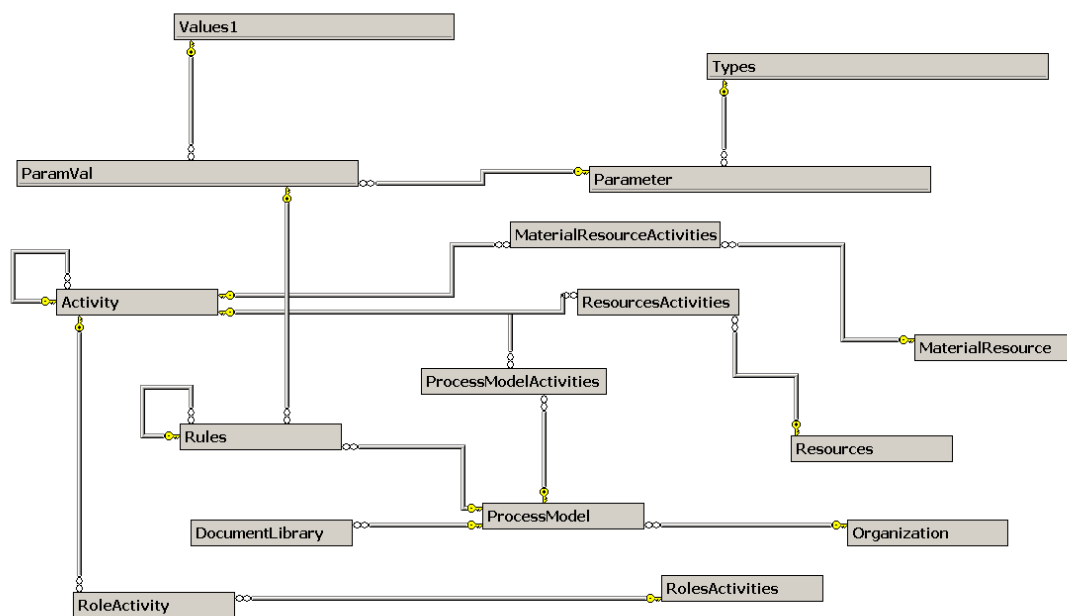


Ilustración 65. Modelo de Datos – PIBOK_PB_ProcessModel

En este modelo de datos, podemos resaltar algunas de las tablas que no han sido utilizadas como pueden ser:

- **MaterialResource:** Recursos materiales de cada una de las actividades, en desarrollos posteriores se añadirá un módulo donde se pueda consultar esta información, así como implementar una herramienta administrativa que gestione estos datos.
- **Resources:** Recursos humanos necesarios para desempeñar una tarea específica, en desarrollos posteriores se añadirá un módulo donde se pueda consultar esta información, así como implementar una herramienta administrativa que gestione estos datos.

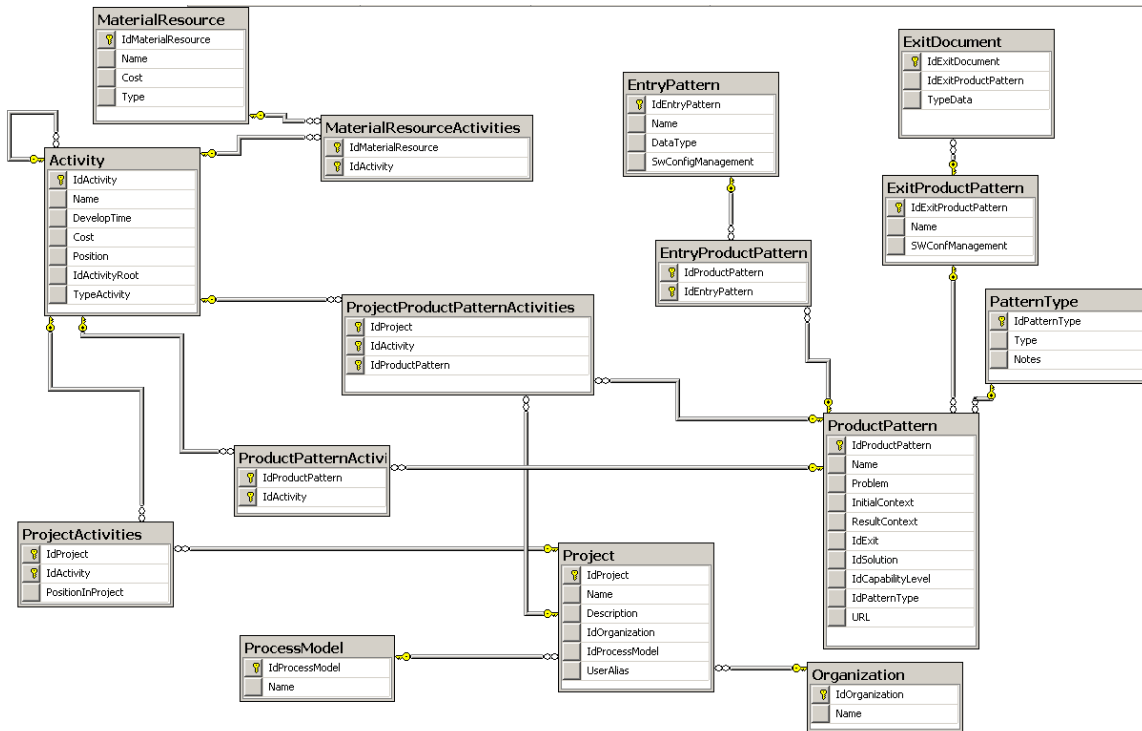


Ilustración 66. Modelo de Datos – PIBOK_PB_CurrentProject

En este modelo de datos, aparecen las entidades relacionadas con proyectos, además de los patrones de producto que están asociados a cada actividad. Destacar que existen otras entidades relacionadas con los patrones de producto (EntryPattern, ExitPattern...), dichas entidades se podrán extender en la aplicación Web con nuevos desarrollos, así como la implementación de alguna herramienta que facilite la gestión de los patrones de producto.



5.4 ARQUITECTURA TRES CAPAS

La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño o de interfaz, un ejemplo básico de esto es separar la capa de datos de la capa de presentación al usuario.

La idea de la “arquitectura orientada a capas” es que tenemos separadas de forma clara las responsabilidades, desacoplando nuestro código.

Si no tuviéramos capas, que es lo mismo que decir que tenemos una sola capa, nuestro código tendría mezcladas todas las responsabilidades. Cualquier cambio que se quiera hacer al código que implementa la interfaz afectaría a todo el resto del código.

Si necesitáramos cambiar de interfaz deberíamos modificar todas las llamadas en nuestro código que cumplen esa tarea. Pero si tenemos "capas", podemos crear un nuevo "paquete o clase" y sustituirlo por el anterior, y nuestro sistema debería poder seguir funcionando sin problemas. Sería exactamente el mismo caso si tuviéramos que cambiar de motor de bases de datos, solo deberíamos poder cambiar de "capa" y nuestro sistema seguir operativo (o por lo menos, las modificaciones solo afectarían un paquete, no al resto).

A continuación se muestra una definición de cada una de las capas que componen esta arquitectura:

1. Capa de presentación: Es la que ve el usuario, presenta el sistema al usuario, le comunica la información y captura la información del usuario dando un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Es lo que conocemos generalmente por la interfaz de usuario (formularios, páginas Web, etc.). Esta capa se comunica únicamente con la capa de negocio.

2. Capa de negocio: Es la encargada de recibir las peticiones del usuario y enviar las respuestas tras el proceso. Se denomina capa de negocio (o de lógica del negocio), pues es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos información o para eliminar, modificar o almacenar datos en él.

En esta capa se aloja el código que resuelve concretamente la problemática de nuestro sistema.

3.- Capa de datos: Es la encargada de comunicarse con la BBDD, es decir montar las instrucciones que realizan todo el almacenamiento de datos, o recuperan información para pasársela a la capa de negocio.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

En el presente proyecto todas estas capas residen en un único ordenador, lo que se denomina “Arquitectura a tres capas con un solo nivel”, aunque lo normal y más correcto sería ubicar cada una de las capas en un servidor (tres niveles), pero es obvio el coste que esto supone, por lo que no es factible para el desarrollo de este proyecto.

Cada una de las capas se implementa como una clase aparte del resto, con un nombre característico que identifica dentro de que módulo se encuentran ubicados.

