



Universidad
Zaragoza

Proyecto Fin de Carrera

Diseño y desarrollo de un sistema de control de la
dedicación basado en tecnología de Portlets

Autor

Luis Miguel Gimeno Caudepón

Director

Pedro Javier Álvarez Pérez-Aradros

Escuela de Ingeniería y Arquitectura
2013

“Diseño y desarrollo de un sistema de control de la dedicación basado en tecnología de Portlets”

RESUMEN

Este Proyecto Fin de Carrera comprende el análisis, diseño e implementación de un sistema accesible vía Web que facilite el control y la previsión del coste temporal de la realización de una amplia variedad de tipos de proyectos. Dada la naturaleza de cualquier proyecto software es complicado prever con antelación el tiempo necesario para su realización. En algunos casos, puede ser la causa para que no se cumpla con la fecha de entrega, con la posible pérdida de confianza por parte del cliente.

El sistema da acceso a los gestores y a los participantes en los proyectos. Un gestor puede configurar un proyecto y añadirle tareas y participantes. También puede configurar alarmas que permitan avisarle, tanto a él como a los participantes, de que el tiempo previsto para la realización de una tarea del proyecto ha sido superado. Además podrá generar informes de la dedicación de los proyectos que gestione.

Por otro lado, un participante en un proyecto podrá introducir su dedicación diaria realizada a cada una de las tareas del proyecto en el que participe. También podrá ver las alarmas que el gestor haya configurado, así como crear informes de la dedicación que ha realizado en los proyectos.

El sistema está compuesto por dos aplicaciones:

- Una aplicación Web que permitirá a los gestores configurar los proyectos y los participantes.
- Una aplicación Portlet que permitirá a los participantes en los proyectos introducir su dedicación diaria.

La tecnología de Portlet permite crear aplicaciones que se integran fácilmente en un portal Web. Un gestor de un proyecto podrá añadir esta aplicación en su página Web para facilitar el acceso a los participantes.

AGRADECIMIENTOS

Quisiera agradecer a Pedro Álvarez por haberme dirigido este proyecto y a la Universidad de Zaragoza que me ha brindado la oportunidad de realizarlo.

A todos mis profesores que tanto me han enseñado en estos años.

Agradecer a mi familia por su apoyo incondicional y por la paciencia que han tenido durante todos estos años, en especial a Pilar.

Recordar a todos mis amigos y compañeros que me han acompañado en los buenos y los malos tiempos, Virginia, Quique, Rafa, Juan, Cristina, Ángel, Luis, Pilar, No hay puntos suficientes para representarlos a todos, ni los puntos los representan como se merecen.

Quiero dar mi más sincero agradecimiento a todo el personal de la EINA, que me ha adaptado en todo momento las salas, ordenadores y aulas para que pudiera realizar mis estudios.

Y un especial agradecimiento a la ONCE, que sin su apoyo personal y tecnológico no podría haber realizado esta carrera.

A todos muchas gracias.

ÍNDICE

RESUMEN	i
AGRADECIMIENTOS	iii
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS	x
1 INTRODUCCIÓN	1
1.1 CONTEXTO DEL PROYECTO	1
1.2 MOTIVACIÓN	1
1.3 ESTRUCTURA DE LA MEMORIA	2
2 ANÁLISIS DEL PROBLEMA	3
2.1 OBJETIVOS GENERALES	3
2.2 ESPECIFICACIÓN DE REQUISITOS	4
2.2.1 REQUISITOS FUNCIONALES	4
2.2.2 REQUISITOS NO FUNCIONALES	5
2.3 ANÁLISIS DE LAS TECNOLOGÍAS.....	6
2.3.1 REQUISITOS TECNOLÓGICOS	6
2.3.2 SELECCIÓN DE TECNOLOGÍAS	7
3 DISEÑO DEL SISTEMA.....	11
3.1 CONTEXTO DEL SISTEMA	11
3.2 ARQUITECTURA DEL SISTEMA	12
3.2.1 DESPLIEGUE DEL SISTEMA	13
3.3 CAPA DE DATOS.....	15
3.4 CAPA DE NEGOCIO	17
3.4.1 EJEMPLO DE FUNCIONAMIENTO.....	19
3.5 CAPA DE PRESENTACIÓN	21
3.5.1 APLICACIÓN WEB.....	22
3.5.2 APLICACIÓN PORTLET.....	26
4 GESTIÓN DEL PROYECTO	29
4.1 METODOLOGÍA	29
4.2 PLANIFICACIÓN	31
4.3 ESFUERZO REAL DEDICADO	32

5 CONCLUSIONES	35
5.1 CONCLUSIONES TÉCNICAS	35
5.2 CONCLUSIONES PERSONALES	36
5.3 TRABAJO FUTURO	37
ANEXO A DOCUMENTO DE ANÁLISIS DE REQUISITOS	39
A.1 INTRODUCCIÓN.....	41
A.2 REQUISITOS	43
A.2.1 REQUISITOS FUNCIONALES	43
A.2.2 REQUISITOS NO FUNCIONALES	44
A.3 CAPTURA DE REQUISITOS.....	45
A.3.1 APLICACIÓN WEB.....	45
A.3.2 APLICACIÓN PORTLET	47
ANEXO B DOCUMENTO DE DISEÑO DEL SISTEMA	49
B.1 INTRODUCCIÓN.....	51
B.1.1 OBJETIVOS Y ALCANCE DEL DOCUMENTO	51
B.2 DISEÑO DEL SISTEMA	53
B.2.1 DESCOMPOSICIÓN DEL SISTEMA	53
B.2.2 DESPLIEGUE DEL SISTEMA	54
B.2.3 DISEÑO BOTTOM-UP	54
B.3 DISEÑO DE LA BASE DE DATOS	55
B.3.1 DISEÑO CONCEPTUAL	55
B.3.1.1 DESCRIPCIÓN DE LAS ENTIDADES.....	56
B.3.2 DISEÑO LÓGICO.....	60
B.3.2.1 DESCRIPCIÓN DE LAS RELACIONES	60
B.4 ESTRUCTURA DE LOS DATOS	65
B.4.1 CLASES	66
B.4.2 DESCRIPCIÓN DE LAS ALARMAS	67
B.4.2.1 ALARMAS DE TIPO TAREA	68
B.4.2.2 ALARMAS DE TIPO PARTICIPANTE	69
B. 5 CAPA DE DATOS.....	71
B.5.1 ESTEREOTIPO DAO	71
B.6 CAPA DE NEGOCIO	73
B.6.1 COMPONENTES	73

B.6.1.1 GESTIÓN DE LOS PROYECTOS	74
B.6.1.2 GESTIÓN DE LOS USUARIOS.....	77
B.6.1.3 GESTIÓN DE LAS DEDICACIONES.....	78
B.6.1.4 GESTIÓN DE LAS ALARMAS.....	78
B.6.1.5 GENERACIÓN DE INFORMES ESTADÍSTICOS.....	79
B.6.1.6 GENERACIÓN DE INFORMES DE LA DEDICACIÓN	80
B.6.2 COMPORTAMIENTO DE LAS ALARMAS.....	83
B.6.2.1 CREACIÓN Y MODIFICACIÓN DE UNA ALARMA	83
B.6.2.2 COMPORTAMIENTO DE UNA ALARMA	84
B.7 OTROS COMPONENTES	87
B.7.1 CONFIGURACIÓN	87
B.7.1.1 NIVEL 1: CONFIGURACIÓN BÁSICA	88
B.7.1.2 NIVEL 2: CONFIGURACIÓN POR PROPIEDADES.....	88
B.7.2 GESTIÓN DE LOS LOGS	88
B.8 CAPA DE PRESENTACIÓN DE LA APLICACIÓN WEB	91
B.8.1 CABECERA.....	92
B.8.1.1 MENÚ DEL ADMINISTRADOR.....	92
B.8.1.2 MENÚ PARA LOS GESTORES	94
B.8.1.3 OTROS MENÚS.....	96
B.8.2 PIE DE PÁGINA	97
B.8.3 INTERFACES DEL ADMINISTRADOR	97
B.8.3.1 GESTIÓN DEL SISTEMA.....	97
B.8.3.2 GESTIÓN DE LOS USUARIOS.....	99
B.8.3.3 ESTADÍSTICAS DEL USO DEL SISTEMA.....	105
B.8.4 INTERFACES PARA LOS GESTORES	105
B.8.4.1 GESTIÓN DE LOS PROYECTOS	106
B.8.4.2 GESTIÓN DE LOS PARTICIPANTES	110
B.8.4.3 GESTIÓN DE LAS ALARMAS DE UN PROYECTO	112
B.8.4.4 DEDICACIÓN AGREGADA.....	116
B.8.5 INTERFACES COMUNES.....	117
B.8.5.1 ENTRADA A LA APLICACIÓN WEB	117
B.8.5.2 INFORMACIÓN.....	117
B.8.5.3 GESTIÓN DEL PERFIL DEL USUARIO	118

B.8.5.4 PANTALLAZO DE ERROR	121
B.8.6 AYUDA INTEGRADA	121
B.9 CAPA DE PRESENTACIÓN DE LA APLICACIÓN PORTLET	123
B.9.1 CABECERA.....	123
B.9.2 PIE DE PÁGINA	123
B.9.3 INTERFACES PARA LOS USUARIOS.....	123
B.9.3.1 MENÚ PRINCIPAL.....	124
B.9.3.2 PARTICIPACIONES EN LOS PROYECTOS.....	125
B.9.3.3 GESTIÓN DE LAS ALARMAS.....	128
B.9.3.4 GENERAR INFORMES DE LA DEDICACIÓN.....	132
B.9.3.5 GESTIÓN DEL PERFIL DEL USUARIO	133
B.9.3.6 CONTACTO CON EL ADMINISTRADOR	135
B.9.3.7 ACCESO AL PORTLET	135
B.9.3.8 PANTALLAZOS DE ERROR	136
B.9.4 AYUDA INTEGRADA	136
B.10 COPIAS DE SEGURIDAD	137
B.10.1 REALIZAR UNA COPIA DE SEGURIDAD.....	137
B.10.2 RESTAURAR UNA COPIA	137
ANEXO C DOCUMENTO DE TEST DE PRUEBAS	139
C.1 INTRODUCCIÓN.....	141
C.2 CAPA DE DATOS	143
C.2.1 COMPONENTE “DAO”	143
C.2.1.1 TEST DAO HIBERNATE.....	143
BIBLIOGRAFÍA Y REFERENCIAS.....	147

ÍNDICE DE FIGURAS

Figura 3.1 Contexto general del sistema	11
Figura 3.2 Descomposición del sistema	12
Figura 3.3 Despliegue del sistema	14
Figura 3.4 Diagrama Entidad-Relación simplificado	16
Figura 3.5 Dependencias entre componentes	17
Figura 3.6 Diagrama de secuencia del método “actualizarAlarmas”	20
Figura 3.7 Interfaz “Información de un proyecto”	23
Figura 3.8 Interfaz “Listado de los proyectos gestionados”	24
Figura 3.9 Mapa Web de los gestores.....	24
Figura 3.10 Mapa Web del administrador.....	25
Figura 3.11 Menú principal.....	26
Figura 3.12 Mapa Web del Portlet.....	27
Figura 3.13 Interfaz “Listado de las participaciones”	28
Figura 3.14 Interfaz “Introducción de la dedicación diaria”	28
Figura 4.1 Diagrama de Gantt de la planificación.....	32
Figura 4.2 Diagrama de Gantt con el esfuerzo realizado.....	33
Figura 4.3 Distribución de la dedicación real.....	34
Figura A.3.1 Casos de uso relacionados con la aplicación Web	46
Figura A.3.2 Casos de uso relacionados con la aplicación Portlet	47
Figura B.2.1 Secuencia de una petición	53
Figura B.2.2 Despliegue del sistema	54
Figura B.3.1 Diagrama Entidad-Relación de la BD.....	55
Figura B.3.2 Notación del diagrama Entidad-Relación	56
Figura B.4.1 Diagrama de clases	66
Figura B.6.1 Componentes de la capa de negocio.....	74
Figura B.6.2 Diagrama de estado 1 de las alarmas.....	85
Figura B.6.3 Diagrama de estado 2 de las alarmas.....	85

ÍNDICE DE TABLAS

Tabla 2.1 Herramientas y tecnologías	9
Tabla 4.1: Tiempo planificado (horas)	32
Tabla 4.2: Tiempo real dedicado (horas).....	33
Tabla A.1.1 Notación de los requisitos	41
Tabla A.2.1 Requisitos funcionales	43
Tabla A.2.2 Requisitos no funcionales	44
Tabla A.2.3 Requisitos de seguridad.....	44
Tabla B.3.1 Descripción de las entidades	56
Tabla B.3.2 Notación del diseño Relacional	60
Tabla C.2.1 Particiones de equivalencia del Dao	143

1 INTRODUCCIÓN

Este documento contiene la memoria del Proyecto Fin de Carrera que el alumno ha realizado como finalización de sus estudios de Ingeniería en Informática cursados en la Escuela de Ingeniería y Arquitectura [1] (EINA), de la Universidad de Zaragoza [2].

En el apartado 1.1 de este capítulo se describe el contexto en el que se ha desarrollado el PFC. El apartado 1.2 explica las razones que han motivado la realización del PFC. Por último, en el apartado 1.3, se describe la estructura del presente documento.

1.1 CONTEXTO DEL PROYECTO

Durante el siglo XX, las empresas tecnológicas comenzaron a darse cuenta de la importancia de la planificación y la gestión de esfuerzos y recursos. En el contexto de los proyectos software la gestión de estos aspectos puede considerarse como un punto crítico. Las empresas requieren de aplicaciones que les faciliten la gestión del esfuerzo dedicado a sus proyectos, de manera que puedan realizar una mejor planificación. Cada empresa u organización suele tener su propio sistema de gestión, que se adapta a sus propias necesidades.

Por parte del cliente, se propuso la creación de una herramienta para ser utilizada por miembros del grupo GIDHE [3], que forma parte del DIIS [4], de la Universidad de Zaragoza [2]. Esta herramienta debía ajustarse a las necesidades del grupo y tenía que ser fácil de instalar y de utilizar por los usuarios.

La plataforma Web se presenta como una buena solución para este problema. Es fácil de utilizar y un gran número de personas la usan de forma diaria. Los usuarios solo necesitan un ordenador con un navegador Web y una conexión a la red, por lo que no será necesario instalar ninguna otra aplicación. Se planteó como una necesidad que una parte del sistema se desarrollara con tecnología de Portlets [5], la cual facilita a los usuarios la integración en sus portales Web [6].

1.2 MOTIVACIÓN

Los miembros del GIDHE [3] desarrollan proyectos Software en su trabajo diario y requieren de una herramienta que les facilite su gestión. Necesitan poder realizar un control del tiempo empleado en todo el proceso y

tener la posibilidad de generar informes de la dedicación que les permita acumular cierto conocimiento para ayudarles en el control del coste de las tareas y los proyectos con los que trabajan. Por ejemplo, saber que actividades requieren más tiempo o recursos para su realización.

Desde el punto de vista de los participantes en los proyectos, será interesante disponer de una herramienta que les permita introducir su dedicación diaria, que sea fácil de utilizar y rápida de acceder, sin que les supongan costes innecesarios para su utilización, permitiéndoles centrarse en el control de la dedicación. Además les permitirá poder conocer y controlar el número de horas que dedican a cada tarea y proyecto y la realización de informes de su dedicación.

1.3 ESTRUCTURA DE LA MEMORIA

El documento se ha dividido en cinco capítulos y tres anexos. En este capítulo se presenta una introducción al PFC, describiendo el contexto y las motivaciones que han llevado a su desarrollo.

En el segundo capítulo se presenta el análisis realizado al comienzo del PFC. Se describen los objetivos generales y los requisitos que debía cumplir. En el último apartado del capítulo se encuentra un resumen del análisis de tecnologías con una lista de la selección realizada y la justificación de las principales decisiones que se tomaron.

El tercer capítulo comprende el contexto y el diseño del sistema que se ha desarrollado. Describe como se ha realizado el despliegue del sistema y describe cada una de las partes, presentando un ejemplo de su funcionamiento.

En el cuarto capítulo se presentan los aspectos relativos a la gestión del PFC. Se describe el modelo de gestión realizado, la planificación y el coste real de su desarrollo.

En el quinto capítulo el autor ha expresado sus conclusiones finales. Expone las conclusiones técnicas y las conclusiones personales. Además, incluye una serie de posibles mejoras para el PFC.

Además de los cinco capítulos nombrados anteriormente, se han incluido tres anexos que contienen los documentos técnicos que se crearon durante el desarrollo del PFC. El anexo A contiene el “Documento de Análisis de Requisitos” (DAR), el anexo B presenta el “Documento de diseño del Sistema” (DDS) y, por último, el anexo C incorpora el “Documento de Test de Pruebas” (DTP).

2 ANÁLISIS DEL PROBLEMA

En este capítulo se presenta un resumen del análisis realizado al comienzo del PFC con el objetivo de resolver el problema planteado. En el apartado 2.1 se enumeran los objetivos generales del PFC derivados de las necesidades de los usuarios y de la naturaleza del mismo. En el apartado 2.2 se muestran los requisitos concretos correspondientes a los objetivos anteriores. Por último, en el apartado 2.3 se describen las tecnologías empleadas para su desarrollo.

2.1 OBJETIVOS GENERALES

El objetivo de este PFC es el diseño e implementación de un sistema Web que permita a los gestores y participantes en proyectos gestionar de forma sencilla y adecuada el control de su esfuerzo.

De forma más precisa, se pretende:

- Analizar como funciona el control de tiempo en la gestión de los proyectos.
- Diseñar y desarrollar una aplicación Web capaz de:
 - Configurar proyectos, añadir tareas y participantes.
 - Configurar alarmas de la dedicación asociadas a tareas y participantes que muestren un aviso cuando el tiempo máximo especificado haya sido superado.
 - Generar informes sobre la dedicación de los proyectos y los participantes.
 - Gestionar a los usuarios del sistema.
- Diseñar y desarrollar una aplicación Portlet capaz de:
 - Consultar los proyectos.
 - Introducir la dedicación diaria en las tareas de los proyectos.
 - Recibir notificaciones de las alarmas de la dedicación.
 - Configurar informes de la dedicación.
 - Ser fácilmente integrable en un portal Web.

2.2 ESPECIFICACIÓN DE REQUISITOS

Este apartado detalla con mayor precisión los requisitos descritos en los objetivos generales (apartado 2.1). Los requisitos del sistema se han dividido en funcionales y no funcionales. Los primeros describen la funcionalidad y comportamiento del sistema. Los requisitos no funcionales definen otros aspectos como las tecnologías que se deben emplear o el nivel de seguridad que se debe alcanzar.

2.2.1 REQUISITOS FUNCIONALES

Los requisitos para cada aplicación son:

Requisitos desde el punto de vista de la aplicación Web

1. Dar de alta y baja a los usuarios del sistema.
2. Configurar el sistema.
3. Consultar estadísticas de utilización del sistema.
4. Configurar y modificar el perfil de un usuario del sistema.
5. Dar de alta un nuevo proyecto.
6. Establecer la lista de tareas asociadas a un proyecto.
7. Establecer que usuarios registrados en el sistema van a participar en un proyecto concreto.
8. Consultar el listado de proyectos de los cuales es responsable el usuario.
9. Consultar el esfuerzo dedicado en un proyecto aplicando criterios temporales.
10. Consultar el esfuerzo dedicado a cada tarea de un proyecto aplicando criterios temporales.
11. Consultar el esfuerzo que un participante ha dedicado a un proyecto aplicando criterios temporales, según tareas o combinando ambos criterios.
12. Generar informes en formato PDF relativos a la dedicación de un proyecto concreto.
13. Modificar, añadir o dar de baja a los participantes de un proyecto.

14. Configurar alertas de dedicación máxima asociadas a una tarea o a un participante de un proyecto.
15. Detectar e informar al gestor del proyecto cuando se sobrepase el tiempo asignado a una alerta de la dedicación de un proyecto.

Requisitos desde el punto de vista de la aplicación Portlet

1. Configurar y modificar el perfil del usuario.
2. Consultar los proyectos.
3. Consultar las tareas de un proyecto.
4. Introducir la dedicación diaria en una tarea de un proyecto.
5. Consultar la dedicación en un proyecto aplicando criterios temporales, según tareas o combinando ambos criterios.
6. Generar informes en formato PDF relativos a la dedicación en un proyecto concreto.
7. Informar al participante si sobrepasa, en algún momento, alguna de las alertas de dedicación máxima establecidas.

2.2.2 REQUISITOS NO FUNCIONALES

Los siguientes requisitos describen los aspectos que no tienen que ver con la funcionalidad ni el comportamiento del sistema.

1. La parte del sistema de gestión de la dedicación que utilizarán los participantes en los proyectos será desarrollado con tecnología de Portlets.
2. La parte del sistema desarrollada con tecnología de Portlet deberá integrarse en una página Web con un esfuerzo mínimo.
3. El sistema dispondrá de un portal Web para que el administrador o el gestor de proyectos configure su actividad.
4. Al finalizar el PFC deberá estar operativo y poder ser utilizado por al menos el director del proyecto.
5. El acceso al sistema será seguro, por medio de un mecanismo de login y contraseña.

6. El sistema deberá ser capaz de realizar copias de seguridad de manera periódica.
7. El sistema almacenará ficheros de "log" de la actividad del sistema. El formato del fichero deberá ser apropiado para futuros análisis de su contenido.

2.3 ANÁLISIS DE LAS TECNOLOGÍAS

Este apartado pretende describir brevemente las tecnologías elegidas para el desarrollo y despliegue del PFC.

2.3.1 REQUISITOS TECNOLÓGICOS

Para el desarrollo del PFC se han necesitado una serie de tecnologías que resolvieran las distintas necesidades planteadas por los requisitos. A continuación se describen los principales requisitos tecnológicos:

1. Plataforma que permita desarrollar aplicaciones con múltiples capas y componentes. Deberá coordinar las capas y los componentes, así como facilitar la configuración del sistema.
2. Tecnologías que den soporte al almacenamiento y al acceso a la información. Ambas aplicaciones requieren compartir un repositorio permanente de los datos del dominio del problema. Se requiere de un mecanismo que facilite la conexión entre el sistema de almacenamiento y los datos de la aplicación.
3. Tecnologías que permitan desarrollar interfaces Web para los usuarios y que se adapten fácilmente a los navegadores más utilizados.
4. Tecnología Web orientada a servicios que facilite la integración en portales. Deberá facilitar a los gestores la integración de la aplicación en sus portales.
5. Tecnologías que faciliten el despliegue del sistema. Para ello será necesario un servidor Web que permita alojar las aplicaciones.

2.3.2 SELECCIÓN DE TECNOLOGÍAS

A continuación se listan las tecnologías que se han seleccionado en función de los requisitos tecnológicos descritos en el apartado 2.3.1.

Plataforma de desarrollo

Se ha elegido Java Platform Enterprise Edition [7] (JEE) como la plataforma de desarrollo. JEE es una plataforma basada en un lenguaje orientado a objetos, con un amplio repositorio de librerías. Uno de los factores más relevantes para esta decisión es el soporte que facilita para la creación de Portlets [5] (uno de los requisitos del PFC). Dado el soporte que ofrece para la creación de Servlets parece más que conveniente que también se utilice para crear la aplicación Web.

Spring [8] es el framework que ha sido elegido para dar soporte al sistema. Se trata de un enorme y completo marco de trabajo que facilita el desarrollo de aplicaciones Web, dando soporte a muchas otras tecnologías y modelos de proyectos.

Los motivos para elegir a Spring pueden resumirse en tres: el framework favorece la adopción de buenas prácticas de programación, permite construir aplicaciones basadas en componentes con bajo acoplamiento entre sí, gracias al principio de inversión de control (IoC) y la inyección de dependencias [9]. Además, da soporte a un gran número de otras tecnologías (Hibernate [10], Log4j [11],...).

El incentivo de aprender el uso de un nuevo framework con tanta difusión y soporte debería ser suficiente para añadirlo como el cuarto motivo para utilizarlo.

Tecnologías de soporte para el almacenamiento de la información

Como repositorio de la información del sistema se ha utilizado MySQL [12]. Esta decisión se ha tomado en función del SGBD [40] que se encontraba previsto para éste y otros proyectos.

Aunque no se hubiese dado esta situación, MySQL hubiera sido la principal opción para este objetivo. Esta aplicación es un SGBD muy completo, se basa en el lenguaje de consulta estructurado (SQL) con un gran soporte por parte de los programadores y es gratuito.

Tecnologías de mapeo Objeto-Relacional (ORM)

Considerando que la BD será relacional y que la plataforma utilizada está basada en objetos, se debe incluir una tecnología que permita convertir las tuplas en objetos y los objetos en tuplas.

Dada la buena experiencia utilizando Hibernate Framework [10] para este propósito se ha optado por tomarlo como el ORM [13] más adecuado. Como en las asignaturas de la carrera se utilizó esta tecnología con XML, se planteó como reto utilizar anotaciones para el mapeo de las clases.

Tecnologías Web

Desde un punto de vista técnico, un Portlet [5] es una pequeña aplicación que permite, de forma modular, ser añadida a un portal Web. Los Portlets suelen estar orientados a mostrar información a diferencia que otras aplicaciones Web. Se ha optado por tomar la última versión Portlet 2.0 [5] (JSR 286 [14]) por ser más completa que su predecesora (JSR 168 [15]).

Aunque la decisión de utilizar un Portlet ha venido dada por los requisitos del PFC, se puede decir que la decisión de emprender este proyecto se debe al ímpetu de querer aprender esta tecnología. Durante la fase previa al desarrollo se realizó un pequeño estudio de la misma.

Para el despliegue del Portlet es necesaria una aplicación que lo contenga. Se eligió Open Portal Portlet Container [16] como la mejor opción para este fin. Soporta numerosos contenedores Web como Tomcat [17], JBoss [18], Jetty [19], Oracle WebLogic [20] y GlassFish [21]. Para el despliegue de la aplicación Web se seleccionó Apache Tomcat [17], que es un contenedor de aplicaciones Web (Servlets [22] [23]) ligero y fácil de utilizar, que se adapta a las necesidades del PFC.

Tecnologías de desarrollo de interfaces Web

Para el desarrollo de la capa de presentación se decidió utilizar Java Server Pages [24] (JSP) como la solución más sencilla y completa. Permite la creación de HTML de forma dinámica con el código Java incrustado dentro del código de marcado.

Junto con JSP se ha utilizado la tecnología de etiquetas que ofrece Java. Esta tecnología permite crear etiquetas que se pueden añadir dentro del HTML y que se implementan con código Java, facilitando crear código sin emborronar el código HTML.

Para la comunicación asíncrona entre los servidores y los usuarios se ha empleado la tecnología Asynchronous JavaScript And XML (AJAX) [25]. Para ello se ha incorporado la librería Prototype Framework [26]. Este marco de trabajo facilita la adaptación del código de script a los diferentes navegadores (problema de no adaptarse al estándar).

Tabla 2.1 Herramientas y tecnologías	
CATEGORÍA	SOFTWARE/TECNOLOGÍA
Soporte al análisis y diseño del sistema	Edge Diagrammer 4.16 [27] Argo UML 0.32 [28]
Tecnologías de desarrollo	<u>General</u> JEE 6 [7] Apache Commons [29] iText [30] Log4j [11] XML <u>Lógica de negocio y control</u> Spring Framework 3 [8] <u>Datos y mapeo Objeto-Relacional</u> Hibernate Framework 3 [10] Java Validation API C3P0 <u>Presentación</u> HTML [31] CSS [31] AJAX [25] Prototype 1.7 [26] JSP [24] <u>Pruebas</u> JUnit 4 [32]
Software de desarrollo	JDK 6u24a [33] NetBeans 6.9.1 [34]
Despliegue	Apache Tomcat 6 [17] Open Portal Portlet Container 2 [16] Firefox 22.0 [35] Internet Explorer 7 [36]
Almacenamiento y datos de prueba	MySQL 5 [12] MySQL Workbench 5.2 [37]

3 DISEÑO DEL SISTEMA

En este capítulo se presenta el diseño del sistema. El primer apartado muestra el contexto del proyecto. En el apartado 3.2 se describe la arquitectura empleada en el sistema, incluyendo la división del problema y la distribución de cada una de las partes. Los últimos tres apartados (3.3, 3.4 y 3.5) se dedican a la descripción de cada una de las partes en que se ha dividido el sistema e incluye una secuencia del funcionamiento de una de sus operaciones.

3.1 CONTEXTO DEL SISTEMA

Como se observa en la Figura 3.1, el sistema está compuesto por una aplicación Web y una aplicación Portlet. Los usuarios podrán consultar los proyectos en los que participan e introducir su dedicación diaria a través de los distintos portales Web que hayan incorporado la aplicación Portlet entre sus contenidos. El administrador del sistema y los usuarios de tipo gestor compartirán la aplicación Web que les permitirá la gestión del sistema y de los proyectos respectivamente. Ambas aplicaciones compartirán una base de datos, que les permitirá almacenar y recuperar la información del sistema.

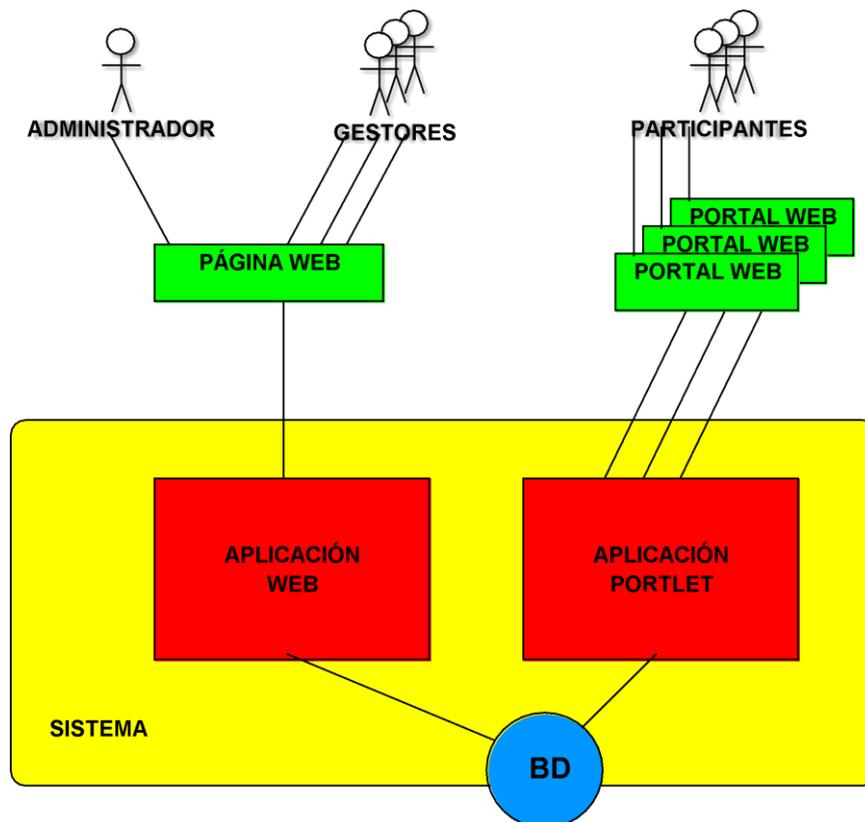


Figura 3.1 Contexto general del sistema

3.2 ARQUITECTURA DEL SISTEMA

La arquitectura del sistema se basa en un diseño multicapa [38]. Los diseños multicapa son muy utilizados en las aplicaciones de gestión ya que permiten separar claramente las tres funciones principales presentes en la mayor parte de este tipo de aplicaciones: tratamiento de los datos persistentes, lógica de negocio e interfaz del usuario. Esto permite reducir las dependencias entre capas facilitando el mantenimiento del sistema, dado que los cambios en una parte del mismo sólo afectarán a la capa concreta, sin necesidad de tener que modificar el resto de capas.

El sistema diseñado consta de tres capas que se pueden ver en la Figura 3.2. Estas capas se corresponden con las tres funcionalidades descritas en el párrafo anterior. Cada capa se construye en términos de la capa inmediatamente inferior. Este modelo se corresponde con el patrón MVC [38] [39], donde el modelo viene dado por la capa de datos, el control por la capa de negocio y la vista por la capa de presentación. Este patrón facilita la división entre capas, encapsulando su comportamiento a través de una interfaz y permite definir la interacción entre ellas.

