

TRABAJO FINAL DE CARRERA .NET

Aplicación de gestión de tareas

11/06/2012

TFC ETIS

Alumno: José Antonio García Andreu

Consultor: Jairo Sarrias Guzman



Universitat Oberta
de Catalunya

Contenido

1.- Resumen del proyecto	4
2.- Especificación del trabajo	5
2.1.- Descripción del trabajo	5
2.2.- Objetivos	5
2.3.- Estudio de la idoneidad del proyecto	6
3.- Organización del proyecto	7
3.1.- Planificación inicial	7
3.2.- Planificación real	7
3.3.- Diagrama de Gantt	7
3.4.- Producto final	9
4.- Análisis de requerimientos	10
5.- Arquitectura	12
5.1.- Lenguaje	12
5.2.- Persistencia y base de datos	12
5.3.- Presentación	14
5.4.- Complementos	15
6.- Diagramas	17
6.1 Diagrama de casos de uso	17
6.2.- Descripción de los casos de uso	19
6.3.- Diagrama de clases	22
6.4.- Diagrama de secuencia	24
6.4.1 Diagrama de creación de la tarea	24
6.4.2 Diagrama de actualización de una tarea	25
6.5 Diagrama de base de datos	26
6.6.- Descripción de las tablas	27
7.- Diseño de la interfaz de usuario	31
7.1 Pantallas comunes a todos los roles	31
7.1.1 Pantalla de acceso a la aplicación	31
7.1.2 Pantalla de cambio de contraseña	31
7.1.3 Pantalla de búsqueda de tareas	32
7.1.4 Pantalla de creación de tareas	33
7.1.5 Pantalla de creación de tarea desde una plantilla	33
7.1.6 Pantalla de configuración del usuario	34

7.2 Pantallas comunes a los Usuarios y a Sistemas.....	35
7.2.1 Pantalla de eliminación de una tarea.....	35
7.2.2 Pantalla de comentario a una tarea.....	35
7.2.3 Pantalla de fichero a una tarea	36
7.2.4 Pantalla de detalles de la tarea	37
7.2.5 Pantalla de notificaciones	38
7.3 Pantallas de los usuarios de Sistemas	38
7.3.1 Pantalla de editar tarea.....	38
7.3.2 Pantalla de inicio	39
7.4 Pantallas de los usuarios	40
7.4.1 Pantalla de editar tarea.....	40
7.4.2 Pantalla de lista de tareas	41
7.5 Pantallas del Administrador	41
7.5.1 Pantalla de lista de plantillas.....	41
7.5.2 Pantalla de creación de plantilla	42
7.5.3 Pantalla de edición de plantilla	42
7.5.4 Pantalla de eliminación de plantilla	43
7.5.5 Pantalla de detalles de la plantilla.....	43
7.5.6 Pantalla de lista de estados.....	44
7.5.7 Pantalla de creación de estados.....	44
7.5.8 Pantalla de edición de estados.....	44
7.5.9 Pantalla de eliminación de estados.....	45
7.5.10 Pantalla de ordenación de estados	46
7.5.11 Pantalla de listado de departamentos	46
7.5.12 Pantalla de creación de departamentos	47
7.5.13 Pantalla de edición de departamentos	47
7.5.14 Pantalla de detalles de un departamento.....	47
7.5.15 Pantalla de eliminación de departamentos	48
7.5.16 Pantalla de lista de proyectos	48
7.5.17 Pantalla de creación de proyectos	49
7.5.18 Pantalla de edición de proyectos	49
7.5.19 Pantalla de detalles de proyectos	50
7.5.20 Pantalla de eliminación de proyectos	50
7.5.21 Pantalla de lista de usuarios.....	51

7.5.21 Pantalla de creación de usuarios.....	51
7.5.22 Pantalla de eliminación de usuarios.....	52
7.5.23 Pantalla de visualización de informes	52
8.- Instalación y configuración del software	55
8.1.- Pre-requisitos:.....	55
8.2.- Restauración de la base de datos:	55
8.3.- Uso de los fuentes:.....	56
8.4.- Ponerlo en producción:.....	56
8.5.- Usuarios de prueba	57
9.- Posibles mejoras	58
10.- Evaluación de costes	59
11.- Conclusiones	60
12.- Bibliografía	61
MVC.....	61
Linq	61
Ajax,JavaScript,Jquery,Json	61
Hojas de estilo	61

1.- Resumen del proyecto

El siguiente proyecto de fin de carrera, tiene como objetivo el de intentar solucionar la problemática que tienen los departamentos que ofrecen servicios dentro de una empresa, como puede ser el del departamento de sistemas.

Debido a la diversidad de puestos de trabajo desde los cuales se puede hacer uso de la aplicación, incluyendo ordenadores basados en sistemas Mac (como los que hay en los departamentos de marketing o diseño) o incluso usuarios con “terminales tontos”, se ha optado por realizar una aplicación web, ya que solo necesitaremos hacer uso de un navegador web para poder usarla.

Como consecuencia de esta decisión se ha desarrollado la aplicación haciendo uso de las tecnologías que nos ofrece Microsoft, basándonos en el modelo MVC (Modelo Vista y Controlador) para desarrollar el software, ya que estos tipos de patrones nos permiten un desarrollo ágil y estructurado.

Se ha optado por la tecnología de Microsoft, basándonos en su amplia experiencia en sistemas operativos, servidores web y bases de datos. Además, una empresa como está siempre nos ofrece una buena cantidad de documentación y unas herramientas para desarrollar, que aúnan en un solo paquete todo el software que necesitamos.

También se ha intentado hacer una aplicación lo mas completa y rica al usuario posible, añadiendo controles web que mejoran la experiencia en la aplicación, y ajustándonos a la realidad, se ha incluido también software para la generación de reportes.

2.- Especificación del trabajo

2.1.- Descripción del trabajo

Las empresas que disponen de un departamento propio de sistemas, lo utilizan como un departamento que ofrece servicios. Es un departamento que ofrece recursos para solucionar o mejorar procesos que se requieren desde otros departamentos, por eso surge la necesidad de proporcionarles las herramientas adecuadas para la petición, supervisión y seguimiento de estos requerimientos.

Por lo tanto, el software que se pretende desarrollar, está claramente enfocado a la gestión de las tareas y sobre todo a unificar la vía por la que son requeridas, ya que es muy normal que estas peticiones se realizarán unas veces por email, otras por teléfono, o incluso abordando al responsable de la tarea cuando camina por la empresa. Estas tareas pueden provenir de todas las áreas de la empresa, incluso desde el mismo departamento de sistemas, con lo que obtenemos una solución global y única para todos los usuarios. Lo que se pretende conseguir es que una vez un usuario ha demandado una necesidad (mediante la creación de la correspondiente tarea), se le pueda hacer un seguimiento de esta y se sepa en todo momento en el estado en el que se encuentra (mediante las notificaciones que se pueden activar en la aplicación e incluso por email), y quien es el responsable de realizar la tarea (ya que en todo momento se sabe que persona del departamento se ha hecho cargo de realizar la tarea). Incluso la aplicación proveerá un sistema de comunicación entre las dos partes interesadas en la resolución de la tarea. Adicionalmente desde el departamento de sistemas se está llevando un control sobre los proyectos en los que se trabaja (podemos ver la carga de trabajo de un recurso del departamento), el uso que se hace de los recursos físicos y la planificación de estos (gracias al calendario que el usuario de sistemas va generándose), con lo que conseguimos una optimización del personal.

Además, por medio de unos informes implementados en la aplicación, se puede obtener mucha mas información, preparada de una forma mas clara y mas enfocada a las necesidades de ese momento, que la que obtenemos si usamos las búsquedas que hay dentro de la aplicación.

2.2.- Objetivos

El objetivo a nivel personal es alcanzar los conocimientos necesarios para desarrollar el proyecto, tanto a nivel de analista como de desarrollador. La elección de este proyecto viene determinada por la necesidad e inquietud personal de seguir aprendiendo algunas de las tecnologías más modernas y con más futuro dentro de la empresa privada. Además, me apetecía mucho el poder empezar a usar tecnologías destinadas a realizar páginas web y todo lo que las envuelve, como son las hojas de estilo o librerías Ajax y JavaScript, las cuales no había tenido oportunidad de usar.

2.3.- Estudio de la idoneidad del proyecto

Actualmente, las empresas deben optimizar los recursos de los que disponen para poder competir en un mercado en clara recesión. El departamento de sistemas no es una excepción, y es por ello por lo que surge la necesidad de elaborar un software que nos permita relacionarnos con los demás departamentos de la empresa, a los cuales se les puedan ofrecer soluciones rápidas y satisfactorias, ahorrando tiempo y dinero al mismo tiempo.

Debido a la naturaleza de nuestro trabajo, en el cual se requiere una concentración y dedicación casi exclusiva, las interrupciones hacen que se pierda esta concentración, por lo cual, lo que requiera de nuestra atención debería venir únicamente por una vía. Esta vía es el software que se pretende realizar.

Además, en todo momento se optimiza el uso de los recursos del departamento, generándose una base de conocimiento y manteniendo un control sobre los proyectos que se están realizando.

3.- Organización del proyecto

3.1.- Planificación inicial

Para la planificación del proyecto nos hemos basado en los hitos que marcaba la entrega de cada pac, intentando ajustarnos lo máximo posible. Hemos creado un diagrama de Gantt para hacernos la planificación, descomponiendo las fases del desarrollo para tener una mejor referencia temporal .

3.2.- Planificación real

Personalmente pienso que la parte mas difícil es la de la implementación, cuya realización estaba pensada para la pac3, por lo que durante la pac1 y la pac2 he ido adelantando tareas que eran propias de esta pac. Una vez terminado el TFC pienso que fue una elección acertada por que me ha permitido desarrollar todas las funcionalidades que se habían planteado y alguna extra.

3.3.- Diagrama de Gantt

Id	Nombre de tarea	Duración	Comienzo	Fin	01 febrero		01 marzo		01 abril		01 mayo		01 junio		01 jul
					23/01	06/02	20/02	05/03	19/03	02/04	16/04	30/04	14/05	28/05	11/06
1	TFC	75 días	jue 01/03/12	mié 13/06/12											
2	PAC1-Plan de trabajo	9 días	jue 01/03/12	mar 13/03/12											
3	Estudio previo del proyecto	2 días	jue 01/03/12	vie 02/03/12											
4	Propuesta del TFC	1 día	lun 05/03/12	lun 05/03/12											
5	Revisión de la propuesta del TFC	1 día	mar 06/03/12	mar 06/03/12											
6	Instalación del software	2 días	mié 07/03/12	jue 08/03/12											
7	Elaboración y documentación del plan de trabajo	3 días	vie 09/03/12	mar 13/03/12											
8	PAC2-Análisis y Diseño	22 días	mar 13/03/12	mié 11/04/12											
9	Análisis	12 días	mar 13/03/12	mié 28/03/12											
10	Análisis de la arquitectura	2 días	mar 13/03/12	mié 14/03/12											
11	Diagrama casos de uso y actores	4 días	jue 15/03/12	mar 20/03/12											
12	Diagrama de clases	3 días	mié 21/03/12	vie 23/03/12											
13	Diagramas Entidad-Relación	3 días	lun 26/03/12	mié 28/03/12											
14	Diseño	10 días	jue 29/03/12	mié 11/04/12											
15	Planificación de las funcionalidades y requerimientos	2 días	jue 29/03/12	vie 30/03/12											
16	Diseño de la estructura del proyecto	1 día	lun 02/04/12	lun 02/04/12											
17	Diseño de las pantallas	3 días	mar 03/04/12	jue 05/04/12											
18	Diseño de la base de datos	4 días	vie 06/04/12	mié 11/04/12											
19	Documentación	1 día	mié 11/04/12	mié 11/04/12											
20	Documentación de la fase	1 día	mié 11/04/12	mié 11/04/12											
21	PAC3-Implementación	27 días	jue 12/04/12	vie 18/05/12											
22	Implementación del código del proyecto	19 días	jue 12/04/12	mar 08/05/12											
23	Pruebas unitarias	4 días	mié 09/05/12	lun 14/05/12											
24	Documentación	3 días	mar 15/05/12	jue 17/05/12											
25	Memoria y presentación	19 días	vie 18/05/12	mié 13/06/12											
26	Ajustes finales de codificación del proyecto	5 días	vie 18/05/12	jue 24/05/12											
27	Redacción de la memoria	7 días	vie 25/05/12	lun 04/06/12											
28	Desarrollo y elaboración de la presentación	7 días	mar 05/06/12	mié 13/06/12											

Planificación del proyecto

3.4.- Producto final

Cuando se termine el proyecto obtendremos una aplicación web dividida en una serie de ficheros, organizados en las siguientes carpetas:

- **Bin:** carpeta donde se almacenan todas las dll's necesarias para el correcto funcionamiento del proyecto, entre ellas encontramos "*MVC_Tareas_TFC.dll*", que es la dll donde se almacena toda la lógica de nuestra aplicación.
- **Content:** carpeta en la que se almacenan todos los recursos CSS, imágenes y temas JQuery que hemos usado.
- **Informes:** carpeta donde se encuentran todos los informes que vamos a usar en la aplicación.
- **Scripts:** en esta carpeta se almacenan los ficheros JavaScript que hemos usado.
- **Views:** aquí encontramos las carpetas donde se almacenan las vistas, que además se corresponden con los controladores que habíamos definido.
- Tres ficheros de configuración, que son:
 - Global.asax
 - Packages.config
 - Web.config

Estos ficheros una vez copiados en el ISS (*Internet Information Server*) conformarán la página web, que será accesible desde cualquier navegador.

4.- Análisis de requerimientos

La aplicación que estamos desarrollando va destinada a las organizaciones con un departamento de sistemas propio, el cual hay que organizar correctamente para poder optimizar los recursos económicos y físicos.

En esta aplicación por tanto tenemos que distinguir los tipos de usuarios que tienen acceso a ella, este acceso se realiza a través de la introducción del usuario y la contraseña correspondiente, pudiendo diferenciar tres roles distintos:

Administrador:

Este rol es el encargado de la creación y administración de la mayoría de las opciones que están disponibles dentro de la aplicación. Entre las que podemos distinguir:

- Departamentos: Son los distintos departamentos que hay dentro de la organización y a los cuales pertenecen cada uno de los usuarios.
- Plantillas de tareas: Son plantillas destinadas a ahorrar tiempo a la hora de crear tareas que son repetitivas por su uso. De esta manera, creando plantillas de tareas, las opciones más comunes para cada tarea ya las tenemos introducidas, con lo que solo hay que rellenar el campo del detalle de la tarea.
- Estados de las tareas: Cada tarea puede pasar por un número indeterminado de estados hasta su resolución y puesta en marcha, por lo que el administrador está capacitado para la creación, modificación y eliminación de estos estados. Una vez los estados están creados, es posible ordenarlos para que se ajuste a las necesidades las tareas.
- Proyectos: Cada tarea está obligatoriamente relacionada con un proyecto. Y pueden haber tantos proyectos como sean necesarios.
- Usuarios: Una parte fundamental de la aplicación es que hayan usuarios que la utilicen, por lo tanto el administrador puede crear estos usuarios. Al crearlos debe elegir uno de los departamentos, al cual pertenece, y debe asignarle también el tipo de rol que quiere para el usuario entre los disponibles:
 - Administrador: Administrador de la aplicación.
 - Sistemas: Son los encargados de resolver las tareas, por lo que tienen algunas funcionalidades adicionales a los usuarios.
 - Usuario: Son el resto de usuarios de la aplicación y su función es únicamente la de crear tareas de acuerdo a sus necesidades

Sistemas:

Son los destinatarios de las tareas, y por tanto son los encargados de resolverlas. Por eso su pantalla inicial de la aplicación es una lista con todas las tareas que tiene pendiente de resolver, pudiendo también ver las tareas de otros compañeros o las que no tienen un usuario especificado. Los usuarios con el rol de sistemas tienen habilitadas algunas opciones extra en las tareas que no tienen los usuarios normales. Entre ellas se encuentran:

- Finalizar una tarea: Cuando ha terminado con la tarea se puede finalizar, para que ya no aparezca entre las que hay que realizar, obligatoriamente se debe rellenar el campo de minutos empleados en la aplicación para después poder sacar las estadísticas de tiempo por proyectos y por usuario de Sistemas.
- Cambiar un estado: Como hemos indicado antes, son los distintos estados en los que se puede encontrar una tarea hasta que se finaliza.
- Cambiar destinatario: Al crear una tarea, ya sea desde plantilla como desde el principio, hay que elegir un destinatario de esta. Si no se tiene claro cual es la persona responsable de la tarea, se puede elegir el usuario genérico del departamento de sistemas, y de este modo la tarea les aparece a todos los usuarios del departamento de sistemas. Por lo tanto, estos usuarios pueden cambiar el destinatario y asignárselos a ellos mismos o por el contrario a un compañero para que sea él, el que realice la tarea.

Usuarios:

Son los encargados de crear tareas para satisfacer alguna de sus necesidades. Al entrar a la aplicación, lo primero que ven son las tareas que tienen abiertas o las finalizadas con la notificación de terminada, y las siguientes opciones:

- Crear tarea (o crear tarea desde plantilla): sirve para crear tareas nuevas.
- Buscar tareas: sirve para buscar entre todas las tareas que el usuario ha creado con la ayuda de filtros
- Requerir la atención del destinatario sobre la tarea.
- Añadir comentarios en la tarea: para así comunicarse con el destinatario sobre algún tema relacionado con la tarea.
- Añadir ficheros en la tarea: para así poder adjuntar imágenes o ficheros para la correcta resolución de la tarea.

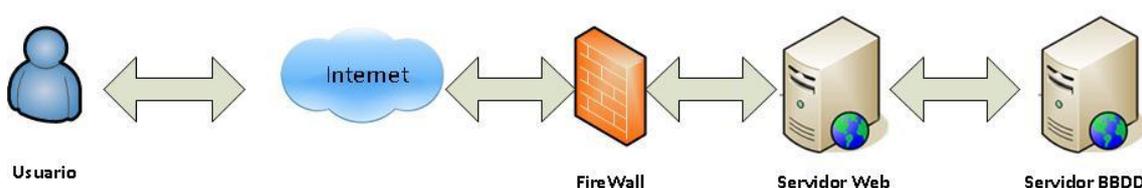
También disponen de otras opciones a realizar como son:

- Cambiar las opciones iniciales al crear una tarea sin la ayuda de las plantillas, como pueden ser:
 - Notificar al leer
 - Notificar al cambiar el estado
 - Notificar al finalizar la tarea
 - Notificar además de por la aplicación por email
- Cambiar su contraseña para iniciar la aplicación

5.- Arquitectura

La arquitectura del proyecto está enfocada claramente a Internet, por lo tanto vamos a usar una arquitectura cliente-servidor. En este tipo de arquitectura el cliente accede a la aplicación a través del navegador web, que es el encargado de dar soporte a las peticiones. Este tipo de arquitectura nos da unos beneficios, como son:

- Accesibilidad desde cualquier ordenador conectado a Internet.
- No hay que instalar aplicaciones en los clientes.
- Datos en tiempo real entre los diferentes clientes de la aplicación.



En el sistema podemos ver los siguientes elementos:

- Cliente: es el usuario que va a hacer uso de la aplicación.
- Firewall: Elemento que nos protegerá de los posibles ataques del exterior.
- Servidor web: Ordenador encargado de proporcionar la aplicación web, en nuestro caso utilizaremos Internet Information Server 7.5 de Microsoft
- Servidor de base de datos: Ordenador encargado de almacenar los datos de la aplicación, en nuestro caso nos decantamos por un Sql Server 2008 R2 también de Microsoft

Dependiendo de varios factores (tanto económicos como de disponibilidad física) los elementos Firewall, Servidor web y Servidor de base de datos se pueden unir en un mismo servidor. En este caso estamos perdiendo rendimiento y seguridad.

5.1.- Lenguaje

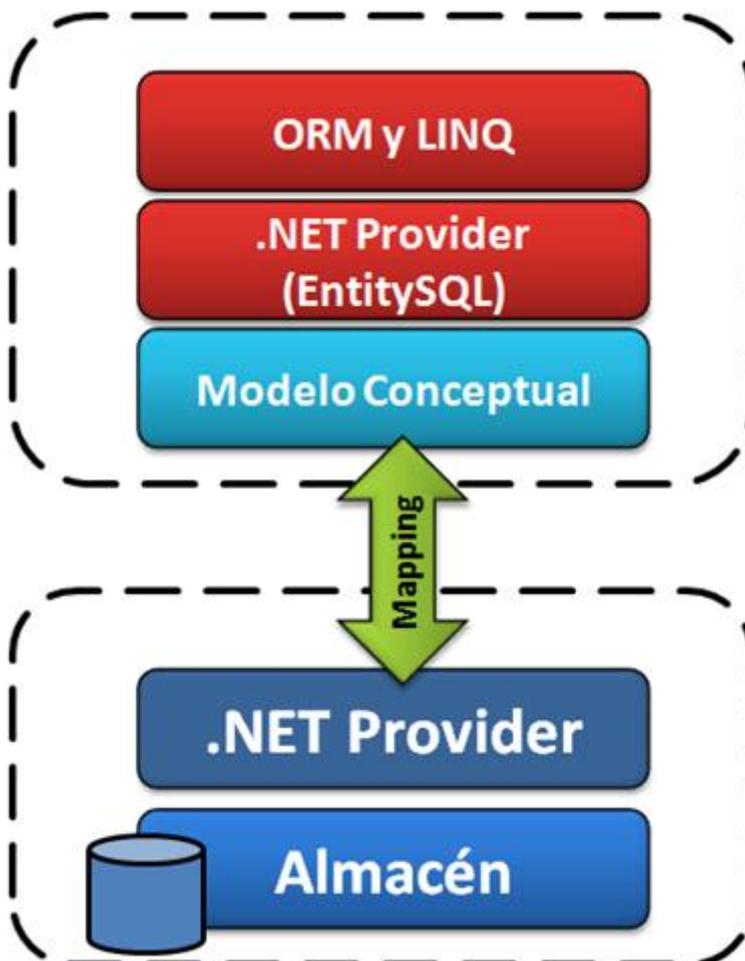
Como lenguaje de programación hemos elegido C# con el Framework 4.0, ya que es el lenguaje creado específicamente para desarrollar la tecnología .NET. Además nos aprovechamos de la gran documentación que tenemos disponible en internet sobre este lenguaje.