A continuación se detalla la nomenclatura específica de cada una de ellas, así cómo las labores que realizan:

- **Clases de la capa de interfaz:** UC3M.PIBOK.Web.Website.[Nombre del módulo]

Se encargan de establecer cuál va a ser el diseño de los controles que se van a mostrar en el formulario al usuario (tamaño, color, tipo de letra, posición, etc.)

Por otra parte recoge los datos introducidos por el usuario en los formulario y, tras validarlos (formato correcto, valores dentro de unos determinados rangos, campos marcados como obligatorios rellenos, etc.), se los pasa a la capa de negocio para que ésta realice con ellos lo que crea oportuno.

Además de todo esto, la capa de interfaz recibe de la lógica de negocios los datos que resultan de llevar a cabo diversas operaciones, para mostrárselos al usuario en los controles correspondientes.

- **Clases de la capa de lógica de negocio:** UC3M.PIBOK.Web.BL.[Nombre del módulo]

Se encargan principalmente de transformar los valores introducidos por el usuario en el formulario, los cuales les llegan a través de la capa de interfaz, en datos válidos para ser introducidos en la BBDD.

Es la encargada de controlar todo lo que se refiere a la lógica de negocio. De esta manera nos aseguramos que todas las instrucciones SQL que montamos en la capa de datos van a devolver un resultado satisfactorio, sin que se produzca ninguna excepción.

También es la encargada de gestionar las transacciones, ejecutar las sentencias SQL que recibe de la capa de datos, así como realizar las comprobaciones oportunas de los campos marcados como FK de las tablas.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

- **Clases de la capa de datos:** UC3M.PIBOK.Web.DAL.[Nombre del módulo]

La única funcionalidad que se ha albergado en esta capa, es la de montar las sentencias SQL que posteriormente van a ser ejecutadas en la capa de lógica de negocio.

Por lo tanto todos los métodos encuadrados en estas clases reciben como parámetros los campos necesarios para construir la sentencia SQL y devuelven la sentencia montada.



5.5 GESTIÓN DE TRANSACCIONES

5.5.1 PROCESAMIENTO DE TRANSACCIONES

A menudo se dice que una transacción es una unidad lógica de trabajo, lo que quiere decir que es una forma de agrupar y asociar un cierto conjunto de acciones en una única acción, que se ejecuta en la forma <<todo o nada>>.

Como ejemplo se expone el de una transacción bancaria en la que se traspasan 900 euros desde una cuenta corriente a una cuenta de ahorros. Esta transacción incluye dos acciones distintas: una resta de la cuenta corriente y un incremento de la cuenta de ahorro. Piense en el impacto que tendría en su situación financiera que el servidor del banco se estropeará justo después de completar la primera etapa y no se llegara a ejecutar la segunda. Al agrupar las dos operaciones, en forma de transacción, o ambas tienen éxito o fallan simultáneamente.

Una vez visto la importancia de añadir al proyecto el procesamiento de transacciones, a continuación se detallan las principales características o propiedades que debe obligatoriamente cumplir una transacción. Estas propiedades son conocidas como ACID (por sus iniciales inglesas):

- **Atomicidad:** Las modificaciones asociadas son proposiciones del tipo todo o nada. O bien se realizan todas, o no se realiza ninguna.
- **Coherencia:** después de que finalice una transacción, los datos se encuentren en el estado en que deberían encontrarse, todas las estructuras internas son correctas y todo refleja con precisión el efecto de la transacción que ha tenido lugar.
- **Aislamiento:** Ninguna transacción puede interferir con los procesos de otra transacción.
- **Permanencia:** Una vez que la transacción ha finalizado, los cambios realizados son permanentes.

La responsabilidad de asegurar el cumplimiento de estas características es compartida tanto por los desarrolladores, es decir por mí mismo, como por SQL Server. El desarrollador es responsable de asegurar que las modificaciones se agrupan correctamente y de que los datos van a quedar en estado coherente. SQL Server asegura que la transacción estará aislada y será permanente, se asegura de proporcionar la naturaleza atómica a la transacción y garantiza la coherencia de las estructuras de datos finales.



5.5.2 TRANSACCIONES Y BLOQUEOS

SQL Server realiza y mantiene una serie de bloqueos durante la transacción, para asegurar el aislamiento de la misma y la coherencia de las modificaciones.

Existen cuatro tipos diferentes de bloqueo: READ COMMITTED, READ UNCOMMITTED, REPEATABLE READ y SERIALIZABLE.

Para el desarrollo del proyecto se ha empleado el tipo de bloqueo que SQL Server usa como comportamiento predeterminado, es decir READ COMMITED.

Este tipo de bloqueo hace que la transacción mantenga bloqueos compartidos sobre los datos durante el tiempo que dure la lectura. Sin embargo estos bloqueos no tienen efecto más allá de la página que esté siendo leída; la transacción puede encontrarse con lecturas no repetibles o datos fantasma, porque no existe un bloqueo que impida a otras transacciones modificar los datos una vez que los hemos leído.

5.5.3 CODIFICACIÓN DE TRANSACCIONES EFECTIVAS

Pero no todo a la hora de emplear transacciones es beneficioso, ya que las transacciones tienen un efecto negativo en la concurrencia del acceso a los datos. SQL Server puede bloquear una serie de recursos mientras la transacción está abierta; las filas codificadas tienen bloqueos exclusivos y pueden mantenerse también otros bloqueos. Por lo que para reducir la contienda de bloqueo sobre los recursos, deben mantenerse las transacciones tan cortas y efectivas como sea posible.

A continuación se enumeran las consideraciones que se han tenido en cuenta a la hora de programar las transacciones con el fin de reducir el problema:

- No devolver conjuntos de resultados dentro de una transacción. Dicha acción prolonga la transacción innecesariamente. La recuperación de los datos y su análisis la realizo fuera de la transacción.
- Cuando una transacción falla se debe deshacer antes de hacer aparecer el cuadro de mensaje que informa al usuario de que se ha producido un error.

De no realizarlo de esta manera, la transacción no se desharía hasta que el usuario no pulsara aceptar en el cuadro de error, e imaginemos lo que supondría que el usuario se marchará a tomar café dejando abierto dicho cuadro....



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

- Mantener el inicio y el final de la transacción juntos, en el mismo lote.
- La transacción es lo más corta posible, se da comienzo a la misma en el punto donde se necesita llevar a cabo las modificaciones. Las tareas preliminares necesarias son realizadas de antemano.
- Como esquema de bloqueo se emplea el usado por SQL Server 2005 por defecto, el READ COMMITED.
- Se emplean bloqueos optimistas. El bloqueo optimista hace uso de la cláusula WHERE para actualizar los datos en lugar de bloquear una página o una tabla.
- Se agrupan múltiples transacciones en una única transacción siempre que es posible, ya que reduce la cantidad de trabajo que SQL Server tiene que realizar para iniciar y terminar una transacción.

5.5.4 OPERACIONES REALIZADAS HACIENDO USO DE TRANSACCIONES

Dentro de la aplicación existen algunas operaciones que es necesario que se lleven a cabo haciendo el uso de las transacciones con el nivel de bloqueo arriba descrito.

Realmente esto no sería necesario realizarlo de esta manera, ya que con las restricciones de integridad controladas en el diseño físico de la BBDD mediante los triggers bastaría. Pero se ha decidido establecer un doble control de integridad, es decir, que a través del código fuente de la aplicación también se va a controlar que se cumplan todas las restricciones de integridad, con el fin de poder ejecutar esta misma aplicación contra otra base de datos que carezca de dicho control.

A continuación se enumeran todas aquellas operaciones que se han llevado a cabo dentro de una transacción:

- Implementación de todos los borrados en cascada.
- Inserción y borrado en SuperTipo y SubTipo.
- Para la implementación de un borrado restringido controlado por código, se ejecuta la instrucción *delete* indicándole en la cláusula *where* de la misma las restricciones. Si esa ejecución devuelve 0 registros significa que el borrado no ha podido llevarse a cabo y se aborta la transacción, en caso contrario se continúan con todos los borrados asociados en cascada.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

5.6 PRINCIPIOS DE DISEÑO DE LA INTERFAZ DE USUARIO

En esta actividad se especifican las interfaces entre el Sistema de Información y el usuario: formatos de pantallas y diálogos, principalmente. El objetivo es realizar un análisis de los procesos del Sistema de Información en los que se requiere la interacción con el usuario con el fin de crear una interfaz que satisfaga todos los requisitos establecidos.