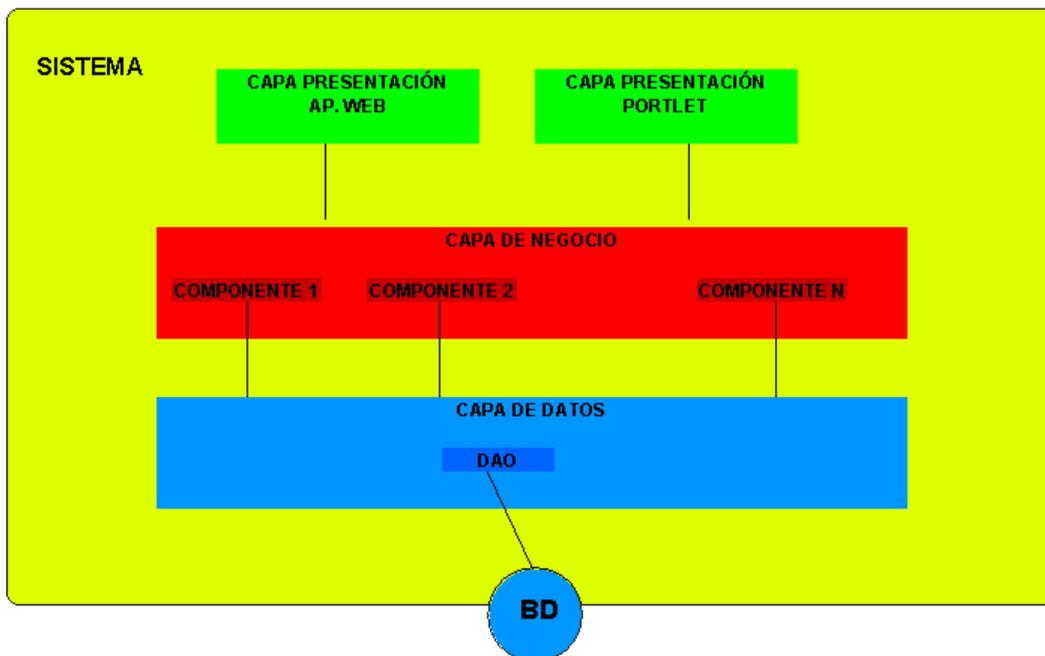


Figura 3.2 Descomposición del sistema

La capa de datos está formada por un componente, el “Data Access Object” (DAO), que se encargará de todas las operaciones relacionadas con la BD, incluido el mapeo Objeto-Relacional entre la BD y los datos de la aplicación. La capa de negocio está formada por varios componentes, cada uno de ellos asume una parte de las funciones del sistema. Los principales objetivos que se han pretendido alcanzar con esta capa son los siguientes:

1. Abstractar a la capa de presentación de las operaciones con los datos.
2. Proveer métodos que encapsulen las funciones con cierto grado de complejidad (funciones compuestas de varias operaciones).
3. Facilitar un medio transaccional que atomice las operaciones.
4. Comprobar que el usuario tiene permiso para realizar una operación sobre ciertos datos.
5. Comprobar que los datos son válidos antes de guardarlos.

La capa de presentación encapsula todos los componentes de la interfaz de usuario: las pantallas de cada aplicación y la lógica para el tratamiento de todos los eventos que pueda generar el usuario en cada pantalla.

Las capas de datos y de negocio son compartidas por ambas aplicaciones (Web y Portlet). El diseño de la capa de presentación será distinto para cada aplicación, dado que las interfaces de cada una, sus pantallas y eventos no son iguales para ambas.

En el diagrama de la Figura 3.2 se puede ver que la capa de presentación no tiene acceso directo a la capa de datos, teniendo que interactuar con la capa de negocio para obtenerlos. También hay que aclarar que la capa de negocio depende de la capa de presentación para lanzar sus operaciones, es esta última la que tiene que lanzar peticiones a la primera para que se actualice la interfaz, esto se debe al contexto Web en el que se desarrollan las aplicaciones. Las capas interactúan de arriba hacia abajo, siendo la capa de presentación la que inicia los procesos y se comunica con la capa de negocio enviándole peticiones. A su vez la capa de negocio se comunica con la capa de datos para acceder a la información almacenada en el repositorio.

3.2.1 DESPLIEGUE DEL SISTEMA

En la Figura 3.3 se puede observar el diagrama de despliegue a nivel de software. Cada nodo, representado por una caja tridimensional, se corresponde con un elemento software. La caja central representa el contenedor Web que almacenará las aplicaciones descritas en los apartados anteriores. El Contenedor Web alojará la aplicación Web y el contenedor de Portlets (“Open Portal”, véase el apartado “2.3 Requisitos tecnológicos”). El contenedor de Portlets alojará a la aplicación Portlet. De esta forma las dos aplicaciones que componen el sistema se ejecutarán en aplicaciones distintas, pero podrán compartir los ficheros de configuración. La caja de la derecha representa el Sistema de Gestión de Bases de Datos (SGBD [40]) que gestionará la BD que almacenará los datos de la aplicación. El sistema está preparado para que

ambos nodos residen físicamente o virtualmente en la misma máquina o por separado, en función de la configuración requerida por el cliente. En cualquier caso, ambos nodos deberán integrarse dentro de la infraestructura que disponga para ello.

Los otros tres nodos no forman parte del proyecto, y deberán ser aportados por terceras personas. Los usuarios de la aplicación vienen representados por los nodos “Navegador Web Gestor / Admin” y “Navegador Web Participante”. Ambos nodos representan los navegadores Web que utilizarán los usuarios para acceder a las aplicaciones. El nodo “Portal” representa el conjunto de portales Web que contendrán la aplicación Portlet entre sus contenidos.

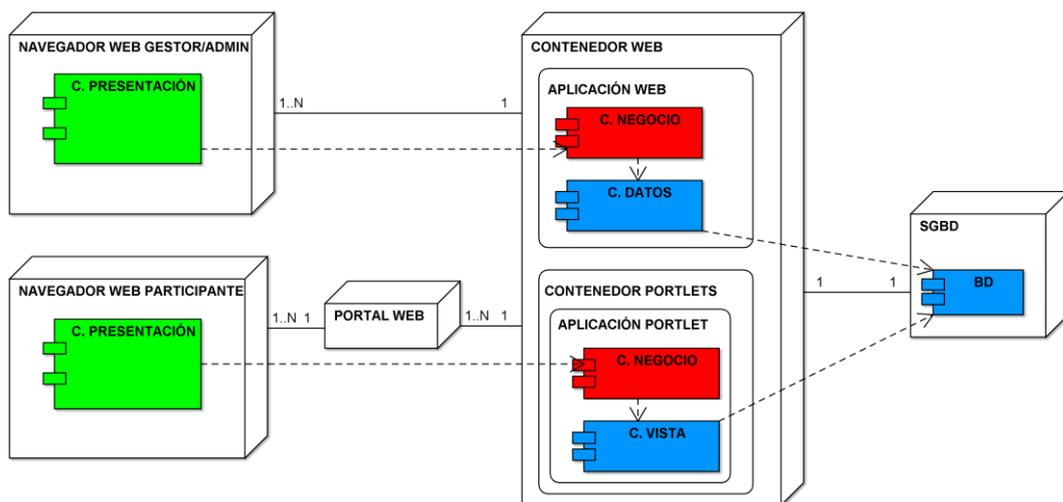


Figura 3.3 Despliegue del sistema

La comunicación entre nodos se representa dentro del diagrama como una línea continua. Los usuarios de tipo gestor y el administrador se podrán comunicar directamente con el contenedor Web para acceder a la aplicación. Por el contrario, los participantes deberán utilizar uno de los portales Web que se pongan a su disposición y éstos, a su vez, se conectarán al contenedor de Portlets. En la parte derecha del diagrama se observa que el nodo “Contenedor Web” se comunica con la base de datos para obtener la información del negocio de la aplicación.

A nivel de componentes, se observa la distribución de las capas nombradas en los apartados anteriores. Cada componente viene representado por un rectángulo relleno con dos tiras en su lado izquierdo, y sus dependencias se denotan por flechas de trazo discontinuo. La capa de presentación se repartirá por los navegadores de los usuarios. Como ya se dijo anteriormente, esta capa se especializa en función de la aplicación que utiliza

el usuario. Las dos capas de presentación son dependientes de la capa de negocio y ésta de la capa de datos. Copias de ambas capas se distribuyen por las dos aplicaciones desarrolladas. Por último, la capa de datos utiliza la base de datos para sus funciones.

3.3 CAPA DE DATOS

La capa de datos está formada por un repositorio relacional, un modelo de datos orientado a objetos y un componente que facilita el acceso y gestión de la información. El repositorio consiste en una base de datos relacional cuyas tablas almacenan la información del negocio del sistema. Cada una de las tablas que la componen se relaciona directamente con una clase de objetos del modelo de datos de la aplicación. Para ello, el componente “Dao” (“Data Access Object”) se encarga de realizar el mapeo Objeto-Relacional y facilita la relación que se produce entre el repositorio y los datos de la aplicación. Las funciones del Dao son:

1. Crear nuevos objetos.
2. Cargar los datos de un objeto.
3. Guardar los datos de un objeto.
4. Borrar un objeto definitivamente del sistema.
5. Realizar listados de los objetos en función de un criterio de búsqueda.
6. Realizar operaciones agregadas sobre los datos (sumar, contar,...)
7. Validar los datos.
8. Facilitar las estadísticas de uso de los datos.

En la Figura 3.4 se puede observar la estructura a nivel conceptual de los datos de la aplicación. Se puede ver la versión completa del diagrama Entidad-Relación en el capítulo “B.3.1 Diseño conceptual”. Los recuadros representan las entidades que se pueden ver como objetos del mundo real. Las relaciones entre entidades se muestran como rombos que se unen con las entidades mediante líneas. Cada relación tiene asociada una multiplicidad que indica la cantidad de objetos de una entidad que participan en la relación.

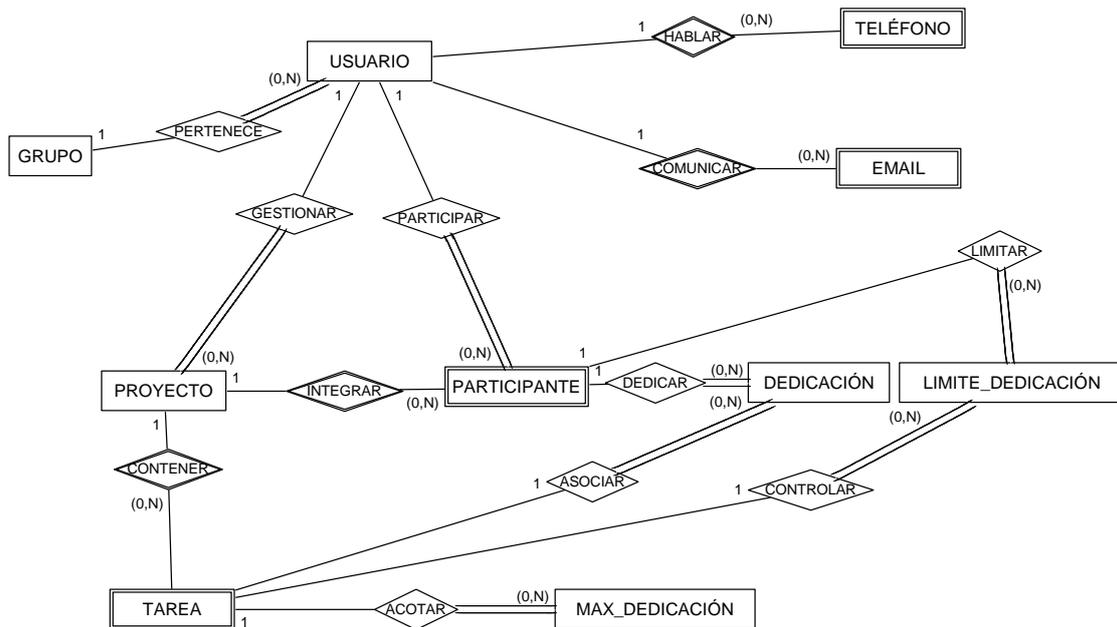


Figura 3.4 Diagrama Entidad-Relación simplificado

Desde un punto de vista funcional, el diagrama se divide en tres partes. Las entidades “Grupo” y “Usuario”, junto con “Teléfono” y “Email”, permiten gestionar las personas físicas que utilizan la aplicación. La entidad “Usuario” se corresponde con los datos personales de una persona real que utiliza el sistema e incluye los datos que le permiten acceder a la misma (login y password). Cada usuario se integra en un grupo de trabajo.

Las entidades “Proyecto”, “Tarea” y “Participante” forman una segunda división que agrupa la información relacionada con los proyectos. Cada proyecto estará asociado con un conjunto de participantes y dispondrá de una lista de tareas que se puedan realizar dentro del mismo. Los usuarios de tipo gestor se relacionan con los proyectos que gestionan a través de la entidad “Proyecto”, mientras que los usuarios participantes en los proyectos se relacionarán con ellos a través de la entidad “Participante”.

Por último, las entidades “Dedicación”, “Limitededicación” y “Maxdedicación” representan la información relacionada con la gestión y control del tiempo dedicado por los participantes en las tareas de los proyectos. La entidad “Dedicación” permite a los participantes almacenar el tiempo que han dedicado en una tarea de un proyecto durante un día concreto. Las otras dos entidades permiten configurar alarmas que permitan avisar al gestor y a los usuarios de que el tiempo establecido para una tarea ha sido superado.

Para el diseño de los objetos se ha seguido la filosofía “Plain Old Java Object” (POJO) [41] que se basa en la idea de crear clases simples donde los objetos tienen sus atributos ocultos y se utilizará un conjunto de métodos públicos para su acceso, evitando añadir otro tipo de métodos. Esto permite simplificar el manejo de los datos, evitando el uso de frameworks o la implementación de complejas interfaces.

3.4 CAPA DE NEGOCIO

La capa de negocio se encarga de realizar las distintas funciones de la lógica de negocio que ofrece el sistema. Se ha dividido en seis componentes, cada uno de ellos cubre un conjunto de funciones relacionadas entre sí. La Figura 3.5 muestra las dependencias entre los componentes.

Cada componente dispone de una interfaz que le facilita la comunicación con otros componentes. En la Figura 3.5 los componentes se encuentran representados por rectángulos con marcas en su lado izquierdo. Las dependencias entre componentes se indican con flechas de trazo discontinuo, el componente desde el que sale la flecha depende del componente al que apunta.

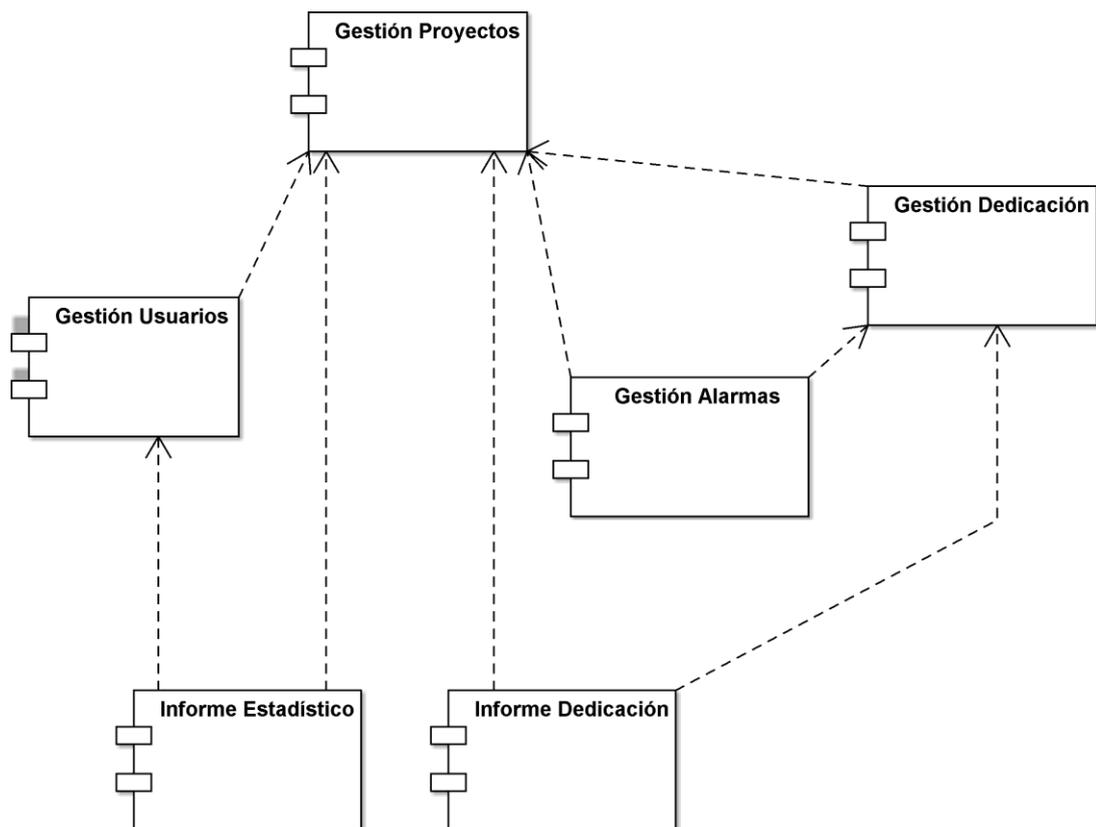


Figura 3.5 Dependencias entre componentes

El componente “Gestión Proyectos” permite a los gestores configurar sus proyectos, definir las tareas y dar de alta y baja a los usuarios participantes en los mismos. Los usuarios pueden consultar la información de los proyectos en los que participan. Éste es el componente central, todos los demás dependen de él para realizar parte de sus tareas.

El componente “Gestión Usuarios” se encarga de la gestión de las cuentas de los usuarios y facilita la configuración de su perfil. Permite al administrador del sistema añadir, modificar, dar de baja y alta a los usuarios, también puede crear nuevos grupos, que permiten distribuir a los usuarios. Los demás usuarios podrán configurar su perfil (nombre, dirección, teléfonos, direcciones de correo electrónico,...).

Los participantes en los proyectos pueden introducir su dedicación diaria en las tareas que realizan utilizando el componente “Gestión Dedicación”. Permite la introducción de las horas dedicadas a una tarea por un usuario que participa en un proyecto. También facilita calcular la dedicación que se ha realizado en una tarea y/o por un participante de un proyecto.

El componente “Gestión Alarmas” facilita la configuración de las alarmas de la dedicación. Los gestores podrán consultar, crear y modificar las alarmas de sus proyectos. Los participantes podrán consultar las alarmas que estén relacionadas con ellos.

Los cuatro componentes descritos se relacionan entre ellos con el objeto de intercambiar los datos que necesitan y poder comprobar ciertas restricciones. Por ejemplo, cuando se introduce una dedicación es necesario comprobar que exista el proyecto, la tarea y que el usuario participe en el proyecto. Para este objetivo el componente “Gestión Dedicación” deberá comunicarse con “Gestión Proyectos” para obtener esta información. Además de las funciones descritas anteriormente, cada componente puede realizar búsquedas filtradas por distintos parámetros, contar los elementos que cumplen una condición y realizar funciones de agregación.

El componente “Informes Estadísticos” permite a los gestores y los participantes en proyectos configurar informes de la dedicación que realizan. El componente es capaz de recolectar la información comunicándose con otros componentes en función de la configuración del informe pedido por el usuario. Por ejemplo, se puede crear un informe de la dedicación de cada semana entre dos fechas que ha realizado un participante en un proyecto en una determinada tarea.

Por último, el componente “Informes Estadísticos” es utilizado por el administrador para crear informes sobre el uso del sistema. Esta información puede incluir el número de usuarios en el sistema o las veces que se accede a una tabla de la BD.

3.4.1 EJEMPLO DE FUNCIONAMIENTO

Para ilustrar el funcionamiento de la capa de negocio se presenta un ejemplo de interacción. La Figura 3.6 muestra un diagrama de secuencia formado por distintos componentes que interactúan entre sí. Los componentes están representados por líneas discontinuas verticales que en su parte superior muestran un recuadro con su nombre. La línea vertical se convierte en un rectángulo del color del componente mientras se procesa un mensaje recibido desde otro componente (desde que se inicia hasta que se finaliza el proceso). Las flechas continuas horizontales representan los mensajes que se pasan entre ellos y las flechas discontinuas horizontales representan la finalización del subproceso iniciado por la recepción de un mensaje (en algunos casos se ha omitido por motivos de simplificación), estas flechas pueden incluir una referencia a los datos devueltos por el mensaje. La secuencia del proceso se ordena cronológicamente según se desciende verticalmente.

El mensaje “actualizarAlarmas” forma parte de la interfaz del componente “Gestión de las alarmas”. Este mensaje se produce después de que un participante introduce su dedicación en un proyecto y se utiliza para comprobar las alarmas de la dedicación afectadas. Por ejemplo, si se ha establecido un límite de 30 h. para la realización de una tarea en un determinado mes, y hasta el momento la tarea ha llevado 27 h. de trabajo, si un participante añade 5 h. más, la suma total será de 32 y el sistema deberá cambiar el estado de la alarma y mostrar un aviso para que los usuarios se den cuenta de que el tiempo máximo establecido ha sido superado.

En la Figura 3.6 se pueden ver los componentes que participan en el diagrama de secuencia. En los extremos se localizan las capas de presentación (verde) y de datos (azul) cuyos componentes han sido abstraídos. Entre ellas se encuentran “Gestión Alarmas”, “Gestión Proyectos” y “Gestión Dedicación” (en rojo) componentes de la capa de negocio (que fueron presentados en la Figura 3.5). Por su mayor papel en el proceso, el componente “Gestión Alarmas” se ha dividido en tres de los objetos que lo componen.

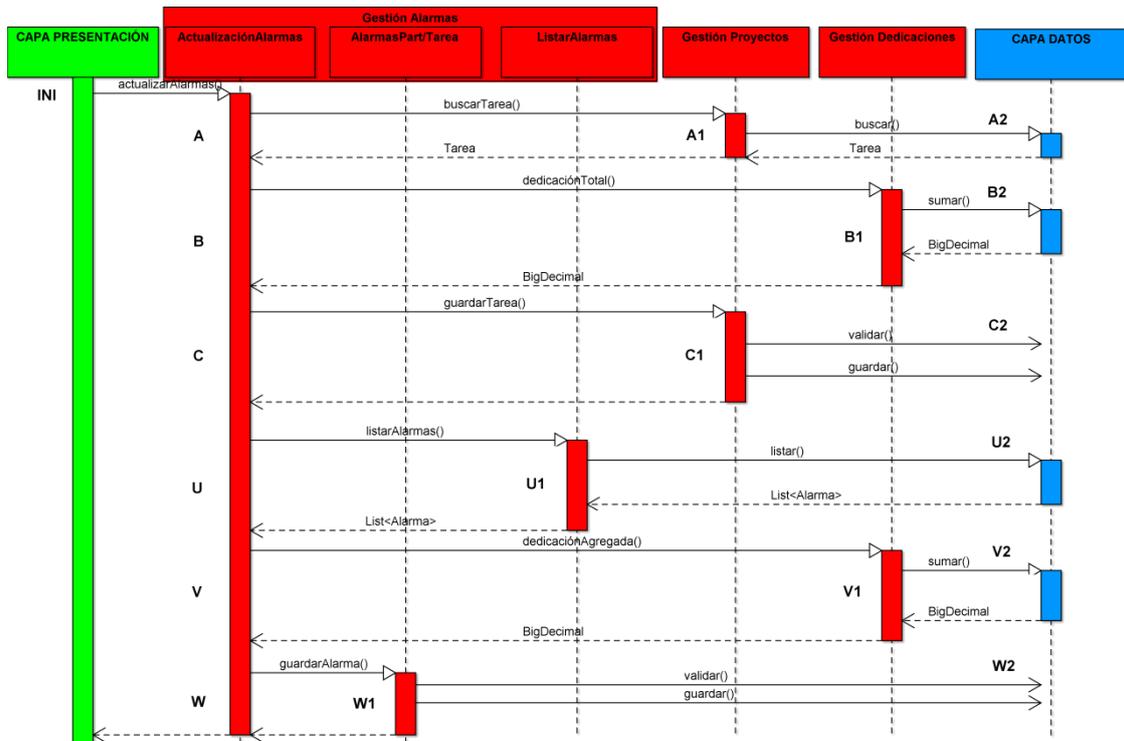


Figura 3.6 Diagrama de secuencia del método “actualizarAlarmas”

La capa de presentación siempre se encarga de iniciar los procesos del sistema. En este ejemplo le envía un mensaje a “Gestión Alarmas” de la capa de negocio (INI). Internamente, el objeto “ActualizaciónAlarmas” se encarga de gestionar la petición recibida por la interfaz del componente (por motivos de representación, la flecha apunta directamente sobre el objeto, en vez del componente).

El proceso se realiza en dos fases divididas en tres pasos cada una. La primera fase pretende actualizar la alarma que controla toda la dedicación de la tarea en la que el participante añadió su dedicación. La segunda fase se dedica a actualizar las alarmas que tienen una duración limitada a un mes o una semana.

La primera fase se divide en tres pasos (A, B y C). Primero se realiza la búsqueda de los datos de la tarea, para ello se envía un mensaje al componente “Gestión Proyectos” (A1) que tiene los mecanismos para llevarla a cabo, éste prepara la búsqueda y delega en la capa de datos para acceder al repositorio (A2). Una vez se haya recuperado la información se envía hacia atrás (flecha discontinua). Después de comprobar que la alarma se encuentra activada (en cualquier otro caso se saltarían los pasos siguientes), se le pide al componente “Gestión Dedicación” que calcule el número total de horas empleadas en la tarea (B1), entonces el componente delega en la capa de datos (B2) y el valor es devuelto sucesivamente hasta el principio. En ese momento se puede comparar el valor máximo asignado a la tarea con el

número de horas reales que se han empleado. Entonces, si es necesario, se modifica el estado de la alarma y se procede a guardarla (C), para ello se inicia una secuencia parecida a la de la búsqueda (C1 y C2).

La segunda fase también se divide en tres pasos (U, V, W) con funciones parecidas a las anteriores. Primero se le pide al objeto “Alarmas Participantes” (de forma interna al componente) la lista de alarmas semanales y mensuales que se encuentren activadas (U1), entonces utilizará la capa de datos para este fin (U2), este paso solo se realizará una vez. A continuación los pasos V y W se repetirán para cada alarma que se haya devuelto en el paso anterior. El paso V calculará la dedicación realizada en la tarea durante el periodo de tiempo indicado por la alarma enviando un mensaje al componente “Gestión Dedicación” (V1), que como anteriormente) se comunicará con la capa de datos (V2). En caso de que no se modifique el estado de la alarma se seguirá con la siguiente, por el contrario se pasará al paso W que le pedirá al objeto “Alarmas Participante” (W1) que guarde el nuevo estado, entonces se comunicará con la capa de datos (W2). Los pasos V y W se repetirán para cada alarma que se haya encontrado en el paso U.

En resumen, las distintas partes que conforman el sistema dependen entre ellas para la realización de sus tareas. La capa de presentación utiliza a la capa de negocio (INI) y ésta utiliza a la capa de datos (A2, B2,..., W2). Dentro de la capa de negocio, cada componente está especializado en un conjunto de tareas, para desempeñarlas necesitan de otras tareas que no son capaces de realizar y para ello envían mensajes a los componentes que se han especializado en ellas (A1, B1,..., W1). Todos los componentes acaban comunicándose con la capa de datos (A2, B2,..., W2) que se encarga de conectar con el repositorio de los datos.

3.5 CAPA DE PRESENTACIÓN

La capa de presentación ofrece la interfaz gráfica de usuario (GUI). Captura los eventos que produce el usuario (pulsar un botón, desplegar una lista de selección,...), realiza un pequeño filtrado de los datos introducidos y envía un mensaje a la capa de negocio para que realice las operaciones oportunas. Cuando la capa de negocio termina sus tareas y le da una respuesta, entonces se genera la nueva interfaz con los parámetros indicados por la capa de negocio. El funcionamiento de esta capa viene dado por la página en la que se encuentra el usuario, los eventos que éste genera y por el control de la capa de negocio que se encarga de indicarle que debe mostrar y de restringir que se puede hacer en cada página. La capa ha sido dividida en dos partes, una para la aplicación Web y otra para el Portlet. La primera

permite el acceso a los gestores y al administrador y la segunda facilita la gestión de la dedicación de los participantes en los proyectos.

La capa de presentación genera dinámicamente las páginas que los usuarios pueden ver. Ambas aplicaciones codifican las páginas en Unicode, permitiendo un amplio abanico de caracteres, y pueden dar soporte a múltiples idiomas. El administrador puede configurar distintos parámetros que modifican las páginas, como el número de entradas que se muestran en una tabla o los valores por defecto que filtran las búsquedas. También puede configurar, para cada tipo de usuario, el tema que define el aspecto de las páginas, incluyendo las imágenes utilizadas y las hojas de estilo. Los usuarios podrán configurar sus datos personales, incluyendo el idioma con el que desean que se muestren las páginas.

3.5.1 APLICACIÓN WEB

Los usuarios de tipo “Gestor” y el administrador del sistema podrán utilizar un navegador para visualizar la página de acceso de la aplicación Web, en ella podrán acceder a las demás páginas identificándose con su login y contraseña. Cada tipo de usuario dispone de sus propias páginas Web, especializadas en las funciones que debe desarrollar.

En ambos casos la estructura de las páginas se compone de una cabecera y de un pie de página, entre los dos se encuentra el contenido de la sección. La cabecera contiene tres menús y un botón para ver la ayuda. Los menús se configuran en función del tipo de usuario y la sección en la que se encuentre.

La Figura 3.7 muestra la imagen de un navegador Web con una página que contiene la información de un proyecto gestionado por un usuario. La parte superior de la página muestra la cabecera, arriba a la izquierda se muestra el nombre de la aplicación y a la derecha el login del usuario con el menú auxiliar, que contiene tres botones para ver la información de contacto, modificar la información del perfil del usuario y poder salir de la aplicación. Un poco más abajo se ven dos filas de pestañas, la fila superior (algo mayor) es el menú principal que permite acceder a las distintas secciones de la aplicación. Las pestañas que se encuentran debajo son el menú secundario, que contiene enlaces para ir a las distintas páginas donde el usuario puede realizar ciertas operaciones sobre el proyecto mostrado, como son añadir nuevas tareas y/o participantes o ver el listado de participantes del proyecto. El menú principal se personaliza para cada tipo de usuario y nunca se modifica. Por el contrario, el menú secundario se adaptará a la sección en la que se encuentre el usuario.

En la Figura 3.8 se puede observar que el menú secundario se encuentra vacío, debido a que el listado de proyectos no requiere de ninguna opción. Por último las cabeceras disponen de un botón para acceder a la ayuda, el usuario puede consultarla en cualquier momento y moverse por las distintas páginas que ofrece y pulsar el botón para volver sin perder la información que previamente hubiera introducido en los formularios de la página en que se encontrase.

La opción de emplear una cabecera para albergar los menús frente a la alternativa de un menú lateral le ofrece mayor espacio al resto del contenido de la página, facilita la utilización de pestañas que permiten mostrar todas las opciones de un menú sin que se salgan de la ventana, son bastante estéticas, se emplean en muchos sitios Web y son muy fáciles de utilizar por parte de los usuarios.

Debajo de la cabecera se encuentra el contenido de la página. En la Figura 3.7 se puede ver la información del proyecto “1002” llamado “Analizador Léxico”, debajo se dispone de botones para realizar operaciones como cerrar o reabrir el proyecto, seguido de su lista de tareas, la cual se extiende fuera de la parte visible de la ventana. El contenido de la página de la Figura 3.8 muestra el listado de los proyectos que gestiona el usuario donde puede ir avanzando y retrocediendo en la lista utilizando los botones marcados con los símbolos “<<” y “>>”, también puede modificar los parámetros de la lista con el formulario dispuesto a tal fin. Cada entrada de la lista dispone de un enlace que lleva a la sección que muestra la información del proyecto (ver Figura 3.7) y botones para realizar distintas operaciones con los proyectos. Otras páginas de la aplicación Web contienen formularios para introducir datos en el sistema o configurar informes.

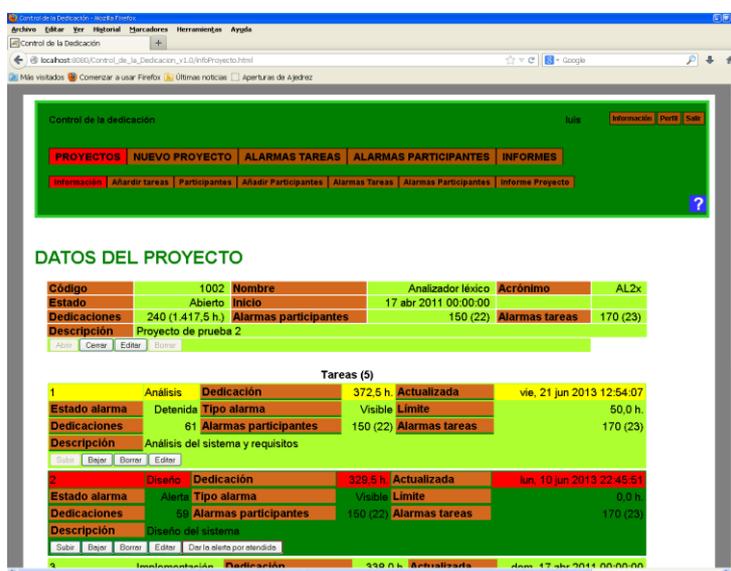


Figura 3.7 Interfaz “Información de un proyecto”

En la parte inferior de la página de la Figura 3.8 se encuentra el pie de página donde el administrador, mediante la configuración del sistema, puede mostrar una dirección de correo que permita a los usuarios reportarle problemas en el sistema.



Figura 3.8 Interfaz “Listado de los proyectos gestionados”

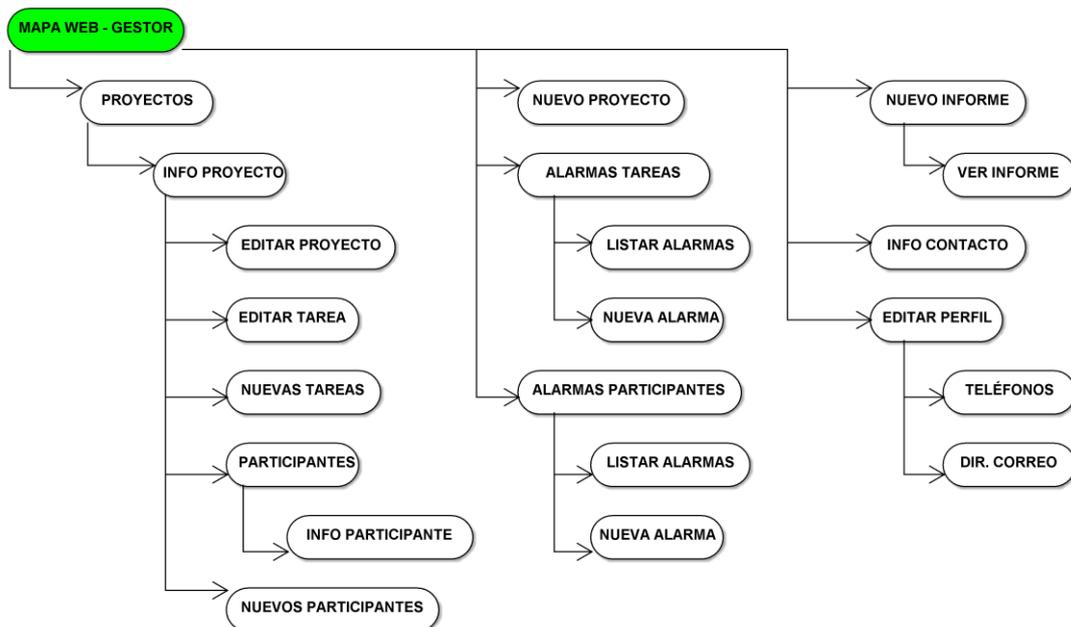


Figura 3.9 Mapa Web de los gestores

Las Figuras 3.9 y 3.10 muestran los mapas Web de las distintas páginas de los gestores y el administrador del sistema respectivamente. Desde el nodo inicial (en color verde) los usuarios pueden utilizar sus menús principales para acceder a una de las páginas, una vez en ella se puede utilizar un enlace o las pestañas del menú secundario para seguir navegando. Por ejemplo, un gestor puede editar una tarea entrando en “Proyectos” con el menú principal, seleccionar el proyecto en la lista accediendo a “Info Proyecto” y pulsar el botón de edición para acabar en “Editar tarea”. De igual manera, el administrador puede utilizar la pestaña del menú para acceder a “Lista usuarios” y, utilizando el menú secundario, acceder a “Nuevo Grupo” y rellenar el formulario para crear un nuevo grupo de usuarios.