5.2.- Persistencia y base de datos

Para la capa de persistencia de datos vamos a utilizar Entity Framework 4.1 atacando a una base de datos Microsoft Sql Server 2008 mediante Linq. Se ha elegido esta tecnología por ser la última que ha desarrollado Microsoft, permitiendo a los desarrolladores crear aplicaciones de acceso a datos programando con un modelo de aplicaciones conceptuales en lugar de programar directamente con un esquema de almacenamiento relacional. El objetivo es reducir

la cantidad de código y el mantenimiento necesarios para las aplicaciones orientadas a datos. Las aplicaciones de Entity Framework ofrecen las siguientes ventajas:

- Las aplicaciones pueden funcionar en términos de un modelo conceptual más centrado en la aplicación, que incluye tipos con herencia, miembros complejos y relaciones.
- Las aplicaciones están libres de dependencias de codificación rígida de un motor de datos o de un esquema de almacenamiento.
- Las asignaciones entre el modelo conceptual y el esquema específico de almacenamiento pueden cambiar sin tener que cambiar el código de la aplicación.
- Los desarrolladores pueden trabajar con un modelo de objeto de aplicación coherente que se puede asignar a diversos esquemas de almacenamiento, posiblemente implementados en sistemas de administración de base de datos diferentes.
- Se pueden asignar varios modelos conceptuales a un único esquema de almacenamiento.
- La compatibilidad con Language Integrated Query (LINQ) proporciona validación de la sintaxis en el momento de la compilación para consultas en un modelo conceptual.



Funcionamiento de LINQ

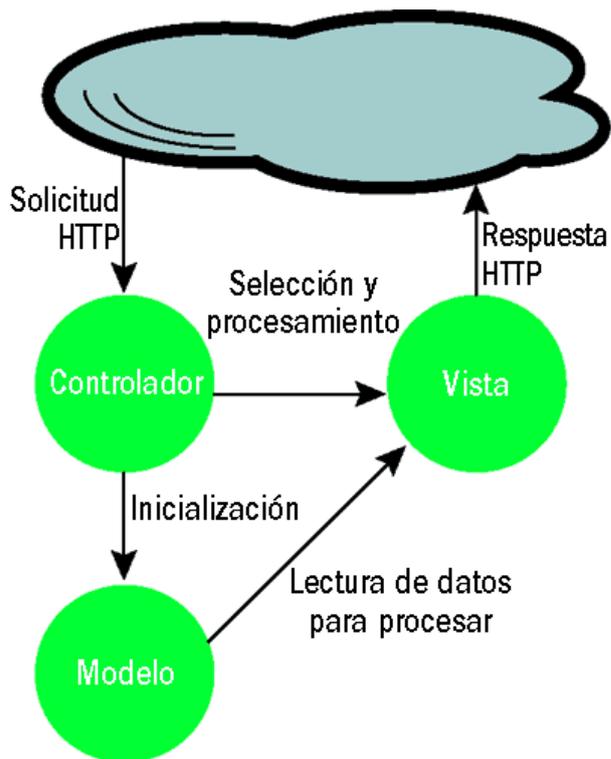
5.3.- Presentación

Para las páginas web vamos a utilizar ASP.NET MVC 3 , ya que el modelo MVC nos ayuda a crear aplicaciones que separan los aspectos diferentes de la aplicación (lógica de entrada, lógica comercial y lógica de la interfaz de usuario), proporcionando un vago acoplamiento entre estos elementos. El modelo especifica dónde se debería encontrar cada tipo de lógica en la aplicación. La lógica de la interfaz de usuario pertenece a la vista. La lógica de entrada pertenece al controlador. La lógica comercial pertenece al modelo. Esta separación le ayuda a administrar la complejidad al compilar una aplicación, ya que le permite centrarse en cada momento en un único aspecto de la implementación. Por ejemplo, se puede centrar en la vista sin estar condicionado por la lógica comercial.

El acoplamiento vago entre los tres componentes principales de una aplicación MVC también favorece el desarrollo paralelo. Por ejemplo, un desarrollador de software puede trabajar en la vista, un segundo desarrollador puede ocuparse de la lógica del controlador y un tercero se puede centrar en la lógica comercial del modelo.

Los componentes de la tecnología son:

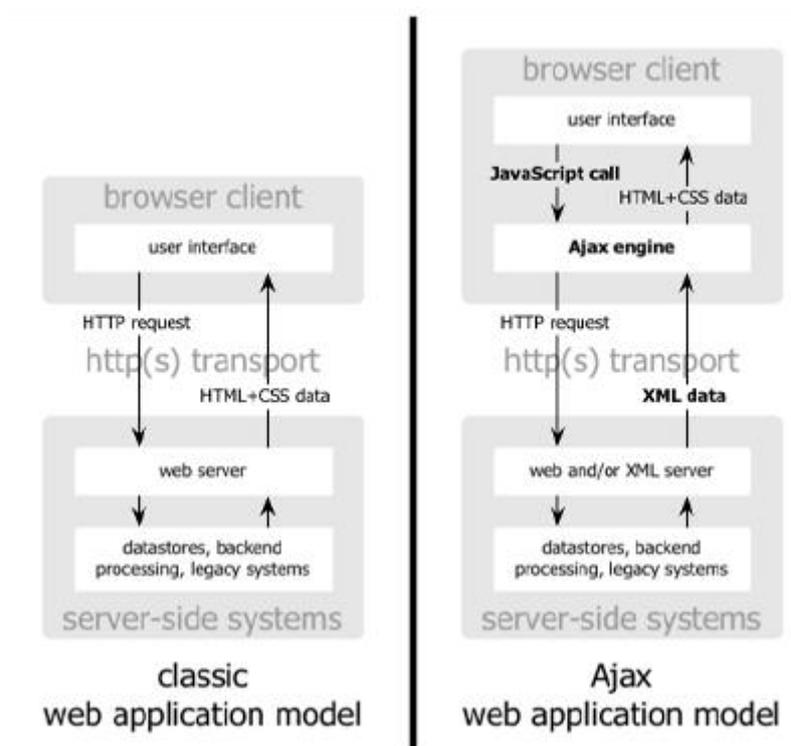
- **Modelos.** Los objetos de modelo son las partes de la aplicación que implementan la lógica del dominio de datos de la aplicación. A menudo, los objetos de modelo recuperan y almacenan el estado del modelo en una base de datos. En las aplicaciones pequeñas, el modelo es a menudo una separación conceptual en lugar de física. Por ejemplo, si la aplicación solo lee un conjunto de datos y lo envía a la vista, la aplicación no tiene un nivel de modelo físico y las clases asociadas. En ese caso, el conjunto de datos asume el rol de un objeto de modelo.
- **Vistas.** Las vistas son los componentes que muestra la interfaz de usuario de la aplicación. Normalmente, esta interfaz de usuario se crea a partir de los datos de un modelo.
- **Controladores.** Los controladores son los componentes que controlan la interacción del usuario, trabajan con el modelo y por último seleccionan una vista para representar la interfaz de usuario. En una aplicación MVC, la vista solo muestra información; el controlador administra y responde a los datos proporcionados por el usuario y su interacción. Por ejemplo, el controlador administra los valores de la cadena de consulta y pasa estos valores al modelo, que a su vez podría utilizarlos para consultar la base de datos.



Funcionamiento de la tecnología MVC

5.4.- Complementos

Además vamos a añadir al código JavaScript y llamadas AJAX, para hacer más rica la experiencia del usuario.



Diferencia entre el modelo clásico y con llamadas Ajax

6.- Diagramas

6.1 Diagrama de casos de uso

Los diagramas de casos de uso documentan el comportamiento de un sistema desde el punto de vista del usuario. Por lo tanto los casos de uso determinan los requisitos funcionales del sistema, es decir, representan las funciones que un sistema puede ejecutar.

Su ventaja principal es la facilidad para interpretarlos, lo que hace que sean especialmente útiles en la comunicación con el cliente.

Elementos básicos:

- Actores: Los actores representan un tipo de usuario del sistema. Se entiende como usuario cualquier cosa externa que interactúa con el sistema. No tiene por qué ser un ser humano, puede ser otro sistema informático o unidades organizativas o empresas.
- Caso de uso: Es una tarea que debe poder llevarse a cabo con el apoyo del sistema que se está desarrollando. Se representan mediante un óvalo. Cada caso de uso debe detallarse, habitualmente mediante una descripción textual.
- Asociaciones: Hay una asociación entre un actor y un caso de uso si el actor interactúa con el sistema para llevar a cabo el caso de uso
- Un escenario: es una interacción entre el sistema y los actores, que puede ser descrito mediante una secuencia de mensajes. Un caso de uso es una generalización de un escenario.

Vamos a ver el nuestro:

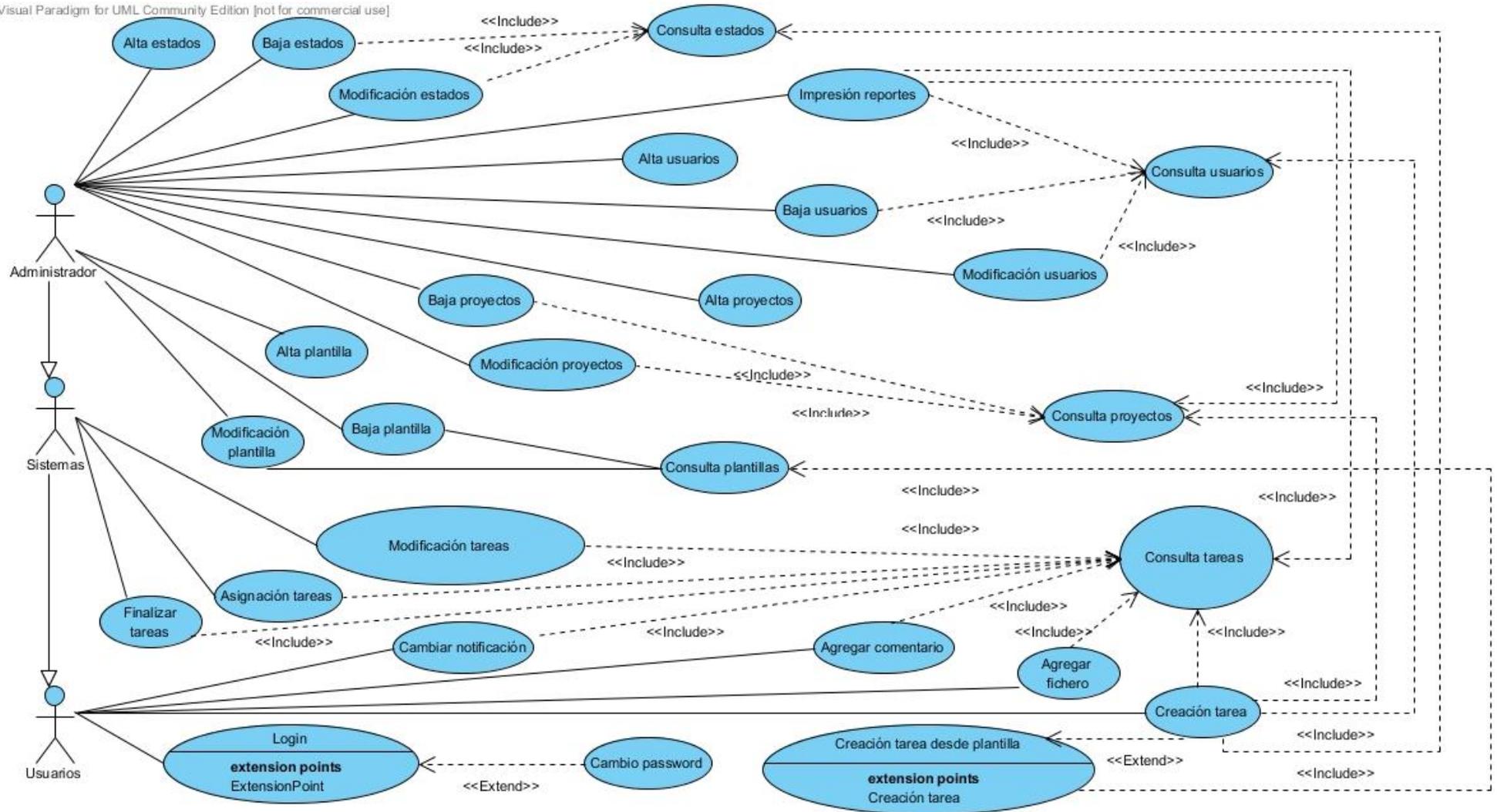


Diagrama de caso de uso

6.2.- Descripción de los casos de uso

Vamos a explicar los casos de uso más importantes:

Descripción	Valor
Caso de uso	Creación de tarea.
Actores	Usuarios, Sistemas, Administrador.
Descripción	Creación de una tarea para satisfacer una necesidad hardware o software del usuario por el departamento de sistemas.
Pre-condiciones	Estar autenticado en el sistema.
Post-condiciones	El sistema crea la tarea.
Flujo principal	<ol style="list-style-type: none"> 1.- El usuario a través de la página principal de la aplicación, seleccionará la creación de una tarea con o sin plantilla. 2.- Si a elegido con plantilla seleccionará la plantilla adecuada. 3.- Llegamos a la ventana de introducción de datos de la tarea. 4.- Completamos los datos que no estén rellenos. 5.- Pulsamos el botón “<i>crear</i>” y se crea la tarea.
Flujo alternativo	<p>Cancelar operación: Antes de pulsar el botón de crear podemos abortar la creación de la tarea simplemente pulsando el enlace hacia la página principal de la aplicación</p> <p>Error en los datos: Si a la hora de crear la tarea se produce algún error el sistema nos informará .</p>

Descripción	Valor
Caso de uso	Edición de tarea.
Actores	Usuarios, Sistemas, Administrador.
Descripción	Permite la modificación de la tarea para completar o actualizar los datos de esta.
Pre-condiciones	Estar autenticado en el sistema y seleccionar una tarea.
Post-condiciones	Obtenemos la tarea modificada
Flujo principal	<ol style="list-style-type: none"> 1.- Dependiendo de si somos Usuario o Sistemas modificamos los datos que se nos presentan en la página. 2.- Pulsamos el botón “<i>guardar</i>”.
Flujo alternativo	<p>Cancelar operación: Antes de pulsar el botón de guardar podemos abortar la modificación de la tarea simplemente pulsando el enlace hacia la página principal de la aplicación</p> <p>Error en los datos: Si a la hora de modificar la tarea se produce algún error el sistema nos informará.</p>

Descripción	Valor
Caso de uso	Agregar comentario.
Actores	Usuarios, Sistemas, Administrador.
Descripción	Permite agregar un comentario a una tarea para así poder interactuar con los otros usuarios que intervienen en la resolución o petición de la tarea.
Pre-condiciones	Estar autenticado en el sistema y seleccionar una tarea.
Post-condiciones	Obtenemos un comentario en la tarea
Flujo principal	1.- Rellenar las opciones que aparecen en la página de creación de

	comentarios 2.-Pulsar el botón “guardar”
Flujo alternativo	Cancelar operación: Antes de pulsar el botón de guardar podemos abortar la creación del comentario, pulsando el botón de cancelar. Error en los datos: Si a la hora de modificar la tarea se produce algún error el sistema nos informará.

Descripción	Valor
Caso de uso	Agregar fichero.
Actores	Usuarios, Sistemas, Administrador.
Descripción	Permite agregar un fichero a una tarea para así poder explicar mejor la tarea que se quiere realizar o pedir.
Pre-condiciones	Estar autenticado en el sistema y seleccionar una tarea.
Post-condiciones	Obtenemos un fichero adjunto en la tarea
Flujo principal	1.- Seleccionar la ruta en la que se encuentra el archivo 2.- Pulsar el botón de “subir fichero”
Flujo alternativo	Cancelar operación: Antes de pulsar el botón de guardar podemos abortar la creación del comentario, pulsando el botón de cancelar. Error en los datos: Si a la hora de modificar la tarea se produce algún error el sistema nos informará.

Descripción	Valor
Caso de uso	Agendar tarea.
Actores	Sistemas.
Descripción	Permite agendar una tarea en el calendario personal del usuario de Sistemas.
Pre-condiciones	Estar autenticado en el sistema y seleccionar una tarea.
Post-condiciones	Tarea agendada.
Flujo principal	1.- Seleccionar la vista que mejor se ajuste a nuestras necesidades entre mensual, semanal y diaria. 2.- Arrastrar la tarea hasta la fecha seleccionada 2.1.- Si la vista es semanal o diaria podemos seleccionar la hora de comienzo o que sea para todo el día. 2.2.- Si la vista es mensual la tarea se agendará a las 8.00 AM
Flujo alternativo	Cancelar operación: Solo se puede cancelar antes de soltar la tarea. Error en los datos: Si a la hora de agendar sucede un error el sistema nos informará.

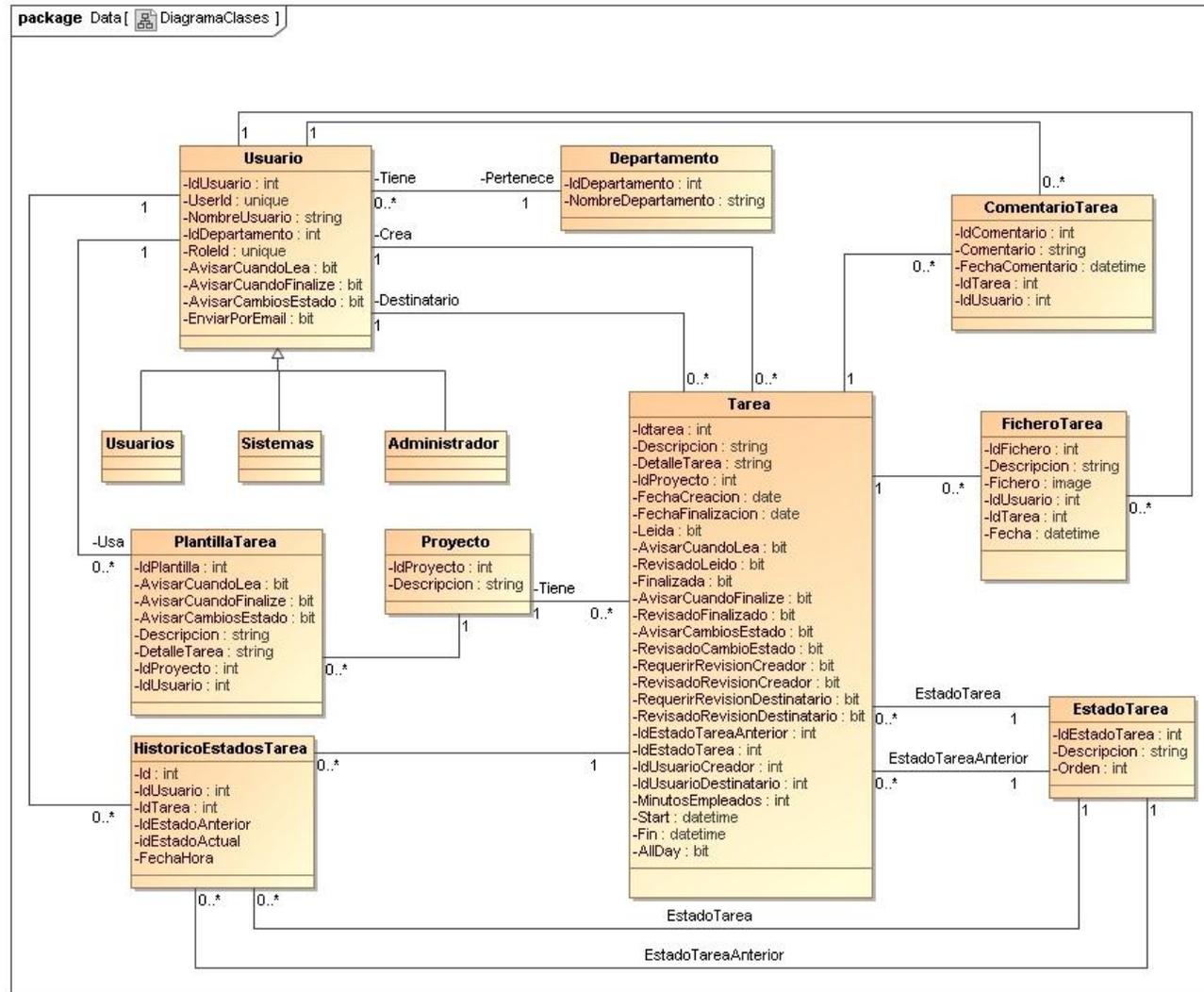
Descripción	Valor
Caso de uso	Crear usuario.
Actores	Administrador.
Descripción	Permite la creación de usuarios dentro de la aplicación.
Pre-condiciones	Estar autenticado y que no exista el usuario que queremos crear.
Post-condiciones	Tenemos un usuario mas creado en la aplicación.
Flujo principal	1.- Dirigirnos a la pestaña de usuarios

	<p>2.- Pulsar sobre el icono de crear usuarios nuevos</p> <p>3.- Rellenar los datos del usuario</p> <p>4.- Pulsar el botón de “registrarse”</p>
Flujo alternativo	<p>Cancelar operación: Antes de pulsar el botón de “registrarse” podemos abortar la creación del usuario, pulsando cualquier enlace.</p> <p>Error en los datos: Si a la hora de crear el usuario se produce algún error el sistema nos informará.</p>

Descripción	Valor
Caso de uso	Autenticarse en la aplicación.
Actores	Usuarios, Sistemas, Administrador.
Descripción	Permite entrar a la aplicación con el usuario y la contraseña personal.
Pre-condiciones	No estar autenticado en la aplicación.
Post-condiciones	Estar autenticado en la aplicación.
Flujo principal	<p>1.- Entrar en la aplicación, al no estar autenticado seremos dirigidos automáticamente a la pagina de registro.</p> <p>2.-Introducir los datos de usuario y contraseña</p>
Flujo alternativo	Error en los datos: Si la contraseña y el usuario no coinciden veremos el mensaje de error.