El propósito es construir una interfaz de usuario acorde a sus necesidades, flexible, coherente, eficiente y sencilla de utilizar.

Para cada proceso que requiere de la intervención del usuario se especifica el tipo de información que es necesaria y se realiza una descomposición de diálogos.

Finalmente, se define el formato y contenido de cada uno de los interfaces de pantalla especificando su comportamiento dinámico.

Para la realización de los diversos formularios que van a formar parte de la aplicación se han tenido en cuenta todos los principios de ingeniería de la usabilidad así como de diseño estudiados en INTERFACES DE USUARIO.

5.6.1 PRINCIPIOS GENERALES DE INGENIERÍA DE LA USABILIDAD

A continuación se exponen los principios de la ingeniería de la usabilidad y cómo la aplicación responde a ellos.

1. Que sea efectiva: Que la aplicación realice bien el trabajo que se supone que debe realizar.

Para comprobar el cumplimiento de este principio, tras la finalización de la misma se realizará un plan exhaustivo de pruebas unitarias y de integración descrito más adelante.

2. Que sea eficiente: La aplicación debe ayudar en todo momento al usuario a llevar a cabo sus tareas.

Para ello se emiten mensajes de información, se emplean iconos, títulos y mensajes explicativos, distinción de colores entre campos obligatorios y campos opcionales, etc.

3. Que sea seguro: Se refiere a que el usuario esté protegido de condiciones peligrosas y de situaciones indeseables.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

Para ello a lo largo de toda la aplicación se capturan las posibles excepciones que puedan saltar a la hora de trabajar con la BBDD (tiempo de espera de respuesta sobrepasado, final inesperado de una transacción, campo obligatorio vacío...), y dado que el usuario no tiene por que entender el lenguaje empleado por SQL Server para anunciar tales excepciones, se cambian estos mensajes por unos más explicativos del error que acaba de producirse.

4. Que sea útil: Se refiere a que el sistema proporciones el tipo de funcionalidades correctas, de manera que el usuario pueda hacer lo que necesita y lo que quiera hacer.

5. Que se pueda aprender fácilmente: Se refiere al esfuerzo que requiere aprender a usar un sistema.

La aplicación se desarrolla respetando la unicidad de todos sus módulos, es decir los pasos a realizar para llevar a cabo una funcionalidad son siempre los mismos, de manera que con aprender a usar uno de los módulos de la aplicación (ej. Insertar una verificación) se aprenden a usar todos, no requiriendo el aprendizaje de éste más de 10 minutos. (Regla de los 10 minutos).

6. Que sea fácil el recordar cómo se usa: Se refiere al esfuerzo que requiere recordar un sistema después de que se haya aprendido como se usa y no se haya utilizado durante un tiempo.

Este principio también se cumple dado que sólo es necesario recordar el funcionamiento de uno de los módulos para ser capaces de recordar cómo se usa la aplicación al completo.



CAPITULO 6. PRUEBAS

6.1 ESPECIFICACIÓN DEL PLAN DE PRUEBAS

Se trata de la definición del plan de pruebas, a alto nivel, que sirva como guía para la realización de las pruebas que permitan verificar que la aplicación cumple las necesidades establecidas por el usuario, con las debidas garantías de calidad.

En este punto, se define el marco general, estableciendo los requisitos de prueba de aceptación relacionados directamente con la especificación de requisitos por parte del usuario.

La aplicación web deberá someterse a una serie de pruebas de diversa índole antes de realizar su puesta en producción.

Con esto, las pruebas a realizar se agruparán en distintos niveles según la característica del sistema en que se centran, distinguiéndose los siguientes:

1. Pruebas unitarias: Cada uno de los módulos desarrollados estará probado para que funcione correctamente según las entradas y salidas diseñadas.

2. Pruebas de integración: Se realizarán las pruebas oportunas que permitan la perfecta integración de todos los elementos que componen el sistema definitivo: subsistemas, herramientas, componentes...

3. Pruebas de sistema: Pruebas de rendimiento y recursos del sistema para que la ejecución sea aceptable.

4. Pruebas de implantación: Pruebas para la correcta puesta en explotación del software desarrollado.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

6.2 CATÁLOGO DE PRUEBAS

ID	PRUEBA	RESULTADO ESPERADO	RESULTADO PRUEBA	FECHA	VALIDADO
MODULO	Descripción Prueba	Resultado	Resultado	dd/mm/aa	Ok ó vacío
PROY0001	Inserción de un nuevo proyecto.	Nuevo proyecto con el campo IdProject incrementado en uno con respecto al anterior existente.	Nuevo proyecto insertado en la Base de Datos.	12/10/2009	OK
PROY0002	Consulta de un proyecto.	Visualización de la información del proyecto seleccionado en el listado, con la opción de escritura deshabilitada.	Visualización de los datos del proyecto seleccionado.	12/10/2009	OK
PROY0003	Borrado de un proyecto.	Aviso de que el borrado se realizó correctamente.	Resultado correcto	12/10/2009	OK
PROY0004	Modificación de un proyecto.	Visualización de la información del proyecto seleccionado en el formulario, de manera que pueda modificar la información que desee, siempre y cuando siga un formato correcto (Nombre y Descripción deben estar completos)	Operación realizada correctamente	12/10/2009	OK
ORG0001	Inserción de una nueva organización	Nueva organización con el campo IdOrganization incrementado en uno con respecto al anterior existente.	Nueva organización insertada en la Base de Datos.	12/10/2009	OK
ORG0002	Consulta de una organización	Visualización de la información de la organización seleccionada.	Visualización de los datos de la organización seleccionada.	12/10/2009	OK



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

ORG0003	Borrado de una organización.	Borrado de la organización de la BBDD.	Resultado correcto	12/10/2009	OK
ORG0004	Modificación de una organización.	Visualización de la información de la organización seleccionada, de manera que pueda modificar la información que desee.	Operación realizada correctamente	12/10/2009	OK
ACT0001	Inserción de una nueva actividad	Nueva actividad añadida al modelo de proceso seleccionado.	Nueva actividad insertada en el modelo de proceso en cuestión.	12/10/2009	OK
ACT0002	Modificar posición en el árbol	Modificación de la posición en el árbol de actividades escogido por el usuario..	Resultado correcto	12/10/2009	OK
ACT0003	Borrado de una actividad.	Borrado de una actividad del modelo de proceso escogido por el usuario.	Resultado correcto	12/10/2009	OK
MOD0001	Inserción modelo de proceso personalizado	Inserción del modelo de proceso personalizado, añadiendo todas las actividades en la posición que el usuario desee.	Operación realizada correctamente	12/10/2009	OK
MOD0002	Consulta de un modelo de proceso.	Visualización del modelo de proceso escogido por el usuario.	Resultado correcto	12/10/2009	OK



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

PAR0001	Inserción de nuevo parámetro.	Inclusión de un nuevo parámetro en las reglas heurísticas del sistema experto, dicho parámetro resulta ser vital para el usuario.	Parámetro añadido a la búsqueda.	12/10/2009	OK
PAR0002	Borrado de un parámetro.	Eliminación del parámetro para la búsqueda que quiera realizar el usuario (no selección de dicho parámetro).	Parámetro no seleccionado para búsqueda.	12/10/2009	OK
PAR0003	Modificación de un parámetro.	Selección de otro valor dentro del rango permitido para un parámetro en concreto.	Resultado correcto	12/10/2009	OK
DOC0001	Consulta de la documentación referenciada.	Visualización de los documentos relacionados con el modelo de proceso seleccionado.	Operación realizada correctamente	12/10/2009	OK
PAT0001	Añadir patrón de producto.	Selección de un nuevo patrón de producto en el árbol de actividades escogido por el usuario.	Operación realizada correctamente	12/10/2009	OK
PAT0002	Modificación de un patrón de producto.	Selección de otro patrón de producto diferente al seleccionado anteriormente para una actividad perteneciente al proyecto escogido por el usuario.	Resultado correcto	12/10/2009	OK
PAT0003	Eliminación de un patrón de producto.	Eliminación de una asociación entre una actividad y un patrón de producto en el momento que se está creando el proyecto.	Patrón de producto eliminado correctamente	12/10/2009	OK