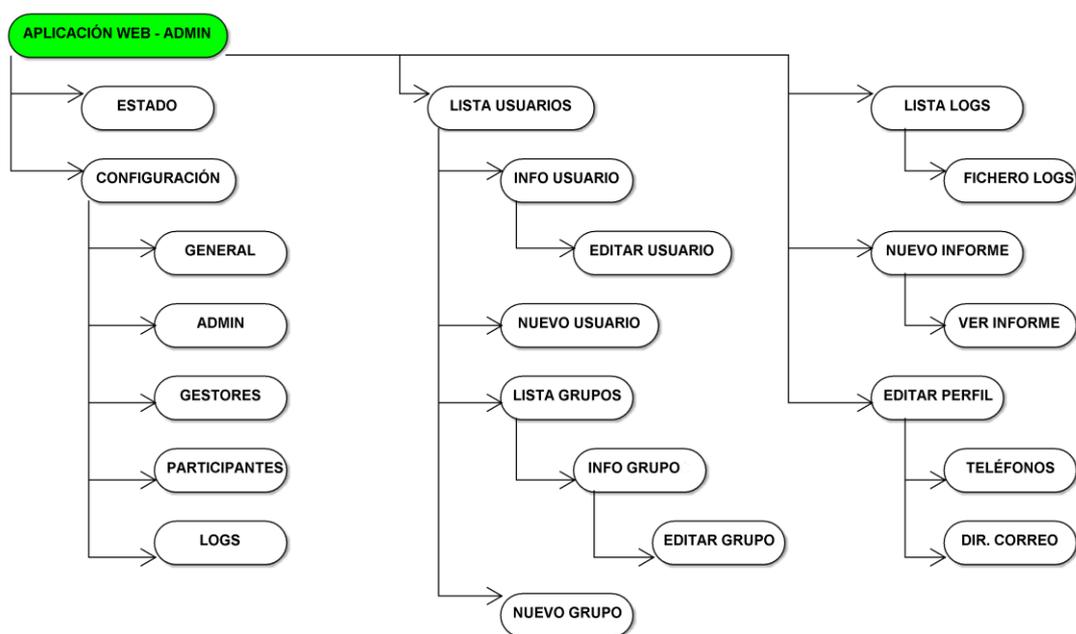


Figura 3.10 Mapa Web del administrador

3.5.2 APLICACIÓN PORTLET

La aplicación Portlet permite gestionar la dedicación de los participantes en los proyectos, esto incluye, sin ningún tipo de diferencia, a los usuarios de tipo gestor y de tipo normal.

Los Portlets van dentro de otras páginas Web (portales), por ello el espacio que disponen suele ser algo menor. Para reducir el tamaño de la cabecera se ha creado una sección “Menú” (ver Figura 3.11) que incluye un menú principal que facilita el acceso a las demás secciones. La aplicación Portlet también incluye un sistema de ayuda integrada, similar al de la aplicación Web, que podrá ser consultado por los usuarios a través del botón azul con un interrogante.

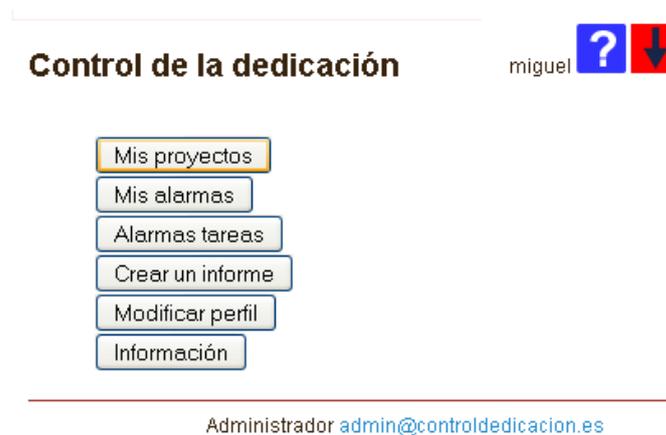


Figura 3.11 Menú principal

En la Figura 3.12 se ve el mapa Web del Portlet. Se puede observar que, como ya se ha dicho, se accede a todas las secciones a través de “Menú”. Por ejemplo, un usuario puede ir a la lista de proyectos (“Mis proyectos”), seleccionar uno de ellos (“Proyecto”) y entrar a la página donde puede introducir su dedicación diaria (“Mi dedicación”).

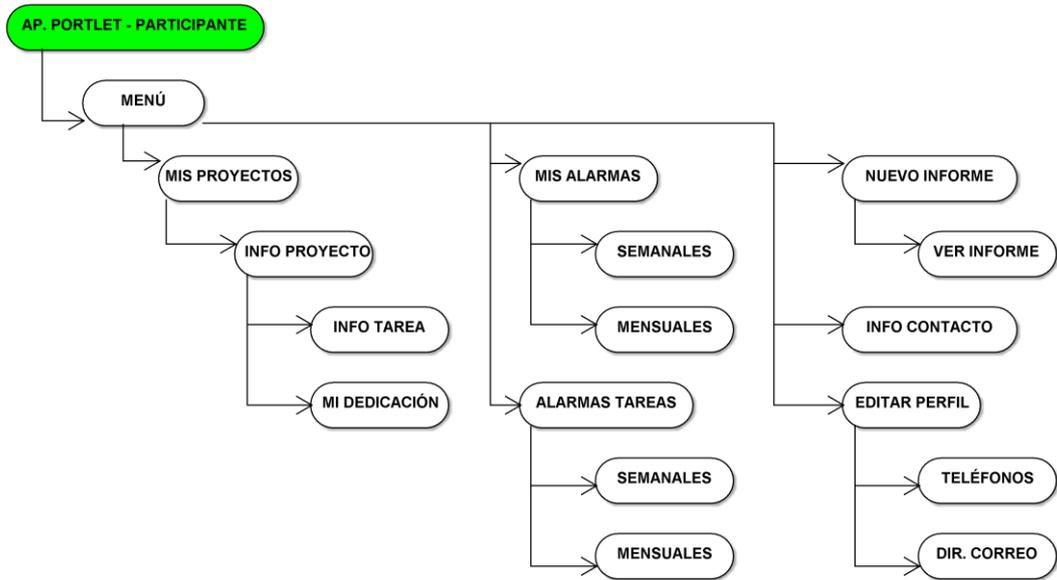


Figura 3.12 Mapa Web del Portlet

Las Figuras 3.13 y 3.14 muestran dos ejemplos de GUI del Portlet. La cabecera de cada página contiene un botón para ver la ayuda y otro para salir de la aplicación y de una línea con la ruta de navegación que ha seguido el usuario, que le informa donde se encuentra y que puede utilizar para regresar a las secciones superiores. Por ejemplo, en la Figura 3.14 la ruta es “Menú/Mis Proyectos/Dedicación”. El contenido de la página de la Figura 3.13 muestra la lista de proyectos en los que participa el usuario. La Figura 3.14 muestra la lista de tareas de un proyecto y permite al usuario introducir su dedicación diaria, se puede seleccionar un día, ver la dedicación total del día, del usuario, de las tareas, etc.

Activos [No activos](#)

/ 3

#	Código	Nombre	Acrónimo			
1	1002	Analizador léxico	AL2x			
2	1004	Simulador $\alpha\beta\gamma$	Sim $\alpha\beta\gamma$			
3	8003	Ajedrez 	Aj 			

Administrador admin@controdedicacion.es

Figura 3.13 Interfaz “Listado de las participaciones”

[Información](#) [Tareas](#) [Dedicación](#) [Alarmas](#) [Crear informe](#)

8003 - Ajedrez  - Aj 

Se han guardado 3 dedicaciones.

/ /

#	tarea	Dedicación	Agregada	Total
1	Mov. 	<input type="text" value="1,0"/>	h. 1,0 h.	1,0 h.
2	Aperturas  e4	<input type="text" value="2,0"/>	h. 2,0 h.	2,0 h.
3	Finales   contra 	<input type="text" value="1,5"/>	h. 1,5 h.	1,5 h.
		4,5 h.	4,5 h.	4,5 h.

Administrador admin@controdedicacion.es

Figura 3.14 Interfaz “Introducción de la dedicación diaria”

4 GESTIÓN DEL PROYECTO

Este capítulo describe el proceso seguido para la gestión del PFC. El apartado 4.1 explica la metodología empleada para su desarrollo. En el apartado 4.2 se expone la planificación realizada en las primeras etapas del desarrollo. Por último, el capítulo 4.3 muestra el esfuerzo real realizado, comparando los tiempos empleados con los estimados y justificando sus diferencias.

4.1 METODOLOGÍA

Para la realización de este PFC, se ha seguido una metodología de desarrollo iterativo e incremental [42]. En este proceso, el desarrollo del software se divide en una serie de iteraciones, cada una de ellas cumple con unos objetivos definidos. Cada iteración incrementa la funcionalidad del sistema o, visto de otra manera, cubre algunos de los requisitos especificados para el PFC. Desde las primeras etapas del proceso el cliente puede ver su aplicación con la mínima funcionalidad y en cada iteración puede comprobar cómo va creciendo.

Como paso previo al desarrollo del PFC se realizó un proceso de formación y selección de tecnologías y herramientas. Principalmente se centró en la tecnología de Portlets. El proceso terminó con una presentación ante el director del PFC sobre los Portlets.

Tras el proceso de formación y de búsqueda de tecnologías, se comenzó con las fases de análisis y diseño del sistema. Primero se realizó el análisis que permitió definir los requisitos funcionales y no funcionales del problema. Posteriormente se realizó el diseño en alto nivel de la solución. Se incluyó la arquitectura del sistema, el modelo de los datos, los componentes con sus interfaces y se modeló la interfaz gráfica de los usuarios.

En la fase de diseño se tomó la decisión de dividir el desarrollo en ocho iteraciones, planificando el orden que llevaría su inicio y fin. Cada iteración fue realizada en un modelo en cascada en cuatro fases. Primero se refinaba el análisis y el diseño de la parte que se acometía, definiendo de forma más precisa sus requisitos y detallando con más precisión los componentes, luego se acometía la implementación de las funcionalidades requeridas y, por último, se realizaba un juego de pruebas unitarias y de integración. Durante este proceso se fue realizando parte de la documentación, que finalmente se juntó en tres documentos que se han añadido en los anexos de la memoria. Estos

documentos son el “Análisis de Requisitos”, el “Documento de Diseño del Sistema” y el “Documento de Test de Pruebas”.

En la primera iteración se crearon la base de datos, la estructura de los datos del negocio y el “Data Access Object”. Estos pasos no añadían funcionalidad directa a la aplicación, pero eran imprescindibles para que los demás componentes, que se abordaban en otras iteraciones, pudieran funcionar. También se desarrolló el esqueleto de las aplicaciones y sus interfaces, incluyendo el sistema que facilitaba el acceso a los usuarios y un sistema de apoyo a la depuración que permitía iniciar las aplicaciones como si un usuario hubiera accedido al sistema y estuviese en una página concreta. En este momento un usuario podía utilizar la aplicación sin ningún contenido ni función. Durante el análisis y diseño realizado en esta iteración se valoraron dos posibles riesgos, los problemas podían venir en el mapeo Objeto-Relacional y en la configuración de Hibernate, ambos eran puntos críticos que afectarían al funcionamiento de otros componentes, por ello se fue lo más cuidadoso posible en este aspecto.

La segunda iteración pretendía cubrir la gestión de los proyectos y las alarmas que realizan los gestores. Para ello se empezó desarrollando las funciones necesarias de la capa de negocio y se adaptaron las interfaces de los usuarios de la capa de presentación. Al terminar esta iteración un usuario de tipo gestor podía manejar sus proyectos y añadirles tareas y alarmas.

En la tercera iteración se cubrieron los requisitos de la aplicación Portlet con respecto a la consulta de los proyectos, tareas, alarmas y la gestión de su dedicación. Igual que en la iteración anterior, primero se desarrolló la capa de negocio y luego se desarrollaron las interfaces de los usuarios. Al terminar la iteración, los usuarios del sistema podían gestionar su dedicación en los proyectos en los que participaban.

La iteración 4 se dedicó a la generación de los informes de la dedicación. Tras una reunión con el cliente, se realizó una etapa de análisis que concluyó con el diseño de dos modelos de informes, uno para los proyectos y otro para los participantes. Tras el desarrollo del componente encargado de estos informes se añadieron las funciones necesarias en las interfaces de ambas aplicaciones para permitir su configuración y generación. Esta iteración requirió algo más de tiempo de análisis y diseño para su desarrollo, incluyendo la necesidad de aprender a utilizar las tecnologías de creación de documentos de tipo “Portable Document Format” (PDF). En este momento se habían llevado a cabo todas las funciones imprescindibles del sistema, que incluían todo lo relacionado con la gestión de la dedicación y las páginas Web necesarias para su utilización.

La quinta iteración se dedicó a la gestión de los usuarios y de sus datos personales. Para este objetivo se desarrollaron las funciones de la capa de negocio y de presentación que permitían al administrador del sistema manejar los usuarios y grupos y que permitía a todos los usuarios modificar sus datos personales.

La sexta iteración se dedicó a la realización de los dos componentes de apoyo al sistema. El primero permitió el acceso y modificación de la configuración de las aplicaciones y el segundo abordó la gestión de los ficheros de logs.

En la séptima iteración se realizó la generación de informes estadísticos del sistema. Con los datos de las reuniones con el cliente se diseñó un modelo de informe y se adaptaron las interfaces de la aplicación Web.

Con todas las funcionalidades del sistema terminadas, la octava iteración se dedicó a la mejora de las interfaces de los usuarios. Se prestó especial atención para que el código que se ejecutaba en los ordenadores de los usuarios fuera compatible con los navegadores, en especial con Internet Explorer.

Tras la realización de todas las iteraciones el cliente podía ver cada una de las aplicaciones terminadas. Entonces se comenzó con las fases de pruebas y aceptación, la documentación y el despliegue del sistema construido durante este PFC.

4.2 PLANIFICACIÓN

En las primeras etapas del desarrollo del PFC se realizó la planificación del mismo. Para cada tarea se estimó el tiempo necesario para su realización. Cada una de las iteraciones se diseñó con un modelo en cascada con cuatro tareas: análisis, diseño, implementación y pruebas.

El proceso de planificación sirvió como una guía para saber cómo iba el desarrollo del PFC. Para su mejor entendimiento se realizó un diagrama de Gantt como se puede ver en la Figura 4.1 (el tiempo está escalado en semanas). Se decidió dedicar 6 h. al día y los cinco días laborales de la semana, con lo que se considera que una semana equivale a 30 h. de trabajo.

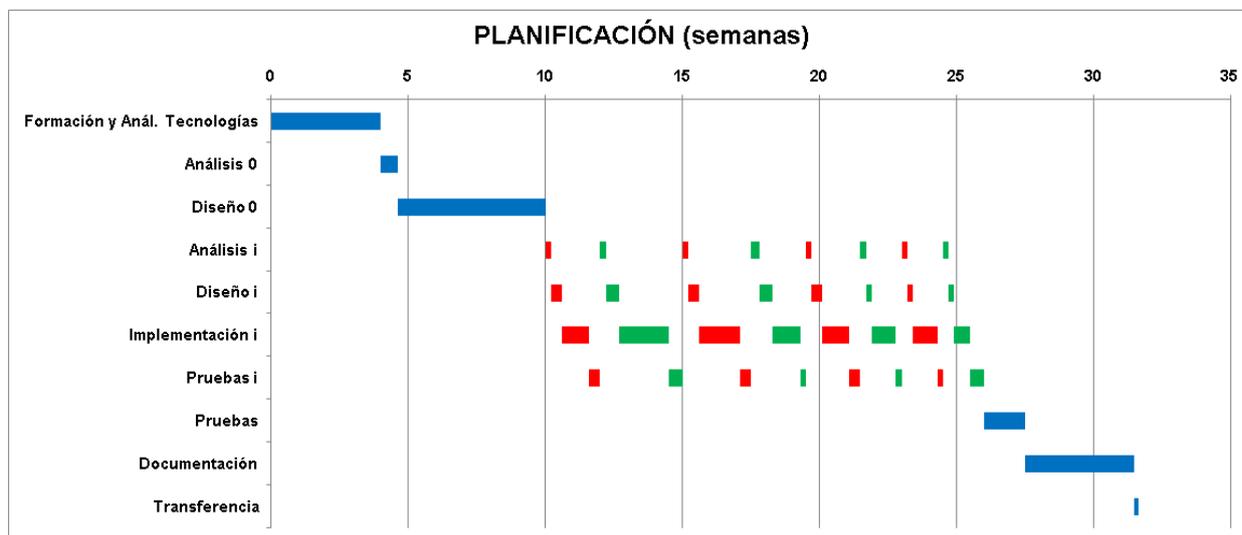


Figura 4.1 Diagrama de Gantt de la planificación

A continuación, la Tabla 4.1 muestra el desglose de tiempo planificado para cada tarea.

Tabla 4.1: Tiempo planificado (horas)												
Tarea	Inicio	Iteración								Final	Total	
		1	2	3	4	5	6	7	8			
Formación y Anál. Technolog.	120											120
Análisis	18	6	6	6	9	6	6	6	6		69	
Diseño	162	12	15	12	15	12	6	6	6		246	
Implementación		30	54	45	30	30	27	27	18		261	
Pruebas		12	15	12	6	12	6	6	15	45	129	
Documentación										120	120	
Transferencia										5	5	
Total	300	60	90	75	60	60	45	45	45	170	950	

4.3 ESFUERZO REAL DEDICADO

Durante la realización del PFC se fue tomando valores del tiempo diario dedicado a la tarea que se realizaba. Esto ha permitido la realización de un segundo diagrama de Gantt con la distribución del esfuerzo realizado. (Figura 4.2, también escalado en semanas).

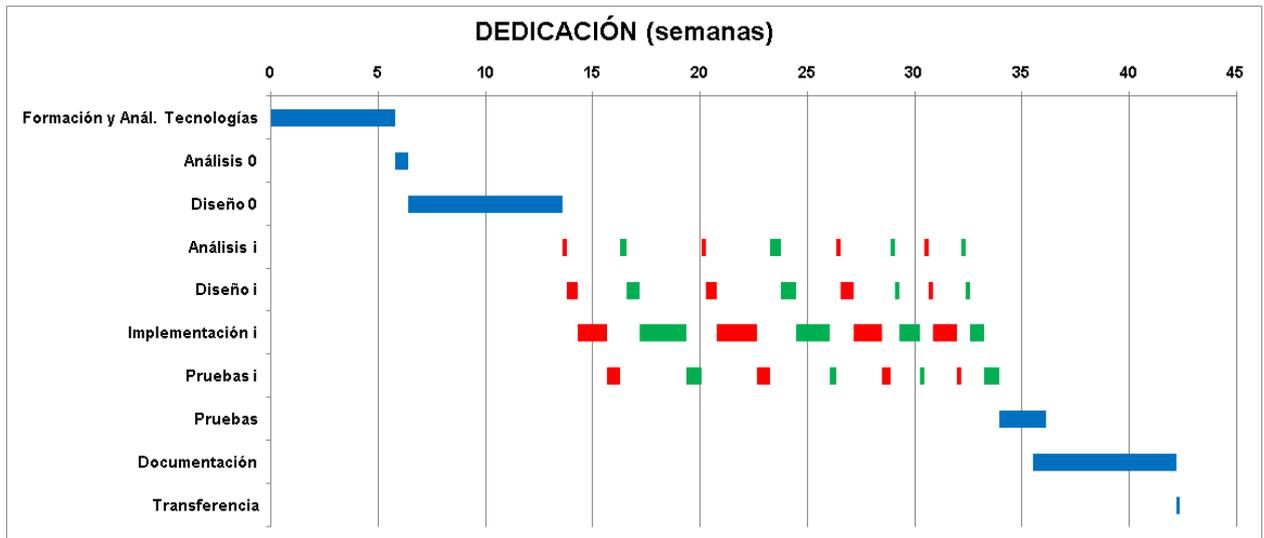


Figura 4.2 Diagrama de Gantt con el esfuerzo realizado

En la Tabla 4.2 se muestran los valores de tiempo empleados en cada una de las tareas.

Tabla 4.2: Tiempo real dedicado (horas)												
Tarea	Inicio	Iteración								Final	Total	
		1	2	3	4	5	6	7	8			
Formación y Anál. Technolog.	174											174
Análisis	18	6	9	6	15	6	6	6	6		78	
Diseño	215	15	18	15	21	18	6	6	6		320	
Implementación		42	66	57	48	39	30	33	21		336	
Pruebas		18	21	18	9	12	6	6	21	65	176	
Documentación										200	200	
Transferencia										5	5	
Total	407	81	114	96	93	75	48	51	54	270	1289	

Los principales desfases de tiempo se han producido en varias de las tareas. El desconocimiento de algunas de las tecnologías obligó a realizar un mayor esfuerzo de formación, esto también se manifestó en la fase de implementación, que requirió de un número mayor de consultas de los manuales y la realización de pruebas independientes del PFC con el objeto de comprender su funcionamiento.

La Figura 4.3 muestra un diagrama circular con el porcentaje del coste de cada una de las tareas. Para su composición se han sumado los valores de cada tarea en todas las etapas e iteraciones.

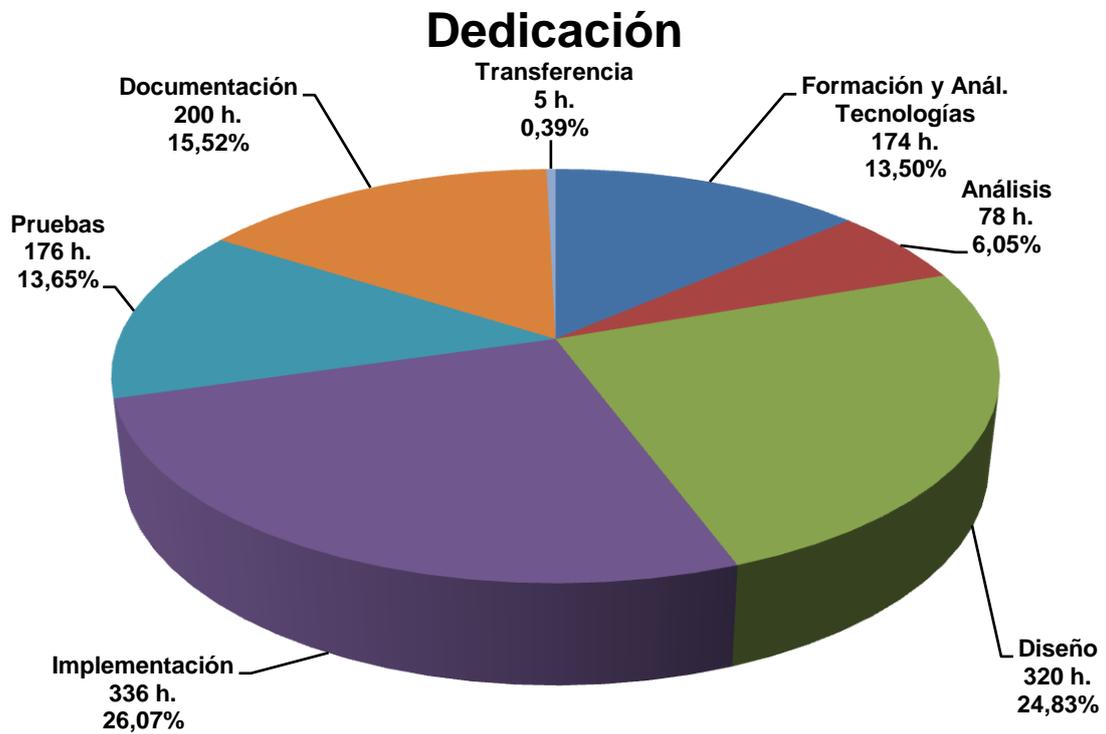


Figura 4.3 Distribución de la dedicación real

5 CONCLUSIONES

En este capítulo se exponen las conclusiones finales a las que ha llegado el autor de este PFC, abarcando tanto las conclusiones técnicas propias del PFC como las conclusiones personales. En el último apartado se presentan algunas ideas aplicables al PFC en un futuro, con el objetivo de ampliar su funcionalidad y mejorar la que hay.

5.1 CONCLUSIONES TÉCNICAS

El objetivo de este PFC era crear una aplicación Web y otra Portlet que diera soporte a la gestión de los proyectos realizados por los miembros del GIDHE y que les permitiera controlar el tiempo dedicado a sus proyectos con una herramienta que se adaptara a sus necesidades.

La aplicación Web permite a los responsables de los proyectos su gestión, configurar sus participantes y tareas, les permite crear informes de la dedicación realizada por los participantes de los proyectos y configurar el límite de tiempo dedicado a cada tarea y poder ser advertidos cuando se sobrepase.

A su vez, la aplicación Portlet facilita a los participantes en los proyectos introducir su dedicación diaria, comprobar que han sobrepasado el límite de tiempo previsto para la realización de una tarea y configurar informes de su dedicación en los proyectos.

A nivel técnico el PFC ha supuesto un reto interesante dado el número de tecnologías que se han utilizado para su diseño e implementación. Algunas de ellas como Java, HTML, JavaScript (Prototype), AJAX, JSP, Servlet,... ya eran conocidas por el autor del PFC. Otras tecnologías como los Portlets y frameworks como Spring se han incorporado a su currículum. Además el PFC ha permitido mejorar en el conocimiento de algunos frameworks como Hibernate y Log4j con los que el autor ya había realizado un contacto previo. En el caso de Hibernate, ha permitido introducirse en la tecnología de las anotaciones, que le han dado un nuevo punto de vista del uso de la meta-información, de la que había leído pero no puesto en práctica.

Aunque no haya cabido en esta memoria, el autor ha tenido la posibilidad de conocer otras tecnologías y herramientas que han ido apareciendo a lo largo del PFC y que al final han sido descartadas por diversos motivos. Entre ellas podemos nombrar las Java Server Faces [43], las posibilidades del Framework Spring para desarrollar otros proyectos que no

tienen que ver con la Web y los servidores de aplicaciones Jonas [44] y JBoss [18].

A nivel de diseño y de desarrollo se han podido consolidar conocimientos de UML, diseño Web, diseño de base de datos y los conocimientos aprendidos en distintas asignaturas como “Ingeniería del Software”, “Proyectos”, “Bases de Datos”, “Sistemas Informáticos”,...

Otro aspecto positivo sería el grado de conocimiento que se ha alcanzado con los servidores y las herramientas asociadas a éstos, por ejemplo Tomcat, Open Portal y MySQL.

Los aspectos más negativos se pueden dividir en dos partes. Todas las tecnologías, frameworks y herramientas citadas anteriormente disponen de su propia documentación, pero en la mayoría de los casos es demasiado larga, poco concisa, repetitiva y los ejemplos disponibles suelen ser demasiado superficiales. Esto y el grado de complejidad al que han llegado algunas de ellas hacen que aprenderlas sea un proceso largo, que obliga a realizar un gran esfuerzo a los ingenieros.

Por otra parte, se encontrarían las herramientas de desarrollo e implementación. Dada la discapacidad visual del autor, la elección de utilizar NetBeans como IDE de desarrollo fue una decisión inadecuada. NetBeans es una aplicación que no cumple con las normas de accesibilidad, esto ha repercutido en el tiempo de desarrollo. El autor reconoce que hubiera sido mucho mejor la utilización del IDE Eclipse [45], que como se descubrió demasiado tarde, sí era accesible, pero que se descartó por motivos técnicos y personales.

También se quiere hacer notar la inestabilidad de otras herramientas utilizadas como son Open Portal y Argo UML. En el caso de Open Portal se ha experimentado un alto grado de auto-bloqueo y en el caso de Argo UML se ha padecido una alta corrupción en los esquemas que ha obligado a modificarlos de manera repetitiva.

5.2 CONCLUSIONES PERSONALES

Independientemente de los problemas técnicos descritos anteriormente y del sobrecoste de tiempo empleado para la realización del trabajo, el autor del PFC considera que el aprendizaje de nuevas tecnologías y la posibilidad de haber realizado un proyecto de cierta envergadura le han permitido madurar como ingeniero y persona.

El autor ha podido aprender y mejorar en ciertas tecnologías que por distintos motivos no se pueden abordar de forma tan intensa en la carrera. Las asignaturas relacionadas con diseño, bases de datos, sistemas de la información y algunas otras, muestran una amplia variedad de conocimientos, pero de una manera superficial. Las prácticas que se realizan en las asignaturas, por motivos evidentes, no se pueden comparar a la realización de un proyecto real.

Por estos motivos, el proyectando considera que la realización de este PFC le ha sido de gran utilidad en diversos aspectos y le ha permitido darse cuenta de los puntos más débiles de su formación.

Entre estos puntos se pueden destacar la falta de formación en distintas tecnologías y la necesidad de introducirse en ellas. También la necesidad de buscar buenas herramientas de desarrollo que le faciliten el trabajo, en este apartado queda todavía muchas cosas pendientes.

En conclusión, el autor, tras haber cursado la carrera, haber realizado prácticas en una empresa (ajena al proyecto) y haber realizado este PFC, cree encontrarse en una posición que le permita desarrollar su carrera profesional de forma adecuada.

5.3 TRABAJO FUTURO

Considerando que se trata de un proyecto que acaba de nacer es difícil saber cuáles serán las necesidades que los usuarios de la misma reclamarán en un futuro cercano y/o lejano.

El PFC ha cubierto todas las necesidades especificadas por el cliente. Aun así sería posible mejorar en algunos aspectos que afectan a la funcionalidad del sistema. A nivel de los usuarios se pueden considerar las siguientes mejoras:

- Sería necesario, tras cierto tiempo de funcionamiento, ampliar las posibilidades que ofrecen los informes de la dedicación, en base a las propuestas que realicen los usuarios.
- En estos momentos la única forma que tienen los usuarios de saber que han sido dados de baja o alta en un proyecto es accediendo a la aplicación. Sería muy interesante para ellos la realización de un sistema que les notificara estos sucesos mediante el envío de un correo electrónico.

- Igualmente sería interesante que los usuarios recibieran notificaciones vía correo electrónico cuando se sobrepase el límite de tiempo dedicado a una tarea de un proyecto. En este caso y el anterior sería conveniente que se pudiera configurar este comportamiento de forma personalizada, utilizando cada usuario su propia configuración.
- Sería muy interesante adaptar las interfaces a otros medios como tabletas y móviles. En este momento la aplicación se encuentra dirigida hacia los navegadores Web de los ordenadores personales.
- Debería permitirse a los usuarios de la aplicación Web poder mantener la configuración de la presentación de cada sección (valores de búsqueda, número de elementos mostrados,...). En estos momentos, esta configuración solo se mantiene durante el tiempo que permanece conectado a la aplicación.

Respecto de la gestión de la aplicación se pensaron varias mejoras durante su desarrollo, pero algunas de ellas se quedaron como posibles ampliaciones del PFC.

- La introducción de nuevos usuarios en el sistema se realiza de uno en uno. Sería más que recomendable permitir al administrador introducir una lista de nuevos miembros. Este proceso podría realizarse tanto manualmente como a través de un fichero.
- También se podría permitir que los gestores pudieran introducir nuevos usuarios, con objeto de que el administrador confirmase su aceptación.
- La introducción de un nuevo miembro obliga al administrador a informar a la persona de su nueva condición de usuario. Sería recomendable automatizar este proceso mediante un sistema de notificación por correo electrónico. Igualmente se podría enviar un aviso cuando se da de baja o alta a un usuario.
- El acceso y modificación de la configuración del sistema se realiza modificando el texto de los ficheros de propiedades (mediante la aplicación Web). Se podría facilitar esta operación cambiando la interfaz para que se mostrara un formulario con las opciones y los valores permitidos.
- La generación de informes del sistema se ha realizado de forma global, podría ser recomendable que se facilitara una manera de limitar el alcance temporal de los valores, permitiendo fijar unas fechas que acotaran los datos utilizados.

ANEXO A

DOCUMENTO

DE ANÁLISIS DE

REQUISITOS

A.1 INTRODUCCIÓN

El Documento de Análisis de Requisitos (DAR) contiene todos los requisitos que se han definido durante el análisis previo al diseño del proyecto. Para capturar los requisitos se han definido casos de uso derivados de los objetivos planteados en función de las necesidades del cliente.

En el capítulo A.2 se muestran las tablas donde se listan los requisitos del sistema. Los requisitos se han clasificado, estructurado y numerado para poder ser referenciados en los demás documentos del proyecto. Cada requisito tiene un nombre que lo identifica unívocamente (basado en su clasificación y orden).