Descripción	Valor
Caso de uso	Búsqueda de tareas.
Actores	Usuarios, Sistemas, Administrador.
Descripción	<p>Permite la búsqueda de tareas, dependiendo del tipo de usuario:</p> <ul style="list-style-type: none"> - Usuarios: permite la búsqueda de tareas que el ha creado. - Resto: permite la búsqueda de todas las tareas.
Pre-condiciones	Estar autenticado en el sistema.
Post-condiciones	Resultado de tareas en base a los filtros que ha seleccionado el usuario.
Flujo principal	<p>1.- Pulsar en el enlace que pone “Buscar tarea”.</p> <p>2.- Rellenar las opciones en base de los criterios que quiere en la búsqueda.</p> <p>3.- Pulsar el botón “Buscar”</p> <p>4.- Navegar por los resultados</p>
Flujo alternativo	Cancelar la búsqueda: Podemos cancelar la búsqueda pulsando en cualquier enlace.

6.3.- Diagrama de clases



Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

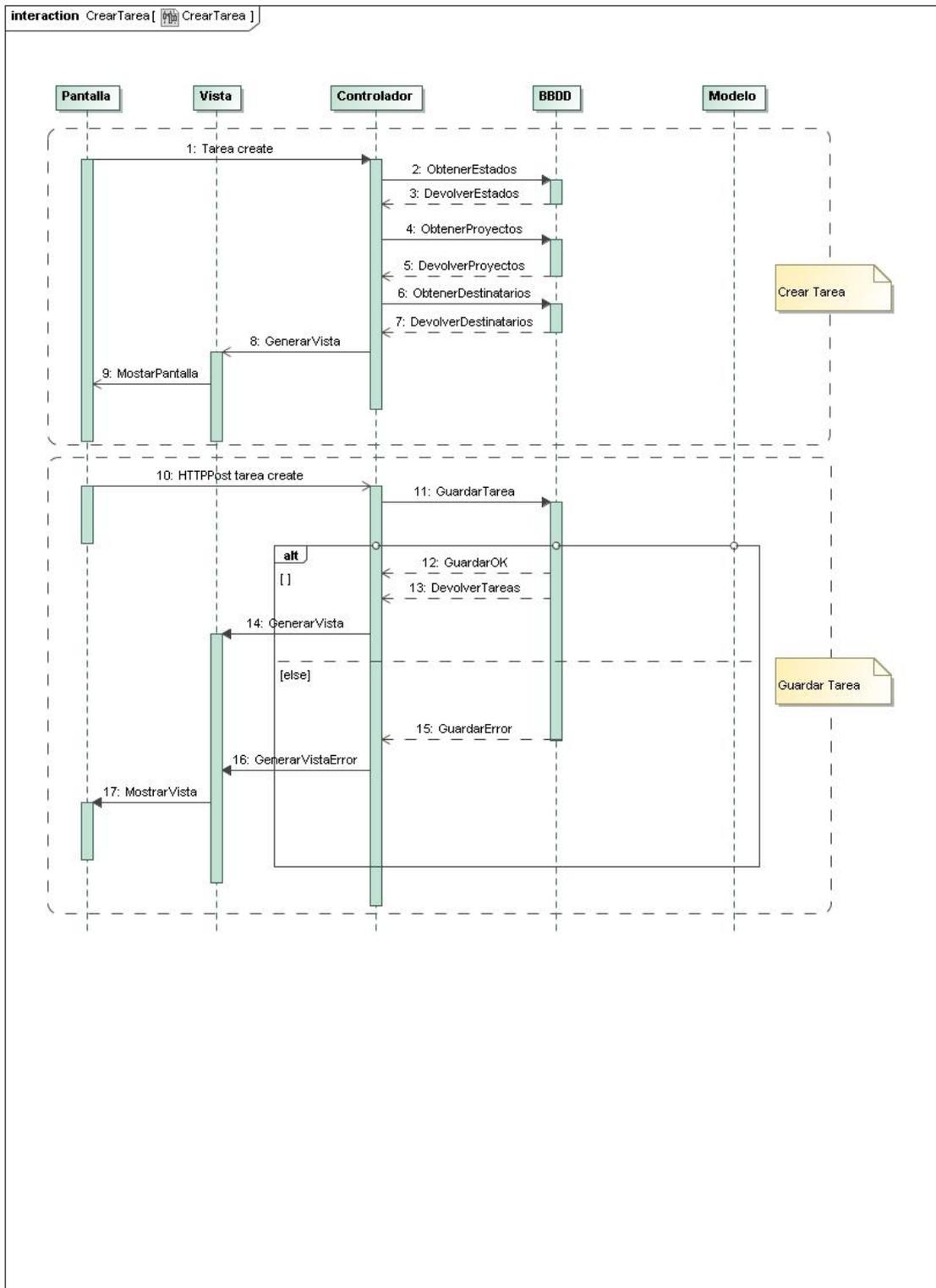
En nuestra aplicación la más importante es la clase tarea, vemos que esta clase se relaciona con casi todas las otras clases por lo que vamos a pasar a explicar su relación con:

- ComentarioTarea: Esta clase contiene los comentarios que se realizan a las tareas, una tarea puede tener cero o muchos comentarios, mientras que un comentario pertenece solo a una tarea
- FicheroTarea: Esta clase funciona igual que la anterior, pero sirve para guardar los ficheros que vamos subiendo a cada tarea
- Proyecto: Esta clase contiene los distintos proyectos que tenemos creados en la aplicación. Cada tarea solo puede pertenecer a un proyecto, y un proyecto puede tener cero o muchas tareas
- EstadoTarea: Esta clase contiene los estados en los que puede estar la tarea, tanto el estado actual como el estado anterior. Ambos campos de la tarea tienen obligatoriamente un estado, y cada estado puede estar en cero o muchas tareas.
- HistoricoTareas: Esta clase contiene todos los estados en los que ha ido estando la tarea, por lo que una tarea puede tener muchos estados históricos y un estado histórico solo puede pertenecer a una tarea.
- Usuario: Esta clase contiene los usuarios registrados en la aplicación, una tarea tiene un usuario creador y un usuario destinatario, y los usuarios pueden tener muchas tareas, tanto como creadores como destinatarios.

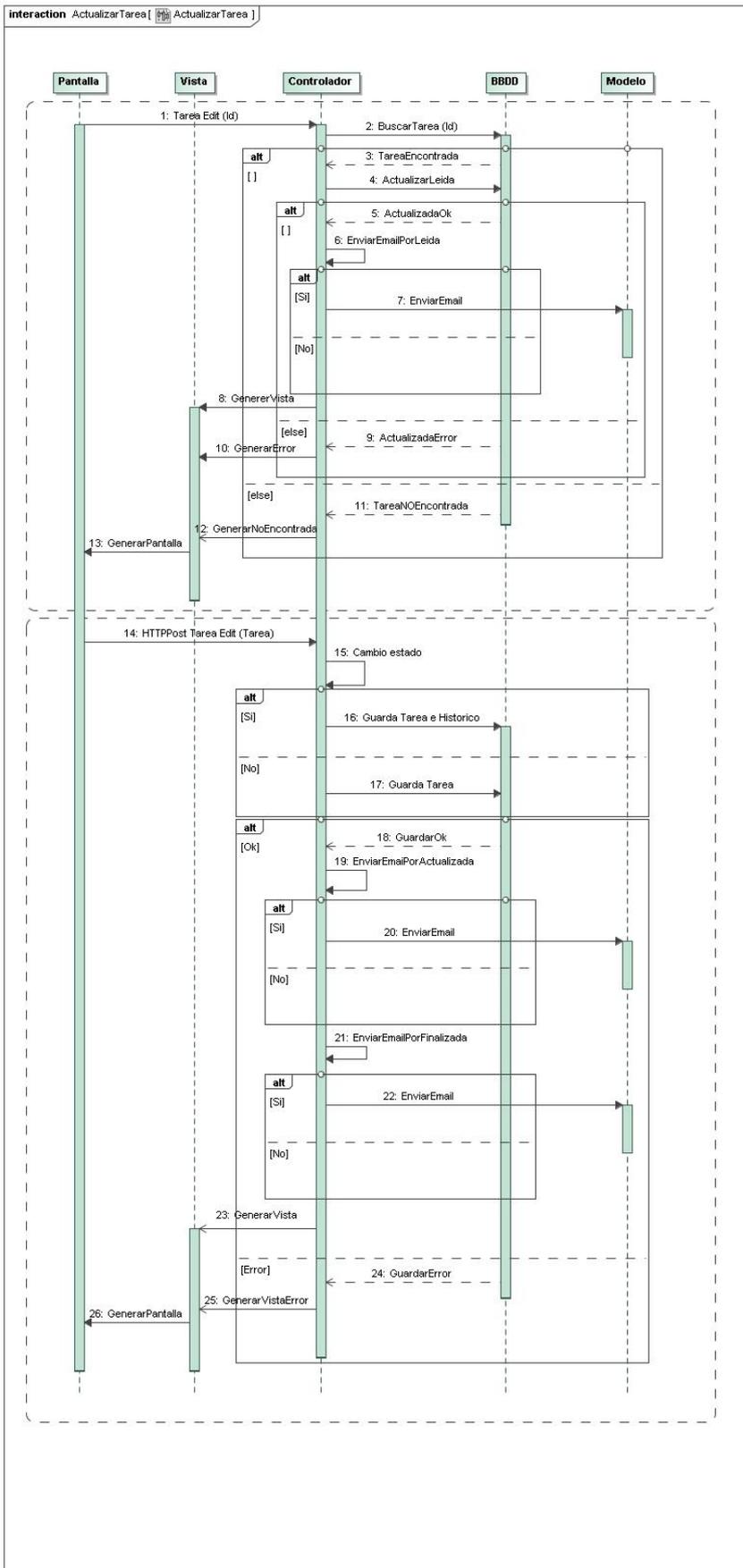
Otra clase muy importante es la Usuario, esta se relaciona con ComentarioTarea, FicheroTarea, PlantillaTarea e HistoricoEstadosTarea de la misma forma: un usuario puede tener muchas de estas clases y estas clases solo pueden tener un usuario. Sin embargo con la clase Departamento funciona a la inversa, es esta la que puede tener muchos usuarios, y un usuario debe pertenecer solo a un departamento.

6.4.- Diagrama de secuencia

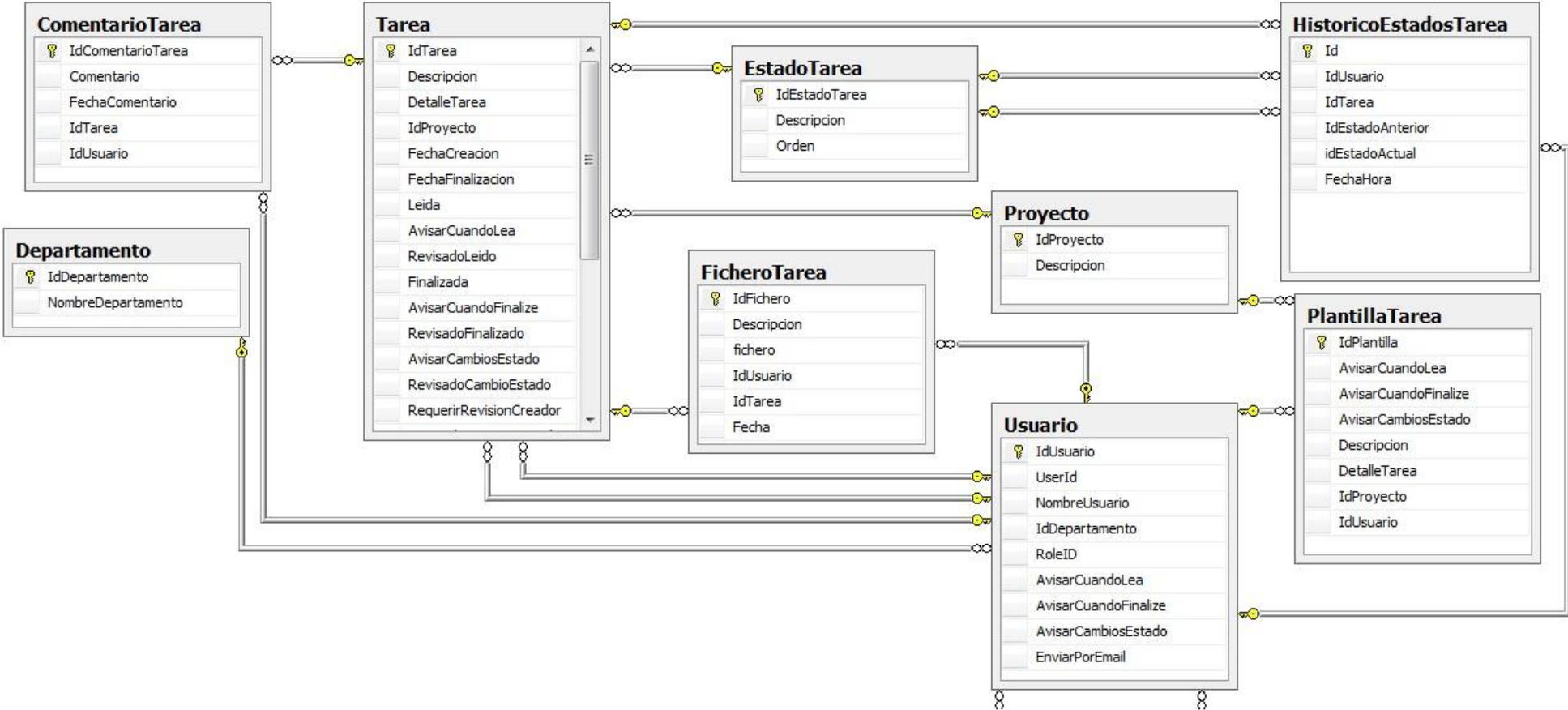
6.4.1 Diagrama de creación de la tarea

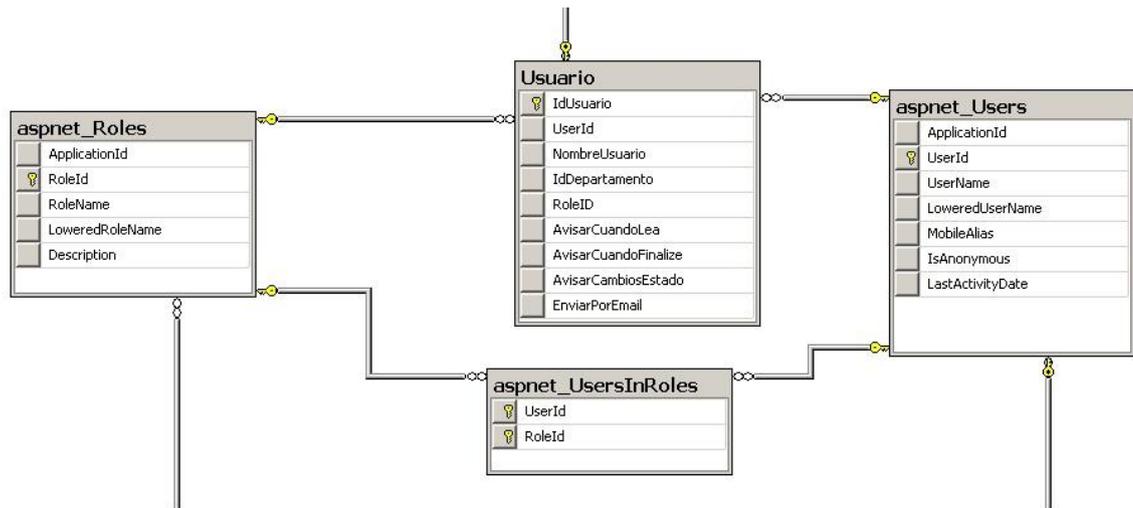


6.4.2 Diagrama de actualización de una tarea



6.5 Diagrama de base de datos





Se han incluido las tablas de ASPNET ya que para la validación de los usuarios utilizamos el servicio de membresía que nos ofrece este.

6.6.- Descripción de las tablas

ComentarioTarea: Tabla en la que almacenamos los comentarios que los distintos usuarios realizan sobre una tarea.

- IdComentarioTarea (PK): identificador del comentario.
- Comentario: donde almacenamos la información del comentario.
- FechaComentario: fecha en la que se realizó el comentario.
- Idtarea (FK): identificador de la tarea de la que se ha realizado el comentario.
- IdUsuario (FK): identificador del usuario que realizó el comentario.

Departamento: Tabla en la que se almacenan los departamentos a los que puede pertenecer cada usuario.

- IdDepartamento (PK): identificador del departamento.
- NombreDepartamento: nombre que le hemos dado a cada departamento.

EstadoTarea: Tabla en la que guardamos los distintos estados por los que puede pasar una tarea desde que se crea hasta que se resuelve.

- IdEstadoTarea (PK): identificador que guarda los distintos estados creados.
- Descripción: Texto que le damos para distinguir los estados
- Orden: valor que le damos a los estados para ordenar temporalmente los estados entre los que puede estar la tarea, y que van desde uno hasta el número de estados distintos, siendo el uno el primer estado que puede tener la tarea.

FicheroTarea: Tabla en la que se almacenan los ficheros que los usuarios adjuntan para explicar algún concepto de la tarea que han demandado

- IdFichero (PK): identifica cada fichero que hemos adjuntado.

- Descripción: texto que adjuntamos al fichero para mejorar la comprensión del fichero.
- IdUsuario (FK): Identifica al usuario que ha subido el fichero a la aplicación.
- Idtarea (FK): identifica la tarea a la que hemos asociado el fichero.
- Fecha: Guarda la fecha en la que se adjuntó el fichero a la tarea.

HistoricoEstadosTarea: Esta tabla es la encargada de guardar las relaciones entre los distintos estados por los que ha pasado una tarea con el usuario y la fecha que lo ha realizado. Por lo que se pueden obtener históricos que nos muestra la evolución en la resolución de la tarea.

- Id (PK): identificador único del histórico.
- IdUsuario (FK): identificador del usuario que modificó la tarea.
- Idtarea (FK): identificador de la tarea que hemos modificado.
- IdEstadoAnterior(FK): estado inicial de la tarea antes de modificarla.
- IdEstadoActual (FK): estado de la tarea después de modificarla.
- FechaHora: nos informa de cuando se realizó el cambio de estados.

PlantillaTarea: En esta tabla guardamos las plantillas que se generan en la aplicación, a partir de las cuales se generarán las tareas.

- IdPlantilla (PK): Guarda el identificador de la plantilla.
- AvisarCuandoLea: Si queremos que nos avise cuando se lea la tarea.
- AvisarCuandoFinalize: Si queremos que nos avise cuando se finalice la tarea.
- AvisarCambiosEstado: Si queremos que nos avise cuando se cambie el estado.
- Descripción: Texto que le ponemos a la plantilla para saber el tema de la tarea.
- DetalleTarea: Texto que le ponemos para explicar con mas detalle de que va a ir la tarea.
- IdProyecto (FK): Identificador del proyecto al que va asociado la tarea cuando se use la plantilla.
- IdUsuario (FK): Identificador del usuario al que va a ir dirigida la tarea.

Proyecto: Tabla en la que almacenamos los proyectos que hay disponibles para usar en las tareas.

- IdProyecto (PK): Identificador único del proyecto
- Descripción: Descripción que aparece cuando seleccionamos un proyecto.