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 6: Pruebas

PAT0004	Consulta de un patrón de producto.	Visualización del patrón de producto seleccionado para una actividad en concreto, justo en el momento de creación del proyecto.	Resultado correcto.	12/10/2009	OK
PLAN0001	Añadir una tarea al plan de proyecto.	Tras la creación del proyecto, se decide crear una tarea adicional que el sistema no tenía almacenado, se asocia directamente con el plan de proyecto en cuestión.	Operación realizada correctamente.	12/10/2009	OK
PLAN0002	Borrado de una tarea en un plan de proyecto.	Borrado de alguna tarea asociada inicialmente a algún plan de proyecto.	Resultado correcto	12/10/2009	OK
PLAN0003	Modificación de una tarea en un plan de proyecto	Modificación de la información de una tarea asociada a un plan de proyecto existente.	Operación realizada correctamente	12/10/2009	OK
PLAN0004	Consulta de una tarea en un plan de proyecto.	Visualización de la información correspondiente a una de las tareas asociadas a un plan de proyecto en concreto.	Operación realizada correctamente	12/10/2009	OK



CAPITULO 7. CONCLUSIONES Y LÍNEAS FUTURAS

7.1 CONCLUSIONES

A continuación, se enumeran todas las conclusiones obtenidas a la finalización del presente proyecto, así como aquellos puntos positivos que aportan una mayor calidad al resultado final:

- En primer lugar, destacar que el proyecto ha logrado cumplir todos los objetivos marcados al inicio del mismo, así como satisfacer los diferentes requisitos de usuario, cubriendo así toda la funcionalidad requerida, al mismo tiempo de haber conseguido modernizar la apariencia visual de la interfaz y dotar a la aplicación de un manejo más rápido e intuitivo.
- Gracias al desarrollo de este proyecto he podido aprender el manejo de diversas herramientas tales como MOSS 2007, Visio, Project, U2U CAML, WebServices, varios plugins de JQUERY, técnicas avanzadas de ASP.NET, SQL Server 2005, diseño de hojas de estilos y por último entendí cómo se pueden optimizar los procesos de un proyecto, utilizando los modelos idóneos, así se podrá ahorrar tiempo y dinero en todas las fases de un proyecto.
- Todo lo citado anteriormente, me ha servido para madurar en mi trayectoria como profesional en el mercado tecnológico, así como el cierre de un ciclo que para mí ha significado algo más que una carrera.
- Todas las operaciones SQL que necesitan ejecutarse de modo atómico (todas o ninguna), han sido implementadas usando transacciones. Para la programación de estas transacciones se ha tenido en cuenta varias consideraciones que minimizan el tiempo de bloqueo y por tanto favorecen la concurrencia del acceso a datos, sin por ello perder su funcionalidad y su seguridad.
- Se ha tenido especial cuidado en minimizar el trasiego de información cliente-BBDD, empleando, siempre que ha sido posible, una única sentencia SQL para devolver todos los datos necesarios, en lugar de una sentencia SQL por tabla. Con ello quiero decir que mediante estructuras de .NET se podía gestionar gran cantidad de información con tal de minimizar las peticiones a la BBDD.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

- Uno de los puntos a destacar en el presente proyecto es la utilización de gran cantidad de código cliente, utilizando AJAX para comunicarse con el servidor lo que le atribuye un alto nivel de cualificación en cuanto a tecnología se refiere. Para aquellos que desconozcan esta nomenclatura, podría traducirse en una web con gran visibilidad haciéndola lo más interactiva posible y dinámica. Estos valores mencionados anteriormente hacen que la aplicación sea agradable y de fácil manejo para cualquier usuario.
- En cuanto al producto utilizado, para aquellos que ya hayan oído hablar de la herramienta y tengan en la mente la visión del producto estándar, apreciará inmediatamente el gran trabajo que se ha invertido en cuanto instalación, configuración y diseño se refiere. Por lo tanto, ya que es tan compleja dicha instalación y configuración se facilitará más adelante un ANEXO con todos los pasos necesarios para obtener la aplicación creada por un servidor.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

7.2 FUTURAS LÍNEAS DE TRABAJO

Una vez finalizado este proyecto, y tras conocer cuál ha sido el alcance del mismo, surgen futuras líneas de ampliación a través de las cuales se puede mejorar el sistema hasta este momento construido. A continuación se enumeran algunas de ellas:

- Mejorar la herramienta de administración, no sólo a la hora de expandir la funcionalidad sino coger este módulo de administración y plasmarlo en la aplicación web ya creada, restringiendo el acceso a esta parte privada.
- Modificar algunos elementos de la herramienta administrativa, para mejorar tanto el diseño como la funcionalidad integrante.
- Ampliar las funcionalidades añadidas a la nueva aplicación, como pueden ser los patrones de producto (aportando más datos de cada uno de ellos), utilización de roles para los distintos proyectos creados, gestión de recursos materiales...
- Mejorar el aprovechamiento de la herramienta como gestor documental, almacenando toda la documentación necesaria para cada uno de los procesos del sistema.
- Implementación de flujos de trabajo necesarios para el correcto funcionamiento de algunos procesos, los cuales necesitan de validaciones de algunas personas en concreto, pasando entonces por distintos estados.
- Asociar un diagrama de actividades por patrón de producto, representando los pasos necesarios para aplicar un patrón específico.
- Implementación de un módulo funcional relativo a las tareas que constituyen un plan de proyecto, donde se aportará información sobre todos los detalles que conlleva esa tarea además de mostrar una guía de procesos específicos.
- Desarrollo de los indicadores six sigma mediante diferentes gráficos disponibles en algún componente gráfico integrado en SharePoint.
- Implementación de un módulo de gestión relativo a la utilización de Reporting Services, los cuales darán como resultado una serie de informes claves para la compañía / proyecto en cuestión.
- Ampliación en cuanto a funcionalidad del plan de proyecto, añadiendo información sobre las incidencias reportadas por el cliente a cada una de las tareas. En definitiva, poder realizar un estudio sobre el rendimiento de todos los recursos involucrados en el proyecto, así se podrán obtener conclusiones más detalladas sobre la evolución de un proyecto en su fase de mantenimiento.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

CAPITULO 8. REFERENCIAS

Título: Lenguaje extendido de Javascript para implementar controles de usuariocomplejos

Autor: [John Resig](#) and the [jQuery Team](#)

Temática: Lenguajes de Programación

URL: <http://jquery.com/>

Título: Plugins de JQuery

Autor: [John Resig](#) and the [jQuery Team](#)

Temática: Lenguajes de Programación

URL: <http://plugins.jquery.com/>

Título: JQuery & SharePoint

Autor: [Vedant Kulshreshtha](#)

Temática: Lenguajes de Programación

URL: <http://blogs.technet.com/vedant/archive/2009/02/08/jquery-and-sharepoint-2007.aspx>

Título: Programación con SQL SERVER 2005

Autor: Vieira, Robert

Ed.: ANAYA MULTIMEDIA – WROX

ed.: 2006



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

Título: AjaxPro – El poder del cliente

Autor: Ajax.NET Professional Google group

Temática: Lenguajes de Programación

URL: <http://www.ajaxpro.info/>

Título: Trabajando con AjaxPro

Autor: Ajax.NET Professional Google group

Temática: Lenguajes de Programación

URL: <http://www.codeproject.com/KB/ajax/AjaxNetProExample.aspx>

Título: Best way to access SharePoint programatically

Autor: <http://www.experts-exchange.com>

Temática: Lenguajes de Programación

URL: http://www.experts-exchange.com/OS/Microsoft_Operating_Systems/Server/MS-SharePoint/Q_21851451.html

Título: Professional SharePoint 2007 Web Content Management development: Building Publishing sites with Office SharePoint Server 2007

Autor: Andrew Connel

Ed.: Wrox Programmer to Programmer

ed.: 2007

Título: Ingeniería del Software de Gestión: Análisis y diseño de aplicaciones

Autor: Amescua de, A., García, L., Martínez, P. y Díaz, P.