Tabla A.1.1 Notación de los requisitos	
Notación	Significado
RFx	Requisito funcional x.
RNFx	Requisito no funcional x.
RSx	Requisitos del sistema x.

En el capítulo A.3 se muestran dos figuras con los diagramas de casos de uso que han permitido capturar los requisitos del capítulo A.2. Para cada caso de uso se definen los requisitos capturados y se describe brevemente la funcionalidad que representan.

A.2 REQUISITOS

A.2.1 REQUISITOS FUNCIONALES

Tabla A.2.1 Requisitos funcionales	
#	Requisito
RF1	El sistema tendrá tres tipos de usuarios: el administrador general del sistema, los gestores de proyectos y los participantes en proyectos.
A	<i>Requisitos punto de vista del administrador</i>
RF2	El administrador dará de alta y baja en el sistema a los gestores de proyectos y a los participantes.
RF3	El administrador será el responsable de la configuración general del sistema (sistema de logs, copias de seguridad, etc.).
RF4	El administrador podrá consultar estadísticas de utilización del sistema: número de proyectos gestionados, número de gestores y participantes, asignación de gestores y participantes a proyectos, etc.
B	<i>Requisitos punto de vista del gestor de proyectos</i>
RF5	Un gestor de proyectos podrá configurar y modificar su perfil de usuario del sistema.
RF6	Un gestor de proyectos podrá dar de alta un nuevo proyecto.
RF7	Dado un nuevo proyecto, un gestor podrá establecer la lista de tareas asociada al mismo (por ejemplo, formación previa, análisis, diseño, implementación, etc.).
RF8	Un gestor de proyectos podrá establecer qué participantes registrados en el sistema van a participar en un proyecto concreto.
RF9	Un gestor de proyectos podrá consultar el listado de proyectos de los cuales es responsable.
RF10	Dado un proyecto, el gestor podrá consultar el esfuerzo dedicado aplicando criterios temporales (en total, por meses o entre dos fechas concretas).
RF11	Dado un proyecto, el gestor podrá consultar el esfuerzo dedicado a cada tarea aplicando criterios temporales (total, meses o entre dos fechas).
RF12	Dado un proyecto, el gestor podrá consultar el esfuerzo que un participante ha dedicado al mismo aplicando criterios temporales (total, meses o entre dos fechas), según tareas o combinando ambos criterios.
RF13	A partir de las posibles consultas, el sistema podrá generar informes en formato PDF relativos a un proyecto concreto.
RF14	El gestor podrá modificar (añadir nuevo o dar de baja uno ya existente) los participantes de un proyecto.
RF15	Un gestor de proyecto podrá configurar alertas de dedicación máxima, por ejemplo, a nivel de tarea de un proyecto, a nivel de tarea/mes, a nivel de tarea/participante o a nivel tarea (participante/mes).

RF16	El sistema deberá ser capaz de detectar cuando una dedicación máxima previamente configurada es excedida e informar de ello al gestor del correspondiente proyecto.
C	Requisitos punto de vista del participante en un proyecto
RF17	Un participante podrá configurar y modificar su perfil de usuario del sistema.
RF18	Un participante podrá consultar los proyectos en los que está participando o en los que participó.
RF19	Dado un proyecto en el que participa, un participante podrá consultar las tareas en las que trabaja.
RF20	Dado un proyecto en el que participa, un participante podrá introducir para una tarea su dedicación diaria (en número de horas o fracciones).
RF21	Un participante podrá consultar su dedicación en un proyecto aplicando criterios temporales (total, por meses o entre dos fechas), según tareas o combinando ambos criterios.
RF22	A partir de las posibles consultas, el sistema podrá generar informes en formato PDF relativos a un participante concreto.
RF23	Un participante será informado si sobrepasa en algún momento alguna de las alertas de dedicación máxima establecidas.

A.2.2 REQUISITOS NO FUNCIONALES

Tabla A.2.2 Requisitos no funcionales

#	Requisito
RNF1	El sistema de gestión de la dedicación será desarrollado con tecnología de Portlets.
RNF2	El sistema deberá integrarse en la página Web de un participante cualquiera con un esfuerzo mínimo.
RNF3	El sistema dispondrá de un portal Web para que el administrador o el gestor de proyectos configuren su actividad.
RNF4	El sistema deberá estar instalado y operativo en un servidor que favorezca el uso por parte de una amplia comunidad de usuarios en un futuro.
RNF5	Al finalizar el proyecto deberá estar operativo y siendo utilizado por al menos el director del proyecto.

Tabla A.2.3 Requisitos de seguridad

#	Requisito
RS1	El acceso al sistema será seguro, por medio de un mecanismo de clave y palabra clave.
RS2	El sistema deberá ser capaz de realizar copias de seguridad de manera periódica.
RS3	El sistema almacenará ficheros de "log" de la actividad del sistema. El formato del fichero deberá ser apropiado para futuros análisis de su contenido.

A.3 CAPTURA DE REQUISITOS

En este capítulo se describen los casos de uso que han permitido capturar los requisitos funcionales del sistema. Cada caso de uso encierra un conjunto de funcionalidades del sistema que podrían derivar en nuevos casos de uso, pero, por motivos de simplificación, se ha preferido utilizar casos de uso más generales que abarquen las principales partes que se esperan de la aplicación. Los distintos casos de uso han sido creados en función del contexto del sistema y de los objetivos definidos en función de las necesidades especificadas por el cliente.

El sistema se relacionará con tres tipos de actores “Admin”, “Gestor” y “Participante”, requisito capturado en el RF1. El sistema se dividirá en dos partes (requisitos RNF3 y RNF1), los actores “Admin” y “Gestor” harán uso de la aplicación Web y el actor “Participante” se relacionará con la aplicación Portlet. En los siguientes apartados del capítulo se muestra como se han capturado los requisitos funcionales en cada aplicación. Cada caso de uso dispondrá de un nombre abreviado “CUxx”, donde xx serán dos valores numéricos, que permitirán referenciarlos en la documentación.

A.3.1 APLICACIÓN WEB

En la Figura A.3.1 se puede ver el diagrama que define los casos de uso que han permitido capturar los requisitos de RF2 a RF16. El diagrama muestra dos actores “Gestor” y “Admin”, representados por la figura de un señor, que se unen mediante líneas con los casos de uso con los que se relacionan, representados por óvalos amarillos.

El actor “Gestor” se relaciona con cuatro casos de uso. El caso de uso “Gestión Proyectos” (CU01) permite capturar los requisitos RF6, RF7, RF8, RF9 y RF14, facilitando la gestión de los proyectos, las tareas y los participantes. El caso de uso “Gestión Alarmas” (CU02) permite capturar los requisitos RF15 y RF16, facilitando la configuración de las alarmas de la dedicación y permitiendo comprobar cuando se ha superado el límite establecido por la alarma. Los requisitos RF10, RF11, RF12 y RF13 han sido capturados a partir del caso de uso “Informes Dedicación” (CU03) que permitirá crear informes del esfuerzo realizado por los participantes en los proyectos.

El caso de uso “Gestión Perfil” (CU04) se relaciona con ambos actores y captura el requisito RF5. Este caso de uso atiende a la necesidad de los usuarios de poder configurar sus datos personales, como puede ser cambiar su contraseña.

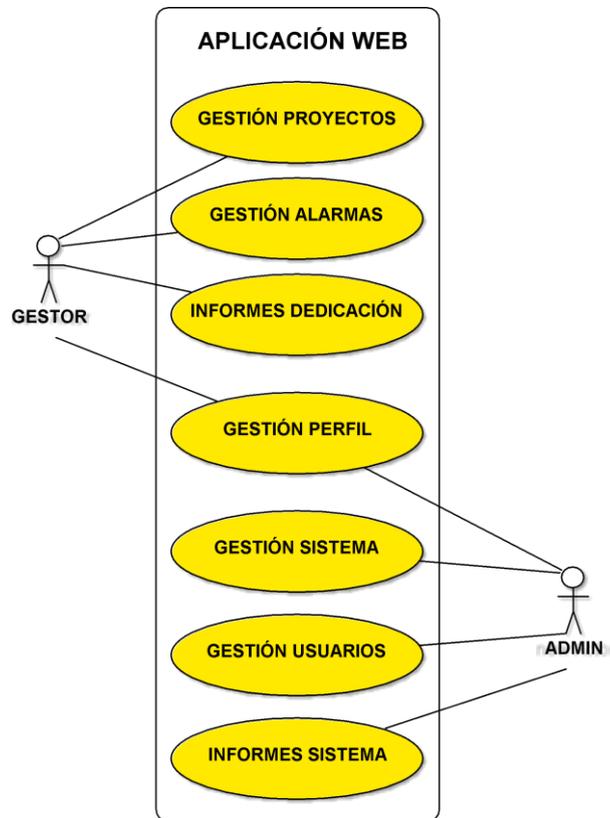


Figura A.3.1 Casos de uso relacionados con la aplicación Web

El actor “Admin” se relaciona con cuatro casos de uso. “Gestión Sistema” (CU05) se encarga de la configuración de las aplicaciones y de la gestión de los ficheros de Logs. Este caso de uso ha permitido capturar el requisito RF3. El caso de uso “Gestión Usuarios” (CU06) permite capturar el requisito RF2 y permitirá la introducción de nuevos usuarios y la gestión de los existentes. El caso de uso “Informes Sistema” (CU07) se encargará de crear informes sobre la utilización del sistema y ha permitido capturar el requisito RF4.

A.3.2 APLICACIÓN PORTLET

Los casos de uso contenidos en el Portlet han permitido capturar los requisitos del RF17 al RF23. A continuación, la Figura A.3.2 muestra el diagrama con los casos de uso de la aplicación Portlet.

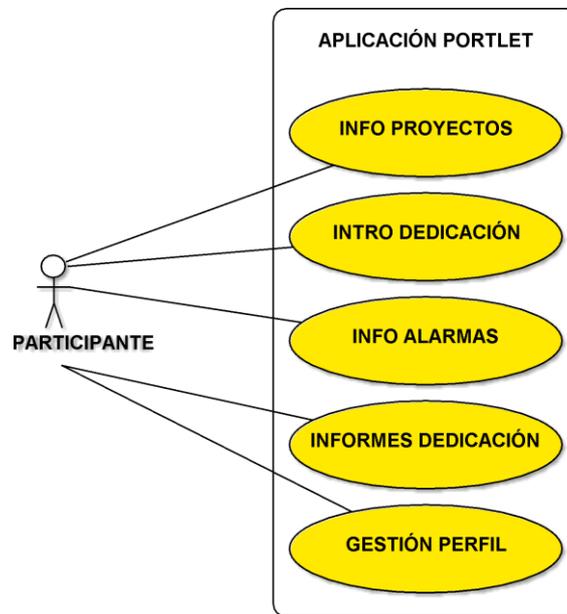


Figura A.3.2 Casos de uso relacionados con la aplicación Portlet

El actor “Participante” se relaciona con cuatro casos de uso. “Info Proyectos” (CU08) ha permitido capturar los requisitos RF18 y RF19, permitiendo consultar la información relacionada con los proyectos en los que participa. El caso de uso “Intro Dedicación” (CU09) permite introducir la dedicación diaria que realiza el participante en sus proyectos, permitiendo capturar el requisito RF20.

El caso de uso “Info Alarmas” (CU10) permite al actor consultar la información de las alarmas de la dedicación y captura el requisito RF23. El caso de uso “Informes Dedicación” (CU03), compartido con la aplicación Web”, permite al “Participante” generar informes de su dedicación y captura los requisitos RF21 y RF22. En último lugar el caso de uso “Gestión Perfil” (CU04), que es compartido con la aplicación Web, permite capturar el requisito RF17, y le permite configurar su perfil de usuario en el sistema.

ANEXO B

DOCUMENTO

DE DISEÑO

DEL SISTEMA

B.1 INTRODUCCIÓN

El Documento de Diseño del Sistema (DDS) contiene la descripción del sistema desarrollado durante el proyecto. A continuación se pretende describir los objetivos que se pretenden alcanzar con el desarrollo de este documento.

B.1.1 OBJETIVOS Y ALCANCE DEL DOCUMENTO

El objetivo principal del “Documento de Diseño del Sistema” es describir formalmente, de manera clara y detallada, la estructura y el funcionamiento de cada una de las partes que componen el sistema desarrollado.

En el capítulo B.2 se presenta el diseño del sistema desarrollado. Los capítulos 3 y 4 describen la estructura de la BD y de los datos del negocio, respectivamente. En el capítulo 5 y 6 se refinan las capas de datos y de negocio, describiendo las interfaces de sus componentes. El capítulo 7 se dedica a los componentes transversales de la aplicación. Los capítulos 8 y 9 describen las interfaces gráficas de los usuarios para la aplicación Web y el Portlet. Por último, el capítulo 10 describe como realizar las copias de seguridad del sistema.

El diseño incluirá diagramas UML que permitan describir las partes del sistema. También se añadirán imágenes que describan la aplicación, p.e. el diseño de una interfaz gráfica.

B.2 DISEÑO DEL SISTEMA

El diseño del sistema se basa en una arquitectura cliente-servidor. Se ha dispuesto un servidor Web para los gestores y el administrador y un servidor de tipo Portlet para los participantes en los proyectos. Los distintos clientes pueden hacer uso de sus navegadores Web para comunicarse con ambos servidores y acceder a las aplicaciones.

B.2.1 DESCOMPOSICIÓN DEL SISTEMA

La arquitectura se basa en un sistema multicapa. El sistema se ha dividido en tres capas, cada una con un objetivo concreto. Su diseño se ha realizado de manera que las funciones de una capa ofrecen servicio a la capa inmediatamente superior.

Cada capa cumple con una necesidad del sistema, evitando la solapación y duplicación de funcionalidades. Se facilita la abstracción entre capas a través de APIs que permiten comunicarse entre ellas. De esta manera, se permite un mejor desarrollo y mantenimiento del software, haciendo más fácil la modificación del contenido interno de las capas sin afectar a las demás.

La capa de presentación facilita las GUIs que permiten a los usuarios comunicarse con la aplicación, las interfaces se basan en páginas Web con contenido dinámico en el lado del cliente. La capa de negocio se encarga de las distintas funciones que ofrece el sistema, dando servicio a la capa de presentación. La capa de datos ofrece el acceso al repositorio de la información del sistema (una base de datos) de forma transparente al medio de almacenamiento. El diseño de las capas se detalla en los capítulos B.5, B.6, B.8 y B.9 de este documento:

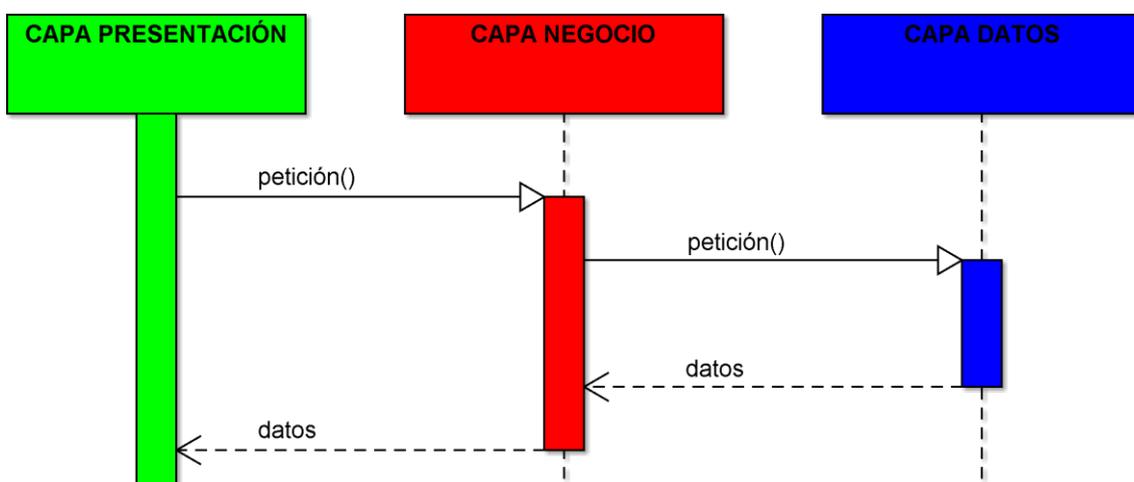


Figura B.2.1 Secuencia de una petición

La Figura B.2.1 muestra una secuencia de una petición de un cliente a una de las aplicaciones. Toda petición comienza en la capa de presentación (GUI) enviando un mensaje a la capa de negocio, la capa de negocio envía uno o más mensajes a la capa de datos para acceder al repositorio. Tras realizar sus funciones, la capa de datos responde a la petición devolviendo los datos pedidos. Entonces, la capa de negocio se encarga de darle una respuesta a la capa de presentación.

B.2.2 DESPLIEGUE DEL SISTEMA

El sistema dispone de dos niveles, el cliente y el servidor. Las capas de negocio y de datos se quedan en el nivel del servidor, los componentes que forman ambas capas son compartidos por las dos aplicaciones que forman el sistema. La capa de presentación se distribuye en el nivel del servidor, personalizándose para cada una de las aplicaciones. La Figura B.2.2 muestra la distribución de las capas en ambos niveles.

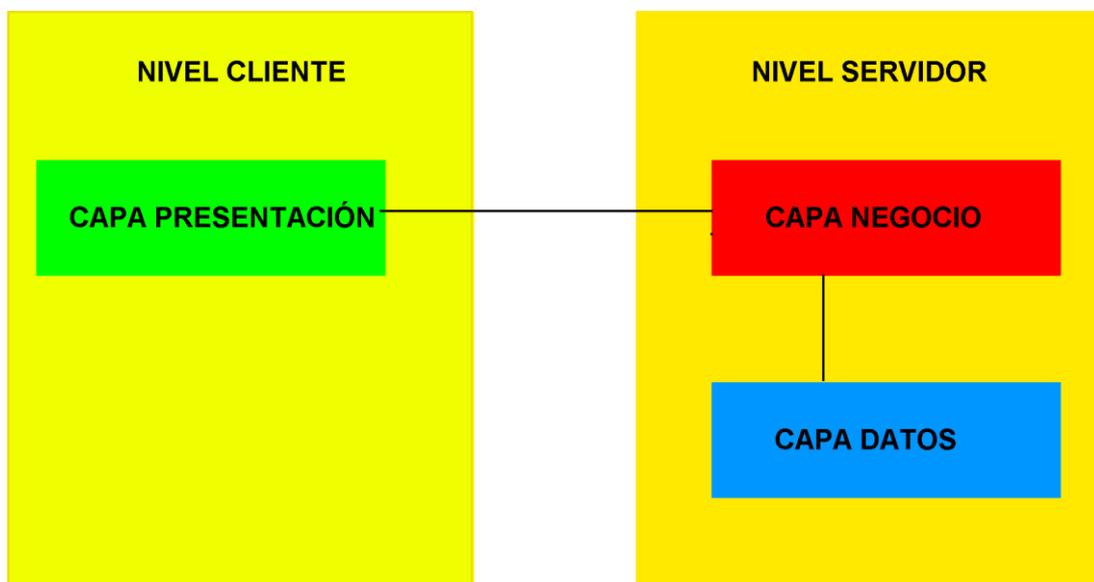


Figura B.2.2 Despliegue del sistema

B.2.3 DISEÑO BOTTOM-UP

En los siguientes capítulos se presenta el diseño de cada una de las partes del sistema. Para su diseño se ha seguido un modelo de abajo hacia arriba, donde primero se han diseñado los componentes inferiores, empezando por la capa de datos y se ha ido ascendiendo hacia la capa de presentación.

B.3 DISEÑO DE LA BASE DE DATOS

En este capítulo se describe el diseño de la base de datos. La BD ofrecerá al sistema un repositorio que almacenará los datos del negocio de forma persistente. La estructura de datos que se describe a continuación influirá en el diseño del resto de la aplicación.

B.3.1 DISEÑO CONCEPTUAL

El diagrama Entidad-Relación de la Figura B.3.1 (y su nomenclatura en la Figura B.3.2) muestra el modelo conceptual de la base de datos. A continuación se describirán con más detalle las entidades y relaciones que lo componen.

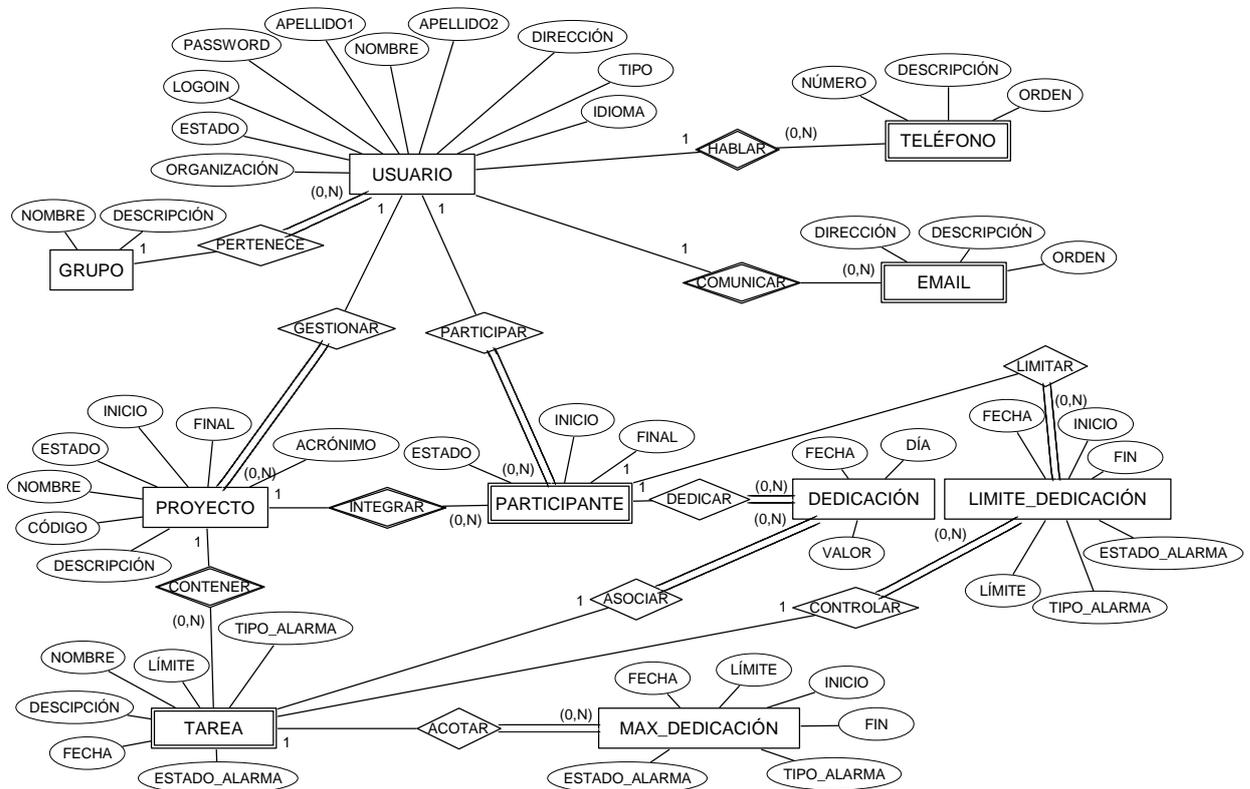


Figura B.3.1 Diagrama Entidad-Relación de la BD

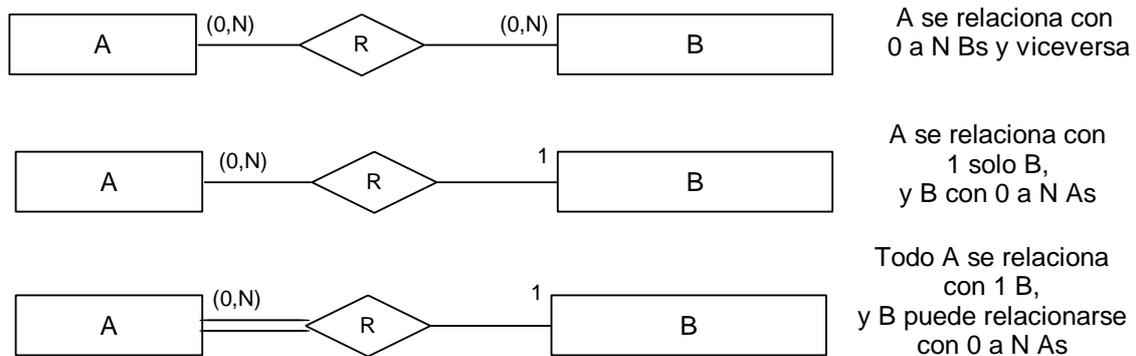
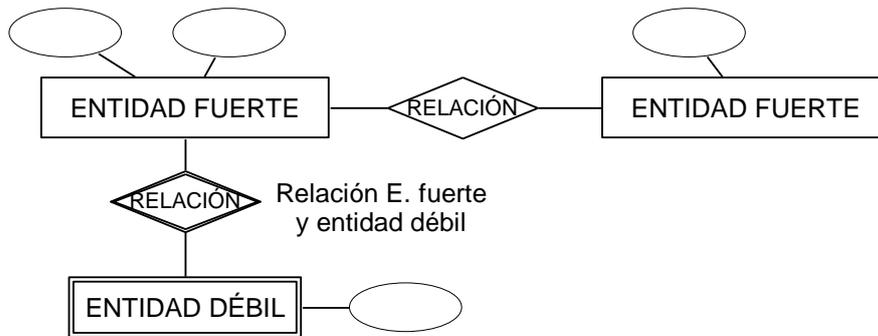


Figura B.3.2 Notación del diagrama Entidad-Relación

B.3.1.1 DESCRIPCIÓN DE LAS ENTIDADES

Las siguientes tablas describen las entidades que aparecen en el diagrama.

Tabla B.3.1 Descripción de las entidades (tabla dividida en varias partes)	
ENTIDAD	
USUARIO	
Descripción	
Datos de un usuario del sistema.	
Atributo	Descripción
Nombre	Nombre de la persona
Apellido1	Primer apellido
Apellido2	Segundo apellido (si existe)
Login	Identificador de acceso al sistema
Password	Clave privada de acceso al sistema
Dirección	Dirección del usuario (calle, edificio, ciudad, código postal)
Idioma	Idioma que se aplicará a la interfaz de la aplicación para este usuario.
Estado	Estado del usuario, posibilita el acceso y uso del sistema (Activo, no activo)
Tipo	Tipo de usuario, lleva asociado ciertos privilegios (Administrador, gestor, usuario normal)

Organización	Organización o empresa a la que pertenece
Restricciones	
El atributo "Login" debe ser único para cada usuario. Todo usuario debe estar asociado a un grupo.	

ENTIDAD	
GRUPO	
Descripción	
Permite agrupar los usuarios y restringir su visibilidad y asociación a un proyecto.	
Atributo	Descripción
Nombre	Nombre dado al grupo
Descripción	Breve descripción del grupo.
Restricciones	

ENTIDAD	
TELÉFONO	
Descripción	
Número de teléfono o fax asociado a un usuario del sistema.	
Atributo	Descripción
Número	Número de teléfono o fax.
Descripción	Breve descripción del teléfono (casa, oficina, móvil,...).
Orden	Orden por el que se mostrará el número de teléfono de un usuario.
Restricciones	

ENTIDAD	
EMAIL	
Descripción	
Dirección de correo electrónico asociada a un usuario del sistema.	
Atributo	Descripción
Dirección	Dirección del correo electrónico.
Descripción	Descripción del correo (Ej. Oficial, personal,...).
Orden	Orden por el que se mostrará la dirección de correo de un usuario.
Restricciones	

ENTIDAD	
Proyecto	

Descripción	
Datos de un proyecto.	
Atributo	Descripción
Código	Código asociado al proyecto.
Nombre	Nombre asociado al proyecto.
Descripción	Descripción del proyecto.
Acrónimo	Acrónimo asociado al proyecto.
Estado	Estado actual del proyecto (Activo, cerrado).
Inicio	Fecha de inicio del proyecto
Final	Fecha de finalización del proyecto
Restricciones	
<p>La fecha final de un proyecto debe ser nula si el estado del mismo no es un estado de finalización.</p> <p>El código de un proyecto debe ser único.</p> <p>Un proyecto debe ir asociado a un usuario de tipo "gestor".</p> <p>El proyecto debe estar relacionado con un usuario gestor que esté activo en el sistema si el proyecto está en el estado abierto.</p>	

ENTIDAD	
Participante	
Descripción	
Asociación de un usuario dentro de un proyecto de su grupo de pertenencia.	
Atributo	Descripción
Estado	Estado actual del participante (activo, no activo).
Inicio	Fecha en que el participante es añadido al proyecto.
Final	Fecha en la que el participante es dado de baja en el proyecto.
Restricciones	
<p>Un participante debe estar asociado con un usuario del sistema.</p> <p>El usuario asociado al participante debe estar en un estado activo si el participante lo está.</p> <p>El participante no puede estar en un estado activo si el proyecto asociado está en un estado de cerrado.</p> <p>El usuario asociado al participante debe pertenecer al mismo grupo que el gestor asociado al proyecto.</p>	

ENTIDAD	
Tarea	
Descripción	
Datos de una tarea asociada a un proyecto.	
Atributo	Descripción
Nombre	Nombre asociado a la tarea.
Descripción	Descripción de la tarea.
Fecha	Fecha de la última actualización de la alarma.
Límite	Número máximo de horas que se deberían emplear en esta tarea (incluyendo la agregación de todas las horas empleadas por todos los participantes del proyecto).
TipoAlarma	Tipo de alarma que se aplicará si se sobrepasa el límite de

	horas asociadas al proyecto.
EstadoAlarma	Estado en el que se encuentra la alarma actualmente (activada, desactivada, limite sobrepasado,...).
Restricciones	
Toda tarea debe estar relacionada con un proyecto.	

ENTIDAD	
Maxdedicación	
Descripción	
Límite máximo de tiempo que se deberá dedicar a una tarea dentro de unas fechas dadas (incluyendo todas las horas de todos los participantes en el proyecto).	
Atributo	Descripción
Fecha	Fecha de la última actualización de la alarma.
Inicio	Fecha inicial del intervalo de tiempo a tener en cuenta para el cálculo de las horas.
Final	Fecha final del intervalo de tiempo a tener en cuenta para el cálculo de las horas.
Límite	Número máximo de horas que se deberían emplear en esta tarea dentro del periodo de fechas indicado (incluyendo la agregación de todas las horas empleadas por todos los participantes del proyecto).
TipoAlarma	Tipo de alarma que se aplicará si se sobrepasa el límite de horas asociadas al proyecto.
EstadoAlarma	Estado en el que se encuentra la alarma actualmente (activada, desactivada, limite sobrepasado,...).
Restricciones	
Toda entidad "Maxdedicación" debe estar relacionada con una tarea de un proyecto.	

ENTIDAD	
Dedicación	
Descripción	
Cantidad de tiempo que un participante ha dedicado a una tarea en un día concreto. NOTA: Por el atributo "Fecha" se entiende el día en el que se insertó el dato en la BD y por el atributo "Día" la fecha asociada con las horas dedicadas a la tarea del proyecto.	
Atributo	Descripción
Fecha	Fecha de introducción de los datos.
Día	Fecha del día en que se realizó la dedicación.
Valor	Cantidad de tiempo empleada en la tarea.
Restricciones	
"Valor" debería ser mayor que 0 e inferior a 24 horas. Toda entidad "Dedicación" tiene que estar relacionada con una tarea y un participante de un mismo proyecto. No puede relacionarse con participantes y tareas de distintos proyectos.	

ENTIDAD	
Limitededicación	
Descripción	
Límite máximo de tiempo que un usuario se debe dedicar a una tarea dentro de unas fechas dadas	
Atributo	Descripción
Fecha	Fecha de la última actualización de la alarma.
Inicio	Fecha inicial del intervalo de tiempo a tener en cuenta para el cálculo de las horas.
Final	Fecha final del intervalo de tiempo a tener en cuenta para el cálculo de las horas.
Límite	Número máximo de horas que el usuario debería emplear en esta tarea dentro del periodo de fechas indicado.
TipoAlarma	Tipo de alarma que se aplicará si se sobrepasa el límite de horas asociadas al proyecto.
EstadoAlarma	Estado en el que se encuentra la alarma actualmente (activada, desactivada, limite sobrepasado,...).
Restricciones	
Toda entidad "Limitededicación" tiene que estar relacionada con una tarea y un participante de un mismo proyecto. No puede relacionarse con participantes y tareas de distintos proyectos.	

B.3.2 DISEÑO LÓGICO

A continuación se describen las relaciones generadas al convertir diagrama Entidad-Relación en un modelo relacional que cumple la 2ª Forma Normal.

B.3.2.1 DESCRIPCIÓN DE LAS RELACIONES

Tabla B.3.2 Notación del diseño Relacional	
PK	Clave Primaria.
NN	No Nulo.
UQ	Único.
FK(x) RF t(y)	X clave Ajena de la tabla t clave y.

GRUPO (
Nombre Cadena PK,
Descripción Cadena
);

USUARIO (

Login Cadena PK NN
Password Cadena NN,
Nombre Cadena NN,
Apellido1 Cadena NN,
Apellido2 Cadena,
Dirección Cadena,
Idioma Cadena NN,
Organización Cadena,
Tipo {SUPERADMIN, GESTOR, NORMAL} NN,
Estado {ACTIVO, INACTIVO} NN,
Grupo Cadena,
FK (Grupo) RF Grupo (Nombre),

Restricción: Solo puede haber un usuario de tipo "SUPERADMIN", éste tendrá el login "admin" y será el único usuario que no tenga asociado un grupo. Los demás usuarios deberán tener asociado un grupo.