Tarea: Esta es la tabla mas importante de la aplicación ya que todo se basa en las tareas, por lo tanto esta es la tabla en la que se van a guardar las tareas que vamos generando.

- IdTarea (PK): Identificador de la tarea.
- Descripción: Da una breve descripción del tema de la tarea.
- DetalleTarea: Da una descripción mas detallada de la tarea.
- IdProyecto (FK): Identificador del proyecto al cual hemos asociado la tarea.
- FechaCreación: Fecha en la que se creó la tarea.
- Leida: Valor que nos dice si ya ha sido leida.
- AvisarCuandoLea: Si queremos que nos avise cuando se lea la tarea.

- RevisadoLeido: Valor que nos dice si se ha revisado la notificación de que se ha leído.
- AvisarCuandoFinalize: Si queremos que nos avise cuando se finalice la tarea.
- RevisadoFinalizado: Valor que nos dice si se ha revisado la notificación de que se ha finalizado.
- AvisarCambiosEstado: Si queremos que nos avise cuando se cambie el estado.
- RevisadoCambioEstado: Valor que nos dice si se ha revisado la notificación de que se ha cambiado de estado.
- RequerirRevisionCreador: Opción que se usa para que el que está resolviendo la tarea mande una notificación al creador de la tarea.
- RevisadoRevisionCreador: Valor que nos dice si se ha revisado la notificación creada por el punto anterior.
- RequerirRevisionDestinatario: Opción que se usa para que el creador notifique al destinatario de la tarea su atención sobre la tarea.
- RevisadoRevisionDestinatario: Valor que nos dice si se ha revisado la notificación que se había creado en el punto anterior.
- IdEstadoTareaAnterior (FK): identificador del estado de la tarea que tenemos antes de modificarla.
- IdEstadoTarea (FK): Identificador del estado actual del estado de la tarea.
- IdUsuarioCreador (FK): Identificador del usuario que ha creado la tarea.
- IdUsuarioDestinatario (FK): Identificador del usuario destinatario de la tarea.
- MinutosEmpleados: Tiempo total que se ha necesitado para terminar la tarea.
- Start: Cuando se agenda una tarea esta es la fecha inicial.
- Fin: Cuando se agenda una tarea esta es la fecha final.
- AllDay: Este valor es verdadero si una tarea está agendada todo el día, y es false si lo está en un intervalo de tiempo.

Usuario: Tabla que guarda los valores de los usuarios que usan la aplicación.

- IdUsuario (PK): Identificador del usuario por que vamos a relacionar con nuestras tablas.
- UserID (FK): Identificador de usuario que usa la membresía de ASP.NET, usado para unir nuestras tablas con las de la membresía.
- NombreUsuario: Nombre que le hemos dado al usuario dentro de la aplicación.
- IdDepartamento (FK): Identificador del departamento al que pertenece el usuario.
- RoleID (FK): Identificador que usamos para relacionar esta tabla con las de la membresía para identificar a un usuario dentro de los grupos:
 - o Administrador.
 - o Departamento Sistemas
 - o Usuario
- AvisarCuandoLea: Opción por defecto que vamos a usar cuando creamos una tarea, en este caso para que nos avisen cuando se lea la tarea.
- AvisarCuandoFinalice: Opción por defecto que vamos a usar cuando creamos una tarea, en este caso para que nos avise cuando se finalice.
- AvisarCambiosEstado: Opción por defecto que vamos a usar cuando creamos una tarea, en este caso para que nos avise cuando se cambie de estado.

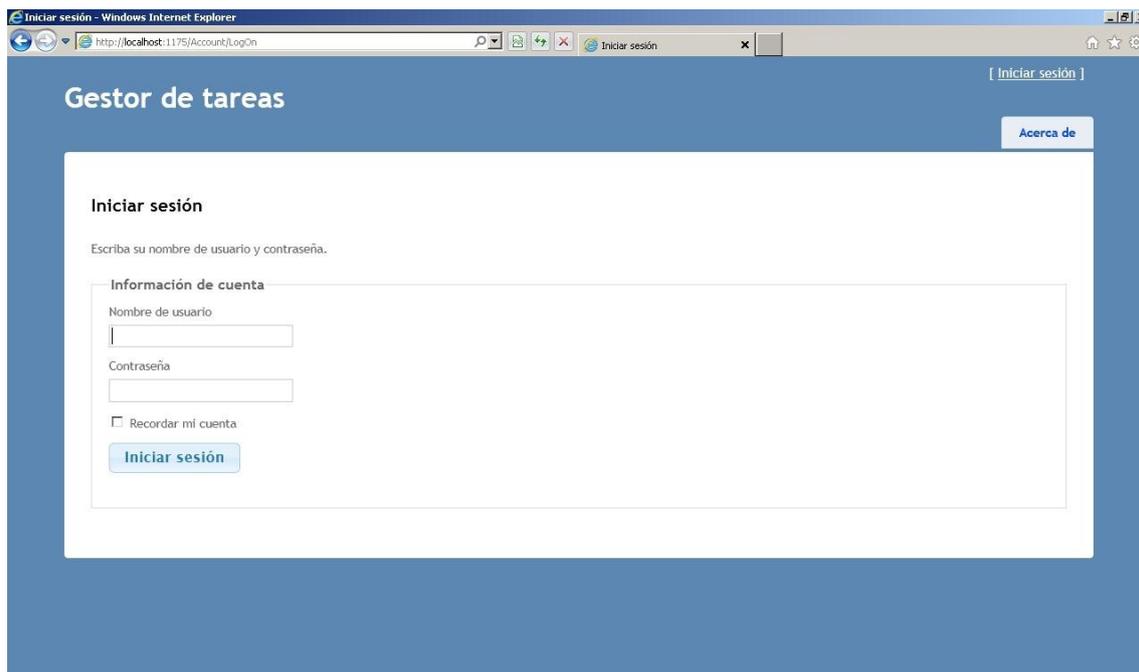
- EnviarPorEmail: Opción que usamos para decirle a la aplicación que además de que nos notifique en la aplicación lo haga por email.

7.- Diseño de la interfaz de usuario

7.1 Pantallas comunes a todos los roles

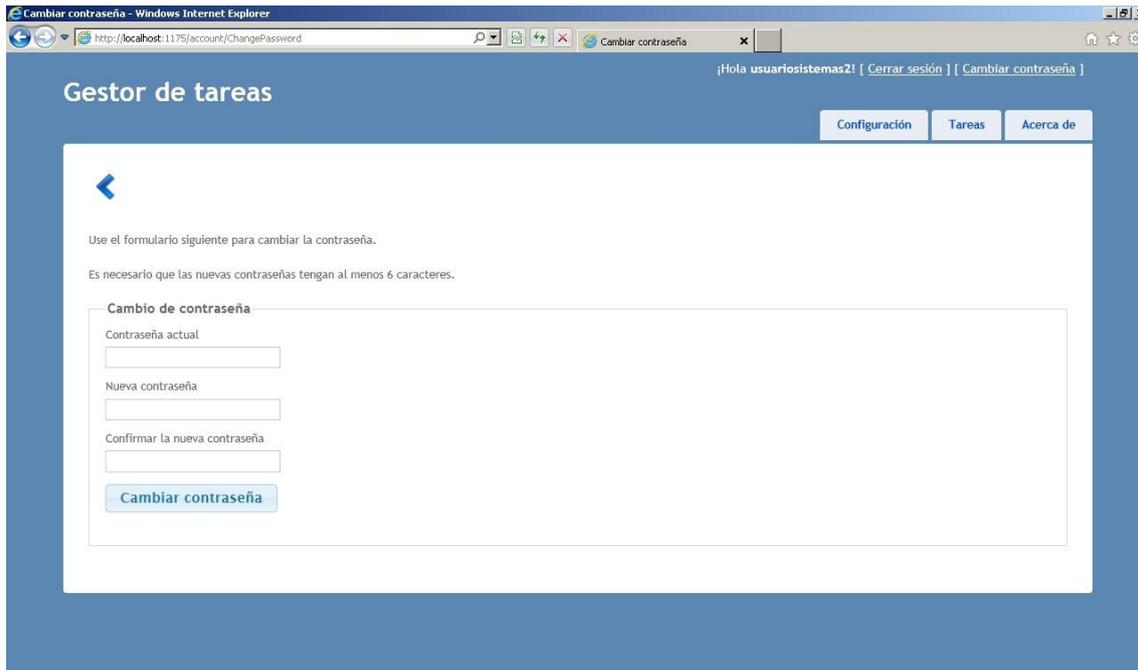
7.1.1 Pantalla de acceso a la aplicación

Para entrar al sistema, debemos de hacer uso del usuario y password que nos ha proporcionado el administrador de la aplicación. Para ello debemos dirigirnos a la aplicación y automáticamente nos aparece la siguiente ventana:



7.1.2 Pantalla de cambio de contraseña

Cuando un usuario se encuentra autenticado en la aplicación puede cambiar la contraseña en el momento que quiera, en la siguiente ventana:

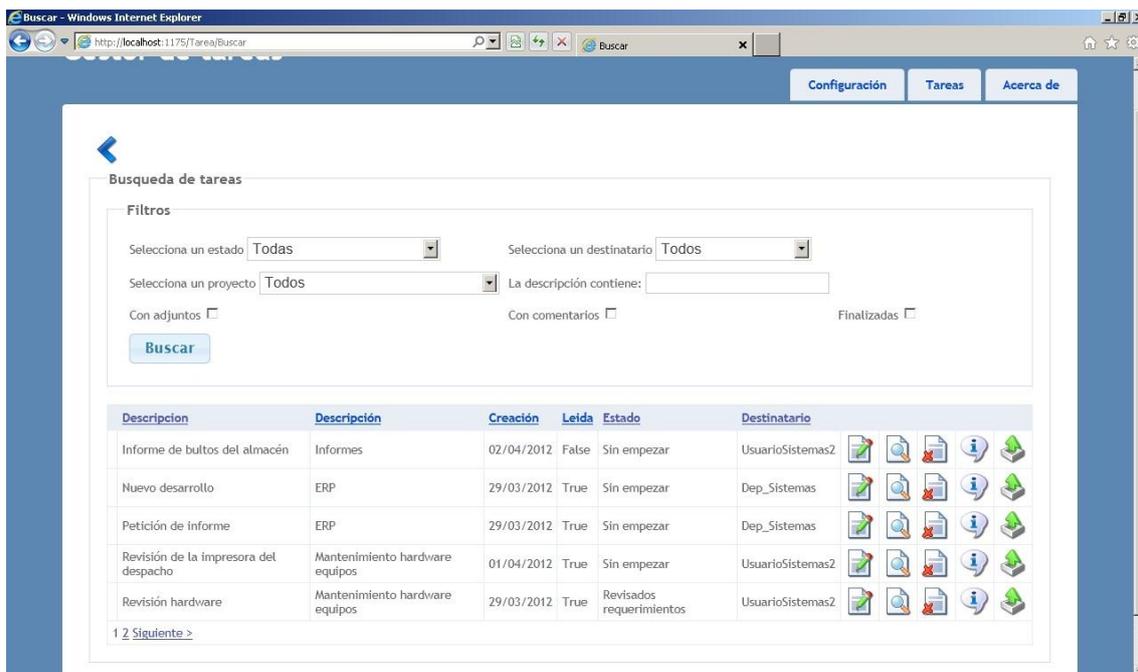


7.1.3 Pantalla de búsqueda de tareas

Cualquiera puede hacer una búsqueda de todas las tareas que hay guardadas en la aplicación, con algunas matizaciones:

- Los usuarios solo pueden buscar las aplicaciones que ellos han creado
- Los usuarios de Sistemas o administradores no tienen esta limitación.

Simplemente deben configurar los filtros que deseen en la siguiente pantalla:



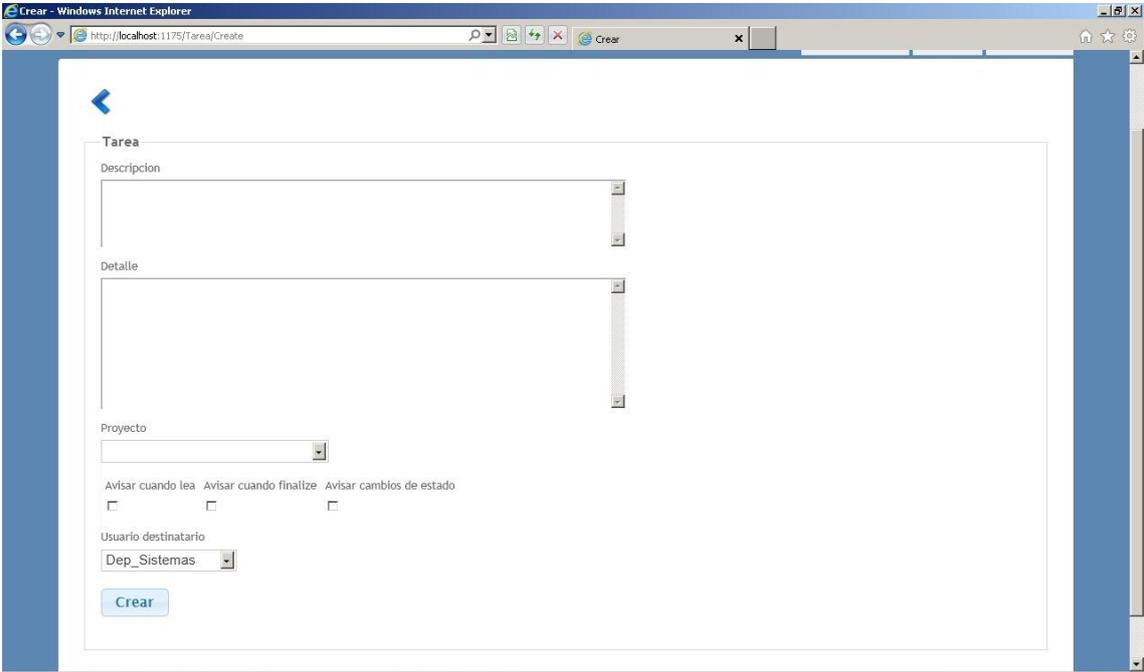
7.1.4 Pantalla de creación de tareas

Podemos crear tareas desde la página principal de la aplicación pulsando el botón de “Crear tarea”, si lo hacemos seremos dirigidos a una página en la que se puede introducir una descripción de lo que trata la tarea, luego podemos introducir un texto explicativo mayor en el detalle.

Podemos seleccionar el proyecto al que se adjuntará la tarea, y al usuario al que va dirigida. Si este último punto no lo sabemos, podemos dejar el usuario genérico de sistemas.

Ya por último seleccionaremos las opciones de notificación, que serán cogidas automáticamente de las que tenemos configuradas en nuestro usuario.

Vemos la imagen:

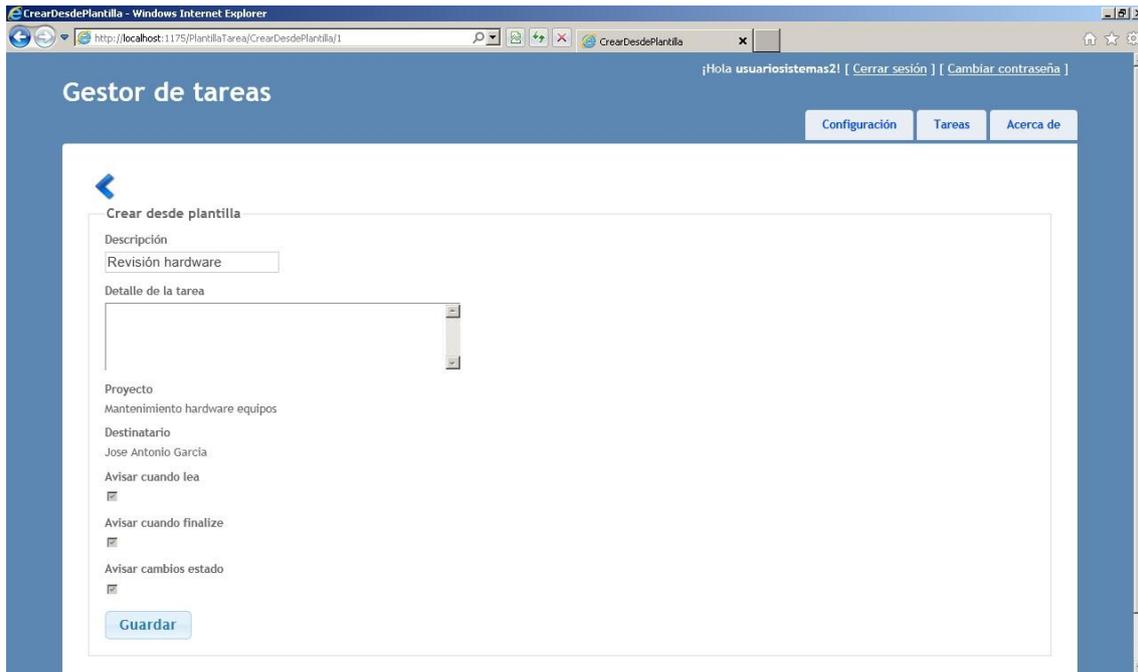


The screenshot shows a web browser window titled "Crear - Windows Internet Explorer" with the address bar showing "http://localhost:1175/Tarea/Create". The page content includes a blue navigation bar on the left with a back arrow. The main form area is titled "Tarea" and contains the following elements:

- A "Descripcion" text area.
- A "Detalle" text area.
- A "Proyecto" dropdown menu.
- Three notification checkboxes: "Avisar cuando lea", "Avisar cuando finalice", and "Avisar cambios de estado".
- A "Usuario destinatario" dropdown menu with "Dep_Sistemas" selected.
- A blue "Crear" button.

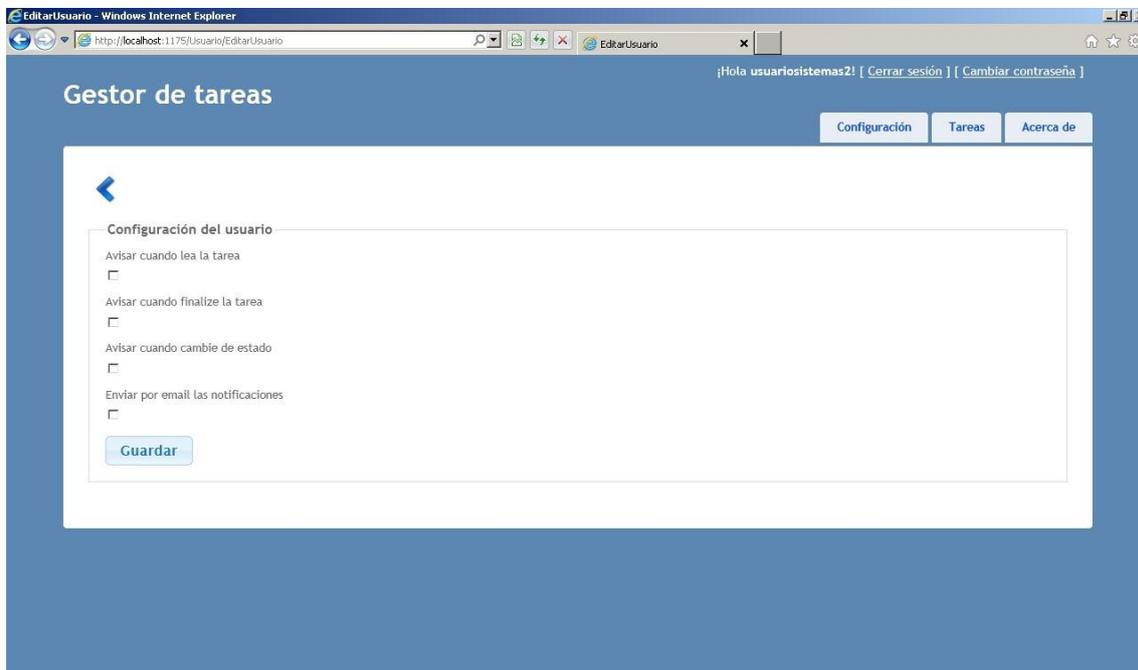
7.1.5 Pantalla de creación de tarea desde una plantilla

Desde la página principal de la aplicación podemos crear una tarea desde una plantilla configurada para hacer más rápida la creación de tareas redundantes o habituales. Para ello seleccionamos “Crear desde plantilla”. Y veremos una selección de las plantillas disponibles, simplemente debemos elegir una.