Ed.: Paraninfo

ed.: 1998



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

Título: Gestión del Proceso Software

Autor: Cuevas, G

Ed.: Centro de Estudios Ramón Areces

ed.: 2002

Título: IEEE 1074, Standard for Developing Software Life Cycle Processes Document

Autor: --

Ed.: --

ed.: 1997

Título: Métrica 3 (2000) Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de información

Autor: --

Temática: Metodologías de Programación

URL: <http://www.csi.map.es/csi/metrica3/>

Título: Análisis y Diseño Detallado de Aplicaciones Informáticas de Gestión

Autor: Mario G. Piattini, José A. Calvo-Manzano, Joaquín Cervera, Luís Fernández

Ed.: RA-MA

ed.: 1996

Título: Profesional C# 2ª Edición

Autor: Simon Robinson, K. Scout Allen, Ollie Cornes, Jay Glynn, Zach Greenvoss, Burton Harvey, Christian Angel, Morgan Skinner, Karli Watson

Ed.: Danypress

ed.: 2004



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

Título: El lenguaje unificado de modelado

Autor: Booch, James Rumbaugh, Ivar Jacobson

Ed.: Addison Wesley

ed.: 2004



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Capítulo 8: Referencias

CAPITULO 9. ANEXOS

A continuación, se argumentarán cada uno de los anexos mencionados en este documento con el fin de aclarar cualquier tipo de duda , ya sea a nivel de código de la aplicación o a nivel funcional, donde se detallan todos los puntos al respecto.



ANEXO 1. EJEMPLOS DE CÓDIGO DE LA APLICACIÓN

- 1) Como podemos ver en este ejemplo, en todos los controles utilizados en la interfaz de usuario se sobrescribe el evento OnInit, la finalidad de este hecho es registrar un componente básico para el sistema llamado AjaxPro. Gracias a este componente, podemos comunicarnos con el servidor sin la necesidad de refrescar la página, además se aprovecha este evento para cargar los scripts necesarios para el correcto funcionamiento del formulario en cuestión:

```
protected override void OnInit(EventArgs e)
{
    base.OnInit(e);

    Utility.RegisterTypeForAjax<typeof(UC3M.PIBOK.Web.ClientServices.ModelProcess)>, this.Page);

    if
    (!this.Page.ClientScript.IsClientScriptIncludeRegistered("InitForm"))
    {
        this.Page.ClientScript.RegisterClientScriptInclude(GetType(),
        "InitForm", "/Client-side/js/InitForm.js");
    }
}
```

- 2) En este caso podemos ver un par de funciones asociadas a un fichero .js utilizando las extensiones de javascript (JQUERY), para hacer referencia a algún objeto de cliente siempre debemos respetar el siguiente formato "\$("[NombreObjeto]")", para más información acude al siguiente link (<http://jquery.com/>). Por último, si existe la necesidad de sobrescribir algún evento de algún control en particular debemos seguir el siguiente formato "\$("[NombreObjeto"]).unbind().click([MetodoEvento])":

```
init: function() {
    $(".btnCustomAddActivity").show();
    $(".btnCustomDeleteActivity").show();
    $(".btnCustomUpActivity").hide();
    $(".btnCustomDownActivity").hide();
    $(".btnCustomAddActivity").attr('disabled', 'disabled');
    $(".btnCustomDeleteActivity").attr('disabled', 'disabled');
    $(".customContainer .tree li
a").unbind().click(CustomProcessModel.onClickNode);
    $(".customContainer
.btnCustomAddActivity").unbind().click(CustomProcessModel.onClickAdd
);
    $(".customContainer
.btnCustomDeleteActivity").unbind().click(CustomProcessModel.onClick
Delete);
    $(".customContainer
.btnCustomUpActivity").unbind().click(CustomProcessModel.onClickUp);
}
```



```
    $(".customContainer
.btnCustomDownActivity").unbind().click(CustomProcessModel.onClickDo
wn);
    },
onClickEdit: function() {
    $(".relatedActions .templateNotSelectedPattern").hide();
    $(".relatedActions .editButton").hide();
    $(".relatedActions .edit").show();

    $(".clicked").css("color", "white");
    $(".clicked").css("background-color", "red");

    var html = $(".relatedActions
.templateSelectedPattern").html();
    var res =
UC3M.PIBOK.Web.ClientServices.ModelProcess.GetProductPatternsActivit
y(idActivity, html, true);

    if (res.value == "") {
        //Cambio propiedad opacity de 0 - 1 y se muestra el
elemento
        $(".relatedActions
.templateNotSelectedPattern").css({opacity:0});
        $(".relatedActions
.templateNotSelectedPattern")[0].style.display="";
        $(".relatedActions
.templateNotSelectedPattern").animate({opacity:1}, 1000, "swing",
function(){ this.style.filter="" });
    }
    else{
        $(".relatedActions .selectedPattern").empty();
        $(".relatedActions
.selectedPattern").append(res.value);

        $(".relatedActions .selectedPattern
#lnkURL").each(function(index, element){
            element.href = element.text;
        });

        //Cambio propiedad opacity de 0 - 1 y se muestra el
elemento
        $(".relatedActions
.selectedPattern").css({opacity:0});
        $(".relatedActions
.selectedPattern")[0].style.display="";
        $(".relatedActions
.selectedPattern").animate({opacity:1}, 1000, "swing", function(){
this.style.filter="" });
    }

    $(".img
img:even").unbind().click(ProductPatterns.onClickImgSelected);
    $(".img
img:odd").unbind().click(ProductPatterns.onClickImgNotSelected);
```



```
        return false;
    },
```

3) Aquí se puede ver la carga del primer control utilizado por la aplicación, InitForm.ascx. Los puntos que podemos destacar de este ejemplo son:

1. Obtención del usuario que accede al sistema mediante el modelo de objetos de SharePoint.
2. Obtención de los proyectos creados por ese mismo usuario.
3. Uso del control de excepciones, todos los métodos contienen las cláusulas try/catch con tal de garantizar el control de la aplicación sin resultados inesperados. En definitiva, gestión de excepciones controladas.

```
protected void Page_Load(object sender, EventArgs e)
{
    try
    {
        if (!Page.IsPostBack)
        {
            string user =
                SPContext.Current.Web.CurrentUser.LoginName;
            this.ddlProjects.DataSource =
                BL_CurrentProject.GetProjects(user);
            this.ddlProjects.DataTextField = "Name";
            this.ddlProjects.DataValueField = "IdProject";
            ListItem item = new
                ListItem(this.GetLocalResourceObject("defaultValue").ToString(), "0");
            this.ddlProjects.DataBind();
            this.ddlProjects.Items.Insert(0, item);
        }
    }
    catch (Exception ex)
    {
        throw;
    }
}
```

4) En este ejemplo vemos una de las clases más importantes de la aplicación, gracias a esta clase se consiguen unos tiempos y rendimiento óptimos. La explicación exacta está redactada a continuación, el empleo de este tipo de clases pertenece a una de las técnicas avanzadas de la programación:

```
public class TaxonomyCache
{
    #region PRIVATE OBJECTS
    private static DataTable Organizations;
    private static DataTable TypeParameters;
```



```
private static DataSet Parameters;
private static DataTable Values;
private static DataTable Param_Val;
private static DataSet ParametersValuesModelProcess;
private static DataTable ProcessModelValues;
#endregion

/// <summary>
/// Constructor estático para simular la función de caché, este
procedimiento consiste en
/// almacenar toda la información una sola vez en objetos que
persistirán en el tiempo, sin tener que
/// realizar consultas frecuentes a la base de datos. Estos
objetos se rellenarán una sola vez cuando
/// se haga referencia a la clase estática.
/// </summary>
static TaxonomyCache()
{
    //Estas estructuras de datos son necesarias para la carga
del primer control de parámetros
    Organizations =
BL_ParametersModelProcess.Load_ddlOrganization();
    TypeParameters =
BL_ParametersModelProcess.GetParameterType();
    Parameters =
BL_ParametersModelProcess.RecoverParametersByType(TypeParameters);
    Values = BL_ParametersModelProcess.GetParameterValues();
    Param_Val = BL_ParametersModelProcess.GetParam_ValTable();
    ParametersValuesModelProcess =
BL_ParametersModelProcess.GetParametersValuesModelProcess(Parameters,
Values, Param_Val);

    //Estas estructuras de datos son necesarias para la carga
de los modelos de proceso que se quieren recomendar
    ProcessModelValues =
BL_ParametersModelProcess.Load_ProcessModelValues();
}

public static DataTable GetOrganizationsCache()
{
    return Organizations;
}

public static DataTable GetTypeParametersCache()
{
    return TypeParameters;
}

public static DataSet GetParametersCache()
{
    return Parameters;
}

public static DataTable GetValuesCache()
{
    return Values;
}
```




```
public static DataTable GetParamValCache ()
{
    return Param_Val;
}

public static DataSet GetParametersValuesModelProcessCache ()
{
    return ParametersValuesModelProcess;
}

public static DataTable GetProcessModelValuesCache ()
{
    return ProcessModelValues;
}
}
```