);

TELÉFONO (

Id Número NN;
Usuario Cadena NN,
Orden Número NN,
Número Cadena NN,
Descripción Cadena,
PK (Usuario, Id),
UQ (Usuario, Orden),
FK (Usuario) RF Usuario (Login)

);

EMAIL (

Id Número NN,
Usuario Cadena NN,
Orden Número NN,
Dirección Cadena NN,
Descripción Cadena,
PK (Usuario, Id),
UQ (Usuario, Orden),
FK (Usuario) RF Usuario (Login)

);

PROYECTO (

Código Número PK,
Nombre Cadena NN,

Acrónimo Cadena NN,
Descripción Cadena,
Estado {ABIERTO, CERRADO} NN,
Inicio tpFecha NN,
Final tpFecha,
Gestor Cadena,
FK (Gestor) RF Usuario (Login),
Restricción: Solo se pueden asociar a Gestor usuarios de tipo
“GESTOR”,
Restricción: Si el estado es “ABIERTO” el estado del gestor asociado no
puede ser “INACTIVO”.
Restricción: ((Final = nulo OR Estado = CERRADO) AND (final <> null
OR estado = ABIERTO)),
Restricción: Inicio <= Final o final es nulo.
);

PARTICIPANTE (

Proyecto Número NN,
Usuario Cadena NN,
Estado {ACTIVO, INACTIVO} NN,
Inicio tpFecha NN,
Final tpFecha,
PK (Proyecto, Usuario),
FK (Proyecto) RF Proyecto (Código),
FK (Usuario) RF Usuario (Login)
Restricción: El usuario debe ser del mismo grupo que el gestor asociado
al proyecto relacionado con el participante.
Restricción: El “Usuario” debe estar asociado con un usuario de tipo
“GESTOR” o “NORMAL”.
Restricción: Si el campo “estado” es “ACTIVO”, el estado del usuario
asociado debe ser “ACTIVO” y el estado del proyecto asociado debe ser
“ABIERTO”.
Restricción: Si el campo “estado” es “ACTIVO” el campo final debe ser
nulo y si es “INACTIVO” no nulo.
Restricción: Inicio <= final o final es nulo.
);

TAREA (

Proyecto Número NN
Nombre Cadena NN,
Descripción Cadena,
Orden Número NN,
Límite Número NN,
TipoAlarma tpAlarmaTarea NN,

EstadoAlarma tpEstadoAlarmaTarea NN,
Fecha tpFecha NN,
PK (Nombre, Proyecto),
FK (Proyecto) RF Proyecto (Código),
Restricción: Límite ≥ 0
);

MAX_DEDICACIÓN (
ID Número PK,
Tarea Cadena NN,
Proyecto Número NN,
Inicio tpFecha NN,
Final tpFecha NN,
Límite Número NN,
TipoAlarma tpAlarmaTarea NN,
EstadoAlarma tpEstadoAlarmaTarea NN,
Fecha tpFecha NN,
FK (Tarea, Proyecto) RF Tarea (Nombre, Proyecto),
UQ (Tarea, Proyecto, Inicio, Final),
Restricción: Límite > 0 ,
Restricción: Inicio \leq Final
);

DEDICACIÓN (
ID Número PK,
Tarea Cadena NN,
ProyectoTarea Número NN,
Participante Cadena NN,
ProyectoParticipante Número NN, Día tpFecha NN,
Día tpFecha,
Valor Número NN,
Fecha tpFecha NN,
FK (Tarea, Proyecto) RF Tarea (Nombre, Proyecto),
FK (Participante, Proyecto) RF Participante (Usuario, Proyecto),
UQ (Tarea, Proyecto, Participante, Día)
Restricción: ProyectoTarea = ProyectoParticipante,
Restricción: $0 < \text{Valor} < 24$
);

LÍMITE_DEDICACIÓN (
ID Número PK,
Tarea Cadena NN,
ProyectoTarea Número NN,
Participante Cadena NN,

ProyectoParticipante Número NN,
Inicio tpFecha NN,
Final tpFecha NN,
Límite Número NN,
TipoAlarma tpAlarmaParticipante NN,
EstadoAlarma tpEstadoAlarmaParticipante,
Fecha tpFecha NN,
FK (Tarea, Proyecto) RF Tarea (Nombre, Proyecto),
FK (Participante, Proyecto) RF Participante (Usuario, Proyecto),
UQ (Proyecto, Tarea, Participante, Inicio, Final),
Restricción: ProyectoTarea = ProyectoParticipante,
Restricción: Límite > 0,
Restricción: Inicio <= Final
);

NOTA: tpAlarmaTarea, tpAlarmaParticipante, tpEstadoAlarmaTarea y tpEstadoAlarmaParticipante serán definidos en el capítulo B.6 “Capa de negocio”, donde se explicará el comportamiento de las alarmas.

B.4 ESTRUCTURA DE LOS DATOS

Este capítulo pretende ofrecer una visión estática de la estructura de los datos, relacionados directamente con el negocio de la aplicación, que manejará el sistema. Todas las capas de la aplicación harán uso de ellos, la capa de datos deberá ser capaz de crearlos, recuperarlos, consultarlos de forma agrupada, borrarlos, validarlos y persistirlos. La capa de negocio los utilizará como contenedores de la información que maneje. Por último, la capa de presentación les dará forma para que los usuarios puedan verlos adecuadamente, así como ser capaz de tomar la información que le pase el usuario y adaptarla a ella.

Para su diseño se va a utilizar el concepto de “Poor Old Java Object”, más conocidos por sus siglas “POJO”. Todos los POJOs cumplirán con un estereotipo “PojoDedicación”, cuyo fin será el de restringir el uso de los métodos de la capa de datos.

Cada POJO dispondrá de un único constructor sin parámetros y de una colección de métodos getter y setter, necesarios para consultar y modificar sus datos. Todos sus atributos permanecerán ocultos (no serán públicos). Además los POJOs dispondrán de la meta-información necesaria para poder ser consultados, validados y persistidos por la capa de datos.

Para facilitar la validación de algunas clases se creará otro estereotipo llamado “PojoFechado”. Esto implicará que las clases que lo implementen deberán tener los métodos “getInicio()” y “getFin()” que devolverán las fechas de inicio y fin de un intervalo de tiempo. Esto se utilizará para poder asociarles la restricción “fechas” (ver restricciones).

Los POJOs están íntimamente relacionados con la base de datos. Esto significa que cada clase comparte un nombre común con una de las tablas de la BD y que los atributos que tengan el mismo nombre que los campos harán referencia a la misma información. Respecto de los nombres de los atributos solo se hará una excepción, el campo de las tablas llamado “final” será sustituido por el atributo “fin”, para evitar problemas con la palabra reservada “final”. En los siguientes apartados se especificarán las clases que permitirán su construcción.

B.4.1 CLASES

La Figura B.4.1 muestra el diagrama de clases, donde las clases están representadas por cajas y sus relaciones se indican mediante flechas marcadas con su multiplicidad. Se han omitido los atributos y los métodos de las clases para facilitar su visibilidad.

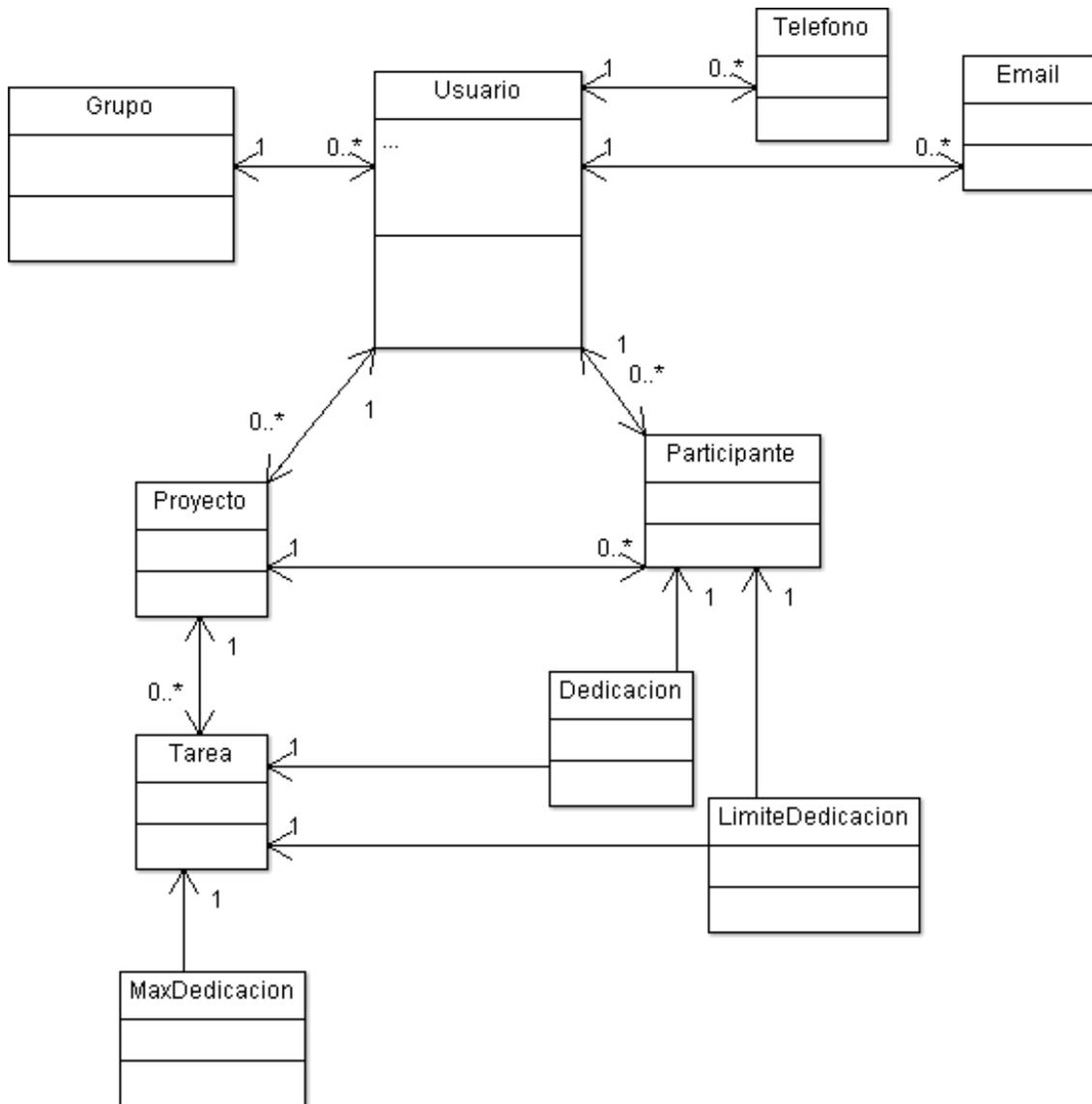


Figura B.4.1 Diagrama de clases

Los atributos de las clases se corresponden con los campos utilizados en las tablas (Ver el capítulo B.3 “Diseño de la BD”). Igualmente cada clase contendrá un método “get” y otro “set” para poder obtener y almacenar cada atributo.

B.4.2 DESCRIPCIÓN DE LAS ALARMAS

Este apartado describe los diferentes modelos, estados y tipos de alarmas de la dedicación. El funcionamiento de las mismas se describirá en el capítulo B.6 “Capa de negocio”.

Las alarmas permiten notificar al gestor y a los participantes de un proyecto que el tiempo dedicado a una tarea ha sobrepasado el límite de horas que se le habían asignado. Son un método diseñado para controlar el tiempo dedicado a una tarea de un proyecto.

Las alarmas serán creadas, modificadas y borradas por el gestor. Los demás participantes del proyecto apenas tendrán control directo sobre ellas. Un participante podrá modificar implícitamente una alarma al modificar sus valores de dedicación en la tarea dada.

El gestor y los participantes asociados a un proyecto podrán ver una lista con las alarmas que les puedan afectar. Cuando el tiempo especificado en una alarma se sobrepase se producirá una alerta que deberá ser notificada al gestor y a los participantes afectados. Este comportamiento se detallará en el capítulo B.6 “Capa de negocio”.

NOTA: Aunque el significado de "alarma" y "alerta" es el mismo, en el entorno de este proyecto se tomarán como diferentes. El primer concepto atenderá a la idea de controlar el límite que se desea poner a una tarea de un proyecto, pudiendo estar ésta en varios estados y ser de varios modelos. El segundo concepto hará referencia al estado/acciones que se deben realizar cuando una alarma ha sido rebasada.

Hay tres modelos de alarmas, cada una de ellas cumple una función:

1. Alarmas de una tarea en total, están asociadas a una tarea de un proyecto, para su comprobación, se sumará todo el tiempo dedicado por los participantes en la tarea.
2. Alarmas de una tarea en un periodo de tiempo. Están asociadas a una tarea de un proyecto y, para su comprobación, se realizará la suma agregada del tiempo empleado en la tarea por todas las dedicaciones de todos los participantes del proyecto dentro del periodo de tiempo indicado (incluidas las fechas de inicio y fin).
3. Alarmas de un participante en una tarea. Asociadas a un participante y a una tarea del mismo proyecto. Para su comprobación se deberá sumar todas las dedicaciones que realizó el participante dentro del periodo de tiempo indicado (incluido el inicio y el fin).

Hay dos tipos de alarmas que permiten limitar el conjunto de participantes que ven la alarma y los avisos que producen las alertas:

1. Gestor: Solo el gestor del proyecto ve la alarma y los avisos que producen las alertas.
2. General: El gestor y todos los participantes implicados en la alarma la pueden ver, así como los avisos que producen las alertas.

Dependiendo del modelo de alarma se podrán dar algunos de los siguientes estados:

1. Desactivada: La alarma no se ha activado, no dará ninguna alerta.
2. Activada: La alarma está activa, pero no se ha excedido el número de horas.
3. Alerta: El número de horas ha sido excedido, nadie ha atendido el aviso de la alerta.
4. Parada: El gestor y el participante (en su caso) han atendido el aviso de la alerta.
5. Parada-Gestor: El gestor del proyecto ha respondido al aviso de la alerta, pero el participante no lo ha hecho todavía.
6. Parada-Usuario: El aviso ha sido recibido por el participante implicado, pero el gestor no ha respondido al aviso de la alerta.

NOTA: El uso y el significado depende de cada modelo de alarma:

B.4.2.1 ALARMAS DE TIPO TAREA

Las alarmas de tipo tarea formarán los modelos 1 y 2. El primero cubrirá las alarmas asociadas a toda la dedicación de una tarea (incluidas en la clase "Tarea") y el segundo estará compuesto por las alarmas asociadas a la dedicación en una tarea dentro de un periodo dado (clase "Maxdedicación"). A continuación se enumeran los valores de los tipos y estados que se utilizarán para ambos modelos.

Tipos

1. Gestor.
2. General (Todos los participantes del proyecto podrán ver la alarma y recibirán las alertas. solo el gestor podrá dar por atendida una alerta).

Estados

1. Desactivada.
2. Activada.
3. Alerta.
4. Parada.

B.4.2.2 ALARMAS DE TIPO PARTICIPANTE

Estas alarmas conformarán el modelo 3 (asociadas a la clase “Limiteded dedicación”). Cada alarma estará asociada a un único par tarea-participante de un mismo proyecto. A continuación se enumeran los valores de los tipos y estados asociados a este modelo:

Tipos

1. Gestor.
2. General (El participante asociado con la alarma podrá verla y podrá atender una alerta de la misma).

NOTA: en ambos casos el participante podrá cambiar el estado de la alarma de forma implícita modificando sus valores de dedicación en la tarea del proyecto.

Estados

1. Desactivada.
2. Activada.
3. Alerta.
4. Parada.
5. Parada-Gestor (Este estado solo se dará si la alarma es de tipo “General”).
6. Parada-Usuario (Este estado solo se dará si la alarma es de tipo “General”).

B. 5 CAPA DE DATOS

En el modelo de programación por capas, la capa de datos tiene la responsabilidad de facilitar los datos a las demás capas, permitiéndoles abstraerse del método de almacenamiento. La capa de datos está formada por una clase que implemente el estereotipo “Data Access Object” o “Dao” que se describe en el apartado B.5.1.

La utilización del estereotipo puede, en un futuro, facilitar la modificación del modelo de almacenamiento, permitiendo crear otras clases que lo implementen, facilitando el acceso a distintas fuentes de datos. En este momento solo se creará una única clase que lo implemente, permitiendo el acceso a la base de datos que se ha definido en el capítulo B.3. Los objetivos de este estereotipo son los siguientes:

1. Cargar datos.
2. Crear datos.
3. Guardar datos
4. Borrar datos.
5. Realizar búsquedas complejas sobre los datos.

La capa de datos se encargará del mapeo Objeto-Relacional entre los datos de la aplicación y el repositorio de los datos. Tal como se ha mencionado en el capítulo anterior los atributos de los datos del negocio tienen el mismo nombre que los campos de las tablas, igualmente las clases tienen el mismo nombre que las tablas, indicando cómo se realizará el mapeado.

B.5.1 ESTEREOTIPO DAO

La clase que conforme la capa de negocio deberá implementar el estereotipo Dao con los siguientes métodos:

- vacío crear (PojoDedicación pojo)
 - Guarda un POJO que no existía en la aplicación.
- vacío guardar (PojoDedicación pojo)
 - Persiste los datos del POJO (que ya existía) en el medio de almacenamiento.
- vacío guardar (PojoDedicación pojo, Boolean forzarCrear)
 - Persiste los datos del POJO en el medio de almacenamiento. Si “forzarCrear” es verdadero y el POJO no existe en la aplicación entonces se crea.

- vacío guardar (Colección <PojoDedicación> pojos)
 - Guarda una colección de POJO s de forma persistente.
- PojoDedicación cargar (Clase clase, Clave clave)
 - Carga el POJO de la clase que tenga la clave dada. Si no existe devuelve null.
- PojoDedicación cargar (Clase clase, Mapa claves)
 - Carga el POJO de la clase que coincida con el conjunto de claves dado. Claves debe contener un conjunto de pares (clave, valor) que sea único (dos POJO s con las mismas propiedades deberán ser el mismo), si no el comportamiento no será predecible. Si no hay ninguno devuelve null.
- Lista <PojoDedicación> cargar (Clase clase)
 - Carga todos los POJOs de la clase dada.
- Lista buscar (Clase clase, Cadena filtro, Cadena orden)
 - Busca todos los POJOs de la clase mediante un filtro y los ordena.
- Lista buscar (Cadena búsqueda)
 - Realiza una búsqueda directa.
- Long contar (Clase clase, Cadena filtro)
 - Cuenta el número de POJOs de una clase que cumplen el filtro aplicado.
- Long contar (Clase clase, Cadena filtro, Cadena atributo)
 - Cuenta el número de POJOs de la clase dada filtrados por filtro. El parámetro “atributo” indica que atributo de la clase se empleará para contar. Este método se debe emplear para contar elementos de una clase que tenga una clave embebida (claves compuestas).
- Decimal sumar (Clase clase, Cadena filtro, Cadena atributo)
 - Suma los valores del atributo con el nombre pasado en el parámetro “atributo” contenido en los POJOs filtrados en la búsqueda. El atributo obligatoriamente debe ser de tipo numérico y debe pertenecer a la clase.
- vacío borrar (PojoDedicación pojo)
 - Borra de forma permanente el POJO en el medio de almacenamiento.
- Vacío borrar (Colección <PojoDedicación> pojos)
 - Borra de forma permanente todos los POJOs de la colección.
- Errores validar (PojoDedicación, pojo)
 - Valida un POJO y devuelve el conjunto de errores producidos. Si el conjunto es vacío los datos del POJO son correctos.

Nota: en todos los casos el parámetro “clase” será la definición de una clase que deberá implementar “PojoDedicación”.

B.6 CAPA DE NEGOCIO

La capa de negocio da servicio a la capa de presentación facilitándole las operaciones relacionadas con la lógica de negocio que realiza la aplicación. No incluirá las operaciones relacionadas con la configuración ni mantenimiento del sistema (ver capítulo B.7 “Otros componentes”). La capa se comunica con cada una de las otras dos capas de forma distinta. Recibe mensajes desde la capa de presentación para realizar las operaciones que piden los usuarios y envía mensajes a la capa de datos para acceder al repositorio del sistema (base de datos).

La capa de negocio dispone de una interfaz que la capa de presentación utiliza para comunicarse con ella. La interfaz se distribuirá entre varios componentes especializados en los distintos tipos de información. El conjunto cubre los casos de uso relacionados con los gestores, los participantes y el administrador, con la excepción de la gestión del sistema. Los componentes trabajarán con distintos tipos de datos, principalmente con los POJOs descritos en el capítulo B.4.

B.6.1 COMPONENTES

El diagrama de la Figura B.6.1 muestra los distintos componentes en que se divide la capa de negocio y las dependencias entre ellos. Los siguientes apartados describen las clases que los componen (se ha incluido los métodos de la clase “GestiónProyectos” que forman parte de la interfaz del componente “Gestión Proyectos” como un ejemplo del resto de las interfaces, que no se han incluido en este documento).

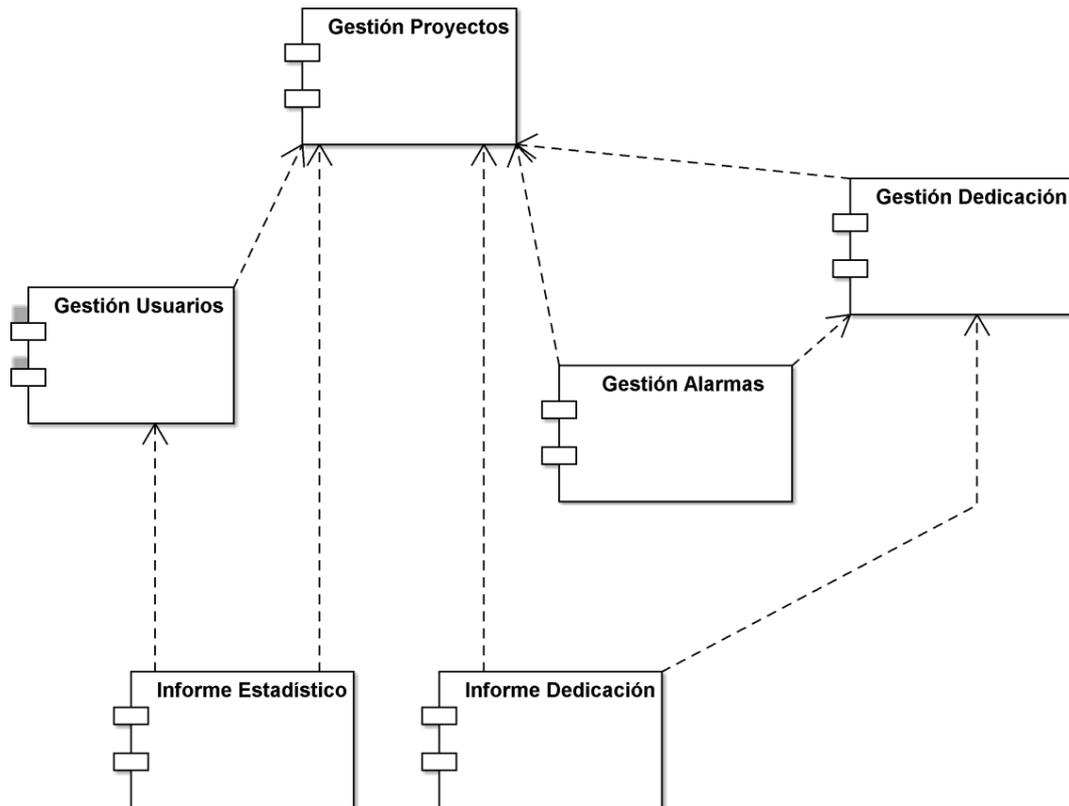


Figura B.6.1 Componentes de la capa de negocio

B.6.1.1 GESTIÓN DE LOS PROYECTOS

Este componente se encarga de la gestión de los proyectos, permitiendo su creación, almacenamiento y destrucción. Facilitará la gestión de los participantes y las tareas de los proyectos. Cubrirá los requisitos de los casos de uso CU01, CU08 y dará soporte a los demás componentes. Se compondrá de las clases que se detallan a continuación.

Gestión de los proyectos

Desarrolla las necesidades de gestión de los proyectos. Cubrirá los requisitos RF6 y RF9.

- 1) void crearProyecto
 - Proyecto proyecto
 - Valida y crea el proyecto.
- 2) Proyecto buscarProyecto
 - Integer código
 - Busca y devuelve el proyecto que tenga el código pasado como parámetro.

- 3) Proyecto buscarProyecto
 - Integer código
 - String login
 - Busca y devuelve el proyecto que tenga el código pasado como parámetro, antes comprueba que el login sea igual al del gestor del proyecto.
- 4) Proyecto buscarProyecto
 - Integer código
 - String login
 - String estado
 - Busca y devuelve el proyecto que tenga el código pasado como parámetro, antes comprueba que el login sea igual al del gestor del proyecto y que el proyecto se encuentre en el estado dado.
- 5) void guardarProyecto
 - Proyecto proyecto
 - Valida y guarda el proyecto.
- 6) void guardarDatosProyecto
 - Proyecto proyecto
 - Valida y guarda el proyecto sin alterar su estado.
- 7) Proyecto abrirProyecto
 - Integer código
 - String loginGestor
 - Busca, cambia el estado del proyecto a “ABIERTO” y lo guarda. Primero comprueba que el login coincide con el del gestor del proyecto.
- 8) Proyecto abrirProyecto
 - Proyecto proyecto
 - Cambia el estado del proyecto a “ABIERTO” y lo guarda, antes se dará de baja a todos sus participantes.
- 9) Proyecto cerrarProyecto
 - Integer código
 - String loginGestor
 - Busca, da de baja a todos sus participantes, cambia el estado del proyecto a “CERRADO” y lo guarda. Primero comprueba que el login coincide con el del gestor del proyecto
- 10) Proyecto cerrarProyecto
 - Proyecto proyecto
 - Da de baja a todos sus participantes y cambia el estado del proyecto a “CERRADO” y lo guarda.
- 11) void borrarProyecto
 - Integer código
 - Busca el proyecto, comprueba que su estado sea “CERRADO” y lo borra del sistema. De forma implícita se borrará a todos los participantes, tareas, dedicaciones y alarmas del proyecto.

- 12) Long numProyectos
 - String estadoProyecto
 - String estadoGestor
 - Devuelve el número de proyectos filtrados por el estado de su gestor y del proyecto.
- 13) Long numProyectos
 - String estadoProyecto
 - String estadoGestor
 - String grupo
 - Devuelve el número de proyectos pertenecientes a un grupo filtrados por el estado de su gestor y el del proyecto.
- 14) Long numProyectosParticipantes
 - String estadoProyecto
 - Long minParticipantes
 - Long maxParticipantes
 - Devuelve el número de proyectos en un estado que tengan asociados un número mínimo y máximo de participantes.
- 15) Long numProyectosTareas
 - String estadoProyecto
 - Long minTareas
 - Long maxTareas
 - Devuelve el número de proyectos que se encuentren en un estado y que contengan un número mínimo y máximo de tareas.
- 16) List <Proyecto> listarProyectos
 - String loginUsuario
 - String estadoProyecto
 - String estadoParticipante
 - Devuelve los proyectos en los que participa un usuario filtrados por el estado del proyecto y el del participante.
- 17) List <Proyecto> listarProyectosGestionados
 - String loginGestor
 - String estado
 - Devuelve una lista de proyectos filtrados por el login del gestor y el estado del proyecto.
- 18) List <Proyecto> listarProyectosGrupo
 - String nombreGrupo
 - String estado
 - Devuelve una lista de proyectos filtrados por el grupo y el estado del proyecto.
- 19) List <Participante> listarParticipantesProyecto
 - Integer código
 - String estado
 - Devuelve una lista con los participantes de un proyecto que se encuentre en el estado dado.

20) Long numParticipantesProyecto

- Integer código
- String estado
- Devuelve el número de participantes en un proyecto, que se encuentre en el estado dado.

Gestión de los participantes

Esta clase se encarga de gestionar las participaciones de los usuarios en los proyectos. Cumple con los requisitos RF8, RF14, RF18 y Rf19.

Gestión de las tareas

Contiene los métodos necesarios para gestionar las tareas de los proyectos. Resuelve el requisito RF7.

B.6.1.2 GESTIÓN DE LOS USUARIOS

Este componente se encarga de manejar toda la información relacionada con los usuarios del sistema. Permitirá crear, guardar, borrar, dar altas y bajas y listar tanto a los grupos como a los usuarios. También permitirá modificar el perfil de un usuario. Este componente debe cubrir los casos de uso CU04 y CU06.

Gestión de los grupos

Facilitará los métodos para la gestión de los grupos de usuarios. Cumplirá una parte del requisito RF2.

Gestión de los usuarios

Se encargará de la gestión de los usuarios del sistema, permitiendo crearlos, darlos de alta y baja. Deberá resolver los requisitos RF2, RF5 y RF17.

Gestión de la agenda de los usuarios

Contendrá los métodos que permitirán a un usuario crear y modificar los números de teléfono y las direcciones de correo electrónico de su perfil. Ofrecerá métodos para cumplir parte de los requisitos RF5 y RF17.

Listado de los usuarios

Contendrá los métodos para listar a los usuarios del sistema. Cumplirá con los requisitos RF2 y dará soporte a otros componentes para listar los usuarios.

B.6.1.3 GESTIÓN DE LAS DEDICACIONES

Facilitará la creación y modificación de los datos de la dedicación que los usuarios realizan en los proyectos. También devolverá listas y consultas de la dedicación agregada en proyectos y participantes. Este componente debe cubrir el caso de uso CU9 y ofrecer soporte para la consulta de la dedicación (CU02, CU03 y CU10).

Gestión de la dedicación

Ofrecerá métodos para introducir la dedicación que realice un participante de un proyecto. Se encargará del requisito RF20.

Dedicación agregada

Ofrecerá las funciones que permitan la consulta de la dedicación de los participantes en los proyectos, dando soporte a éste y otros componentes.

B.6.1.4 GESTIÓN DE LAS ALARMAS

Ofrece la funcionalidad necesaria para el manejo de las alarmas de la dedicación. Facilitará la creación, guardado, borrado y listado de las alarmas de las tareas y participantes de los proyectos. También permitirá el cambio del estado de las alarmas que hayan sido modificadas de forma implícita cuando los participantes de los proyectos introducen su dedicación. Cubrirá los casos de uso CU2 y CU10.

Alarmas de tipo participante

Permitirá crear, consultar y modificar las alarmas de tipo participante, según lo especificado por el requisito RF15.

Alarmas de tipo tarea

Permitirá crear, consultar y modificar las alarmas de tipo tarea, según lo especificado por el requisito RF15.

Listado de las alarmas

Facilitará los métodos necesarios para listar las alarmas de la dedicación. Cumplirá con los requisitos RF16 y RF23.

Actualizar las alarmas

Contendrá un conjunto de métodos que facilitarán la actualización de las alarmas de la dedicación.

B.6.1.5 GENERACIÓN DE INFORMES ESTADÍSTICOS

Permitirá la generación de informes sobre estadísticas de los datos de la aplicación. Esto permitirá al administrador tener una visión de la utilización del sistema, p.e. el número de usuarios de tipo “NORMAL” o el número de alarmas asociadas a proyectos. Cubrirá el caso de uso CU07.

Generación de un informe estadístico del sistema

Crearé informes con las estadísticas del sistema. Para crear un informe se deberá realizar un diseño (ver la clase siguiente).

Diseño de un informe estadístico

Esta clase contendrá la configuración del contenido de un informe, el cual se utilizará para crearlo. La siguiente lista describe las distintas secciones que podrá incluir un informe. Cada una de las subsecciones se plasmará en una o varias tablas dentro del documento.

- Usuarios
 - Número de usuarios (todos, gestores, normales, activos y no activos).
- Grupos
 - Número de grupos, número de grupos vacíos.
 - Usuarios por grupo (todos, gestores, normales, activos y no activos).

- Proyectos
 - Número de proyectos (total, abiertos, cerrados, gestor activo o inactivo).
 - Número de proyectos en cada grupo (total, abiertos, cerrados) y número de participantes en el grupo (total, media por grupo y por proyecto).
- Alarmas de tipo participante (se mostrarán el número de ellas en cada estado y tipo, así como el total).
 - Número de alarmas (todas, generales o del gestor, proyecto abierto o cerrado)
 - Alarmas por proyecto (todas, generales o del gestor, proyecto abierto o cerrado).
 - Número de alarmas en cada grupo (todas, generales o del gestor, proyecto abierto o cerrado).
 - Alarmas por proyecto en cada grupo (todas, generales o del gestor, proyecto abierto o cerrado).
- Alarmas de tipo tarea (se mostrarán el número de ellas en cada estado y tipo, así como el total).
 - Número de alarmas (todas, generales o del gestor, proyecto abierto o cerrado).
 - Alarmas por proyecto (todas, generales o del gestor, proyecto abierto o cerrado).
 - Número de alarmas en cada grupo (todas, generales o del gestor, proyecto abierto o cerrado).
 - Alarmas por proyecto en cada grupo (odas, generales o del gestor, proyecto abierto o cerrado).
- Uso del sistema
 - Datos del número de operaciones realizadas en el sistema.

B.6.1.6 GENERACIÓN DE INFORMES DE LA DEDICACIÓN

Permitirá crear informes basados en la dedicación realizada en un proyecto, creados por un participante o un gestor del proyecto. Cubrirá el uso de uso CU03.

Generación de un informe de la dedicación

Crearé un informe sobre la dedicación de los proyectos y sus participantes. Para ello se deberá configurar los datos que se desean incluir en el informe. Cumplirá con los requisitos RF10, RF11, RF12, RF13, RF21 y RF22.

Diseño de un informe de la dedicación de un proyecto

Describirá el contenido de un informe de la dedicación en un proyecto.

Diseño de un informe de varios participantes

Describirá el contenido de un informe de la dedicación de varios usuarios. Servirá tanto para los informes de un gestor como los de un participante.