7.1.6 Pantalla de configuración del usuario

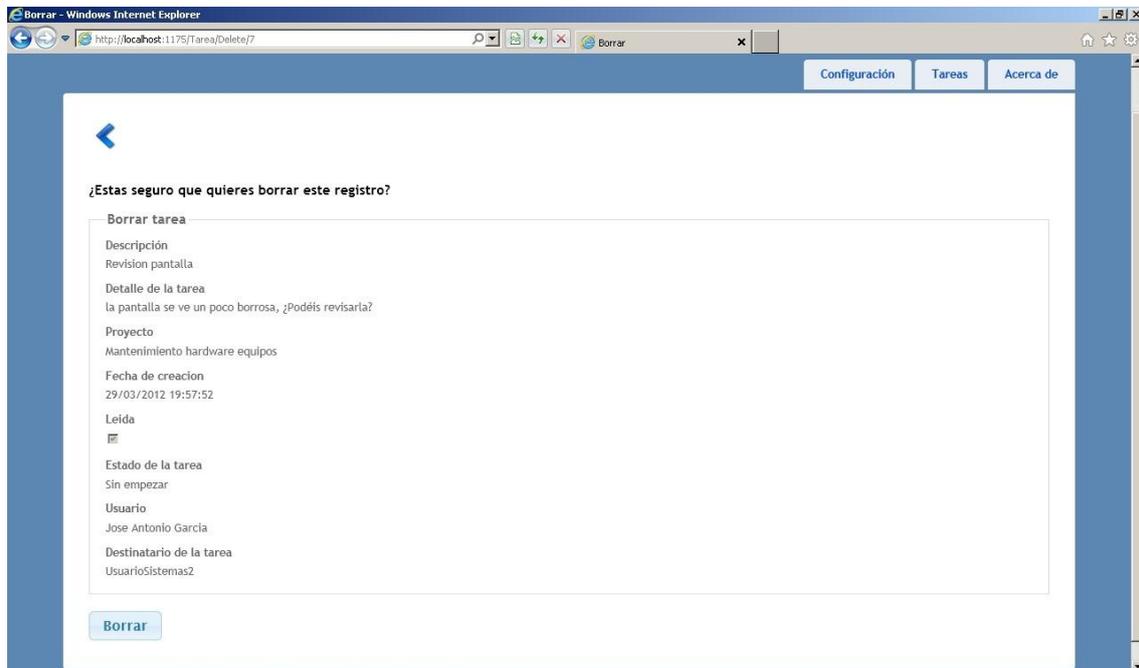
Desde esta pantalla podemos modificar las opciones por defecto que pasaremos a las tareas cuando las creamos, y además podremos activar la opción de recibir las notificaciones por email además de hacerlo en la misma aplicación. Vemos la imagen:



7.2 Pantallas comunes a los Usuarios y a Sistemas

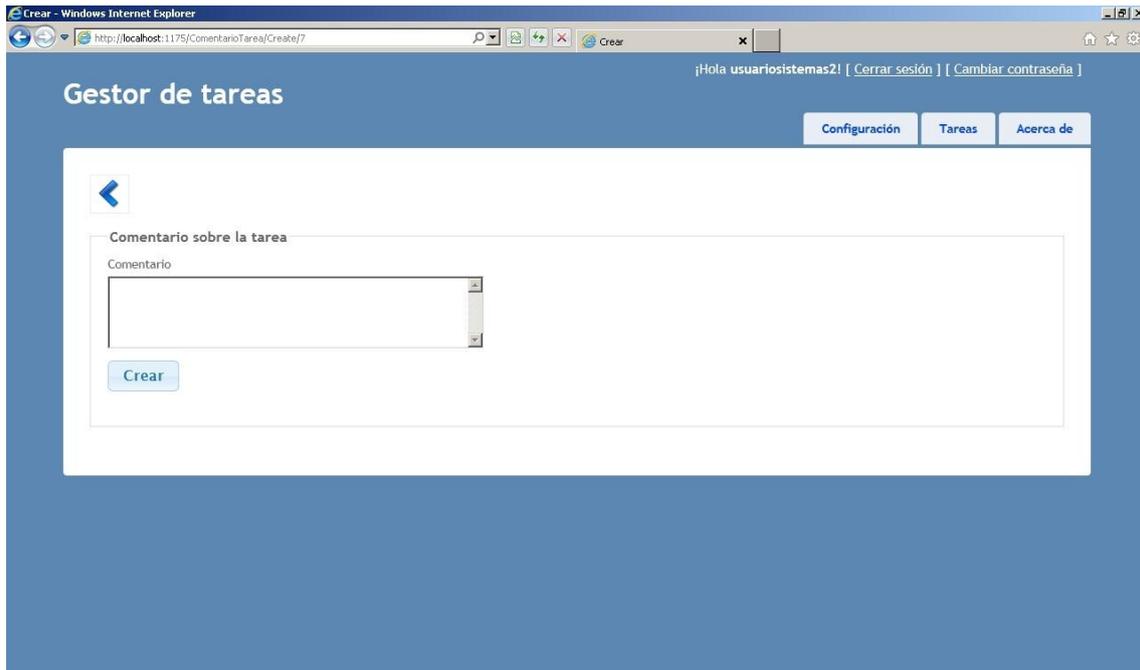
7.2.1 Pantalla de eliminación de una tarea

Siempre y cuando las tareas no estén finalizadas existe la opción de poder eliminarla. Para ello desde la ventana inicial, podemos pulsar sobre el icono  y nos dirigiremos a la página de la eliminación que es la siguiente:



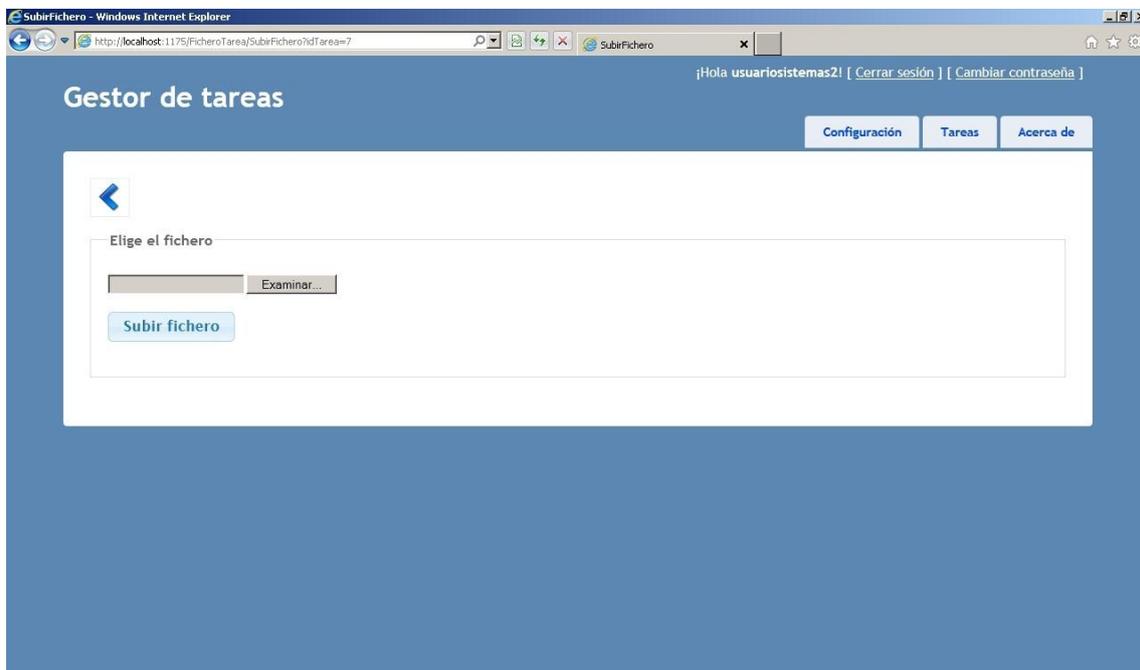
7.2.2 Pantalla de comentario a una tarea

Desde varios puntos de la visualización de una tarea se da opción de añadirle un comentario a una tarea. Para ello debemos pulsar sobre el icono  y se nos abrirá la siguiente ventana:



7.2.3 Pantalla de fichero a una tarea

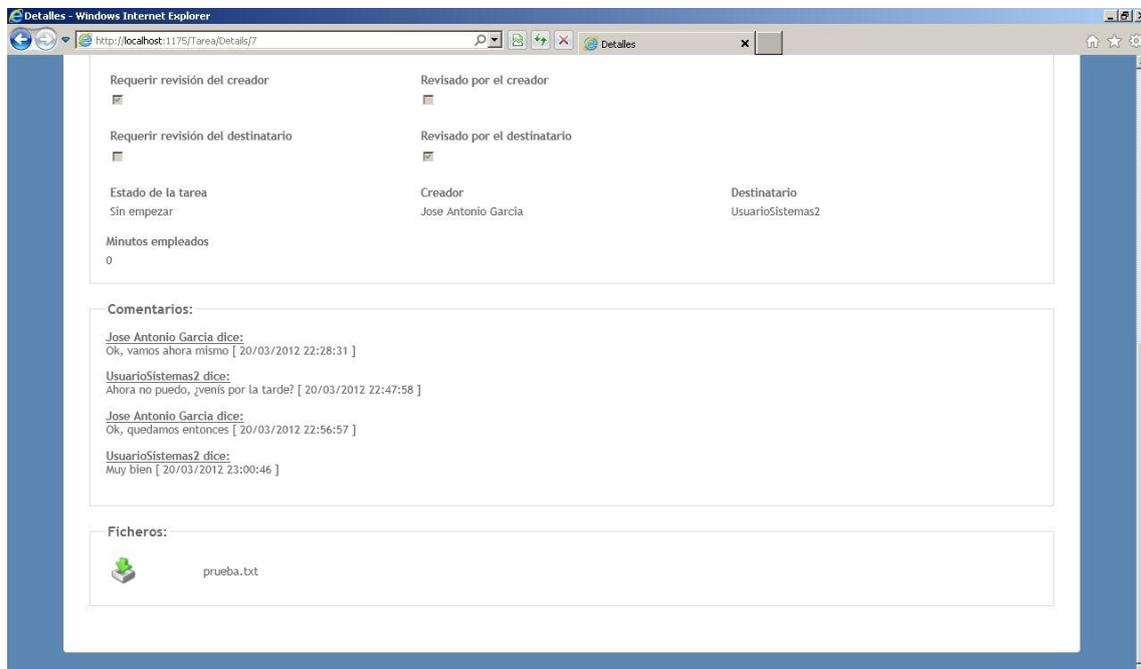
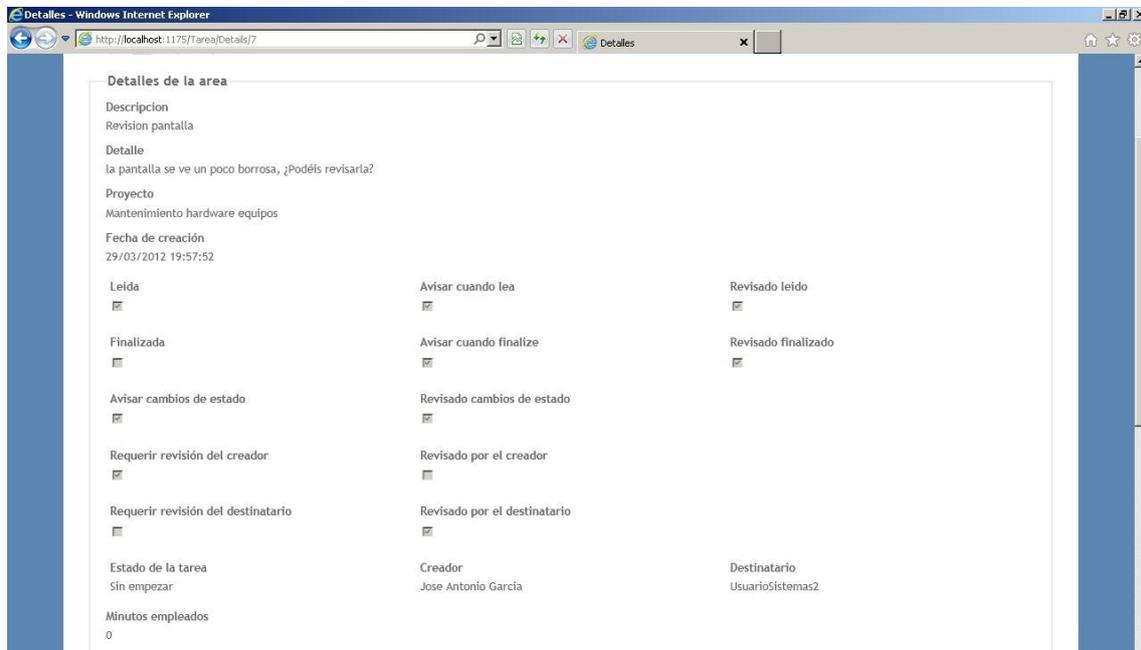
De la misma forma que podemos añadir un comentario a una tarea, podemos adjuntarle un fichero que contenga una captura de pantalla con un error, por ejemplo. Para ello debemos pulsar el botón con el siguiente icono , y eso no abrirá página en la que seleccionaremos el fichero a guardar, se ve en la siguiente imagen:



Hay que decir que el tamaño máximo permitido es de 4 mb.

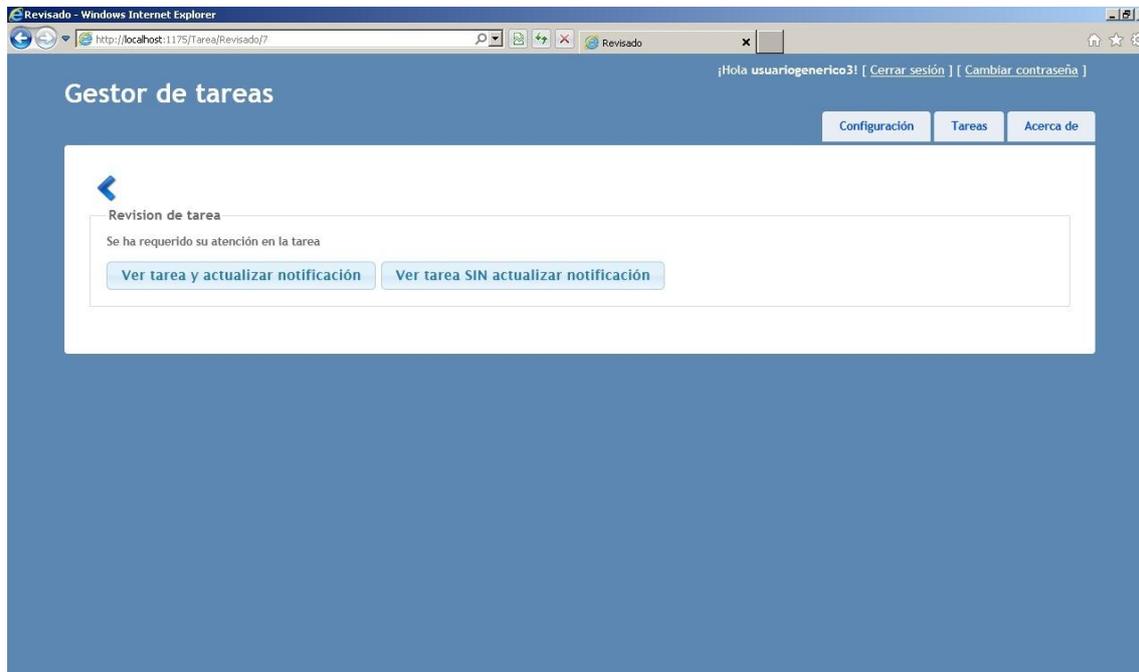
7.2.4 Pantalla de detalles de la tarea

Para ver los detalles de las tareas desde la página principal de la aplicación debemos pulsar sobre el icono , y eso nos lleva a la ventana donde se pueden ver todas las características de la tarea seleccionada así como los comentarios y los ficheros adjuntos, tal y como podemos ver en las siguientes imágenes:



7.2.5 Pantalla de notificaciones

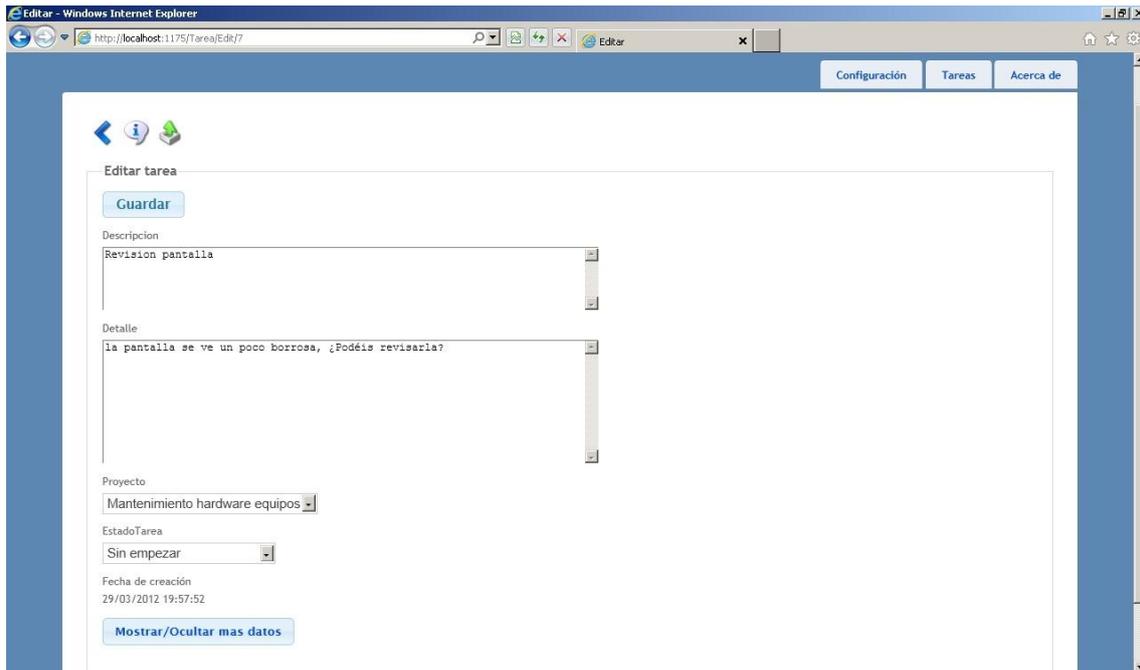
Si navegando por las tareas, encontramos una con el siguiente icono , nos está avisando de que esa tarea tiene por lo menos una notificación para nosotros. Si lo pulsamos nos dirigiremos a una ventana en la cual nos indicará el tipo de notificación que era, y desde aquí podremos navegar a la tarea, eliminando la notificación o manteniéndola. En concreto la página es la siguiente:



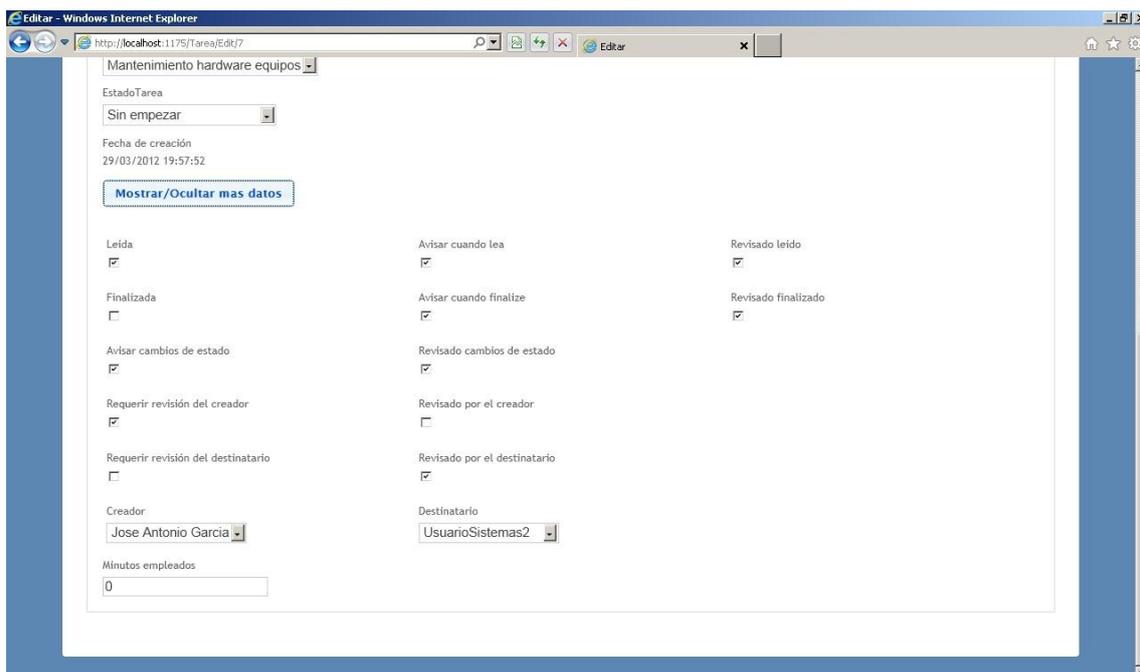
7.3 Pantallas de los usuarios de Sistemas

7.3.1 Pantalla de editar tarea

Los usuarios de sistemas son los encargados de realizar las tareas que han creado los usuarios genéricos, por lo tanto hay una pequeña diferencia entre estos dos grupos a la hora de editar la tarea, ya que estos pueden modificar más propiedades de la tarea. Pulsando sobre el icono siguiente  vemos la página de edición:

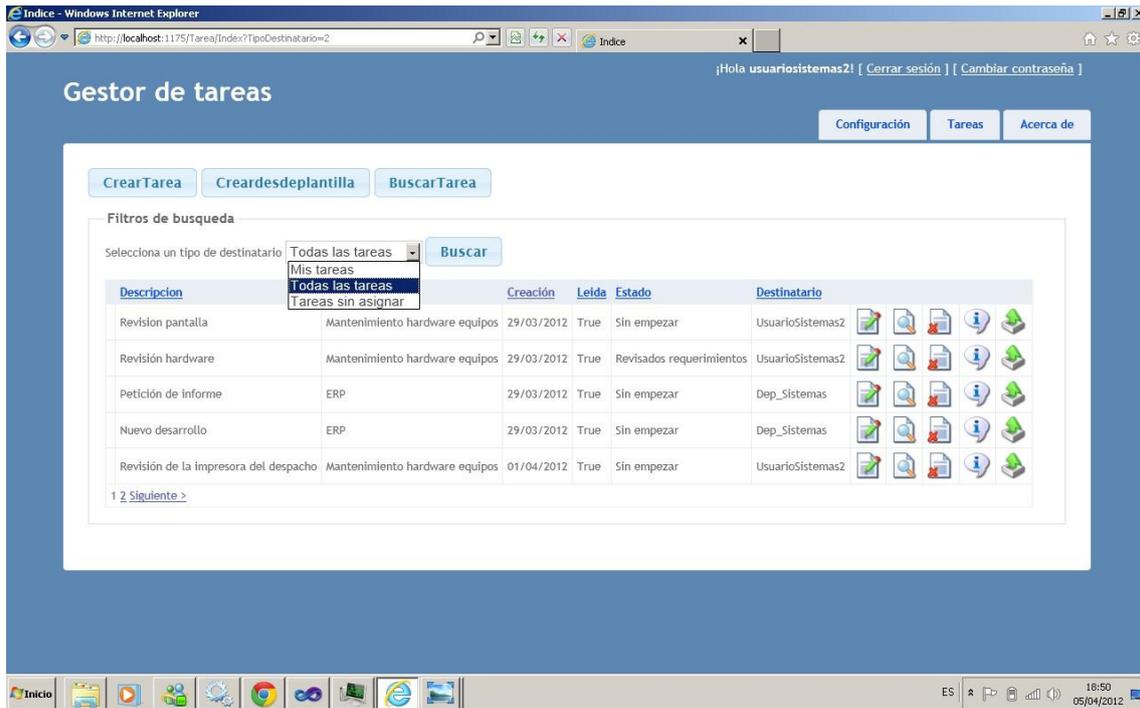


La vista predefinida oculta alguna información, que se puede mostrar si pulsamos en “Mostrar/ocultar más datos”, quedando de la siguiente manera:



7.3.2 Pantalla de inicio

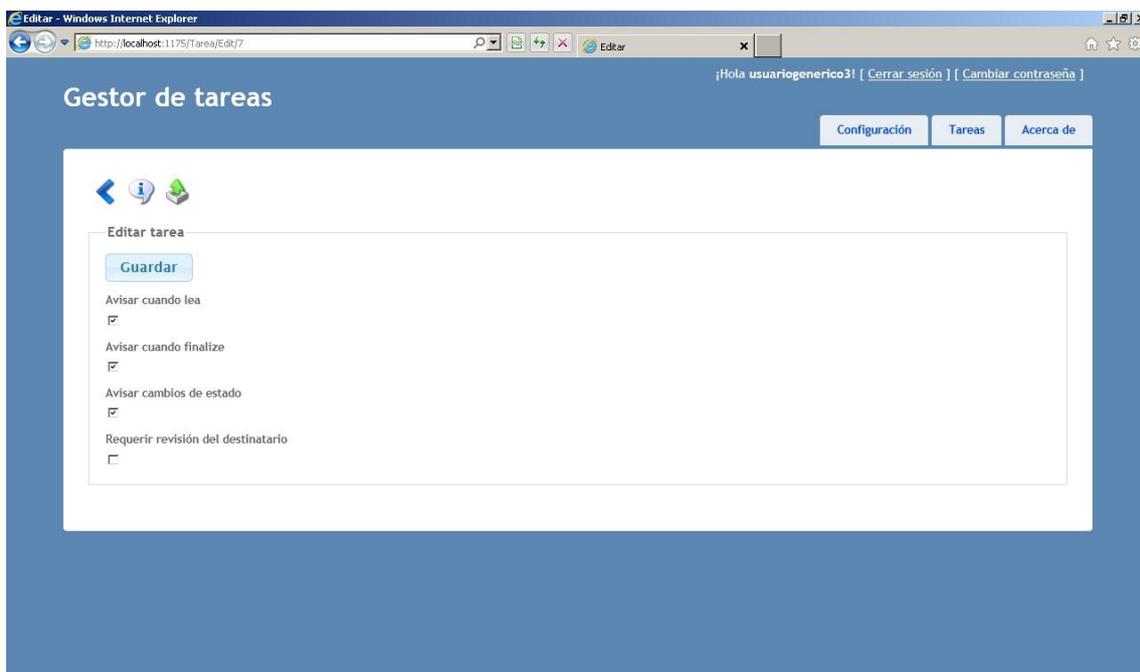
En la pantalla de inicio, por defecto, podemos ver las tareas de las cuales somos destinatarios, aunque usando el desplegable podemos ver, además de estas, las que no tienen usuario definido y todas las tareas abiertas en este momento.



7.4 Pantallas de los usuarios

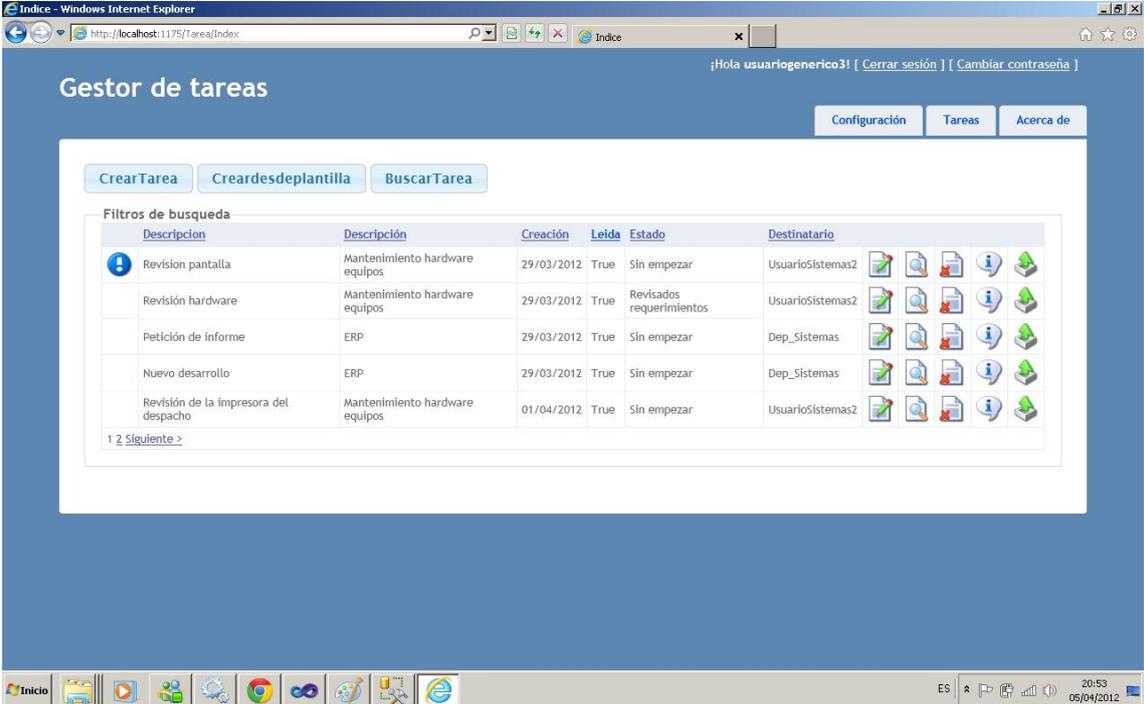
7.4.1 Pantalla de editar tarea

Los usuarios al disponer de menos permisos solo pueden modificar unas pocas características de la tarea tal y como vemos después de pulsar el icono  :



7.4.2 Pantalla de lista de tareas

En esta pantalla el usuario está viendo todas las tareas que están abiertas o con la notificación de terminada, si ya están en ese estado:



The screenshot shows a web browser window titled 'Indice - Windows Internet Explorer' with the URL 'http://localhost:1175/Tareas/Index'. The page header includes '¡Hola usuario generico3! [Cerrar sesión] [Cambiar contraseña]' and navigation buttons for 'Configuración', 'Tareas', and 'Acerca de'. Below the header are buttons for 'Crear Tarea', 'Crearedesdeplantilla', and 'BuscarTarea'. A 'Filtros de búsqueda' section is present above a table of tasks.

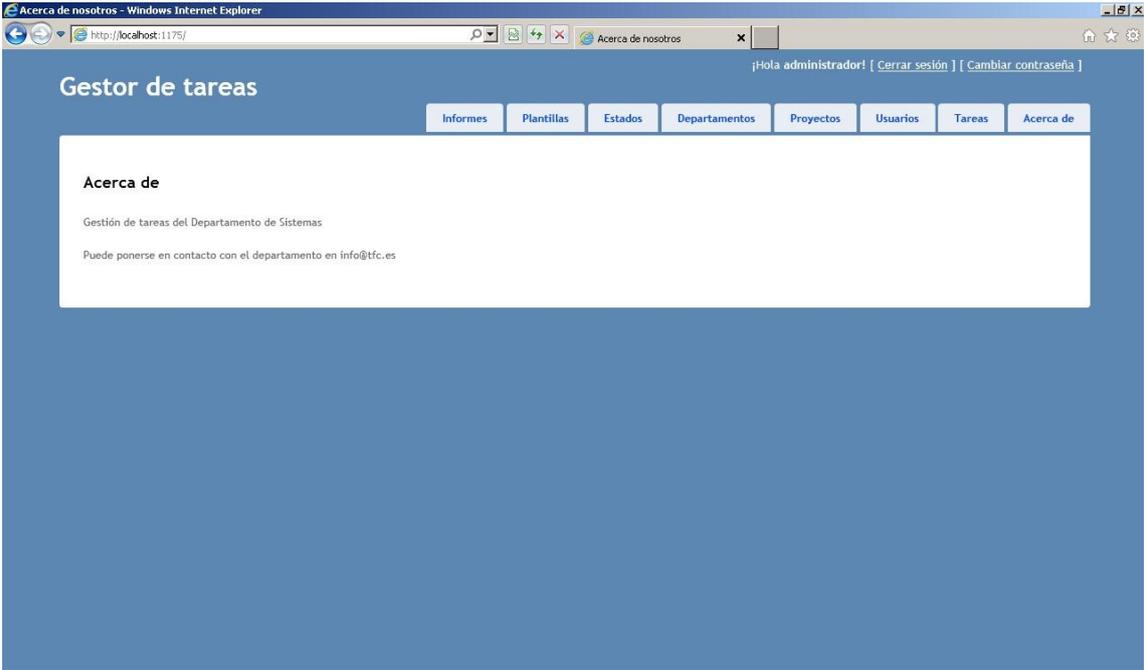
Descripción	Descripción	Creación	Leída	Estado	Destinatario
Revisión pantalla	Mantenimiento hardware equipos	29/03/2012	True	Sin empezar	UsuarioSistemas2
Revisión hardware	Mantenimiento hardware equipos	29/03/2012	True	Revisados requerimientos	UsuarioSistemas2
Petición de Informe	ERP	29/03/2012	True	Sin empezar	Dep_Sistemas
Nuevo desarrollo	ERP	29/03/2012	True	Sin empezar	Dep_Sistemas
Revisión de la impresora del despacho	Mantenimiento hardware equipos	01/04/2012	True	Sin empezar	UsuarioSistemas2

1 2 [Siguiente >](#)

7.5 Pantallas del Administrador

7.5.1 Pantalla de lista de plantillas

Desde esta página se pueden ver todas las plantillas que hay creadas en la aplicación.

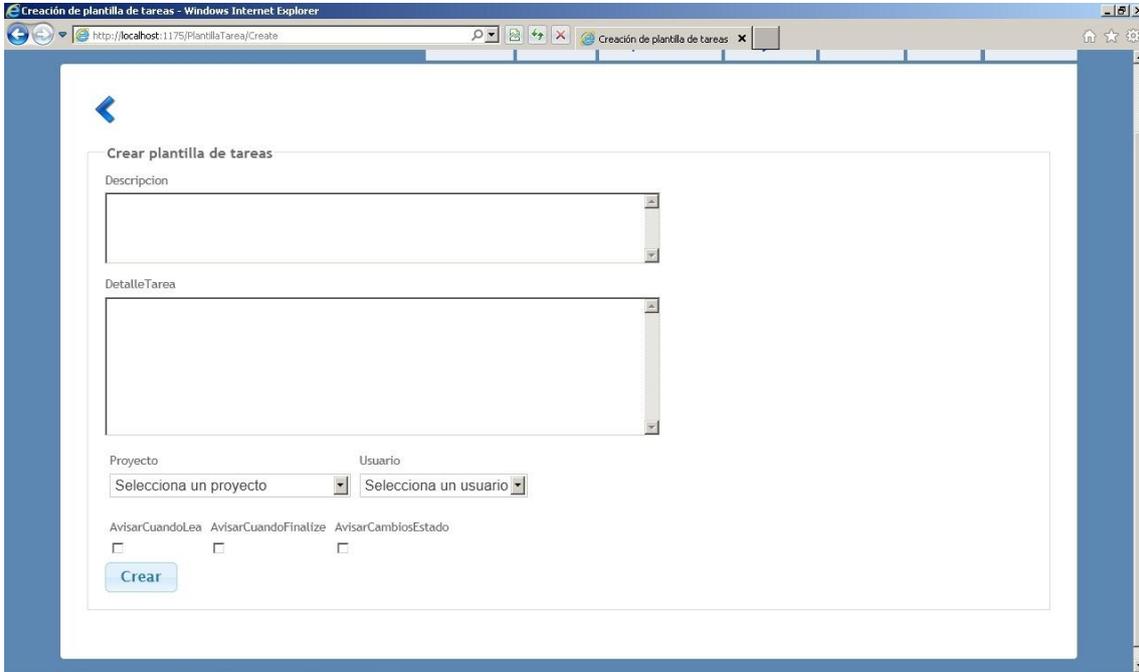


The screenshot shows a web browser window titled 'Acerca de nosotros - Windows Internet Explorer' with the URL 'http://localhost:1175/'. The page header includes '¡Hola administrador! [Cerrar sesión] [Cambiar contraseña]' and navigation buttons for 'Informes', 'Plantillas', 'Estados', 'Departamentos', 'Proyectos', 'Usuarios', 'Tareas', and 'Acerca de'. The main content area is titled 'Acerca de' and contains the following text:

Gestión de tareas del Departamento de Sistemas
Puede ponerse en contacto con el departamento en info@tfc.es

7.5.2 Pantalla de creación de plantilla

Desde la pantalla de lista de plantillas se puede crear una nueva plantilla pulsando sobre el icono , y vemos la siguiente pantalla:



Creación de plantilla de tareas - Windows Internet Explorer

http://localhost:1175/PlantillaTarea/Create

Creación de plantilla de tareas

Crear plantilla de tareas

Descripción

DetalleTarea

Proyecto: Selecciona un proyecto

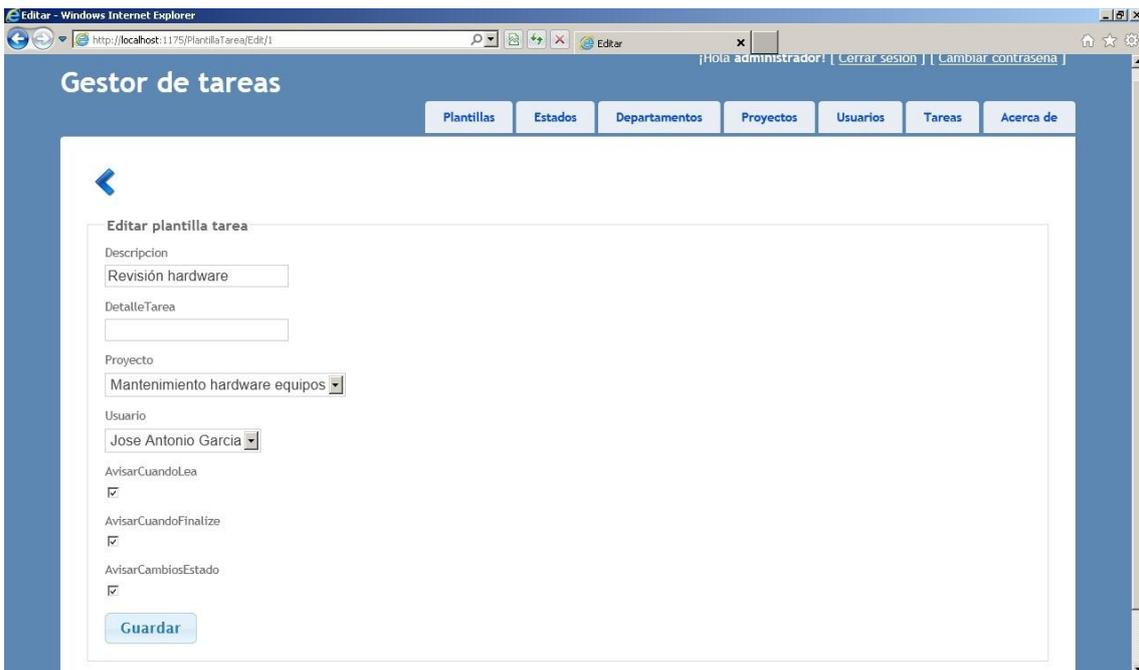
Usuario: Selecciona un usuario

AvisarCuandoLea AvisarCuandoFinalize AvisarCambiosEstado

Crear

7.5.3 Pantalla de edición de plantilla

Podemos editar la plantilla pulsando el icono , y vemos la plantilla en edición:



Editar - Windows Internet Explorer

http://localhost:1175/PlantillaTarea/Edt/1

Editar

¡Hola administrador! | Cerrar sesión | Cambiar contraseña

Gestor de tareas

Plantillas Estados Departamentos Proyectos Usuarios Tareas Acerca de

Editar plantilla tarea

Descripción: Revisión hardware

DetalleTarea

Proyecto: Mantenimiento hardware equipos

Usuario: Jose Antonio Garcia

AvisarCuandoLea

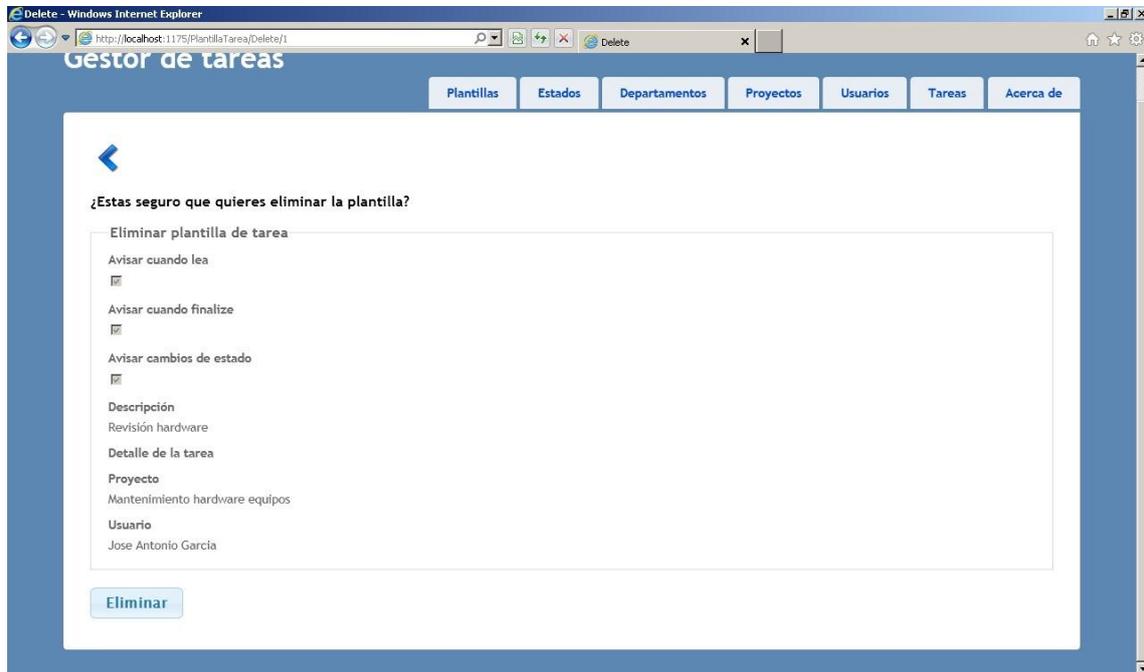
AvisarCuandoFinalize

AvisarCambiosEstado

Guardar

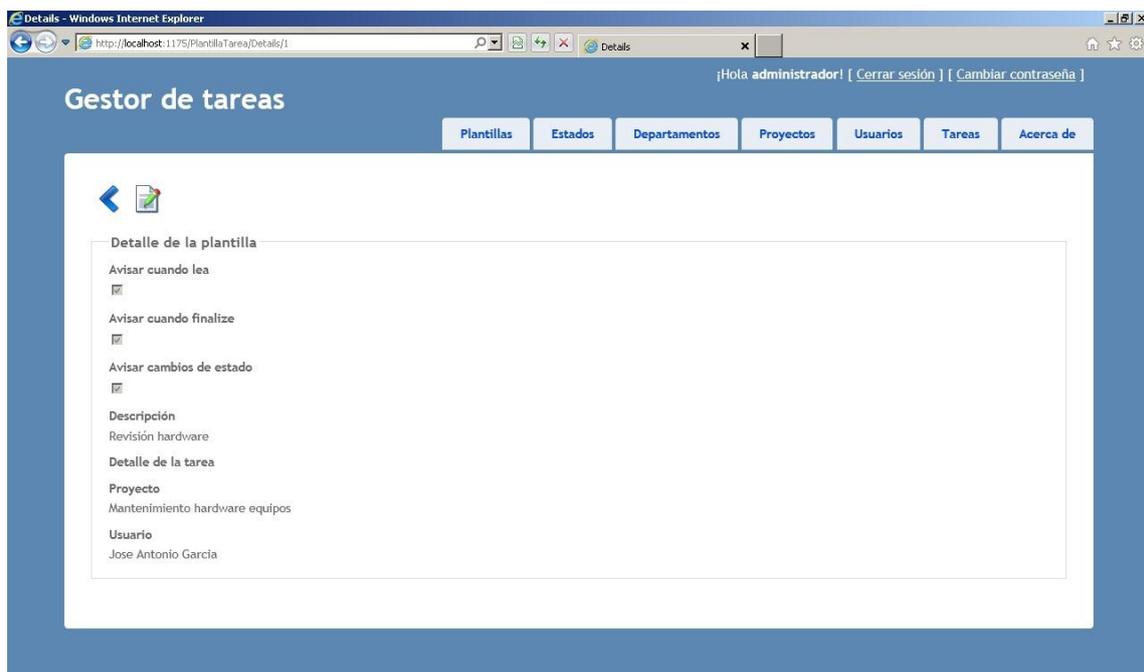
7.5.4 Pantalla de eliminación de plantilla