- 5) En el siguiente ejemplo, se puede observar cómo realizar consultas a objetos del gestor de contenidos (en este caso a una lista), cuyo lenguaje es CAML (Collaborative Application Markup Language):

```
/// <summary>
    /// Creación de la consulta CAML para filtrar por un
    /// proyecto determinado la vista seleccionada.
    /// </summary>
    /// <param name="key"></param>
    /// <returns></returns>
    public static string GetCAMLViewFilter(string idProject)
    {
        StringBuilder sbCAML = new StringBuilder();
        try
        {
            sbCAML.Append("<Where>");
            sbCAML.Append("<Eq>");
            sbCAML.Append("<FieldRef Name='IdProject' />");
            sbCAML.AppendFormat("<Value
Type='Text'>{0}</Value>", idProject);
            sbCAML.Append("</Eq>");
            sbCAML.Append("</Where>");
        }
        catch (Exception ex)
        {
        }

        return sbCAML.ToString();
    }
}
```



- 6) En el siguiente ejemplo, se puede observar cómo implementar los métodos de la capa que actúa como proxy de la aplicación, la cual se encarga de obtener los datos que el cliente solicita mediante peticiones de AJAX (en este caso se devuelven todos los patrones de producto asociados a una actividad en concreto):

```
[AjaxMethod]
public static string GetData(string idActivity, string
selectedProductPatternID)
{
    string result = string.Empty;

    try
    {
        DataTable dtProductPatternsActivities =
BL_ProductPatterns.GetProductPatternActivities(idActivity);

        foreach (DataRow row in
dtProductPatternsActivities.Rows)
        {
            if ((" + row["IdProductPattern"]) ==
selectedProductPatternID)
            {
                result += "<option id=\"" + (" +
row["IdProductPattern"]) + "\" value=\"" + (" + row["Name"]) + "\"
selected=\"selected\">" + (" + row["Name"]) + "</option>";
            }
            else
            {
                result += "<option id=\"" + (" +
row["IdProductPattern"]) + "\" value=\"" + (" + row["Name"]) +
"\">" + (" + row["Name"]) + "</option>";
            }
        }
    }
    catch (Exception ex)
    {
        throw;
    }

    return result;
}
```



- 7) En el siguiente ejemplo, se puede observar cómo crear una webpart (elementos web de SharePoint), con las siguientes líneas de código podemos publicar un control de ASP.NET en cualquier aplicación MOSS:

```
using System;
using System.Runtime.InteropServices;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.WebControls.WebParts;
using System.Xml.Serialization;

using Microsoft.SharePoint;
using Microsoft.SharePoint.WebControls;
using Microsoft.SharePoint.WebPartPages;

namespace UC3M.PIBOK.Web.WebParts
{
    [Guid("cbcafdde-25cd-49c7-b676-b04595715cb5")]
    public class ParametersProcessModel :
        System.Web.UI.WebControls.WebParts.WebPart
    {
        private const string USER_CONTROL_PATH =
            "~/DGSControlTemplates/ParametersModelProcess.ascx";
        private Control _oUserControl = null;

        protected override void CreateChildControls()
        {
            base.CreateChildControls();

            //Load the User control
            _oUserControl =
                this.Page.LoadControl(USER_CONTROL_PATH);

            this.Controls.Add(_oUserControl);
        }
    }
}
```

- 8) Por último, se detalla cómo podemos realizar una impersonación a nivel de programación (usar la identidad de otra persona), este método es muy útil cuando queremos acceder a elementos de la aplicación como listas, bibliotecas, permisos..., también se muestra la forma de invertir el proceso, es decir, restaurar la identidad del usuario:

```
private WindowsIdentity _objOriginalUser;
private AppDomain _objMyAppDomain;
private bool _bImpersonated = false;

public void SetAppPoolIdentity()
{
```



```
WriteEnterTrace("SetAppPoolIdentity");
try
{
    if (_bImpersonated)
        RestoreOriginalCallerIdentity();

    _bImpersonated = true;

    _objOriginalUser =
WindowsIdentity.GetCurrent();
    RevertToSelf();
    WindowsIdentity.GetCurrent().Impersonate();

    AppDomainSetup objAppDomainSetup =
AppDomain.CurrentDomain.SetupInformation;
    Evidence objEvidence =
AppDomain.CurrentDomain.Evidence;
    _objMyAppDomain =
AppDomain.CreateDomain("MyAppDomain", objEvidence, objAppDomainSetup);
}
catch (Exception ex)
{
    WriteError("SetAppPoolIdentity", ex);
    throw new Exception("[SetAppPoolIdentity]:" +
ex.Message, ex);
}
WriteExitTrace("SetAppPoolIdentity");
}

public void RestoreOriginalCallerIdentity()
{
    WriteEnterTrace("RestoreOriginalCallerIdentity");
    try
    {
        if (_bImpersonated)
        {
            _bImpersonated = false;
            AppDomain.Unload(_objMyAppDomain);
            _objOriginalUser.Impersonate();
        }
    }
    catch (Exception ex)
    {
        WriteError("RestoreOriginalCallerIdentity",
ex);
        throw new
Exception("[RestoreOriginalCallerIdentity]:" + ex.Message, ex);
    }
    WriteExitTrace("RestoreOriginalCallerIdentity");
}
```



ANEXO 2. INSTALACIÓN Y CONFIGURACIÓN DEL ENTORNO

En primer lugar, destacar que el proceso de instalación que se va a seguir a continuación resulta algo complejo por lo que deberá abordar cada uno de los pasos verificando que se ha completado todo satisfactoriamente.

En la entrega de la aplicación existe un paquete de instalación llamado **PaqueteInstalaciónUC3M**, en su interior contiene una serie de directorios los cuales se mostrarán a continuación:

Nombre	Tamaño	Tipo	Fecha de modificación
3082		Carpeta de archivos	12/05/2009 20:49
Backup		Carpeta de archivos	13/07/2009 19:50
EstructuraPersonalizada		Carpeta de archivos	12/07/2009 12:40
Features		Carpeta de archivos	13/07/2009 19:53
IIS		Carpeta de archivos	13/07/2009 19:55
SiteTemplates		Carpeta de archivos	13/07/2009 19:54
XML		Carpeta de archivos	12/07/2009 16:10

Ilustración 67. Paquete Instalación - Estructura de carpetas



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

Una vez tenemos la estructura de directorios, según vayamos explicando cada uno de los procesos a implementar sabremos concretamente para que sirven y que contenido albergan. Por tanto, comencemos por los requisitos que deberá tener el servidor donde vayamos a instalar la aplicación:

1. Instalación de Internet Information Services 6.0 ó 7.0
2. Instalación .NET Framework 3.0
3. Instalación y configuración del servidor de correo SMTP/POP3
4. Instalación y configuración SQL Server 2005
5. Instalación Service Pack 2 SQL Server 2005
6. Creación de las cuentas de MOSS (añadir el dominio correspondiente al active directory)
7. Instalación Microsoft Office SharePoint Server 2007 with SP2 (Versión Español)
8. Configurar la seguridad del navegador Internet Explorer para accede a los elementos de MOSS.
9. Iniciar todos los servicios asociados a MOSS
10. Creación de un Shared Services Provider (así se habilitarán las búsquedas de SharePoint)
11. Configurar el servicio de búsqueda.