Diseño de un subinforme de una participación

Describirá el contenido de un subinforme que se añadirá a uno de los informes de varios participantes. Este subinforme contendrá los datos del usuario, los intervalos de tiempo que se incluirán y la lista de las participaciones.

Diseño de un subinforme de una participación de un usuario

Describirá el contenido de un subinforme que se añadirá a uno de los informes de un usuario. Permite añadir una participación en un proyecto al informe con la lista de las tareas que se desean incluir.

Estructura y diseño de un informe

Cada petición de un informe deberá ir acompañada de una descripción de la información que se desea mostrar. Para facilitar la creación de un informe se dispondrá de clases que contengan su diseño. A continuación se detalla la estructura de un informe.

Todo informe se iniciará con una introducción que contendrá un título que por defecto será "INFORME DE LA DEDICACIÓN". Debajo del título se colocará la fecha y hora de la creación del informe. Seguidamente se mostrarán los datos del proyecto al que hace referencia el informe (Código del proyecto, acrónimo, nombre, estado, fecha de inicio y de fin (si su estado es "ABIERTO" se sustituirá por guiones), grupo de trabajo, nombre del gestor (Apellido 1, apellido 2 (si existe) y nombre).

Si el proyecto es creado por un usuario se mostrará su login, su nombre (Apellido 1, apellido 2 (si lo hay) y nombre), su estado y las fechas de incorporación y finalización de la participación en el proyecto (si está activo se sustituirá por guiones).

Tras esta información se añadirá la descripción del informe. Por defecto esta sección no se incluirá.

En el caso de que el informe fuese generado por el gestor, se podrá añadir la lista de participantes en el proyecto. Esto será opcional y por defecto no se incluirá. Por cada participante se mostrará su login, su nombre (Apellido 1, apellido 2 y nombre) y su estado en el proyecto.

El resto del documento se compondrá de uno o más bloques que contendrán diversa información, cada uno con una parte del informe.

Cada bloque dispondrá de un título, una descripción y un cierre (al final del apartado) Por defecto el título será "INFORME". La descripción y el cierre serán opcionales y no se mostrarán si no se especifica ningún texto para ellos.

En los informes generados por un participante cuyo intervalo sea un día, una semana o un mes se mostrará la lista de los nombres de las tareas cuya dedicación será agregada. Después se verá la tabla con la dedicación acumulada. Cada línea contendrá un sub-intervalo de tiempo (un día, una semana, un mes). Se mostrará el inicio y fin del intervalo (el segundo si no es diario), luego la dedicación agregada del participante en todas las tareas en el periodo, seguido de otros datos especificados más adelante. La última línea de la tabla podrá mostrar, según la opción, el total de la dedicación en todo el intervalo.

Por lo contrario, cuando el intervalo de tiempo sea el total, no se mostrarán la lista con las tareas, sino que cada fila de la tabla mostrará el nombre y la dedicación agregada de la tarea en el periodo, el cual se mostrará delante de la tabla. Si la opción mostrar total está activada, la última línea de la tabla mostrará la suma de la dedicación.

En ambos casos, y de forma opcional, se podrá añadir el total de la dedicación y el porcentaje que supone de la dedicación realizada por todos los usuarios en las tareas especificadas dentro del periodo dado, el número de alarmas del participante y el número de ellas que han producido una alerta. El número de alarmas y alertas se referirá solo a las alarmas que sean visibles por el usuario (si es un participante solo verá las de tipo "General" y si es un gestor todas).

Los subinformes generados por un gestor se diferenciarán por el hecho de que podrán incluir a varios participantes. Esto significará que se mostrará la lista de todos los participantes incluidos junto con la lista de las tareas.

B.6.2 COMPORTAMIENTO DE LAS ALARMAS

A continuación se va a describir el funcionamiento de las alarmas de la dedicación. El objetivo de una alarma es advertir al gestor y a los participantes de un proyecto del exceso de horas acumuladas que se han empleado en una tarea del proyecto.

Las alarmas serán creadas por el gestor de un proyecto, éste podrá consultar las alarmas asociadas a los proyectos que gestione. Además podrá modificar sus valores cuando desee (siempre que el proyecto no haya sido cerrado).

Un participante solo podrá ver las alarmas de los proyectos en los que participe cuyo tipo sea “GENERAL”, no tendrá conocimiento de las que tengan por tipo “GESTOR”. Podrá ver las alarmas de tipo tarea (“Maxdedicación”) y aquellas de tipo participante (“Limitededicación”) que estén asociadas con él. No debería poder visualizar las alarmas de tipo participante asociadas con otros usuarios, ni las incluidas en proyectos en los que no participa.

Los modelos y los estados y tipos de alarmas han sido descritos en el apartado B.4.3. A continuación se describe el comportamiento de las alarmas en función de las acciones de los usuarios.

B.6.2.1 CREACIÓN Y MODIFICACIÓN DE UNA ALARMA

Una alarma deberá ser creada o modificada exclusivamente por un usuario gestor de un proyecto, la capa de negocio deberá comprobar que el proyecto se encuentre abierto y que el usuario sea el gestor del mismo.

Las alarmas estarán asociadas a una tarea del proyecto y, en caso de una alarma de tipo participante, también con un participante del proyecto. Las alarmas tendrán asociado un periodo de tiempo, salvo las que incluyen toda la dedicación de una tarea.

Además cada alarma llevará asociado un tipo y un estado. El usuario gestor deberá asignar uno de los siguientes estados “DESACTIVADA” o “ACTIVADA”.

Al inicializar una alarma con el segundo valor se deberá comparar el número de horas dedicadas con el valor límite y, si es necesario, cambiar el estado al de “ALERTA”.

Todas las alarmas tendrán un valor que indicará el límite de horas de la dedicación, este valor deberá ser mayor que 0.0 y múltiplo de 0.5. Si no es positivo se producirá un error, además el sistema lo redondeará a múltiplos de 0.5

Cada modelo de alarma tendrá una forma distinta de contabilizar el tiempo dedicado a la tarea. Una alarma general (modelo 1) tendrá en cuenta todas las dedicaciones realizadas por todos los usuarios del proyecto en la tarea.

Una alarma de tipo tarea (modelo 2) limitará la dedicación al intervalo especificado por los valores “inicio” y “fin”. Una alarma de tipo participante (modelo 3) contabilizará solo la dedicación del usuario asociado realizada dentro del periodo inicio y fin (ambos incluidos).

Los usuarios que participen en un proyecto no podrán modificar los valores de las alarmas con la única excepción de su estado. Cuando un usuario introduzca su dedicación en una tarea de un proyecto, el sistema deberá actualizar el estado de todas las alarmas activas y en alerta que se vean afectadas por la tarea, usuario y fecha de la dedicación. Las alarmas que hayan sido atendidas por el usuario o por el gestor no serán modificadas. Además un usuario podrá dar por atendida una alarma, produciendo el cambio de estado de la misma.

Un gestor podrá cambiar el estado de una alarma de forma explícita activándola o desactivándola, así como de forma implícita al dar por atendida una alerta. El siguiente apartado describe estos comportamientos.

Los valores de los atributos tarea, participante, inicio y fin no serán modificables, para ello se deberá crear una tarea nueva.

B.6.2.2 COMPORTAMIENTO DE UNA ALARMA

En función de lo descrito en el capítulo B.4 y en los apartados anteriores, se pueden derivar los 2 siguientes diagramas de estado para definir el comportamiento de las alarmas.

El diagrama de la Figura B.6.2 se utilizará para los modelos 1 y 2 y para el modelo 3 cuando la alarma sea de tipo “GESTOR”. El diagrama de la Figura B.6.3 se aplicará al modelo 3 cuando la alarma sea de tipo “GENERAL”.

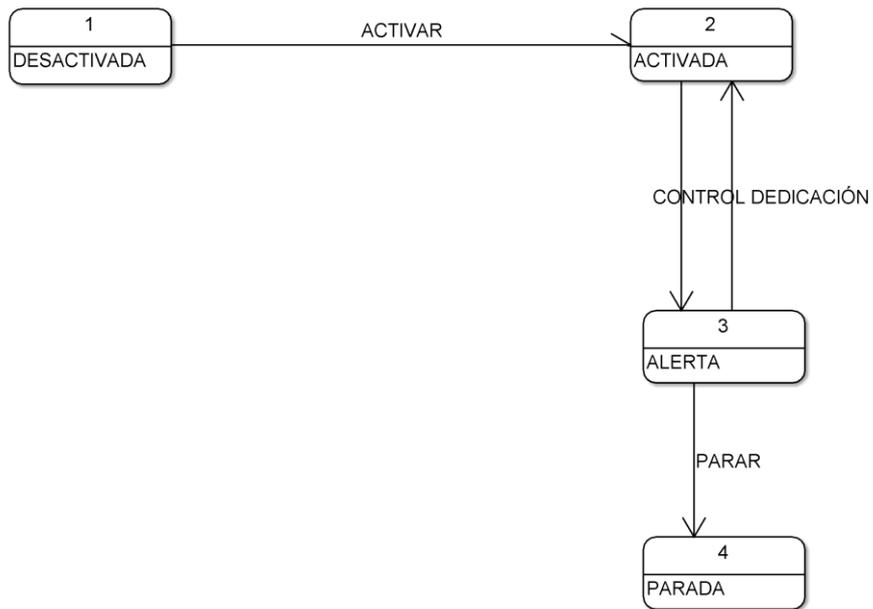


Figura B.6.2 Diagrama de estado 1 de las alarmas

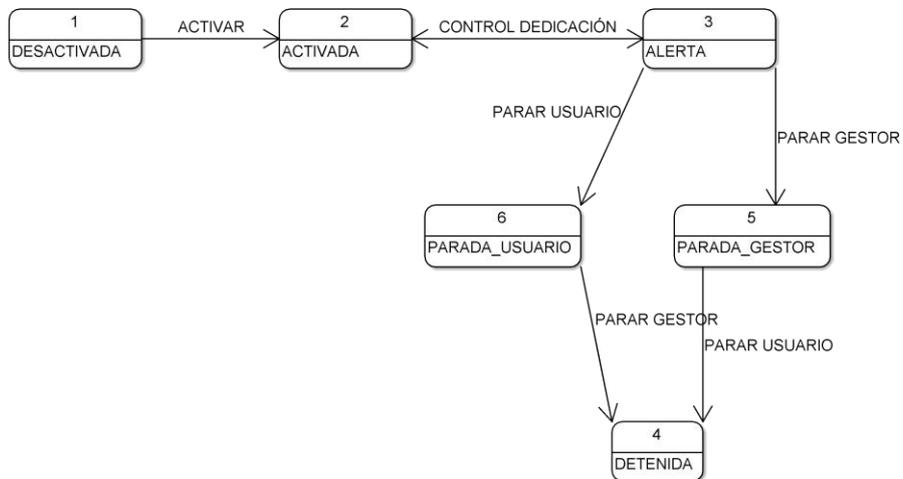


Figura B.6.3 Diagrama de estado 2 de las alarmas

Aunque no aparezca reflejado en los diagramas es posible ir al estado “DESACTIVADA” desde cualquier estado.

Podemos clasificar las transiciones en tres tipos: las generadas explícitamente al guardar los datos de una alarma (solo el gestor), las que se producen explícitamente al dar por atendida una alarma (gestor y participante) y las que se producen implícitamente cuando un participante introduce su dedicación.

B.7 OTROS COMPONENTES

En este capítulo se describen los componentes transversales del sistema que no pertenecen a la lógica de negocio de la aplicación. Estos componentes cubrirán el caso de uso “Gestión sistema”, además de otras funciones que darán soporte a la capa de negocio.

B.7.1 CONFIGURACIÓN

Este componente permitirá acceder y modificar la configuración del sistema. Facilitará la lectura y escritura de los ficheros de configuración y la lectura y escritura de sus propiedades. Deberá dar soporte tanto dentro del Portlet como de la aplicación Web. El componente dará soporte al caso de uso CU05.

El acceso a los ficheros de configuración se podrá realizar en dos niveles, según sea conveniente. El primer nivel solo permitirá leer y sobrescribir el contenido de un fichero de configuración. El segundo nivel dará soporte para el acceso a las propiedades del fichero (lectura y escritura individual).

Cada fichero de configuración consistirá en un archivo de propiedades. En caso de que un fichero de propiedades contenga dos entradas con la misma clave, ambas serán cargadas, pero solo se tendrá en cuenta la última (ver clase “Propiedades”). La función que guarda las propiedades mantendrá todos los comentarios y líneas en blanco, así como el orden de las propiedades tal como estaban al ser leídas. Las nuevas propiedades serán añadidas en una línea nueva al final de la lista.

El sistema deberá disponer de una copia, no modificable por parte de la aplicación para que el administrador pueda restaurar la configuración del sistema. Tanto los ficheros de configuración, como los ficheros con la configuración por defecto se almacenarán en “WEB-INF/config” y “WEB-INF/config/default” respectivamente, ambos tipos de archivos no se incluirán en este componente.

La configuración del programa será compartida por todos los usuarios y los cambios producidos serán visibles por todos ellos. Este componente garantizará la exclusión mutua en el acceso y modificación de las propiedades de la configuración dentro de la misma aplicación. Esto solo garantiza que no se producirán errores al leer o escribir la configuración, pero habrá que tener en cuenta que podrán crearse incoherencias de tipo lector-escritor. Se asumirá

que este problema (común a todos los sistemas) no afectará al correcto funcionamiento de la aplicación y que su solución produciría problemas mucho más graves.

B.7.1.1 NIVEL 1: CONFIGURACIÓN BÁSICA

El nivel uno permitirá el acceso y modificación de los ficheros de configuración... Deberá ser utilizado por aquellas configuraciones que se carguen en componentes que ya disponen de mecanismos para leer los archivos y manejar sus propiedades.

B.7.1.2 NIVEL 2: CONFIGURACIÓN POR PROPIEDADES

Las configuraciones que requieran acceder a las propiedades deberán implementar el segundo nivel. Tanto en la aplicación Web como en el Portlet todos los hilos compartirán las propiedades. Esto significa que solo se realizará una lectura de las mismas y en el caso de ser modificado el fichero se producirá una nueva recarga. Dado que el Portlet no detectará estas modificaciones deberá recargar la configuración por cada lectura.

La implementación se hará de tal manera que la carga de las propiedades sea totalmente transparente a la petición, los cuales se encargarán de comprobar si es necesario hacerlo.

B.7.2 GESTIÓN DE LOS LOGS

El objetivo de este componente es ofrecer las funcionalidades para la gestión de los ficheros de log. Los logs o bitácoras son un medio para registrar los sucesos que ocurren durante la ejecución del programa (errores, advertencias, información, depuración, etc.), los cuales ayudan a la gestión por parte del administrador. Deberá cubrir parte del caso de uso CU05.

La forma de registrar y almacenar estos sucesos vendrá dada por el fichero de configuración, localizado en "WEB-INF/log4j.properties", el cual podrá ser modificado por el administrador (véase el componente "Configuración"). Dada la naturaleza Web de la aplicación, lo lógico es que este registro se almacene en ficheros y que el administrador pueda consultarlos y gestionarlos mediante la aplicación. Esto no significa que no se puedan utilizar otros métodos para su almacenamiento y/o notificación (pues esto dependerá de la configuración dispuesta en "log4j.properties"), tan solo hay que considerar que el alcance de este componente se restringe a los logs almacenados en ficheros. Para utilizar otros medios será necesario crear/usar otras formas de acceso no contempladas en este componente.

El fichero “log4j.properties” deberá cumplir con la especificación dada por log4j y a su vez con el formato de ficheros de propiedades (“.properties”). Consistirá en un fichero de propiedades (“.properties”) que contendrá una o más definiciones de Loggers, cada uno de ellos irá asociado a uno o más appenders, los cuales pueden ser compartidos por más de un Logger. Las funciones de este componente serán:

Clase para la gestión de los logs

Esta clase contendrá los métodos necesarios para la gestión de los ficheros de logs. Permitirá las funciones descritas previamente. La clase mantendrá una lista con todos los appenders. Será recomendable que se llame a la función “actualizar” antes de realizar alguna consulta, especialmente si se ha actualizado la configuración del logs.

B.8 CAPA DE PRESENTACIÓN DE LA APLICACIÓN WEB

Este capítulo y el siguiente presentan el diseño de la capa de presentación. Dentro de este capítulo se mostrará la parte que afecta a la aplicación Web, mientras que en el siguiente se desarrollará la parte que se refiere al Portlet. Cada apartado describe las GUI de una de las secciones de la aplicación.

La GUI de la aplicación Web se compondrá de un Font-End y un Back-End. La parte del Front-End será utilizado por los usuarios de tipo gestor para realizar la gestión de la dedicación y el Back-End será utilizado por el administrador del sistema para su configuración.

Las GUIs serán generadas con código HTML de manera dinámica, utilizando el lenguaje JSP, con cierto apoyo de etiquetas de Java. El código solo incluirá los datos y la estructura del documento, apoyándose en hojas de estilo (CSS) para definir el aspecto de las páginas.

La información mostrada por las GUIs se obtendrá de la capa de negocio añadiéndole los textos e imágenes necesarios para su comprensión, así como formularios para que los usuarios puedan introducir nuevos datos y realizar búsquedas. Para la realización de ciertas funciones en el navegador del cliente se añadirá lenguaje de script con soporte de la tecnología AJAX para la comunicación asíncrona con el servidor.

Las interfaces estarán compuestas de una cabecera (optativa), el contenido principal y un pie de página. Para nombrar a cada una de ellas se utilizará una clave (que se empleará en la implementación) que designará, de forma unívoca, a cada una de ellas y que seguirá las siguientes reglas:

1. Comenzarán por la letra 'V'
2. La segunda letra denotará el usuario
 - 'A' administrador
 - 'G' gestor
 - 'C' común a ambos
3. Una o más letras denotando la sección
4. Dos números comenzando en '01'

B.8.1 CABECERA

Las GUIs de la parte privada del Front-End y del Back-End mostrarán una cabecera con las distintas opciones disponibles, Incluyendo un menú principal (adaptado a cada usuario y que no variará) con enlaces a las secciones a las que puede acceder cada tipo de usuario. Dependiendo de la sección en la que se encuentre el usuario, se podrá ver un menú secundario con las diversas opciones de la sección. En algunas secciones este menú no aparecerá. Además se incluirá un menú auxiliar con enlaces que permitirán salir de la aplicación y acceder a la sección que facilita modificar el perfil del usuario. Dispondrá de un botón que permitirá cambiar el contenido de la sección por el de la ayuda, el contenido de la ayuda se colocará encima del contenido de la sección mostrada hasta que se pulse el botón para ocultarla.

Cada menú podrá ser identificado por su clave, compuesta por un acrónimo (sin puntos) que permitirá identificarlo y estará compuesto según las reglas siguientes:

1. Comenzará por la letra 'M'.
2. El tipo de menú
 - 'P' principal
 - 'S' secundario
 - 'Aux' auxiliar
3. En el caso de un menú principal o secundario
 - 'A' administrador
 - 'G' gestor
4. Una secuencia de números comenzando en '01'

B.8.1.1 MENÚS DEL ADMINISTRADOR

Menú principal del administrador (MPA01)

El menú principal del administrador le ofrecerá acceso a la configuración del sistema, la gestión de logs y la gestión de los usuarios.

Nombre	Evento	Descripción
Estado	infoEstado	Enlace a la sección que muestra la información del estado del sistema.
Configuración	listarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.
Logs	listarLogs	Enlace que permite el acceso a los ficheros de logs.

Nombre	Evento	Descripción
Usuarios	listarUsuarios	Facilita el acceso a la sección que permite gestionar los usuarios del sistema.
Estadísticas	nuevolInformeEstadístico	Permite mostrar la sección dedicada a la generación de estadísticas del sistema.

Menú secundario “Configuración” (MSA01)

El menú dispondrá de una lista con las opciones de configuración disponibles (general, logs,...).

Nombre	Evento	Descripción
General	editarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.
Admin	editarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.
Gestor	editarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.
Participantes	editarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.
Logs	editarConfiguración	Enlace a la sección que facilita la modificación de los ficheros de configuración.

Menú secundario “Logs” (MSA02)

El menú contendrá un enlace que llevará a la sección principal de la gestión de los ficheros de logs.

Nombre	Evento	Descripción
Listado	listarLogs	Enlace que muestra la lista de ficheros de logs disponibles.

Menú secundario “Gestión de los usuarios” (MSA03)

El menú facilitará el acceso a las secciones que permiten la gestión de los usuarios y de los grupos del sistema.

Nombre	Evento	Descripción
Usuarios	listarUsuarios	Enlace a la sección que lista los usuarios.
Nuevo Usuario	nuevoUsuario	Enlace a la sección que permite crear un nuevo usuario.
Grupos	listarGrupos	Enlace a la sección que muestra la lista de los grupos.
Nuevo grupo	nuevoGrupo	Enlace a la sección que facilita crear un nuevo grupo.

Menú secundario “Estadísticas” (MSA04)

Permite acceder a la sección que facilita la generación de informes estadísticos del sistema.

Nombre	Evento	Descripción
Nuevo informe	nuevoInformeEstadístico	Enlace a la sección que permite introducir los parámetros de un nuevo informe.

B.8.1.2 MENÚS PARA LOS GESTORES

Menú principal de los gestores (MPAg01)

El menú principal le permitirá acceder a las secciones disponibles para los gestores.

Nombre	Evento	Descripción
Proyectos	listarProyectos	Enlace a la sección que muestra los proyectos gestionados.
Nuevo proyecto	nuevoProyecto	Enlace que permite iniciar el proceso de creación de un proyecto.
Informes	nuevoInforme Proyecto	Enlace que permite crear un nuevo informe de los proyectos.
Alarmas tareas	listarAlertasTarea	Enlace a la lista de alertas de tipo tarea.
Alarmas participantes	listarAlertas Participante	Enlace a la lista de alertas de tipo participante

Menú secundario “Gestión de un proyecto” (MSP01)

Este menú muestra las acciones que un gestor puede realizar durante la consulta de un proyecto.

Nombre	Evento	Descripción
Información	infoProyecto	Enlace que muestra la información del proyecto.
Añadir Tareas	nuevasTareas	Enlace que permite añadir nuevas tareas.
Participantes	listarParticipantes	Enlace que muestra la lista de participantes del proyecto.
Añadir Participantes	nuevosParticipantes	Enlace a la sección que permite añadir nuevos participantes.
Alarmas Tareas	listarAlarmasTarea	Enlace a la sección que gestiona las alarmas de un proyecto.
Alarmas Participantes	listarAlarmas Participante	Enlace a la sección que gestiona las alarmas de un proyecto.
Informe Proyecto	nuevoInformeProyecto	Enlace a la sección que facilita la creación de informes de la dedicación del proyecto.

Menú secundario “Gestión de las alarmas de los participantes” (MSP02)

Este menú muestra las acciones que un gestor puede realizar mientras consulta las alarmas de los participantes.

Nombre	Evento	Descripción
Recientes	listarAlertasParticipante	Enlace que muestra la lista de las alarmas de tipo participante más recientes y que hayan producido una alerta.
Listar	listarAlarmasParticipante	Enlace que muestra la lista de alarmas de tipo participante.
Nueva	nuevaAlarmaParticipante	Enlace que permite crear una alarma de tipo participante.

Menú secundario “Gestión de las alarmas de las tareas” (MSPg03)

Este menú muestra las acciones que un gestor puede realizar mientras consulta las alarmas de las tareas.

Nombre	Evento	Descripción
Recientes	listarAlertasTarea	Enlace que muestra la lista de las alarmas de tipo tarea más recientes y que hayan producido una alerta.
Listar	listarAlarmasTarea	Enlace que muestra la lista de alarmas de tipo tarea.
Nueva	nuevaAlarmaTarea	Enlace que permite crear una alarma de tipo tarea.

Menú secundario “Generación de informes” (MSP04)

Menú que se muestra mientras se trabaja con informes de los proyectos.

Nombre	Evento	Descripción
Nuevo	nuevoInformeProyecto	Enlace que muestra un formulario para crear un informe de los proyectos.

B.8.1.3 OTROS MENÚS

Menú auxiliar (maux01)

Nombre	Evento	Descripción
Contacto	infoContactoWeb	Enlace que muestra la información de contacto con el administrador.
Perfil	editarPerfilWeb	Enlace que muestra la información personal del usuario.
Salir	SalirWeb	Enlace que cierra la sesión del usuario.

Menú secundario “Gestión del perfil” (MSAux02)

La opción “Perfil” descrita en el menú auxiliar (MAux01) tendrá como consecuencia que cambie el menú secundario. Las opciones que mostrará serán las siguientes.

Nombre	Evento	Descripción
Datos	editarPerfilWeb	Enlace a la sección que permite modificar los datos del perfil del usuario.
Teléfonos	editarTeléfonosWeb	Enlace a la sección que permite modificar los datos del perfil del usuario.

Nombre	Evento	Descripción
Emails	editarEmailsWeb	Enlace a la sección que permite modificar los datos del perfil del usuario.

B.8.2 PIE DE PÁGINA

Las interfaces dispondrán de un pie de página que podrá contener el nombre y correo electrónico del administrador del sistema. El administrador podrá cambiar el contenido de ambos datos mediante la configuración del sistema.

B.8.3 INTERFACES DEL ADMINISTRADOR

A continuación se describirán las distintas GUIs que se utilizarán para la administración del sistema, pensadas para ser utilizadas por usuarios con perfiles de administrador.

B.8.3.1 GESTIÓN DEL SISTEMA

Las siguientes interfaces incluirán las funciones que utilizará un usuario de tipo administrador para la gestión del sistema que se reflejan en el requisito RF3 y que cubre el caso de uso CU03. Estas funciones incluyen la modificación de los ficheros de configuración y la gestión de los ficheros de logs (consulta y borrado).

Estado del sistema (VAS01)

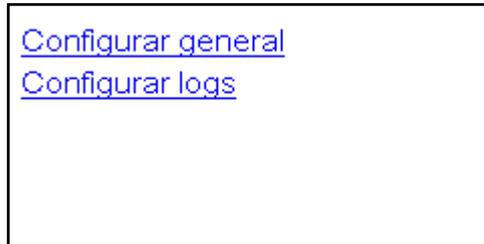
Contendrá un pequeño resumen del estado del sistema y de su configuración. Permitirá abrir y cerrar el acceso a todos los usuarios del sistema (con excepción de los administradores).

Estado del sistema:
ABIERTO

Menú secundario	Ninguno
-----------------	---------

Configuración del sistema (VAS02)

Muestra un listado con todos los ficheros de configuración disponibles, permitiendo acceder a cada uno de ellos para modificarlo.



Menú secundario Ninguno

Fichero de configuración (VAS03)

Muestra el contenido de uno de los ficheros de configuración. Permitirá modificar el texto y dispondrá de botones para guardarlo, restaurarlo y cargar la configuración por defecto.

Previo al texto se podrá incluir una advertencia de error o de confirmación de un intento previo de guardar la configuración del mismo fichero.



Menú secundario MSA01

Listado de los ficheros de logs (VAS04)

Muestra una lista con los nombres de los ficheros de logs asociados a la configuración de los appenders actuales. La lista estará ordenada por nombre.

La GUI permitirá seleccionar un appender concreto (solo se mostrarán los ficheros según la configuración del appender) o mostrarlos todos. Se podrá borrar y acceder al contenido de un fichero de log de la lista, así como borrar una selección de los mismos.

Appender: Depurador ▾

6 /31 elementos

Tamaño lista 5 Actualizar <- ->

Borrar selección

Nombre	
<input type="checkbox"/> depuracion_2012_06_05.log	ver borrar
<input type="checkbox"/> depuracion_2012_06_04.log	ver borrar
<input type="checkbox"/> depuracion_2012_06_03.log	ver borrar
<input type="checkbox"/> depuracion_2012_06_02.log	ver borrar
<input type="checkbox"/> depuracion_2012_06_01.log	ver borrar

Menú secundario MSA02

Contenido de un fichero de log (VAS05)

Muestra el texto contenido en un fichero de log. Permitirá borrar el fichero de log del sistema.

Fichero de log: "depuracion_2012_06_5.log"

Appender: "depuracion"

Borrar

```

21:27:41 - (configuracion.ConfiguracionLogs)
INFO : Cargando la configuración.
21:27:41 - (configuracion.ConfiguracionLogs)
INFO : Cargando la configuración.
21:27:41 - (configuracion.ConfiguracionLogs)
DEBUG : Añadiendo las propiedades básicas.
21:27:41 - (configuracion.ConfiguracionLogs)
DEBUG : Añadiendo las propiedades básicas.
21:27:41 - (configuracion.ConfiguracionLogs)
DEBUG : Leyendo propiedad 'directorio'.
21:27:41 - (configuracion.ConfiguracionLogs)
DEBUG : Leyendo propiedad 'directorio'.

```

Menú secundario MSA02

B.8.3.2 GESTIÓN DE LOS USUARIOS

El siguiente conjunto de GUIs cubrirá los requisitos funcionales RF1 y RF2 y cubrirá el caso de uso CU06, permitiendo la gestión de los usuarios y de los grupos almacenados en el sistema.

Listado de los grupos (VAU01)

Contendrá una tabla con un subconjunto de los grupos ordenados por su nombre. Inicialmente se mostrarán los primeros grupos de la lista completa y se dispondrán de botones para actualizar la lista y para mostrar el siguiente y el anterior tramo de la lista.

Se añadirán dos entradas de texto que permitan indicar el índice del primer elemento de la lista y el número de elementos que se desea mostrar. Al pulsar el botón para actualizar se recargará la tabla con el número de elementos indicados a partir del índice. Si no hubiera suficientes elementos se mostrarán los que estén disponibles. Si cualquiera de las entradas son igual o menor que 0 se tomará el valor por defecto. Ambas entradas serán rellenadas por el último valor introducido o el valor por defecto.

También dispondrá de otra entrada de texto que permitirá filtrar la lista de los grupos a través de su nombre. Al presionar el botón correspondiente se actualizará la tabla, desde el índice descrito anteriormente, con los grupos resultantes de la búsqueda mediante el filtro introducido.

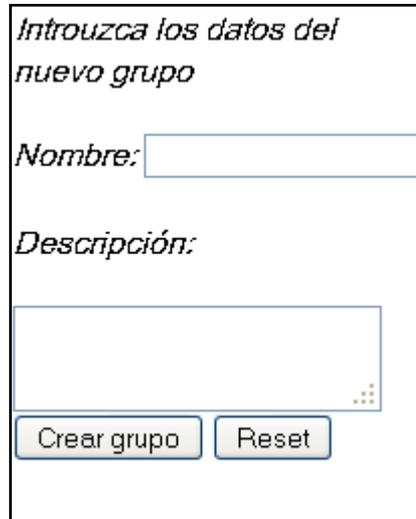
Cada entrada de la lista contendrá el nombre del grupo, el número de usuarios contenidos dentro de él, parte de la descripción del grupo (las primeras palabras) y permitirá la posibilidad de ir a la lista de usuarios del grupo, ver sus datos y borrarlo (si el grupo está vacío)

Filtro nombre: <input type="text" value="Depurador"/>			
<input type="text" value="6"/>		/42 elementos	
Tamaño lista <input type="text" value="5"/>	<input type="button" value="Actualizar"/>	<input type="button" value="←"/>	<input type="button" value="→"/>
Nombre	Descripción	Usuarios	
<input type="checkbox"/> Computación	Estudios de computadores.	55	usuarios info
<input type="checkbox"/> Programación_avanzada	Estudio de la programación	43	usuarios info
<input type="checkbox"/> Computación255	Estudios de computadores en computadores...	0	info borrar
<input type="checkbox"/> ProgGrafica	Gravicos por computador	17	usuarios info
<input type="checkbox"/> Mecanica	Proyectos mecánica.	53	usuarios info

Menú secundario MSA03

Nuevo grupo (VAU02)

Muestra un formulario con los campos necesarios para crear un grupo. Dispondrá de un botón para enviar la información al servidor. El formulario podrá ir precedido de una lista de errores producidos al intentar crear un grupo previamente. En este caso los campos del formulario estarán rellenos con los datos que se enviaron.



Introduzca los datos del nuevo grupo

Nombre:

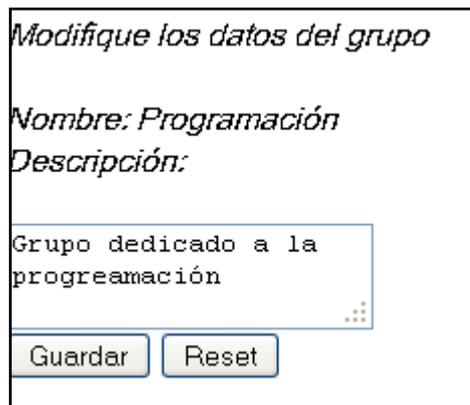
Descripción:

Crear grupo Reset

Menú secundario MSA03

Editar un grupo (VAU03)

Muestra un formulario con los datos de un grupo, junto con un botón que enviará la información al servidor para que remplace a la actual. Al igual que cuando se crea un grupo, el formulario irá precedido de la lista de errores que se produjeron al intentar modificar un grupo, entonces el formulario vendrá inicializado con los nuevos datos.



Modifique los datos del grupo

Nombre: Programación

Descripción:

Grupo dedicado a la programación

Guardar Reset

Menú secundario MSA03

Información de un grupo (VAU04)

Muestra toda la información asociada a un grupo. Su nombre, descripción y número de usuarios. Si el grupo es borrable (no tiene usuarios) se incluirá un botón para borrarlo. Delante de los datos de un grupo podrá ir un mensaje que confirma si un grupo se ha creado o se ha modificado correctamente.

Información del grupo

Nombre: Programación

Descripción:
Grupo dedicado a la programación

Número de usuarios: 30

[Listado de usuarios del grupo](#)

Menú secundario MSA03

Listado de los usuarios (VAU05)

De forma parecida al listado de los grupos, esta interfaz mostrará una tabla con los usuarios del sistema, ordenados por su login. Los usuarios podrán ser filtrados por su login, su estado, su tipo y su grupo. Para ello se incluirá un campo para introducir un filtro para el login y listas de selección con todos los grupos que contenga el sistema, así como las distintas posibilidades de los estados y tipos de usuarios. Al igual que con los grupos, se podrá seleccionar el índice del primer elemento y el tamaño de la lista.