Podemos eliminar una plantilla desde el icono , con lo que nos lleva a la confirmación de la eliminación:



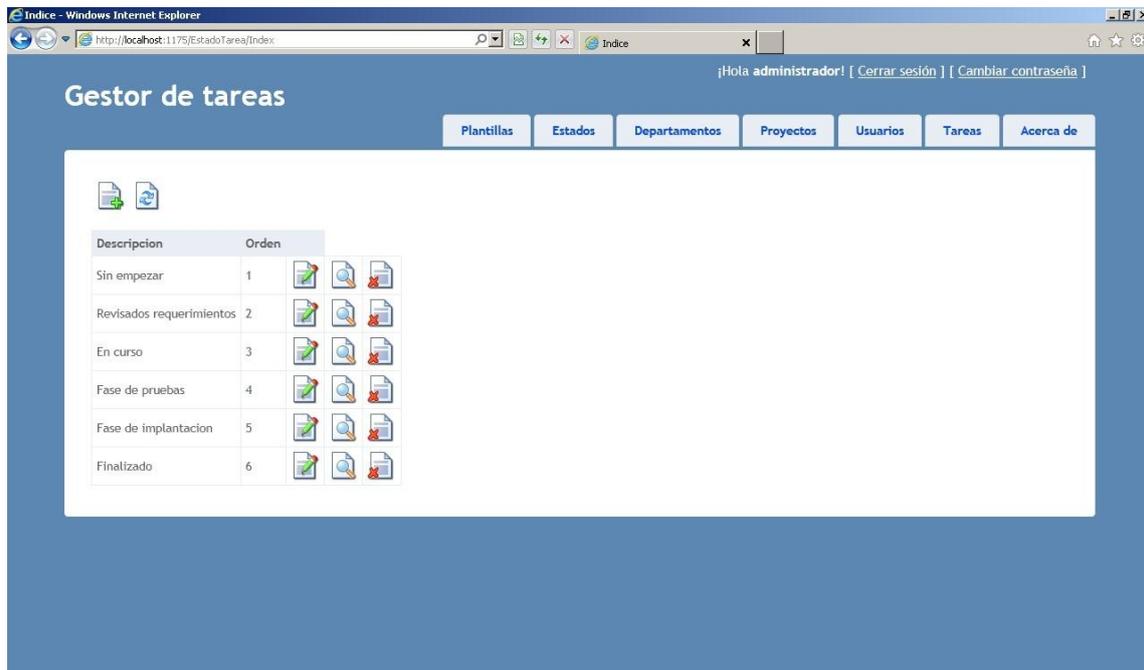
7.5.5 Pantalla de detalles de la plantilla

Podemos ver los detalles de la plantilla desde el icono , como vemos:



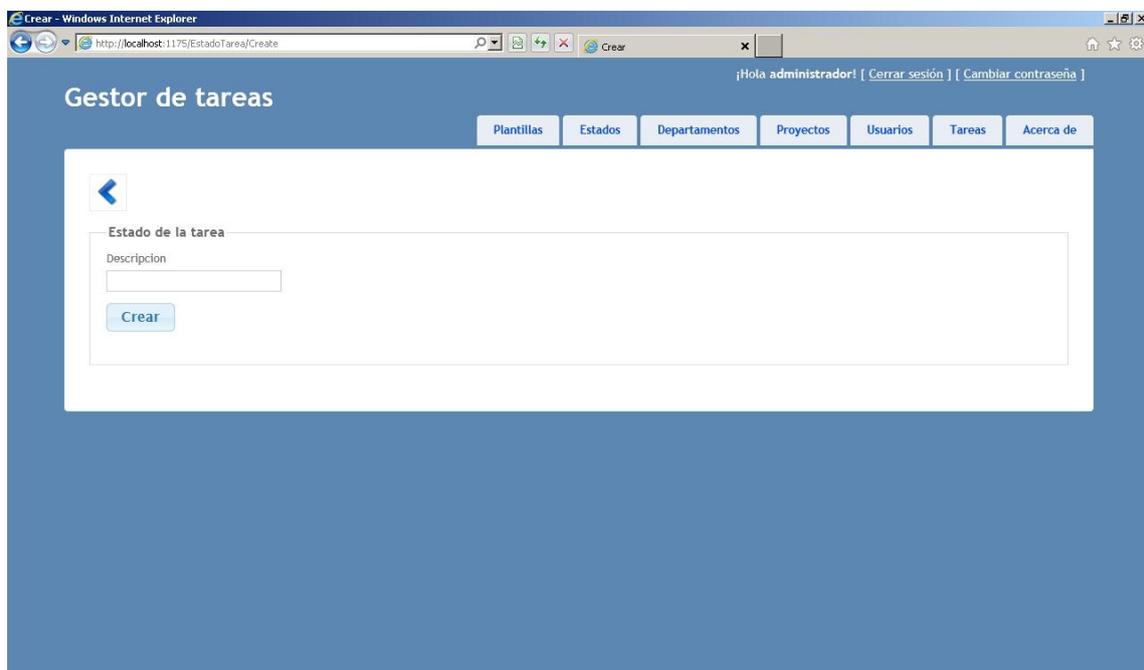
7.5.6 Pantalla de lista de estados

Desde esta pantalla se pueden ver los estados disponibles por los que puede pasar la tarea.



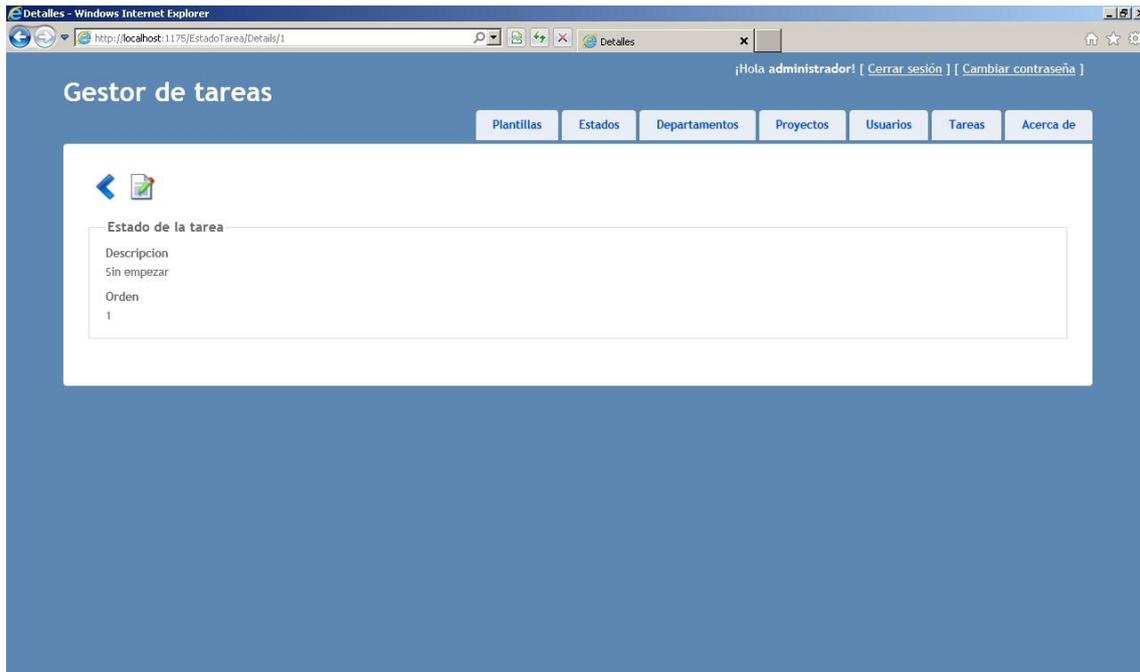
7.5.7 Pantalla de creación de estados

Pulsando el icono  podemos crear un estado nuevo. Lo vemos en la imagen:



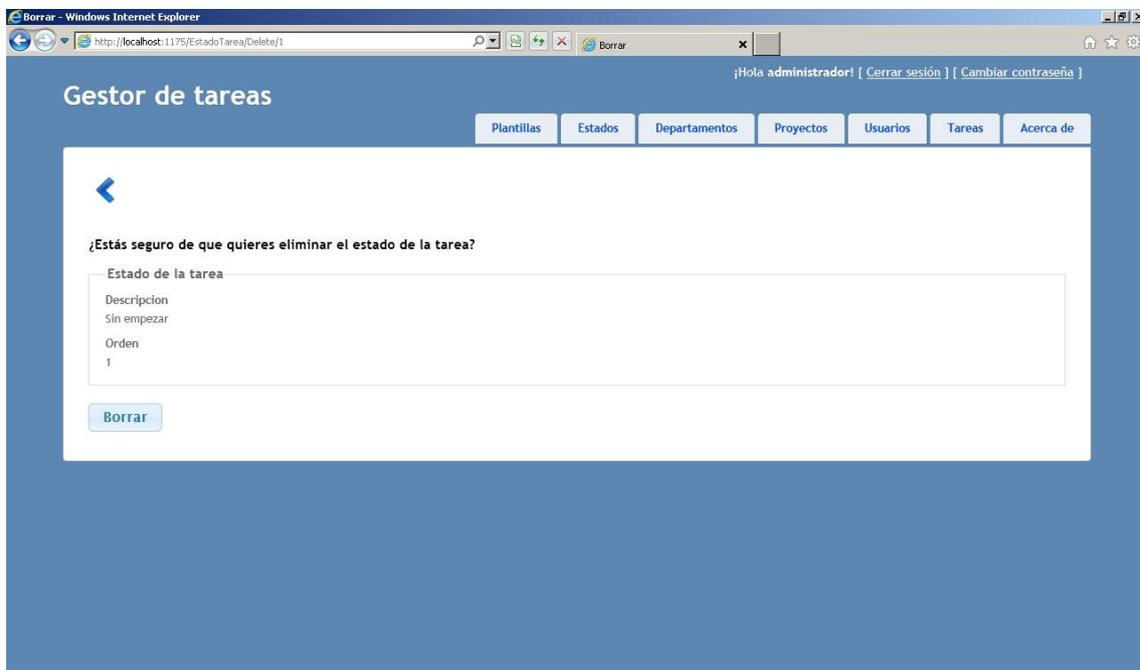
7.5.8 Pantalla de edición de estados

Pulsando el icono  editamos la tarea, como vemos en la imagen:



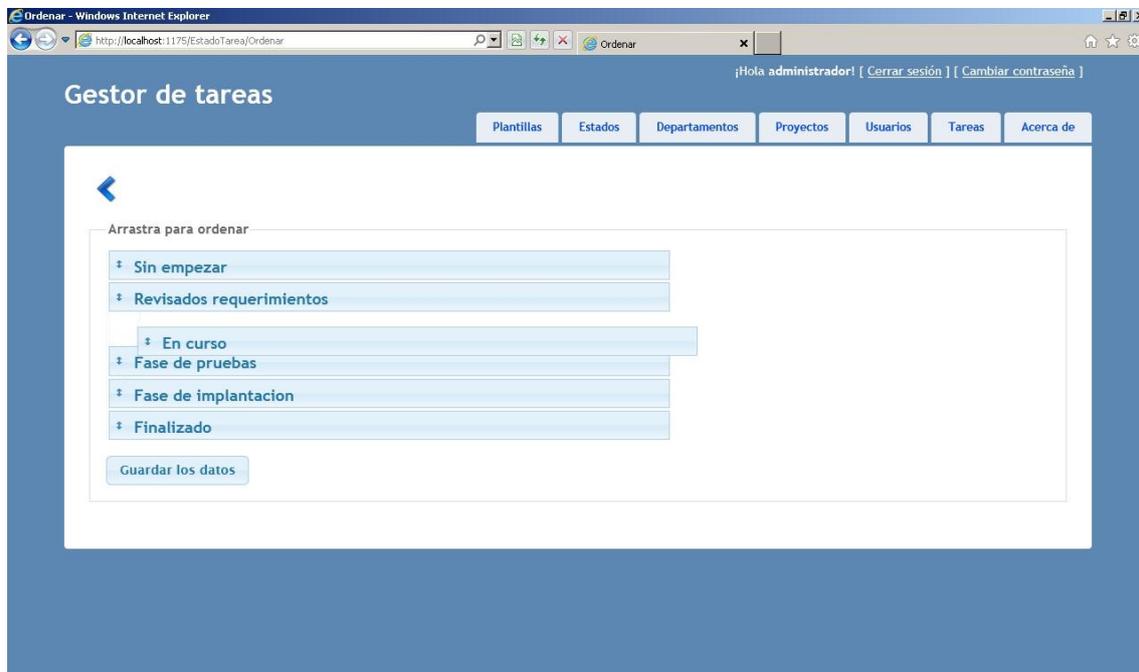
7.5.9 Pantalla de eliminación de estados

Pulsando el icono  podemos eliminar un estado de la tarea. Solo tenemos que confirmar en la siguiente imagen:



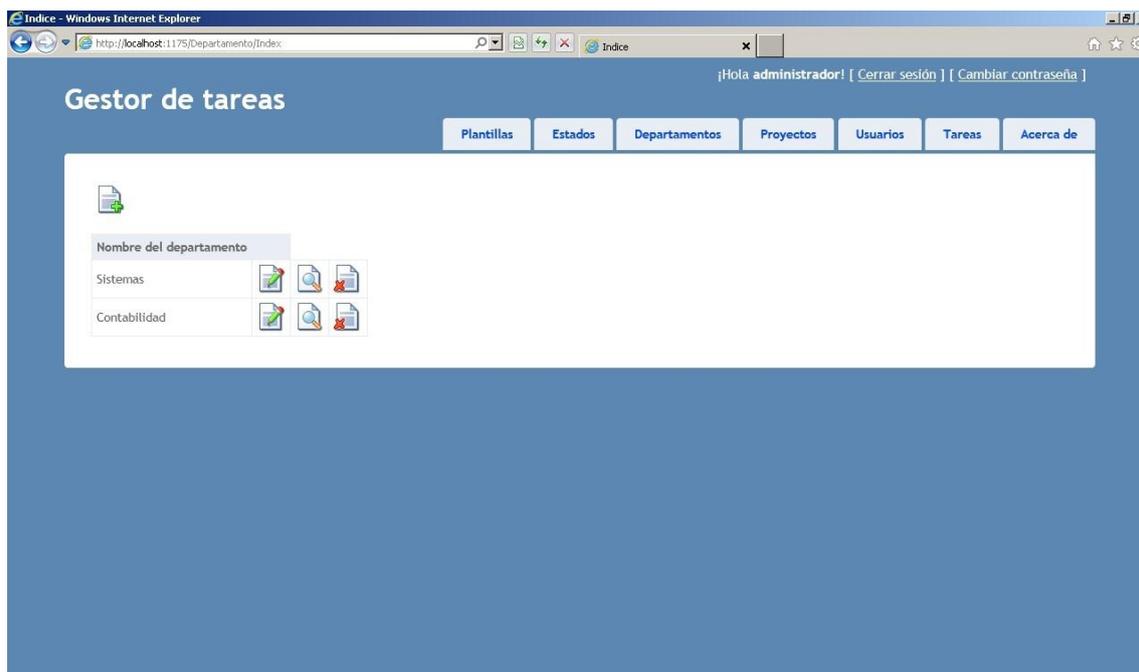
7.5.10 Pantalla de ordenación de estados

Como cuando creamos estados la ordenación viene marcada por la orden de creación, desde esta pantalla se puede modificar esa ordenación. Para hacerlo, solo se necesita arrastrar el estado a la posición deseada en la siguiente ventana:



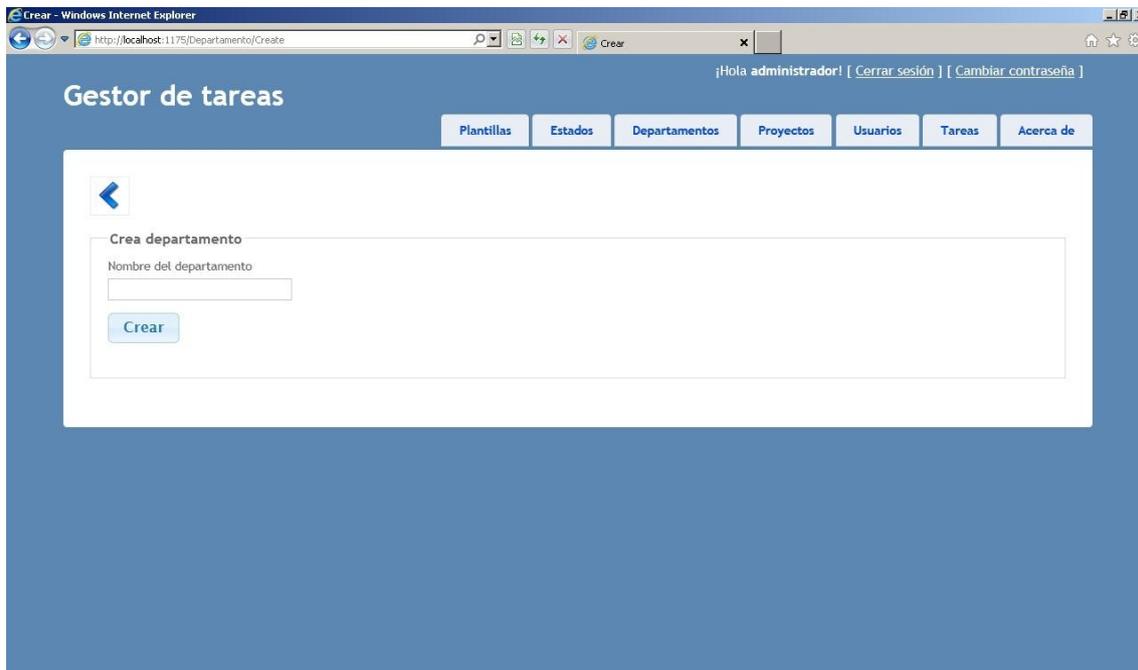
7.5.11 Pantalla de listado de departamentos

Desde esta pantalla podemos ver los departamentos que tenemos creados en la aplicación:



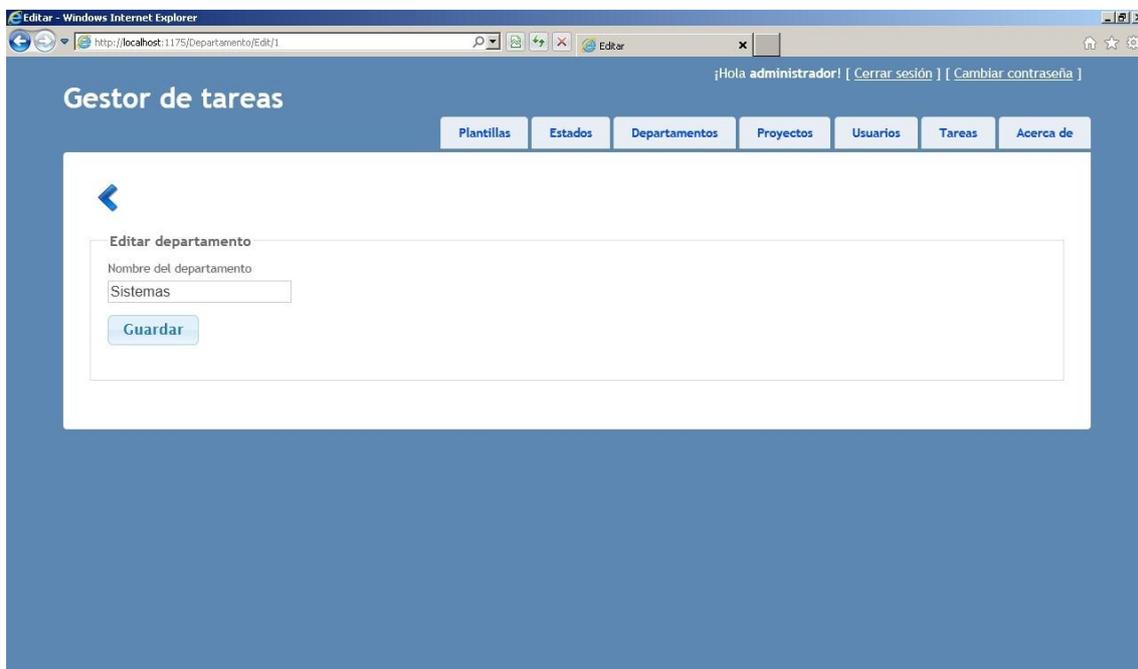
7.5.12 Pantalla de creación de departamentos