Más información aquí:

<http://www.pptspaces.com/sharepointreporterblog/Lists/Posts/Post.aspx?List=7537e639-b4e5-48b6-97c0-a75e44ee9be3&ID=28&Source=http%3A%2F%2Fwww.pptspaces.com%2Fsharepointreporterblog%2FLists%2FPosts%2FAllPosts.aspx>



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

Una vez partimos de la base que se citó anteriormente, el servidor ya está listo para la instalación y configuración de nuestro sistema, por lo que mediante un sencillo manual se enumerarán las acciones que deben efectuarse para garantizar un correcto funcionamiento del mismo:

1. Acceder al directorio "FEATURES", copiar el contenido en la siguiente ruta: **C:\Archivos de programa\Archivos comunes\Microsoft Shared\web server extensions\12\TEMPLATE\FEATURES**
2. A partir de ahora nos referiremos a **C:\Archivos de programa\Archivos comunes\Microsoft Shared\web server extensions\12\TEMPLATE** con la siguiente etiqueta [PATH_MOSS]

Nombre	Tamaño	Tipo	Fecha de modificación	Atributos
DGSContentTypes		Carpeta de archivos	13/07/2009 19:53	A
DGSPageLayoutContentType...		Carpeta de archivos	13/07/2009 19:53	A
DGSProvisioning		Carpeta de archivos	13/07/2009 19:53	A

Ilustración 68. Listado de características

3. En este momento tenemos las características pertenecientes a nuestro sistema en el sitio correcto, para una mayor comprensión se pasará a detallar el contenido de cada una de ellas, antes destacar que todas las características deben contener un **feature.xml** para su instalación:
 - a. **DGSContentTypes:** Esta característica determina qué tipos de contenido vamos a utilizar así como la estructura de campos contenida por cada una de las páginas. En este caso, existirá un tipo de contenido por tipo de página (DGSHomePage, DGSSubPage).



DG5ContentTypes.xml	2 KB	Documento XML	12/07/2009 12:49	A
DG5Fields.xml	3 KB	Documento XML	12/07/2009 12:48	A
Feature.xml	1 KB	Documento XML	12/07/2009 12:46	A

Ilustración 69. Estructura Característica – DG5ContentTypes

- b. DG5PageLayoutContentTypeActivator:** Esta característica determina las páginas que queremos activar en nuestra aplicación, hablando siempre a nivel de contenidos, por tanto, en el momento que queramos crear una página aparecerán como opciones adicionales estos 2 tipos de contenido.

elementManifest.xml	1 KB	Documento XML	12/07/2009 12:46	A
Feature.xml	1 KB	Documento XML	12/07/2009 12:45	A

Ilustración 70. Estructura Característica – DG5PageLayoutContentTypeActivator

- c. DG5Provisioning:** Esta característica es la más compleja de todas ya que contiene todas las páginas maestras, páginas base, imágenes... además determina cuáles son los recursos que queremos publicar, utilizando los tipos de contenido anteriormente citados además asignarle una serie de propiedades a cada recurso. Formalmente estaríamos hablando de *“La definición de los recursos del sitio de tipo Publishing”*.



Catalogs	Carpeta de archivos	13/07/2009 19:53	A
Images	Carpeta de archivos	12/05/2009 20:49	A
Lists	Carpeta de archivos	12/05/2009 20:49	A
PublishingLayoutsPreviewIma...	Carpeta de archivos	13/07/2009 19:53	A
DGSSiteDefinitionFiles.xml	4 KB Documento XML	12/07/2009 12:45	A
Feature.xml	1 KB Documento XML	12/07/2009 12:46	A

Ilustración 71. Estructura Característica - DGSProvisioning

4. Copiar los contenidos de la carpeta SiteTemplates en el siguiente path:
[PATH_MOSS]\SiteTemplates, en este momento se ha instalado el elemento más importante de todo el paquete de instalación, por tanto vamos a pasar a detallar cada uno de los elementos de la plantilla del sitio. Dicha plantilla define cada característica, tipo de contenido, plantillas asociadas... infinidad de elementos que son necesarios para poder crear una aplicación en MOSS, por tanto como se comentó anteriormente se va a proceder a explicar cada elemento de dicha plantilla:
 - a. **ONET.xml:** Para empezar se nombra el fichero que contiene todos los elementos citados anteriormente.
 - b. **NavBars:** Un elemento NavBarLink define vínculos para el área de exploración superior o lateral, un una sección completa de NavBar agrupa los vínculos nuevos en el área lateral. Cada elemento NavBar especifica un nombre para mostrar y un identificador único para la barra de exploración y define cómo mostrar la barra de exploración.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

- c. ListTemplates:** Esta sección especifica las definiciones de lista que forman parte de una definición de sitio.

El elemento ListTemplate también especifica un nombre para mostrar para la definición de lista y si la opción para agregar un vínculo en la barra **Inicio rápido** aparece seleccionada de forma predeterminada en la página **Nuevo**.

Además, este elemento especifica la descripción de definición de lista y la ruta de acceso a la imagen que representa la definición de lista, las cuales se muestran en la página **Crear**. Si se especifica `Hidden="TRUE"`, la definición de lista no aparece como una opción en la página **Crear**.

- d. DocumentTemplates:** La sección DocumentTemplates define las plantillas de documento que se enumeran en la página **Nuevo**.

Cada elemento DocumentTemplate especifica un nombre para mostrar, un identificador único y una descripción para la plantilla de documento.

Si **Default** se establece en **TRUE**, la plantilla es la plantilla predeterminada seleccionada para las bibliotecas de documentos que se crean en los sitios creados con una configuración en la definición de sitio que especifica el archivo Onet.xml. A pesar de su nombre en singular, un elemento DocumentTemplate realmente puede contener una colección de elementos DocumentTemplateFile.

El atributo **Name** de cada elemento DocumentTemplateFile especifica la ruta de acceso relativa a un archivo local que sirve como la plantilla.

El atributo **TargetName** especifica la dirección URL de destino del archivo de plantilla cuando se crea una biblioteca de documentos.

El atributo **Default** especifica si el archivo es el archivo de plantilla predeterminado.

- e. Configurations:** Cada elemento Configuration en la sección Configurations especifica las listas y módulos que se crean de forma predeterminada cuando se crean instancias de la configuración de definición del sitio.

El atributo **ID** identifica la configuración (de forma exclusiva relacionados con las demás configuraciones en la definición de sitio) y se corresponde con el atributo **ID** de un elemento **Configuration** en WebTemp.xml (que será explicado posteriormente).

Cada elemento **List** especifica el título de la definición de lista y la dirección URL donde se va a crear la lista.

El atributo **QuickLaunchUrl** se puede usar para establecer la dirección URL de la página de vista que se va a usar al agregar un vínculo en **Inicio rápido** a una lista creada a partir de la definición de lista.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

El valor del atributo **Type** corresponde con el atributo **Type** de una plantilla en la sección plantillas de lista.

Cada elemento **Module** especifica el nombre de un módulo definido en la sección de módulos.

Los elementos SiteFeatures y WebFeatures contienen las referencias a colección de sitios y características del ámbito del sitio para incluir en la definición del sitio.

Para las capacidades de procesamiento posterior, use un elemento ExecuteUrl dentro de un elemento Configuration para especificar la dirección URL que se llama tras la creación de instancias del sitio.

f. Modules: La colección Modules especifica los módulos para incluir de forma predeterminada al crear una colección de sitios.

Cada elemento Module a su vez especifica uno o más archivos que se van a incluir, normalmente para los elementos web, que se almacenan en la memoria caché en el servidor cliente web, junto con los archivos de esquema.

Puede usar el atributo **Url** del elemento Module para aprovisionar una carpeta como parte de la definición del sitio.

El elemento Module especifica un nombre para el módulo, que corresponde a un nombre de módulo que está especificado en una configuración en ONET.XML.

El atributo **Url** de cada elemento File en un módulo especifica el nombre de un archivo para crear cuando se crea un sitio.

Cuando el módulo incluye un archivo único, como Default.aspx, NavBarHome="TRUE" especifica que el archivo actuará como la página de destino para el vínculo **Home** en las barras de exploración.

El elemento File para Default.aspx también especifica los elementos web que desea incluir en la página principal e información sobre la página principal para otras páginas con vínculos a ella.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

5. Copiar los contenidos de la carpeta 3082 en el siguiente path: **[PATH_MOSS]3082**. Este código identifica el código del idioma en el que queremos instalar MOSS, en el caso de que el producto instalado de MOSS fuese con la versión en Inglés, tendremos la carpeta con el código 1033 por tanto si queremos que funcione la aplicación deberíamos instalar el “Language Pack correspondiente”. Para crear una definición de sitio personalizada, se debe copiar una definición de sitio existente (HECHO) y agregar un archivo WEBTEMP*.XML que defina las configuraciones para crear instancias de sitios. Su estructura principal es la siguiente:

- a. El elemento **Templates** especifica las configuraciones que se pueden usar para crear instancias de sitios.
- b. Cada **Template** especifica un identificador único y un nombre que se corresponde con una subcarpeta de definición de sitios dentro del directorio `\TEMPLATE\[CodigoIdioma]`. Un elemento **Template** puede incluir cualquier número de subelementos **Configuration**, que especifican un identificador único que se corresponde con el identificador de una configuración en un archivo `Onet.xml`, que especifica las listas y los módulos de una definición de sitio. Cada elemento **Configuration** también especifica el título, la descripción y la ruta de acceso virtual a la imagen de vista previa que se muestra en la página de selección de plantillas. Es posible ocultar la configuración en la interfaz del usuario mediante el establecimiento de **Hidden** en **TRUE**.