Cada elemento de la tabla contendrá el login del usuario, su primer y segundo apellido, su nombre, su grupo, su estado y su tipo. Además de botones para dar de baja, alta y para mostrar toda la información de uno de ellos. Además se podrán marcar un subconjunto (casilla de selección) para realizar una operación sobre los elementos seleccionados, para estas operaciones se dispondrá de los botones necesarios.

Delante de la lista de elementos podrá situarse un mensaje que confirme o muestre los errores producidos al realizar una acción sobre los elementos mostrados. Cuando se realice una operación sobre la lista no se modificarán los parámetros de estas (índice inicial, tamaño de la lista).

Filtro login: Tipo **GESTOR** Estado **ACTIVO**

6 /1256 elementos

Tamaño lista 5 Actualizar < ->

Login	Nombre	Grupo	Tipo	Estado	
<input type="checkbox"/> xie8	Gil Perz, Ana	Estadística	GESTOR	ACTIPO	info baja
<input type="checkbox"/> e849	García Perez, Martín	Computación	GESTOR	ACTIPO	info baja
<input type="checkbox"/> u84f	Lopez Plo, María	Alumnos	GESTOR	ACTIPO	info baja
<input type="checkbox"/> ov77	Ovik, Peter	Graficos	GESTOR	INACTIVO	info borrar alta
<input type="checkbox"/> g8389	Gil Perez, Robeto	PorgAvanzada	GESTOR	ACTIPO	info baja

Menú secundario MSA03

Nuevo usuario (VAU06)

Muestra un formulario para introducir los datos de un nuevo usuario. Previo al formulario se puede incluir una lista de errores producidos al haber intentado crear un usuario, en este caso se rellenará el formulario con los datos introducidos previamente.

Introuzca los datos del nuevo usuario

Login:

Password:

Nombre:

Apellido 1:

Apellido 2:

Tipo: **GESTOR**

... mas formulario ...

Menú secundario MSA03

Editar los datos de un usuario (VAU07)

Muestra un formulario con los datos de un usuario. Al pulsar el botón correspondiente los datos serán guardados. De igual forma que al crear un usuario, si se producen errores se mostrarán en una lista previa al formulario.

Se incluirán botones para dar de alta y baja, así como para borrar al usuario.

Modifique los datos del usuario

Login: a893

Password:

Nombre:

Apellido 1:

Apellido 2:

Tipo: ▼

... mas formulario ...

Menú secundario MSA03

Información de un usuario (VAU08)

Muestra toda la información personal asociada a un usuario (nombre, apellidos, teléfonos, emails,...). Se incluirán botones para dar de alta y baja, así como para borrar al usuario. Delante de los datos se podrá incluir un mensaje que confirme la creación o modificación del usuario.

Modifique los datos del usuario

Login: a893

Nombre: Ana

Apellido 1: Gomez

Apellido 2: Gomez

Tipo: GESTOR

Estado: ACTIVO

... mas formulario ...

Menú secundario MSA03

B.8.3.3 ESTADÍSTICAS DEL USO DEL SISTEMA

Esta sección permitirá al administrador generar informes estadísticos del uso que se hace del sistema. Los informes generados contendrán información del número, media y porcentaje de grupos, usuarios,... que contiene el sistema. Cubrirá el requisito RF4 y el caso de uso CU07.

Mostrar un informe estadístico del sistema (VAE01)

Muestra un informe de las estadísticas del sistema que ha sido pedido por el administrador. Facilitará un enlace para generar un recurso en PDF con el contenido del informe que podrá ser guardado por el administrador donde desee.

Menú secundario	MSA04
-----------------	-------

Nuevo informe de las estadísticas del sistema (VAE02)

Esta GUI muestra un formulario mediante el cual el administrador puede configurar un informe estadístico del uso del sistema. Se dispondrá de dos enlaces, uno permitirá ir a la sección que muestra el informe generado en base a la información introducida en el formulario y otro que generará un recurso con el informe en formato PDF.

El formulario mostrará entradas de texto que permitirán introducir el título y una descripción para el informe. Además se incluirán casillas de verificación para indicar que se añadan las estadísticas descritas en el documento de diseño de la capa de negocio. Cada una de las opciones contendrá varios datos según las combinaciones de parámetros indicadas (ver componente “Generación de Informes de la Dedicación”, apartado B.6.1.5).

Menú secundario	MSA04
-----------------	-------

B.8.4 INTERFACES PARA LOS GESTORES

Los usuarios de tipo gestor necesitarán acceder a un conjunto de interfaces que les permitan gestionar la información referente a los proyectos de los que son responsables. Las siguientes interfaces facilitarán la labor de crear, modificar y borrar proyectos, así como seleccionar aquellos usuarios que participen en ellos Además de otras funcionalidades como alarmas e informes de la dedicación.

B.8.4.1 GESTIÓN DE LOS PROYECTOS

Dado el caso de uso CU01 y los requisitos RF6, RF7 y RF9 un gestor podrá gestionar sus proyectos y las tareas y participantes asociados en cada uno de ellos.

Listado de los proyectos gestionados (VGP01)

Muestra una tabla con los proyectos que gestiona el usuario. Al igual que con las tablas descritas anteriormente, solo se mostrará un subconjunto de la lista completa de proyectos gestionados. Se dispondrá de campos para indicar el índice del primer elemento de la lista que mostrará la tabla, el tamaño de la tabla y un filtro por el estado de los proyectos. Se facilitarán botones para ir actualizando los elementos de la tabla.

Cada elemento de la tabla mostrará el código del proyecto, su nombre, su estado y botones para abrir, cerrar o borrar el proyecto (según sea su estado), así como un enlace para mostrar la información del proyecto. Previamente a la tabla se podrá mostrar el resultado de una operación realizada sobre un elemento.

Estado	Todos		
6	/1256 elementos		
Tamaño lista	5	Actualizar < - >	
Código	Nombre	Estado	
1000	Aplicación01	ABIERTO	Cerrar
1003	Videojuego	ABIERTO	Cerrar
2007	Robots	ABIERTO	Cerrar
3009	Inteligencia artivicia	CERRADO	Abrir Borrar
5489	Zx80	ABIERTO	Cerrar

Menú secundario Ninguno

Nuevo proyecto (VGP02)

Muestra un formulario que permite introducir los datos para crear un nuevo proyecto. Delante del formulario se podrá incluir una lista con los errores que se produjeron al intentar crear un proyecto.

Introduzca los datos del nuevo proyecto

Código:

Nombre:

... mas formulario ...

Menú secundario Ninguno

Editar proyecto (VGP03)

Muestra la información de un proyecto en un formulario que permite modificarla. Si se han cometido errores al intentar modificar un proyecto se mostrará una lista delante del formulario.

Modifique los datos del proyecto

Código: 1003

Nombre:

... mas formulario ...

Menú secundario MSG01

Información de un proyecto gestionado (VGP04)

Muestra la información de un proyecto gestionado por el usuario junto con una tabla con la lista de sus tareas. Dispondrá de botones para abrir, cerrar, borrar y editar el proyecto (según su estado).

La tabla de tareas del proyecto incluirá el nombre y la descripción de la tarea, así como el estado y tipo de alarma, las horas totales dedicadas a la misma, límite de la dedicación, número de dedicaciones, alarmas asociadas, alarmas no atendidas. Se añadirán botones para modificarlas, borrarlas y para cambiar su posición. Si la alarma ha producido una alerta que no ha sido atendida se marcará en rojo.

Código: 157
 Nombre: IA
 Estado ABIERTO
 ... mas datos ...

Lista de tareas

Diseño	
Diseño de la solución del problema.	
Tipo alarma	GENERAL
Estado	ACTIVADA
Dedicación	1587 h.
Límite dedicación	1400 h.
Núm. dediciones	149
Alarmas (T/P)	20/80
Alertas (T/P)	1/5

Implementación	
Escribir el código de la solución del problema.	
Tipo alarma	GENERAL
Estado	ALERTA
Dedicación	158 h.
Límite dedicación	130 h.
Núm. dediciones	149
Alarmas (T/P)	20/80
Alertas (T/P)	1/5

Pruebas

Menú secundario MSG01

Nuevas tareas (VGP05)

Presenta un formulario que permite añadir nuevas tareas al proyecto. El formulario estará compuesto por bloques iguales, cada bloque permitirá crear una nueva tarea. Habrá un botón que permitirá añadir nuevos bloques. En el caso de que se hubiera intentado crear nuevas tareas y se hubiera producido algún error, los primeros bloques irían rellenos con los datos de las tareas que los produjeron y cada uno precedidos de una descripción del error. Delante del formulario se mostrará una tabla con los nombres y las descripciones de cada tarea que ya existen en el proyecto.

Proyecto: 157 IA

Tarea	
Diseño	Diseño de la solución
Implementación	Escribir el código de la solución del problema
Pruebas	Comprobación de la solución

Introduzca las nuevas tareas

Tarea 1:

Nombre

Descripción

Tipo Alarma

Activar

Límite horas.

Tarea 2:

Nombre

Descripción

Tipo Alarma

Activar

Menú secundario MSG01

Información de una tarea (VGP06)

Muestra la información de una tarea, permitiendo modificar su descripción y el estado y tipo de la alarma asociada.

Proyecto: 157 - IA

Tarea: Diseño

Descripción:

Diseño del problema

Tipo

Estado:

Menú secundario MSG01

B.8.4.2 GESTIÓN DE LOS PARTICIPANTES

Este conjunto de interfaces permitirá a los usuarios gestionar la lista de usuarios que participarán en cada uno de sus proyectos. Cubrirá los requisitos RF8 y RF14.

Listado de los participantes (VGPa01)

Muestra una tabla con los participantes de un proyecto. Como en otras secciones, se mostrará un subconjunto de todos los participantes del proyecto y se podrá actualizar la tabla con los botones correspondientes.

Cada elemento de la tabla mostrará el login, apellidos, nombre y el estado del participante en el sistema y en el proyecto, así como botones para darlo de alta y baja o borrarlo (según su estado), Además dispondrá de un cuadro de selección para poder realizar estas operaciones sobre un sublistado de participantes, mediante los botones que se dispondrán a tal efecto. Desde cada entrada de la tabla se podrá ir a la sección que muestra los datos del participante. Delante de la lista podrá añadirse un mensaje con el resultado de la última operación realizada sobre los participantes.

2187 - Interfaces animadas

Lista de participantes

Estado

8 / 23 elementos

Tamaño lista

	Login	Nombre	Estado proyecto	Estado usuario	
<input type="checkbox"/>	14456	García Pau, Luisa	ACTIVO	ACTIVO	Info <input type="button" value="Dar baja"/>
<input type="checkbox"/>	346	Sanz, Pablo	ACTIVO	ACTIVO	Info <input type="button" value="Dar baja"/>
<input type="checkbox"/>	3851	Gracia Plo, María	INACTIVO	ACTIVO	Info <input type="button" value="Dar alta"/> <input type="button" value="Borrar"/>
-	a3489	Pool, Charles	INACTIVO	INACTIVO	Info <input type="button" value="Borrar"/>
<input type="checkbox"/>	c999	Tristan, Pablo	ACTIVO	ACTIVO	Info <input type="button" value="Dar baja"/>

Menú secundario MSG01

Nuevos participantes (VGPa02)

Muestra una tabla con los usuarios pertenecientes al mismo grupo que el proyecto y que no participen en él cuyo estado en el sistema es ACTIVO. Se podrá seleccionar un conjunto de ellos y, mediante el botón correspondiente, se les podrá añadir al proyecto.

Igual que en otras tablas, se podrá indicar el índice del primer elemento de la tabla y su tamaño. Y actualizar la lista con los botones correspondientes.

2187 - Interfaces animadas

Seleccione los usuarios y pulse añadir.
Seleccione si desea añadir más usuarios.

1 /23 elementos

Tamaño lista 5

	Login	Nombre
<input type="checkbox"/>	14456	García Pau, Luisa
<input type="checkbox"/>	346	Sanz, Pablo
<input type="checkbox"/>	3851	Gracia Plo, María
<input type="checkbox"/>	a3489	Pool, Charles
<input type="checkbox"/>	c999	Tristan, Pablo

Añadir más participantes

Menú secundario MSG01

Información de un participante (VGPa03)

Muestra los datos de un participante en un proyecto. Permitirá darlo de alta y baja, así como borrarlo (según el estado del participante). Mostrará una tabla con la dedicación agregada de cada tarea del proyecto y la dedicación agregada de todas las tareas.

Proyecto: 157 - IA
Login: f7758
Nombre: Pérez Pérez, Mario
... mas datos ...

[Dar baja en el proyecto](#)

Dedicación	
Diseño	57,0 horas
Pruebas	43,5 horas
Integración	25,5 horas

Alertas no atendidas					
Tarea	Fecha	Dedicación	Límite	Estado	
Resolución problemas	17 Sep 2012	97,0	80,0	ATENDIDA GESTOR	Borrar
	23 Sep 2012				
Integración	23 Ago 2012	23,0	20,0	ALERTA	Atender alerta Borrar
	30 Ago 2012				

Menú secundario MSG01

B.8.4.3 GESTIÓN DE LAS ALARMAS DE UN PROYECTO

Las siguientes GUIs permiten mostrar y gestionar las alarmas asociadas a un proyecto. Cubrirá los requisitos RF15 y RF16, y el caso de uso CU02.

Listado de las alarmas de tipo participante (VGA01)

Muestra dos tablas con las alarmas de tipo participante de un mes. La primera contendrá las alarmas que tengan un periodo semanal y la segunda que tengan un periodo mensual. Cada alarma mostrará el código del proyecto, el login del participante, el nombre de la tarea, selectores con los posibles estados y tipos, una entrada para el valor del límite de horas y la fecha de la última modificación. Además se dispondrá un botón para guardar los cambios y otro para borrar la alarma. El código del proyecto y el login del usuario dispondrán de enlaces para ir a la sección de información del proyecto y del participante.

También se dispondrá de botones para cambiar al mes siguiente y al anterior, así como campos para seleccionar un mes y un año concreto. Además

se dispondrán de selectores que permitan filtrar las alarmas por su estado y tipo. También se añadirán selectores para poder elegir un proyecto y/o participante. Previamente a la tabla se podrá añadir un texto que muestre el resultado de las acciones realizadas.

Lista de alarmas de tipo participatne:

Tipo: Todos Estado: Todos Proyecto: Todos Participante: Todos

<- Septiembre 2012 Actualizar ->

Semanales:

Proyecto	Participatne	Tarea	Semana	Tipo	Estado	Dedicación	Límite	Actualizada			
<input type="checkbox"/>	a775	g859aj	Diseño	27 Sep 2012	GENERAL	ALERTA	27,5 h	20,0	h 30 Sep 2012	Guardar	Borrar
<input type="checkbox"/>	g834	eidk	Implementación	27 Sep 2012	GESTOR	DESACTIVADA	1,5 h	20,0	h 30 Ago 2012	Guardar	Borrar
<input type="checkbox"/>	a348	p458	Revisión	14 Sep 2012	GESTOR	ATENDIDA	13,5 h	10,0	h 28 Sep 2012	Guardar	Borrar

Borrar selección

Mensuales:

Proyecto	Participatne	Tarea	Tipo	Estado	Dedicación	Límite	Actualizada			
<input type="checkbox"/>	a775	g859aj	Diseño	GENERAL	ALERTA	27,5 h	20,0	h 04 Oct 2012	Guardar	Borrar
<input type="checkbox"/>	g834	eidk	Implementación	GESTOR	DESACTIVADA	1,5 h	20,0	h 30 Ago 2012	Guardar	Borrar
<input type="checkbox"/>	a348	p458	Revisión	GESTOR	ATENDIDA	13,5 h	10,0	h 28 Sep 2012	Guardar	Borrar

Menú secundario MSG02

Listado de las alertas de tipo participante (VGA02)

Muestra dos tablas con las alarmas de tipo participante que hayan generado una alerta recientemente (un periodo de la fecha de la última modificación).

La primera tabla mostrará las alarmas semanales y la segunda las alarmas mensuales. Cada alarma mostrará el código del proyecto, login del usuario (ambos con enlaces a la sección que contiene su información), nombre de la tarea, tipo de alarma, estado de la alarma (solo dos posibilidades “ALERTA” y “ATENDIDA PARTICIPANTE”), dedicación, límite de la dedicación, fecha de actualización. Dispondrá de botones para darla por atendida y para borrarla.

Alertas recientes

Semanales:

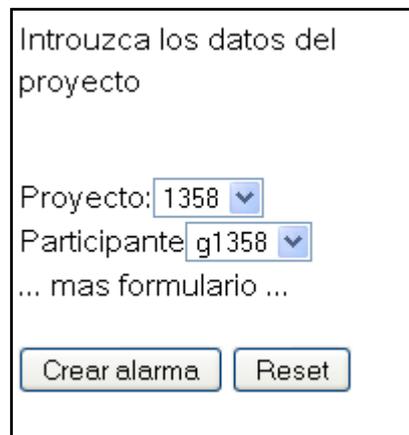
Proyecto	Participatne	Tarea	Semana	Tipo	Estado	Dedicación	Límite	Actualizada		
a775	g859aj	Diseño	27 Sep 2012	GENEARL	ATENDIDAD PART.	27,5 h	20,0 h	30 Sep 2012	Atender	Borrar
g834	eidk	Implementación	27 Sep 2012	GESTOR	ALERTA	21,5 h	20,0 h	30 Ago 2012	Atender	Borrar
a348	p458	Revisión	14 Sep 2012	GESTOR	ALERTA	15,0 h	10,0 h	28 Sep 2012	Atender	Borrar

Mensuales:

Proyecto	Participatne	Tarea	Tipo	Estado	Dedicación	Límite	Actualizada		
a775	g859aj	Diseño	GENERAL	ALERTA	27,5 h	20,0 h	04 Oct 2012	Atender	Borrar
g834	eidk	Implementación	GENERAL	ALERTA	1,5 h	20,0 h	30 Ago 2012	Atender	Borrar
a348	p458	Revisión	GESTOR	ALERTA	13,5 h	10,0 h	28 Sep 2012	Atender	Borrar

Nueva alarma de tipo participante (VGA03)

Muestra un formulario que permite añadir una alarma de tipo participante. Si al crear una alarma se produjeran errores, se mostrarían delante del formulario y éste sería inicializado con los datos introducidos



The screenshot shows a web form titled "Introuzca los datos del proyecto". It contains two dropdown menus: "Proyecto:" with the value "1358" and "Participante" with the value "g1358". Below the dropdowns is the text "... mas formulario ...". At the bottom of the form are two buttons: "Crear alarma" and "Reset".

Listado de las alarmas de tipo tarea (VGA04)

Muestra dos tablas con las alarmas de tipo tarea de un mes. La primera contendrá las alarmas que tengan un periodo semanal y la segunda que tengan un periodo mensual.

Cada alarma mostrará el código del proyecto, el nombre de la tarea, selectores con los posibles estados y tipos, una entrada para el valor del límite de horas y la fecha de la última modificación. Además se dispondrá un botón para guardar los cambios y otro para borrar la alarma. El código del proyecto dispondrá de un enlace que llevará a la sección de información del proyecto.

También se dispondrá de botones para cambiar al mes siguiente y al anterior, así como campos para seleccionar un mes y un año concretos. También se dispondrán de selectores que permitan filtrar las alarmas por su estado y tipo, así como para poder elegir un proyecto.

Lista de alarmas de tipo participatpne:

Tipo: Todos Estado: Todos Proyecto: Todos

< Septiembre 2012 Actualizar >

Semanales:

Proyecto	Tarea	Semana	Tipo	Estado	Dedicación	Límite	Actualizada		
g859a	Diseño	27 Sep 2012	GENERAL	ALERTA	27,5 h	20,0	h 30 Sep 2012	Guardar	Borrar
eidk	Implementación	27 Sep 2012	GESTOR	DESACTIVADA	1,5 h	20,0	h 30 Ago 2012	Guardar	Borrar
p458	Revisión	14 Sep 2012	GESTOR	ATENDIDA	13,5 h	10,0	h 28 Sep 2012	Guardar	Borrar

Borrar selección

Mensuales:

Proyecto	Tarea	Tipo	Estado	Dedicación	Límite	Actualizada		
a775	Diseño	GENERAL	ALERTA	27,5 h	20,0	h 04 Oct 2012	Guardar	Borrar
eidk	Implementación	GESTOR	DESACTIVADA	1,5 h	20,0	h 30 Ago 2012	Guardar	Borrar
p458	Revisión	GESTOR	ATENDIDA	13,5 h	10,0	h 28 Sep 2012	Guardar	Borrar

Borrar selección

Menú secundario MSG03

Listado de las alertas de tipo tarea (VGA05)

Muestra dos tablas con las alarmas de tipo tarea que hayan generado una alerta recientemente (un periodo de la fecha de la última modificación). La primera tabla mostrará las alarmas semanales y la segunda las alarmas mensuales. Cada alarma mostrará el código del proyecto (con un enlace a la sección que contiene su información), nombre de la tarea, tipo de alarma, estado de la alarma (solo hay un valor posible "ALERTA"), dedicación, límite de la dedicación, fecha de actualización. Facilitará botones para que el gestor la dé por atendida o la pueda borrar.

Alertas recientes

Semanales:

Proyecto	Tarea	Semana	Tipo	Estado	Dedicación	Límite	Actualizada		
a775	Diseño	27 Sep 2012	GENEARL	ATENDIDAD PART.	27,5 h	20,0 h	30 Sep 2012	Atender	Borrar
q834	Implementación	27 Sep 2012	GESTOR	ALERTA	21,5 h	20,0 h	30 Ago 2012	Atender	Borrar
a348	Revisión	14 Sep 2012	GESTOR	ALERTA	15,0 h	10,0 h	28 Sep 2012	Atender	Borrar

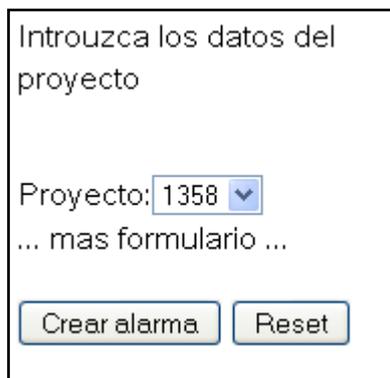
Mensuales:

Proyecto	Tarea	Tipo	Estado	Dedicación	Límite	Actualizada		
a775	Diseño	GENERAL	ALERTA	27,5 h	20,0 h	04 Oct 2012	Atender	Borrar
q834	Implementación	GENERAL	ALERTA	1,5 h	20,0 h	30 Ago 2012	Atender	Borrar
a348	Revisión	GESTOR	ALERTA	13,5 h	10,0 h	28 Sep 2012	Atender	Borrar

Menú secundario MSG03

Nueva alarma de tipo tarea (VGA06)

Muestra un formulario que permite añadir una nueva alarma de tipo tarea al proyecto. Si se hubiera producido un error al crear una alarma, se mostraría la lista con la descripción de los errores y el formulario se iniciaría con los datos introducidos.



Introduzca los datos del proyecto

Proyecto: 1358 ▼

... mas formulario ...

Crear alarma Reset

Menú secundario MSG03

B.8.4.4 DEDICACIÓN AGREGADA

Las siguientes interfaces permiten calcular la dedicación agregada realizada dentro de los proyectos. Cubriendo los requisitos RF10, RF11, RF12 y RF13 y el caso de uso CU03. Permitirá calcular la dedicación agregada que realizan los participantes en los proyectos, Además facilitará la creación de un documento PDF con el contenido de un informe que el usuario podrá guardar donde desee. Para ver la estructura de un informe, consultar el apartado B.6.1.6.

Dedicación agregada de un proyecto (VGD01)

Muestra un informe con la dedicación agregada de un proyecto gestionado por el usuario. La información dependerá de los parámetros pasados en la petición. El informe podrá contener varios apartados, con distinta información

Menú secundario MSG04

Nuevo informe de la dedicación (VGD02)

Esta sección mostrará un formulario que permitirá introducir los parámetros de un informe de la dedicación y poder crearlo posteriormente. Facilitará dos tipos de informes, según lo descrito en el apartado B.6.1.6, uno

sobre la dedicación agregada de un proyecto y otro de la dedicación de varios participantes.

Menú secundario MSG04

B.8.5 INTERFACES COMUNES

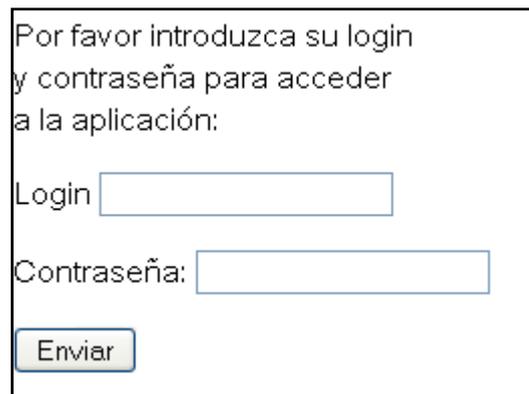
Los siguientes apartados describen el resto de las interfaces de la aplicación Web, las cuales podrán ser utilizadas por todos sus usuarios.

B.8.5.1 ENTRADA A LA APLICACIÓN WEB

Ofrecerá un punto de acceso común a todos los usuarios de la aplicación Web.

Acceso a la aplicación (VCA01)

Mostrará un formulario con dos entradas de texto correspondientes al login y al password, junto con un botón cuya acción mandará los datos al servidor, permitiendo a los usuarios acceder a sus contenidos. Si los datos de identificación suministrados por el cliente Web fueran incorrectos mostrará un mensaje de error y el login se iniciará con el valor introducido. Algo parecido pasará cuando caduque la sesión de un usuario, entonces se mostrará un mensaje advirtiendo el suceso



Por favor introduzca su login y contraseña para acceder a la aplicación:

Login

Contraseña:

Menú secundario Ninguno

B.8.5.2 INFORMACIÓN

Contiene una GUI que permite ver la información de contacto con el administrador.

Información de contacto (VCI01)

Muestra cierta información del administrador que puede servir a los demás usuarios para contactar con él en caso de necesidad. La información mostrada será extraída del perfil del administrador, e incluirá los teléfonos y direcciones de correo electrónico que haya incluido en su perfil.

Control de la dedicación v1.0

Administrador: Pedro Álvarez

Téfonos:

555 25 566 88
Teléfono despacho.

555 33 556 88
Laboratorios.

Emails:

abc@eid.es
Contacto directo

pa@correo.es
Solo emergencias.

Menú secundario Ninguno

B.8.5.3 GESTIÓN DEL PERFIL DEL USUARIO

Permiten a los usuarios gestionar sus perfiles en el sistema, como pueden ser su dirección, teléfonos y direcciones de correo electrónico. Cubrirá el caso de uso CU04 y el requisito RF5.

Datos personales (VCP01)

Muestra un formulario con el perfil del usuario, permitiendo cambiar algunos de sus datos. Delante del formulario podrá aparecer una lista con los errores que se produjeron al intentar guardar el perfil, en este caso el formulario será rellenado con los datos introducidos y no con los que almacena el sistema. También contendrá otro formulario que permitirá cambiar la contraseña.

Login: h48569

Modificar el password:

Actual

Nuevo

Repitalo

Datos personales:

Nombre

Apellidos

Tipo usuario: NORMAL

Organización

...

Menú secundario MSAux01

Listado de teléfonos (VCP02)

Muestra la lista de los teléfonos asociados con el usuario ordenados por el campo "orden". Para cada teléfono se incluirán botones para cambiarlo de posición. También se dispondrá de botones para borrar los teléfonos. Además habrá un formulario que permitirá añadir un nuevo teléfono al final de la lista. El número y la descripción del teléfono podrán ser modificados escribiendo en los campos correspondientes. Un botón permitirá guardar los cambios.

Teléfonos

555 38 40 333	bajar>
Teléfono de contacto oficina	
555 38 40 333	subir
	bajar
Teléfono de contacto móvil	
555 38 40 333	subir
Teléfono de contacto personal	

Guardar Reset Nuevo

Menú secundario MSAux01

Listado de emails (VCP03)

Muestra la lista de direcciones de correo electrónico asociados con el usuario ordenados por el campo "orden". Para cada dirección se incluirán botones para cambiarlo de posición. También se dispondrá de botones para borrar las direcciones. Además habrá un formulario que permitirá añadir una nueva dirección al final de la lista. El email y la descripción de la dirección de uno de los elementos podrán ser modificados escribiendo en los campos correspondientes. Un botón permitirá guardarlos.

Emails

586	@ iie.edi	bajar>
Email de contacto oficina		
586	@ iie.edi	subir
Email de contacto móvil		
586	@ iie.edi	subir
Email de contacto personal		

Guardar Reset Nuevo

Menú secundario MSAux01

B.8.5.4 PANTALLAZO DE ERROR

La mayoría de los errores se gestionarán dentro de las pantallas generadas por las GUIs descritas en estos apartados. Para aquellos errores que no estén previstos en un uso normal de la aplicación se generará una pantalla mostrando una descripción del error que se ha producido.

Pantallazo de error (VCError)

La pantalla mostrará un mensaje que describa de forma breve que ha sucedido y que acción ha generado el error.

B.8.6 AYUDA INTEGRADA

La aplicación Web dispondrá de un sistema de ayuda integrada dentro del sistema. Para ver su contenido se deberá presionar el botón  y con el botón  se podrá volver a la sección que se estaba visualizando. La

aplicación Web mantendrá el contenido introducido en la sección actual durante el tiempo que se consulte la ayuda.

Se podrá navegar dentro de las distintas secciones de la ayuda utilizando una lista de secciones. Habrá una página de ayuda por cada sección a la que pueda acceder el usuario (según su tipo). Las páginas públicas no dispondrán de este servicio.

B.9 CAPA DE PRESENTACIÓN DE LA APLICACIÓN PORTLET

En este capítulo se describe la capa de presentación de la aplicación Portlet. Un Portlet es una pequeña aplicación que forma parte de un conjunto mayor de aplicaciones afines que se muestran dentro de un mismo contenedor (normalmente una página Web).

Un Portlet suele disponer de menos espacio que una aplicación Web de otro tipo, haciendo que sea necesario reducir el número de componentes mostrados. La GUI estará compuesta por una pequeña cabecera, el contenido principal y un pie. Cada GUI será nombrado por una clave (un acrónimo) que la identificará unívocamente. Esta clave cumplirá las siguientes reglas:

- Primera letra 'V'
- Segunda letra 'P'
- Una letra que designará la sección
- 2 números comenzando en 01

B.9.1 CABECERA

Cada sección dispondrá de una cabecera que contendrá botones para salir de la aplicación y para mostrar la ayuda. También mostrará una línea con la ruta que lleva a la sección en la que se encuentra el usuario, cada sección padre dispondrá de un enlace a su página (P.e. "Mis Proyectos/Info Proyecto").

B.9.2 PIE DE PÁGINA

Las GUI dispondrán de un pie de página que podrá contener el nombre y la dirección de correo electrónico del administrador del sistema (será configurable desde la aplicación Web).

B.9.3 INTERFACES PARA LOS USUARIOS

Los participantes en los proyectos podrán hacer uso de estas interfaces para introducir su dedicación en los proyectos en los que participan y poder generar informes sobre su dedicación.

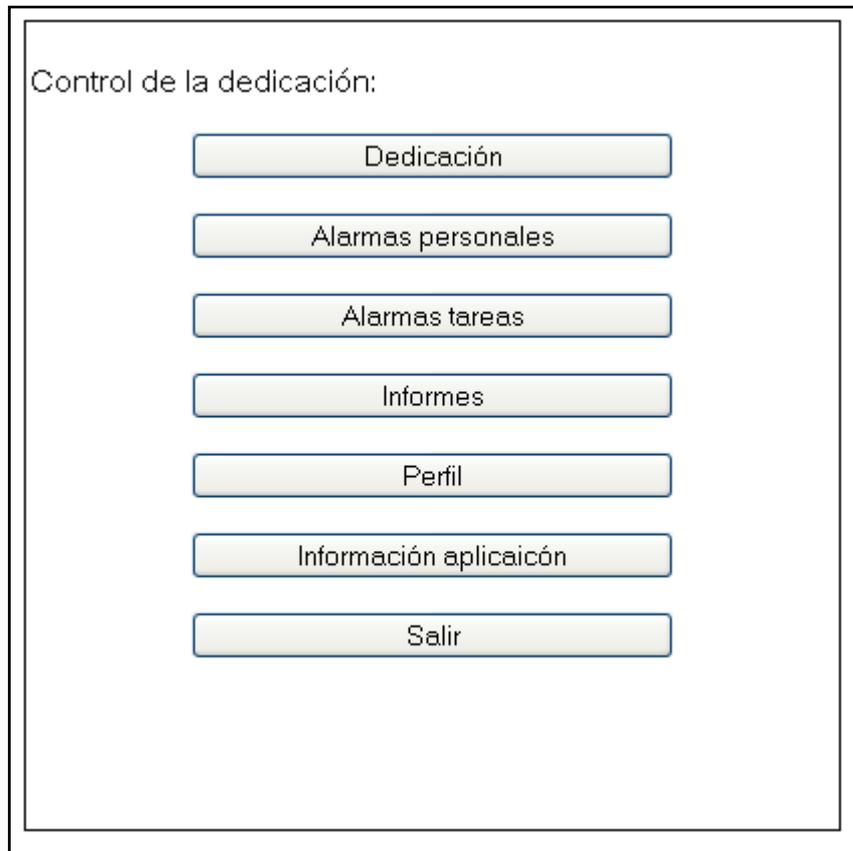
B.9.3.1 MENÚ PRINCIPAL

Cuando un usuario acceda a la aplicación verá una pantalla con varios botones que le permitirán acceder a las distintas secciones.