Para más información acceda al siguiente enlace: <http://msdn.microsoft.com/es-es/library/ms447717.aspx>

6. Copiar los contenidos de la carpeta XML en el siguiente path: **[PATH_MOSS]XML**. Dicho elemento atribuye a una plantilla de sitio creada anteriormente un apartado en los “Publishing Sites” mostrados en la Administración Central cuando queremos crear una aplicación web.
7. Dentro del directorio anterior, se encontraba otro fichero llamado **Fldtypes_LevelColumn.xml**, dicho fichero es necesario para la posterior utilización de la lista del Plan de proyecto, ya que se ha desarrollado un campo personalizado para mostrar los patrones de producto asociados a una actividad.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

8. Copiar los contenidos de la carpeta ControlTemplates en el siguiente path:
[PATH_MOSS]\ControlTemplates. Este elemento es necesario para el correcto funcionamiento de la funcionalidad comentada en el punto anterior. PANTALLAZO DEL CONTENIDO.

Para más información consultar el siguiente enlace:
<http://msdn.microsoft.com/es-es/library/ms474369.aspx>

9. Instalar las características citadas anteriormente, para así tener acceso a toda la información que se detallaba en los puntos anteriores. Para ejecutar comandos en SharePoint, necesitamos utilizar una herramienta que viene integrada con el producto llamada STSADM.EXE, su localización exacta es la siguiente: **C:\Archivos de programa\Archivos comunes\Microsoft Shared\web server extensions\12\BIN**, a partir de este momento ese path se llamará [PATH_TOOL]. Por tanto, por línea de comandos accedemos a ese path y realizamos los siguientes pasos:
 - a. Instalación DGSCContentTypes: stsadm –o installfeature –name [PATH_MOSS\FEATURES\DGSCContentTypes]
 - b. Instalación DGSPageLayoutContentTypeActivator: stsadm –o installfeature –name [PATH_MOSS\FEATURES\ DGSPageLayoutContentTypeActivator]
 - c. Instalación DGSProvisioning: stsadm –o installfeature –name [PATH_MOSS\FEATURES\ DGSProvisioning]

10. Copiar el contenido de la carpeta Assembly en el directorio del sistema “Assembly”. Este contenido es necesario para utilizar el campo de la lista Plan de proyecto que se mencionó anteriormente.

11. Ejecutar por línea de comandos iisreset (reinicio de todos los servicios del IIS), para poder detectar el assembly instalado anteriormente.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

12. Crear la aplicación Web correspondiente, para así tener nuestro sistema actualizado. Se deben seguir los siguientes pasos:

- a. Crear la aplicación Web rellenando todos los datos necesarios, a continuación se enumeran una serie de recomendaciones a la hora de rellenar dichos datos:
 - i. **Nombre:** Utilizar un nombre descriptivo de la aplicación Web en el IIS. Yo siempre opto por la siguiente nomenclatura [URL_WEB] – [PUERTO]
 - ii. **Puerto:** Utilizar el rango de puertos abiertos por el servidor donde se vaya a instalar la aplicación.
 - iii. **Permisos:** Siempre utilizar el usuario administrador de la máquina ya que de esta forma podrá administrar todas las aplicaciones Web existentes.
 - iv. **BBDD:** Por defecto le asigna un GUID a la BBDD que vamos a crear, deberemos renombrarla y ponerle el nombre de la aplicación Web, así en el caso de posibles problemas de la aplicación localizar la BBDD rápidamente.

Ilustración 72. Creación Aplicación Web I



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

Restablecer Internet Information Services Use esta sección para permitir que SharePoint reinicie IIS en otros servidores del conjunto. El servidor local debe iniciarse manualmente para que se complete este proceso. Si esta opción no está seleccionada y tiene más de un servidor en el conjunto, debe esperar hasta que se cree el sitio Web de IIS en todos los servidores y luego ejecutar "Iisreset /noforce" en cada servidor Web. El sitio IIS nuevo no se podrá usar hasta que se haya completado.	<input type="radio"/> Reiniciar IIS automáticamente <input checked="" type="radio"/> Reiniciar IIS manualmente
Nombre y autenticación de base de datos Se recomienda utilizar el servidor de base de datos y el nombre de base de datos predeterminados en la mayoría de los casos. Consulte la guía del administrador para escenarios avanzados donde se requiere especificar información de base de datos. Es muy recomendable que utilice la autenticación de Windows. Para usar autenticación SQL, especifique las credenciales que se usarán para conectarse a la base de datos.	Servidor de bases de datos <input type="text" value="AX-MOSS1"/> Nombre de la base de datos <input type="text" value="WSS_Content_web.uc3m.es"/> Autenticación de base de datos <input checked="" type="radio"/> Autenticación de Windows (recomendada) <input type="radio"/> Autenticación de SQL Cuenta <input type="text"/> Contraseña <input type="text"/>
Servidor de búsqueda	El servicio de búsqueda es proporcionado por: Office SharePoint Server Search
<input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/>	

Ilustración 73. Creación Aplicación Web II

- b. Crear colección de sitios, seleccionar la aplicación Web a la que queremos asociar la colección de sitios, a continuación se enumeran una serie de recomendaciones a la hora de rellenar los datos de la colección de sitios:
 - i. Aplicación Web: Seleccionar la creada anteriormente.
 - ii. Título: Utilizar el nombre de "Home", título estándar para todas las aplicaciones.
 - iii. Seleccionar la plantilla DGS Publishing
 - iv. Asignarle un administrador acorde con el seleccionado anteriormente para la aplicación Web.



Proyecto: Herramienta de Reutilización de Productos Software Basada en Patrones de Producto para Entornos Colaborativos

Fecha: 12/10/09

Alumno: David Gómez Sanabria

Anexo 2: Instalación y configuración del entorno

Aplicación Web
Seleccione una aplicación Web.
Aplicación Web: **http://ax-mossi:1111/**

Título y descripción
Escriba un título y una descripción para el nuevo sitio. El título se mostrará en todas las páginas del sitio.
Título:
Descripción:

Dirección del sitio Web
Especifique el nombre y la ruta de la dirección URL para crear un nuevo sitio o crear uno en una ruta específica.
Dirección URL: **http://ax-mossi:1111/**

Selección de plantilla
Seleccione una plantilla:
Colaboración Reuniones Empresa **DGS Publishing** Publicación
DGS Portal HOME

Administrador de la colección de sitios primaria
Especifique el administrador de esta colección de sitios Web.
Nombre de usuario:

Administrador de la colección de sitios secundaria
Especifique el segundo administrador de esta colección de sitios Web.
Nombre de usuario:

Plantilla de cuota
Seleccione una plantilla de cuota predefinida para limitar los recursos utilizados para esta colección de sitios.
Plantilla de cuota: **Sin cuota**
Límite de almacenamiento:
Número de usuarios invitados:

Ilustración 74. Creación de la colección de sitios

13. Copiar los contenidos de la carpeta IIS en el siguiente path:
C:\inetpub\wwwroot\wss\VirtualDirectories\[NombreAplicacionWeb].
14. Acceder a la aplicación introduciendo el usuario y contraseña especificados anteriormente.
15. Si existen problemas de permisos deberéis ir a las Propiedades del AppPool reservada a nuestra aplicación y cambiar la opción Identidad introduciendo los datos del usuario en cuestión.



UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

HERRAMIENTA DE REUTILIZACIÓN DE PRODUCTOS SOFTWARE BASADA EN PATRONES DE PRODUCTO PARA ENTORNOS COLABORATIVOS

Ingeniería Técnica en Informática de Gestión

David Gómez Sanabria