Menú de inicio (VPM01)

El menú de inicio se mostrará cuando el usuario se haya identificado correctamente en el sistema. Mostrará varios botones que le llevarán a las distintas secciones.

Estas secciones incluirán la gestión diaria de la dedicación, la gestión de las alarmas de tipo participante, la consulta de las alarmas de tipo tarea, la generación de informes de la dedicación, la modificación del perfil del usuario, ver la información de contacto con el administrador y la opción de salir de la aplicación.



B.9.3.2 PARTICIPACIONES EN LOS PROYECTOS

Este conjunto de interfaces mostrará los listados de los proyectos en los que participa el usuario y permitirá mostrar la información de una participación concreta. Cubrirá el caso de uso CU08 y los requisitos RF18 y RF19.

Listado de las participaciones del usuario (VPP01)

Muestra una tabla con las participaciones del usuario en los proyectos ordenados por su código. Habrá un enlace que permitirá cambiar entre la lista de proyectos activos e inactivos. La tabla mostrará un subconjunto de la lista total y el usuario podrá utilizar los botones para ir actualizando los datos mostrados.

Cada entrada de la tabla contendrá el código del proyecto que enlazará con la sección que muestra la información de la participación. También mostrará el nombre y el acrónimo. Se podrá seleccionar un proyecto concreto y con los botones correspondientes se podrá ir a la secciones de alarmas del participante, dedicación y generar informes del proyecto.



[Volver Salir](#)

/Menú/Proyectos

Proyectos:

Activos [Inactivos](#)

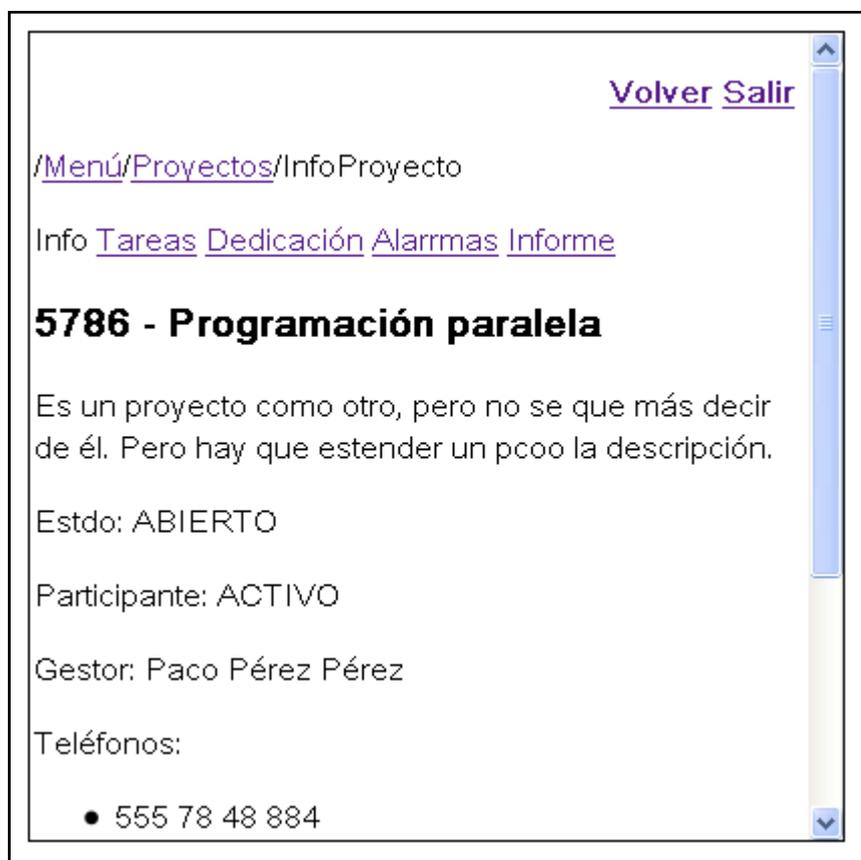
<- 7 de 15 ->

[Dedicación](#) [Alarmas](#) [Informe](#)

Código	Nombre	Acrónimo
1358	Interfaces visuales	A738
1500	Rogótica	A713
2578	Programción Web	3589
3589	Programcación Portlet 1.0 b58	
652685	Alamcenamiento masico.	c8549
789'	Transformación imágenes 78rod	

Información de una participación (VPP02)

Muestra la información de una participación del usuario en un proyecto. Contendrá los datos del proyecto, los del gestor y los del participante. Dispondrá de enlaces que llevarán a las secciones que permiten ver las tareas del proyecto, introducir la dedicación, ver las alarmas y generar informes del mismo.



Listado de las tareas de un proyecto (VPP03)

Muestra un listado con las tareas de un proyecto, en el que participa el usuario, ordenadas tal como impuso el gestor del proyecto. Para cada tarea se mostrará su nombre y descripción, además de la dedicación que el usuario ha realizado en ella, la dedicación de todos los participantes del proyecto, así como el número de alertas y alarmas asociadas con el participante en la tarea del proyecto.

La interfaz contará con enlaces para ver la información del proyecto, introducir la dedicación, ver las alarmas del participante en el proyecto y generar informes de la dedicación en el proyecto.

[Volver](#) [Salir](#)

[/Menú/Proyectos/Tareas](#)

[Info Tareas](#) [Dedicación](#) [Alarrmas](#) [Informe](#)

5786 - Programación paralela

Análisis			
Consta de varias partes...			
Dedicación	Total	Alertas	Alarmas
55,0 h.	2057,5 h 2		25

Diseño			
Consta de varias partes...			
Dedicación	Total	Alertas	Alarmas
55,0 h.	2057,5 h 2		25

Gestión de la dedicación diaria (VPP04)

Mostrará una tabla con todas las tareas y su dedicación en el día correspondiente. Dispondrá de botones para cambiar al día siguiente y al anterior, además de campos para introducir una fecha exacta.

La tabla mostrará el nombre de cada tarea y su dedicación realizada en el día. Si el estado del proyecto y del participante, así como el día, lo permiten, cada valor de la dedicación irá en una entrada de texto que permitirá modificar su valor. Se añadirá una entrada en la tabla que muestre la dedicación total realizada en el día por el participante. Existirá un botón que permitirá guardar la dedicación introducida. Cada entrada de la tabla tendrá, además de la dedicación diaria, la dedicación total del participante y del proyecto. La última fila mostrará el total de las participaciones en el día.

Habrán disponibles enlaces para ver la información del proyecto, la lista de las tareas, las alarmas del participante en el proyecto y para generar informes de la dedicación en el proyecto.

[Volver](#) [Salir](#)

[/Menú/Proyectos/Dedicaciónh](#)

[Info](#) [Tareas](#) [Dedicación](#) [Alarmas](#) [Informe](#)

5786 - Programación paralela

Tarea	Hoy	Participante	Proyecto
Análisis	<input type="text" value="3.0"/> h.	15.5 h.	1578.0 h.
Diseño	<input type="text" value="2.0"/> h.	15.5 h.	1578.0 h.
Implementación	<input type="text" value="0.0"/> h.	15.5 h.	1578.0 h.
Pruebas	<input type="text" value="0.0"/> h.	15.5 h.	1578.0 h.
Total	5.0 h.	62.0 h.	6312.0 h.

B.9.3.3 GESTIÓN DE LAS ALARMAS

Las GUIs descritas a continuación permitirán ver las alarmas asociadas con los participantes. Un usuario concreto podrá ver todas sus alarmas visibles asociadas con él y con las tareas de los proyectos en los que participa. En ambos casos cada uno de los tres estados hace referencia a un conjunto de ellas, siendo por semanas, meses o solo las alarmas que hayan producido una alerta.

Listado de las alarmas personales semanales (VPA01)

Muestra una tabla con todas las alarmas de duración semanal que tiene asociado un usuario en todos sus proyectos. Las alarmas se mostrarán por meses y se dispondrá de botones para cambiar de mes y entradas de texto para indicar un mes y un año concreto. También se dispondrá de un campo para seleccionar un proyecto concreto con el que filtrar las alarmas.

Cada entrada de la tabla contendrá el código del proyecto, el nombre de la tarea, la fecha de inicio del intervalo, la dedicación realizada por el participante en la tarea dentro del periodo, el límite esperado por la alarma, su estado y, en caso necesario un botón para atenderla.

Cada entrada dispondrá de un enlace para ver la información del proyecto al que pertenece. También se dispondrá de enlaces para ver las alarmas mensuales y las alertas (alarmas recientes que deben atenderse).

[/Menú/Alarmas personales](#)
[Alertas](#) [Semanales](#) [Mensuales](#)
 Proyecto
 Estado
 Julio

Proyecto	Periodo	Horas	Estado
1258 Diseño	16/07/2012	15/10 h.	Alerta <input type="button" value="OK"/>
1258 Análisis	16/07/2012	15/10 h.	Atendida
1258 Pruebas	16/07/2012	7/10 h.	Activada

Listado de las alarmas personales mensuales (VPA02)

Es igual que el listado de las alarmas semanales, pero con un periodo mensual.

[/Menú/Alarmas personales](#)
[Alertas](#) [Semanales](#) [Mensuales](#)
 Proyecto
 Estado
 Julio

Proyecto	Periodo	Horas	Estado
1258 Diseño	07/2012	15/10 h.	Alerta <input type="button" value="OK"/>
1258 Análisis	07/2012	15/10 h.	Atendida
1258 Pruebas	07/2012	7/10 h.	Activada

Listado de las alertas personales (VPA03)

Es parecido al listado de las alarmas semanales y mensuales, solo que muestra todas las alarmas que necesitan atención por parte del usuario dentro de un periodo cercano de tiempo.



Proyecto	Periodo	Horas	Estado
1258 Diseño	01/08/2012 31/08/2012	15/10 h.	Alerta <input type="button" value="OK"/>
1258 Análisis	16/07/2012 22/07/2012	15/10 h.	Alerta <input type="button" value="OK"/>
1258 Pruebas	01/07/2012 31/07/2012	7/10 h.	Alerta <input type="button" value="OK"/>
1258 Diseño	16/07/2012 22/07/2012	15/10 h.	Alerta <input type="button" value="OK"/>
1258 Diseño	16/07/2012 22/07/2012	15/10 h.	Alerta <input type="button" value="OK"/>

Listado de las alarmas de las tareas semanales (VPA04)

Muestra una tabla con todas las alarmas de duración semanal que tienen las tareas de los proyectos del participante. Las alarmas se mostrarán por meses y se dispondrá de botones para cambiar de mes y entradas de texto para indicar un mes y un año concreto. También se dispondrá de un campo para seleccionar un proyecto concreto con el que filtrar las alarmas.

Cada entrada de la tabla contendrá el código del proyecto, el nombre de la tarea, la fecha de inicio del intervalo, la dedicación realizada por el participante en la tarea dentro del periodo, el límite esperado por la alarma, su estado.

Cada entrada dispondrá de un enlace para ver la información del proyecto al que pertenece. También se dispondrá de enlaces para ver las alarmas mensuales y las alertas (alarmas recientes).

[Volver Salir](#)

/Menú/Alarmas tareas

[Alertas](#) [Semanales](#) [Mensuales](#)

Proyecto

Estado

<- Julio 2012 Ir ->

Proyecto	Periodo	Horas	Estado
1258 Diseño	16/07/2012	15/10 h.	Alerta
1258 Análisis	16/07/2012	15/10 h.	Atendida
1258 Pruebas	16/07/2012	7/10 h.	Activada

Listado de las alarmas de las tareas semanales (VPA05)

Igual que las alarmas mensuales, pero con un periodo semanal.

[Volver Salir](#)

/Menú/Alarmas tareas

[Alertas](#) [Semanales](#) [Mensuales](#)

Proyecto

Estado

<- Julio 2012 Ir ->

Proyecto	Periodo	Horas	Estado
1258 Diseño	07/2012	15/10 h.	Alerta
1258 Análisis	07/2012	15/10 h.	Atendida
1258 Pruebas	07/2012	7/10 h.	Activada

Listado de las alertas de las tareas (VPA06)

De forma similar a las alarmas de las tareas, muestra una lista con las alarmas en estado de alerta de tipo tarea más recientes. A diferencia que las alarmas de tipo participante, el usuario no podrá realizar ninguna acción sobre ninguna alarma.



Proyecto	Periodo	Horas	Estado
1258 Diseño	01/08/2012 31/08/2012	15/10 h.	Alerta
1258 Análisis	16/07/2012 22/07/2012	15/10 h.	Alerta
1258 Pruebas	13/08/2012 19/08/2012	7/10 h.	Alerta
1258 Diseño	16/07/2012 22/07/2012	15/10 h.	Alerta
1258 Diseño	16/07/2012 22/07/2012	15/10 h.	Alerta

B.9.3.4 GENERAR INFORMES DE LA DEDICACIÓN

Los usuarios podrán utilizar las páginas de esta sección para configurar informes de su dedicación en los proyectos en los que participan.

Informe de la dedicación agregada del participante (VPI01)

Mostrará un informe con la dedicación agregada que el participante ha realizado en el proyecto. Se mostrará el resultado de la petición introducida por el usuario y permitirá copiar el resultado en un archivo externo que generará la aplicación.

Esta sección solo mostrará el resultado del informe de la dedicación agregada, para generar uno nuevo se deberá ir a la sección descrita más abajo (VUD02) con el enlace correspondiente.

Nuevo informe de un participante (VPI02)

Esta GUI consistirá en un formulario donde el usuario podrá configurar las distintas partes de un informe, en base a los proyectos en los que participa. El informe podrá contener la dedicación realizada por el usuario en el proyecto, agregada por semanas, meses o en total.

B.9.3.5 GESTIÓN DEL PERFIL DEL USUARIO

A continuación se describen las GUIs que utilizarán los usuarios para ver y modificar su perfil.

Gestión del perfil del usuario (VPD01)

De forma muy parecida a la aplicación Web, esta GUI mostrará un formulario que le permitirá cambiar alguno de sus datos personales. En otro formulario el usuario podrá cambiar su password. La interfaz contendrá enlaces que permitirán ver la lista de teléfonos y emails del usuario.

[Volver Salir](#)

[/Menú/Datos personales](#)

Datos [Teléfonos](#) [Emails](#)

Login: h48569

Nombre

Apellidos

Tipo usuario: NORMAL

Organización

...

Listado de los teléfonos del usuario (VPD02)

Muestra la lista de teléfonos del usuario. Cada entrada permitirá ser borrada y ser subida o bajada un nivel. Además se dispondrá de un botón para añadir un nuevo teléfono y otro para guardar los cambios. Dispondrá de enlaces para ver el perfil y la lista de emails del usuario.

Volver Salir

/Menú/Datos personales/Teléfonos

Datos Teléfonos Emails

555 38 40 333 bajar>

Teléfono de contacto oficina

555 38 40 333 subir bajar

Teléfono de contacto móvil

555 38 40 333 subir

Teléfono de contacto personal

Lista de los emails del usuario (VPD03)

De forma similar a la lista de teléfonos muestra la lista de direcciones de correo electrónico del usuario.

Volver Salir

/Menú/Datos personales/Emails

Datos Teléfonos Emails

586 @iie.edi bajar>

Email de contacto oficina

586 @iie.edi subir bajar

Email de contacto móvil

586 @iie.edi subir

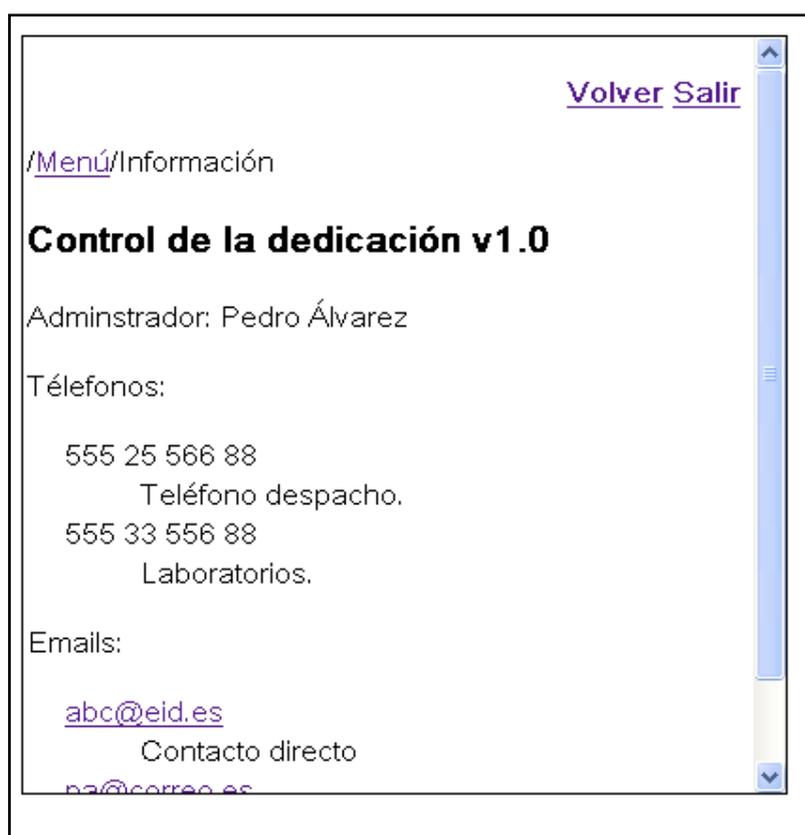
Email de contacto personal

B.9.3.6 CONTACTO CON EL ADMINISTRADOR

Permite mostrar la información para poder contactar con el administrador del sistema.

Información de contacto (VPC01)

Muestra ciertos datos del perfil del administrador junto con la lista de teléfonos y emails. Al igual que en la aplicación Web el administrador del sistema podrá configurar esta información modificando los datos de su perfil.



B.9.3.7 ACCESO AL PORTLET

El Portlet contará con una interfaz que permita acceder a él.

Acceso al sistema (VPS01)

Esta interfaz permitirá a los usuarios que no estén conectados identificarse, mediante su login y contraseña, y poder acceder al sistema como usuarios de tipo participante.

Control de la dedicación

Por favor identifíquese para acceder

Login

Passowrd

B.9.3.8 PANTALLAZOS DE ERROR

Al igual que con la aplicación Web, aquellos errores que el sistema no sepa manejar generarán pantallazos de error.

Pantallazo de error (VPEError)

La pantalla mostrará un mensaje que describa de forma breve que ha sucedido y qué acción ha generado el error. También contendrá un botón que permitirá al usuario ir al menú principal y continuar trabajando.

B.9.4 AYUDA INTEGRADA

Al igual que la aplicación Web, la aplicación Portlet dispondrá de un sistema de ayuda integrada dentro de la misma. Para ver su contenido se deberá presionar el botón  y con el botón  se podrá volver a la sección que se estaba visualizando.

Por motivos de simplificación, el sistema de ayuda del Portlet no mantendrá el contenido de la sección que se estuviera consultando. Dado el bajo número de formularios, este comportamiento apenas afectará al usuario.

Se podrá navegar dentro de las distintas secciones de la ayuda utilizando la lista que se disponga. Habrá una página de ayuda por cada sección a la que pueda acceder el usuario. Las páginas públicas no dispondrán de este servicio.

B.10 COPIAS DE SEGURIDAD

En este capítulo se describe como llevar a cabo las copias de seguridad de los datos de la aplicación, cumpliendo con el requisito RS-2 especificado en el “Documento de Análisis de Requisitos”.

Las copias de seguridad y el proceso de restauración de los datos se llevarán a cabo a través de aplicaciones ya existentes que se gestionarán de forma ajena a las aplicaciones descritas en los demás capítulos.

MySQL ofrece la aplicación “mysqldump” que realiza copias de seguridad de una o más bases de datos alojadas en un servidor MySQL. Con ellas se pueden conseguir archivos con el contenido de la base de datos con un formato de texto plano que incluye el código SQL necesario para su restauración. Esta aplicación está incluida en MySQL [12].

B.10.1 REALIZAR UNA COPIA DE SEGURIDAD

Realizar una copia de seguridad de los datos de la aplicación es muy sencillo con “mysqldump”.

La aplicación dispone de distintos parámetros de configuración, que se pueden listar escribiendo en la línea de comandos “mysqldump –help”, o en el manual en línea en la página:

“<http://dev.mysql.com/doc/refman/5.0/es/mysqldump.html>”.

```
“mysqldump dedicacion01 --single-transaction --quick --extended-insert --
host=<su host> --user=<su login> --pass=<su password> --skip-add-drop-table
--no-create-info --result-file=<archov salida> --tables grupo usuario teléfono
email proyecto tarea participante dedicación limitededicación maxdedicación”
```

Se debe sustituir el contenido entre corchetes angulares (‘<’ y ‘>’) por los datos indicados. Esto generará un fichero de texto plano que contendrá una secuencia de instrucciones SQL “insert” con los datos del sistema.

B.10.2 RESTAURAR UNA COPIA

Para restaurar una copia de seguridad será necesario conectarse al gestor MySQL que contenga la base de datos de la aplicación. Se recomienda que *antes* de iniciar este proceso se cierre el acceso a los demás usuarios de la aplicación.

En primer lugar, se deben borrar los datos existentes escribiendo en la línea de comandos:

```
"delete from dedicacion01.<tabla>
```

Donde "<tabla>" se debe sustituir por el nombre de cada una de las tablas. Se debe realizar esta operación en el siguiente orden de las tablas "dedicación", "maxdedicación", "limitededicación", "participante", "tarea", "proyecto", "teléfono", "email", "usuario" y "grupo".

A continuación se debe lanzar a ejecución el fichero que contiene la copia de seguridad. Desde la línea de comandos de MySQL, esto se realiza de la siguiente forma:

```
"source <nombre_fichero>"
```

Donde se debe sustituir "<nombre_fichero>" por el nombre del archivo que contiene la copia de seguridad.

ANEXO C
DOCUMENTO DE
TEST DE
PRUEBAS

C.1 INTRODUCCIÓN

El Documento de Test de Pruebas (DTP) contiene la descripción del conjunto de pruebas que permiten verificar el correcto funcionamiento de los componentes del sistema desarrollado. Las pruebas han incluido componentes de la capa de datos, de la capa de negocio y componentes transversales.

Las pruebas han sido desarrolladas de forma incremental, comenzando por los componentes inferiores, ascendiendo cada vez. Este modelo permite integrar las pruebas unitarias con las pruebas de integración, evitando tener que utilizar módulos conductores, permitiendo un desarrollo más rápido de las mismas y una mejor comprobación del correcto funcionamiento de las interfaces de cada componente.

Cada conjunto de pruebas irá integrado dentro del subproyecto creado para el componente correspondiente. Para ello se utilizará la carpeta “test”, que no se integrará en el código final de la aplicación. La ejecución de los test de pruebas se realizará desde el entorno del IDE, siendo capaces de verificar el correcto funcionamiento de las distintas partes del sistema sin la necesidad de interactuar con ellas. Para la realización de estas pruebas se utilizarán los datos ficticios creados en la BD.

AVISO IMPORTANTE: En este anexo solo se ha incluido una pequeña parte del DTP, a modo de ejemplo, el documento completo contiene el conjunto de pruebas de los distintos componentes del sistema.

C.2 CAPA DE DATOS

En este capítulo se describe los test de prueba realizados al único componente de la capa de datos.

C.2.1 COMPONENTE “DAO”

El “Data Access Object” ofrece un servicio de acceso al repositorio del sistema, ofreciendo métodos para consultar, modificar y crear los datos de negocio de la aplicación.

C.2.1.1 TEST DAO HIBERNATE

Este conjunto de test pretende comprobar el correcto funcionamiento de la clase “negocio.dao.DaoHibernate”, testeando el correcto funcionamiento de sus métodos. La tabla 2.1 muestra las particiones de equivalencia que se han utilizado para diseñar los test.

Tabla C.2.1 Particiones de equivalencia del Dao			
#	Clase válida	Clase inválida	#
1	Pojo nuevo no existente en la BD.	Pojo nuevo existente en la BD.	1
2	Pojo guardable.	Pojo no actualizable (no existe).	2
3	Lista de pojos guardable.	Clase de pojo no mapeado por Hibernate.	3
4	Pojo, con clave “Id”, existente en la BD.	Pojo nulo.	4
5	Pojo, con clave “Id”, no encontrado en la BD.		
6	Clase de pojo mapeado por Hibernate.		
7	Búsqueda con resultados.		
8	Búsqueda sin resultados.		
9	Lista vacía de pojos guardable.		
10	Pojo borrado.		
11	Lista de pojos borrada.		
12	Filtro válido.		
13	Lista de todos los pojos de una clase.		
14	Pojo válido.		
15	Pojo no válido		
16	Cuenta de pojos mayor que 0.		
17	Cuenta de pojos igual que 0.		
18	Suma de valores mayor que 0.		
19	Suma de valores igual 0.		

Casos de prueba derivados de las clases válidas:

- ⇒ testCrearPojo
 - Crea un nuevo pojo que no existe en la BD. El test es correcto si no se produce ningún error.
 - Clases cubiertas: 1.
- ⇒ testGuardarPojo
 - Guarda un pojo de la clase “Grupo” creado correctamente. El test es correcto si no se produce ninguna excepción. El test no debe modificar la BD.
 - Clases cubiertas: 2.
- ⇒ testGuardarListaPojos
 - Guarda una lista de pojos de la clase “Grupo”. El test es correcto si no se produce ninguna excepción. El test no debe modificar la BD.
 - Clases cubiertas: 3.
- ⇒ testGuardarListaVaciaPojos
 - Guarda una lista de pojos que no contiene ningún pojo. No debería dar ninguna excepción.
 - Clases cubiertas: 9.
- ⇒ testCargarPojo
 - Carga un grupo por su nombre. El pojo existe en la BD y por ello la búsqueda no será nula.
 - Clases cubiertas: 4.
- ⇒ testCargarPojolds
 - Carga un participante por los atributos “usuario.login” y “proyecto.codigo”. El participante existe en la BD y por ello la búsqueda no será nula.
 - Clases cubiertas: 4.
- ⇒ testCargarPojoNoExiste
 - Intenta cargar un pojo de tipo “Usuario” por “Id” que no se encuentra en la BD. El test se cumple si se retorna null.
 - Clases cubiertas: 5.
- ⇒ testCargarPojoldsNoExiste
 - Intenta cargar un pojo de tipo “Participante”, mediante los atributos “usuario.login” y “proyecto.codigo”, que no se encuentra en la BD. El test se cumple si se retorna null.
 - Clases cubiertas: 5.
- ⇒ testCargarTodos
 - Carga todos los grupos de la base de datos. La prueba es correcta si la lista no está vacía.
 - Clases cubiertas: 13.

- ⇒ testBuscarPojos
 - Busca todos los pojos de la clase “Grupo” que empiecen por “Grupo_a”. Debe devolver una lista con longitud mayor que 0.
 - Clases cubiertas: 6.
- ⇒ testBusqueda
 - Realiza la búsqueda de todos los teléfonos del usuario con un login concreto. La búsqueda deberá devolver una lista no vacía.
 - Clases cubiertas: 6.
- ⇒ testBusquedaVacía
 - Busca el usuario con login “noExiste”. La búsqueda deberá devolver una lista vacía.
 - Clases cubiertas: 8.
- ⇒ testFiltrarListaGrupos
 - Realiza el filtrado del listado de todos los pojos. El filtrado debe dar una lista con un número menor de elementos que la original. Se tomará ‘a’ como filtro.
 - Clases cubiertas: 12
- ⇒ testBorrarPojo
 - Borra un pojo de la clase Grupo, que previamente se ha guardado, en la BD.
 - Clases cubiertas: 10.
- ⇒ testBorrarListaPojos
 - Borra una lista de pojos de la clase Grupo, previamente cargados, de la BD.
 - Clases cubiertas. 11.
- ⇒ testPojoValido
 - Crea un pojo de la clase “Grupo” con todos sus valores dentro de los márgenes especificados por las restricciones y lo valida. La validación debe devolver un conjunto de errores vacío (0 errores).
 - Clases cubiertas: 14.
- ⇒ testPojoNoValido
 - Crea un pojo de la clase “Grupo” e inicializa sus valores incumpliendo las restricciones impuestas (nombre con 1 solo carácter y descripción con más de 200 caracteres). La función “validar” del Dao debe devolver un conjunto con 2 errores.
 - Clases cubiertas: 15.
- ⇒ testCuentaMayorCero
 - Cuenta todos los grupos que empiezan por la palabra “Grupo”. La cuenta deberá dar un número mayor que 0. Ésta se realiza con ambas funciones “contar” y se comprueba la igualdad de los resultados
 - Clases cubiertas: 16.

- ⇒ testCuentaIgualCero
 - Cuenta todos los grupos que comienzan por la palabra “NoExisto”. La cuenta debería ser 0. Ésta se realiza con ambas funciones “contar” y se comprueba la igualdad de los resultados
 - Clases cubiertas: 17.
- ⇒ testSumaMayorCero
 - Suma un conjunto de dedicaciones cuyo valor total de dedicación será mayor que 0.
 - Clases cubiertas: 18.
- ⇒ testSumaIgualCero
 - Suma un conjunto vacío de dedicaciones, el resultado deberá ser 0.
 - Clases cubiertas: 19.

Casos de prueba derivados de las clases no válidas:

- ⇒ testCrearPojoYaExistente
 - Crea un pojo que ya existía en la BD produciendo una excepción.
 - Clases cubiertas: 1.
- ⇒ testGuardarPojoNoExistente
 - Intenta guardar un pojo que no existe en la BD produciendo una excepción.
 - Clases cubiertas: 2.
- ⇒ testCargarClaseNoExistente
 - Intenta cargar un pojo cuya clase no existe. La prueba deberá dar una excepción.
 - Clases cubiertas: 3.
- ⇒ testGuardarPojoNulo
 - Guarda un pojo nulo. El test debe dar una excepción.
 - Clases cubiertas: 4.

BIBLIOGRAFÍA Y REFERENCIAS

[1] Escuela de Ingeniería y Arquitectura (EINA)

<https://eina.unizar.es/>

[2] Universidad de Zaragoza (UZ)

<http://www.unizar.es/>

[3] Grupo de Integración de sistemas Distribuidos y Heterogéneos (GIDHE)

<http://www.gidhe.es/>

[4] Departamento de Informática e Ingeniería de Sistemas (DIIS)

<http://diis.unizar.es/>

[5] Portlet

<http://es.wikipedia.org/wiki/Portlet>

[6] Portal Web

[http://es.wikipedia.org/wiki/Portal_\(Internet\)](http://es.wikipedia.org/wiki/Portal_(Internet))

[7] Java Platform Edition (JEE)

<http://www.oracle.com/technetwork/java/javaee/tech/index.html>

[8] Spring Framework

<http://www.springsource.org/>

[9] Inversión del control e Inyección de dependencias

M. Fowler. Inversion of Control Containers and the Dependency Injection pattern.

<http://www.martinfowler.com/articles/injection.html>

[10] Hibernate Framework

<http://www.hibernate.org/>

[11] The Apache Software Foundation. Apache Log4J.

<http://logging.apache.org/log4j/>

[12] MySQL

<http://www.mysql.com/>

[13] Mapeo Objeto Relacional (ORM)

http://es.wikipedia.org/wiki/Mapeo_objeto-relacional (Español)

http://en.wikipedia.org/wiki/Object-relational_mapping (Inglés)

[14] JSR 286 Portlet 2.0

<http://www.jcp.org/en/jsr/detail?id=286>

[15] JSR 168 Portlet 1.1

<http://www.jcp.org/en/jsr/detail?id=168>

[16] Open Portal Portlet Container

<http://openportal.sourceforge.net/>

[17] Apache Tomcat

<http://tomcat.apache.org/>

[18] JBoss

<http://www.jboss.org/overview/>

[19] Jetty

<http://www.eclipse.org/jetty/>

<http://es.wikipedia.org/wiki/Jetty>

[20] Oracle WebLogic

http://es.wikipedia.org/wiki/Oracle_WebLogic

[21] Proyecto GlashFish

<http://glassfish.java.net/>

[22] Oracle Java Servlet

<http://www.oracle.com/technetwork/java/javasee/servlet/index.html>

[23] JSR 154 Servlet

<http://jcp.org/en/jsr/detail?id=154>

[24] Java Server Pages

<http://www.oracle.com/technetwork/java/jsp-138432.html>

[25] Asynchronous JavaScript And XML (AJAX)

<http://es.wikipedia.org/wiki/AJAX>

[26] Prototype Framework

<http://prototypejs.org/>

[27] Edge Diagrammer

<http://www.pacestar.com/>

[28] ArgoUML

<http://argouml.tigris.org/>

[29] Apache Commons

<http://commons.apache.org/>

[30] iText Library

<http://www.itextpdf.com/>

[31] W3C

<http://www.w3.org/>

[32] JUnit

<http://junit.org/>

[33] Java Development Kit (JDK)

<http://www.oracle.com/technetwork/java/javase/overview/index.html>

[34] NetBeans

<https://netbeans.org/>

[35] Mozilla Firefox

<http://www.mozilla.org/es-ES/firefox/fx/>

[36] Internet Explorer

<http://windows.microsoft.com/es-ES/windows/home>

[37] MySQL Workbench

<http://www.mysql.com/products/workbench/>

[38] M. Fowler. Patterns of Enterprise Application Architecture. Addison-Wesley, Indiana, 2002

[39] Modelo Vista Controlador

http://es.wikipedia.org/wiki/Modelo_vista_controlador

[40] Sistema de gestión de bases de datos (SGBD)

<http://es.wikipedia.org/wiki/SGBD>

[41] Plain Old Java Object

<http://es.wikipedia.org/wiki/POJO>

[42] Desarrollo iterativo y creciente

http://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente

[43] JavaServer Faces

http://es.wikipedia.org/wiki/JavaServer_Faces

[44] JOnAS

<http://es.wikipedia.org/wiki/JOnAS>

[45] Eclipse

[http://es.wikipedia.org/wiki/Eclipse_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))