Pulsando el icono  podemos crear departamentos nuevos:



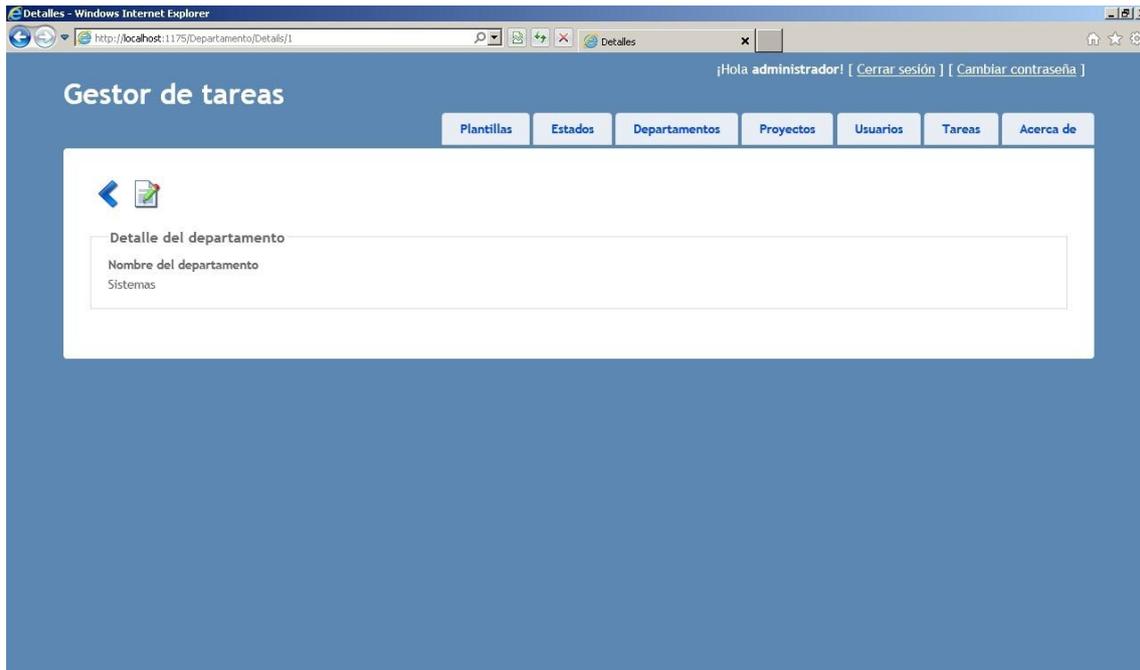
7.5.13 Pantalla de edición de departamentos

Pulsando el icono  podemos editar un departamento:



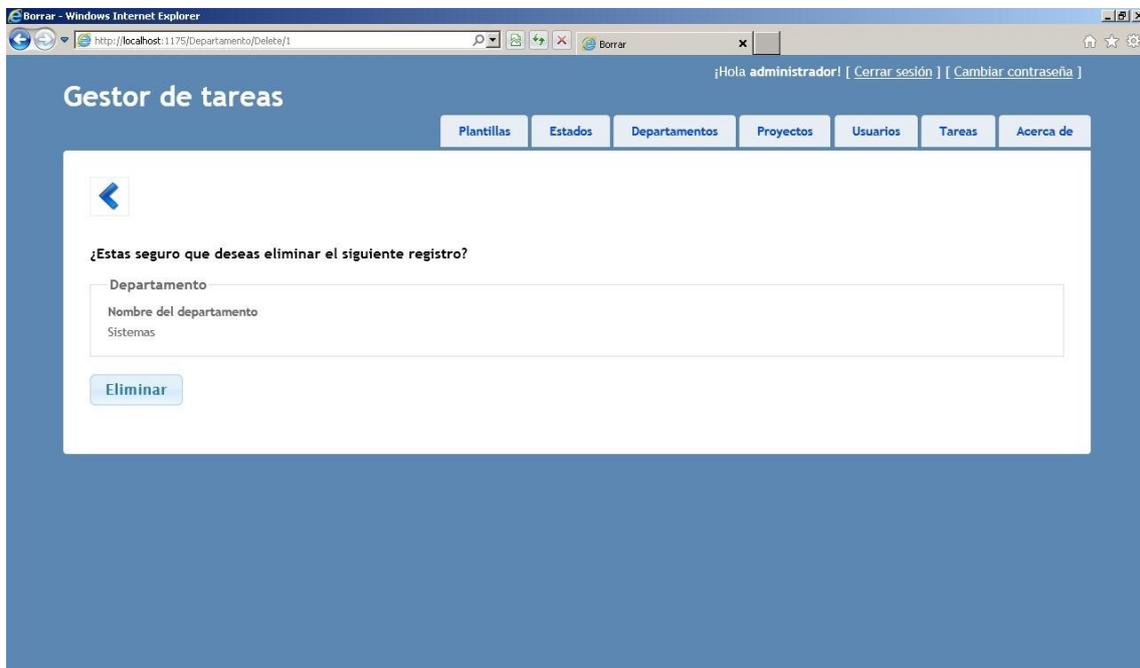
7.5.14 Pantalla de detalles de un departamento

Pulsando el icono  podemos ver los detalles de un departamento:



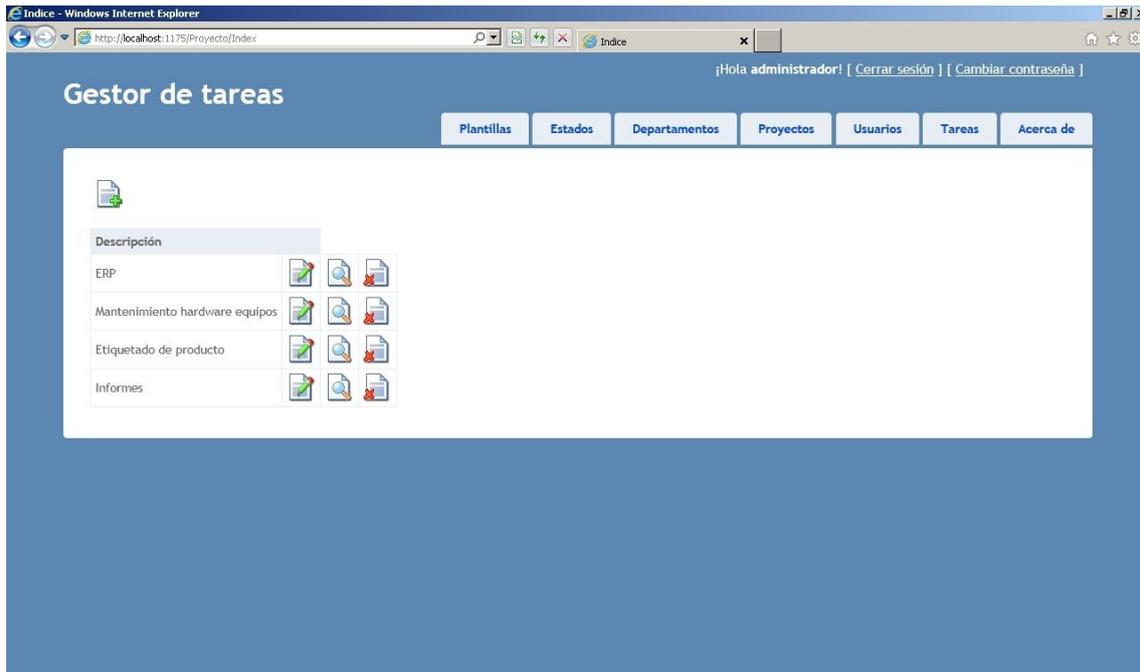
7.5.15 Pantalla de eliminación de departamentos

Pulsando el icono  podemos eliminar un departamento:



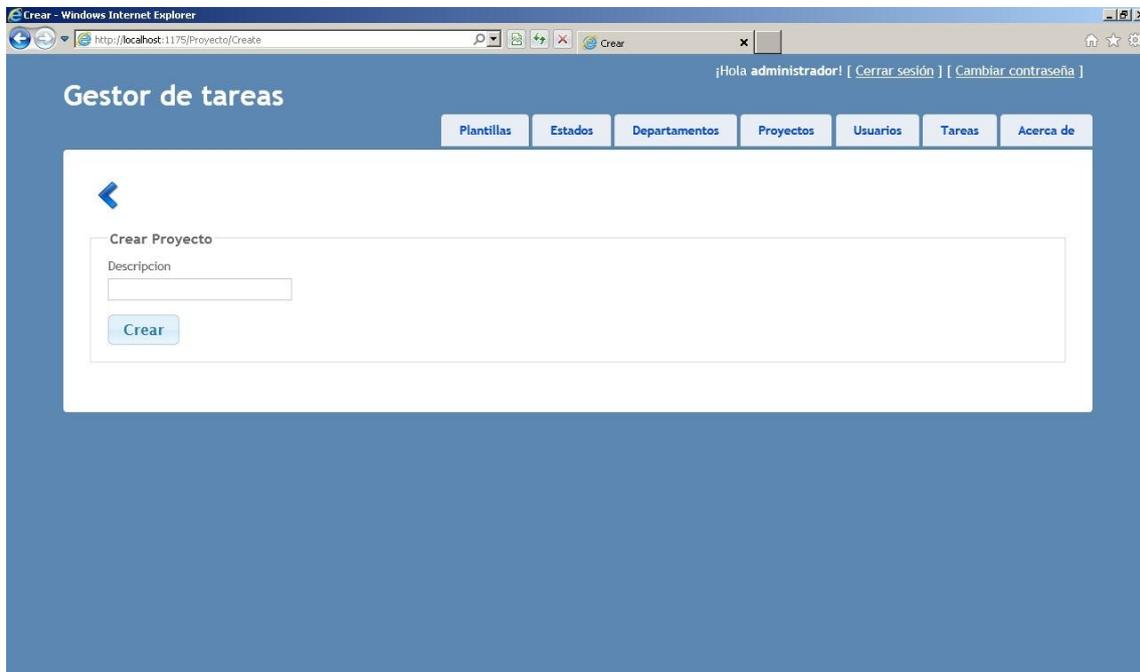
7.5.16 Pantalla de lista de proyectos

En esta pantalla podemos ver todos los proyectos de la aplicación:



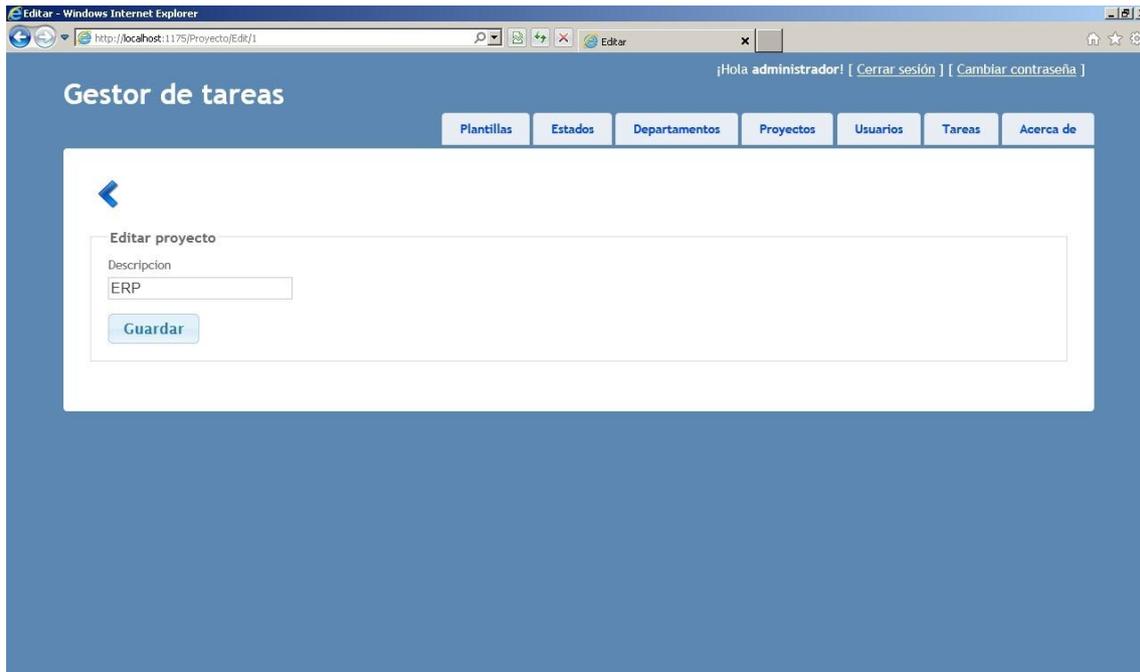
7.5.17 Pantalla de creación de proyectos

Pulsando el icono  podemos crear proyectos:



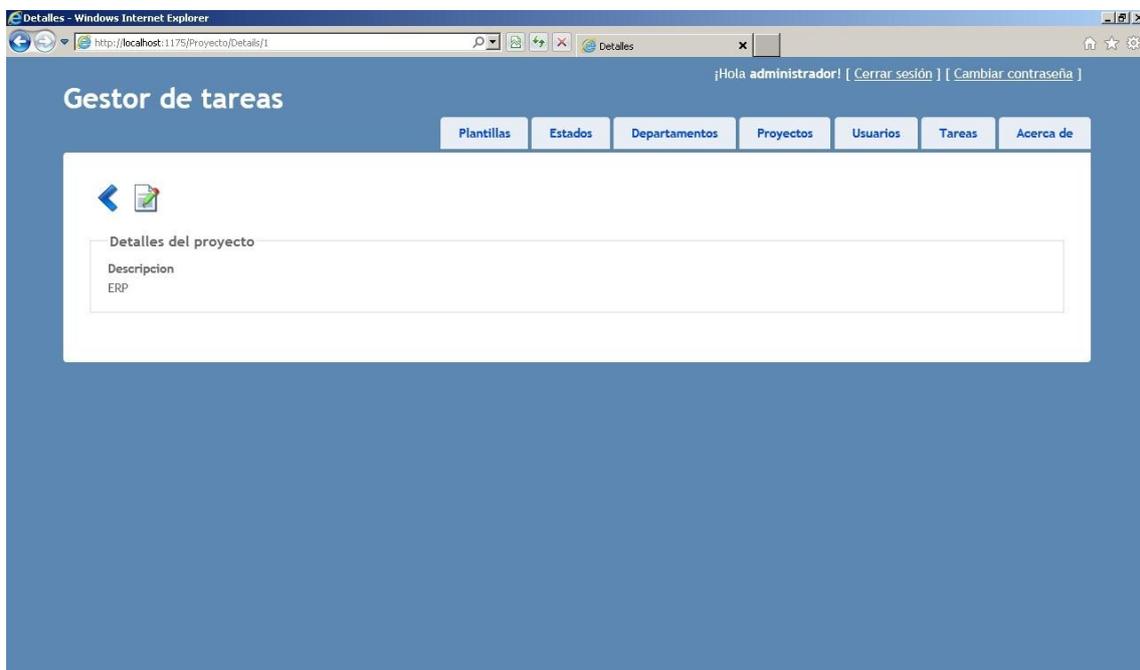
7.5.18 Pantalla de edición de proyectos

Pulsando el siguiente icono  podemos editar un proyecto:



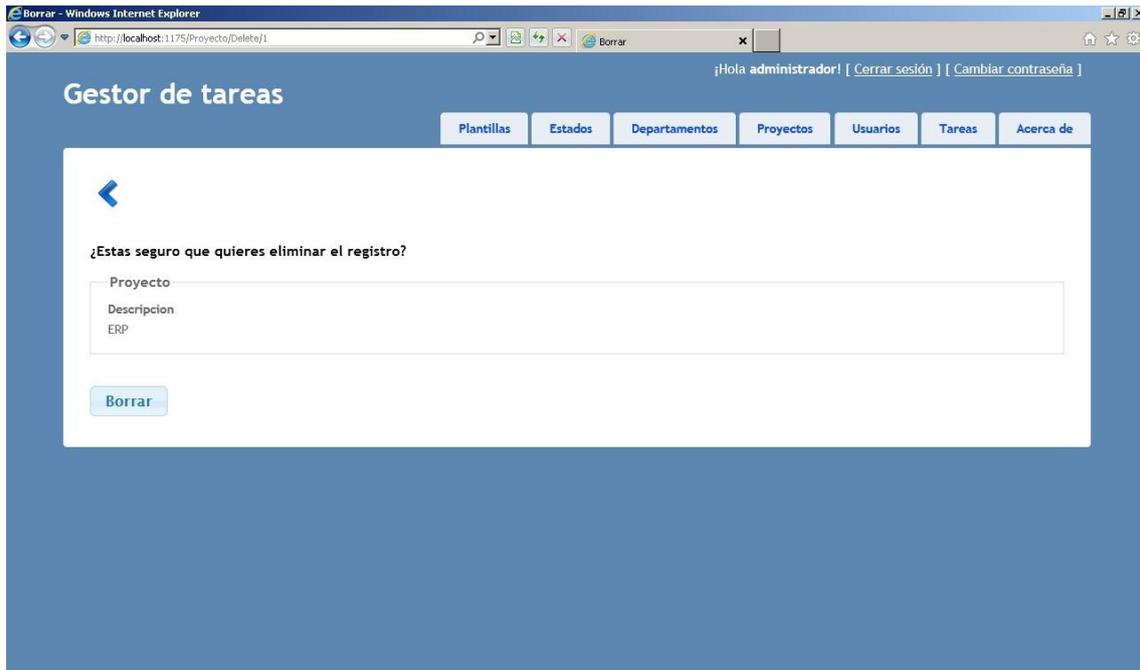
7.5.19 Pantalla de detalles de proyectos

Pulsando el siguiente icono  podemos ver los detalles de un proyecto:



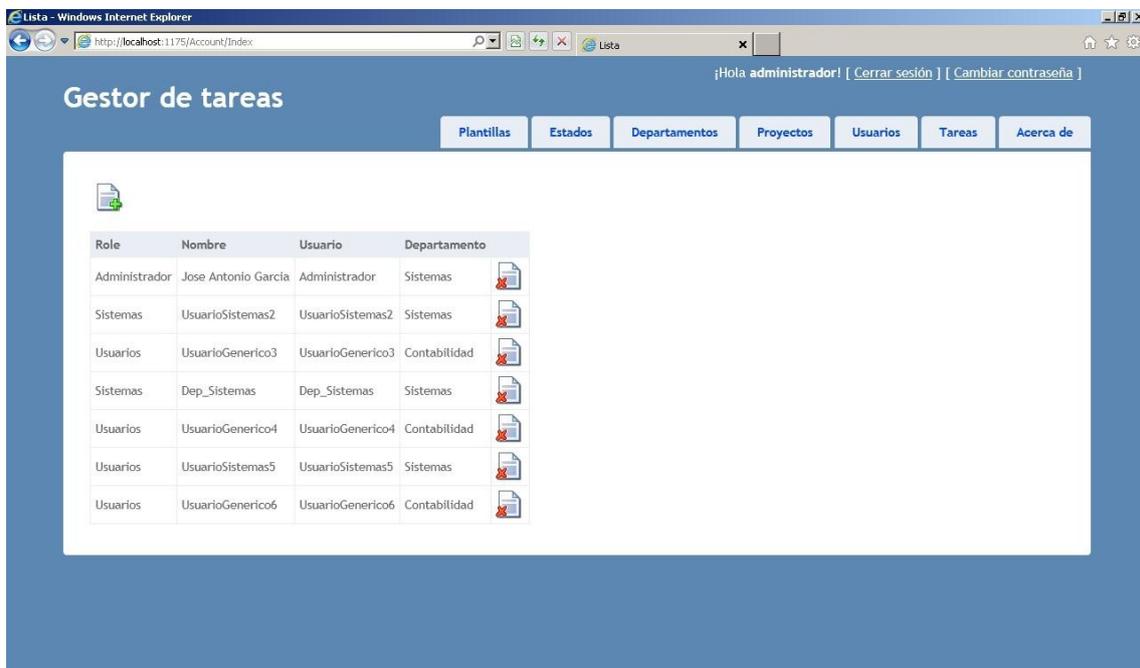
7.5.20 Pantalla de eliminación de proyectos

Pulsando el icono  podemos eliminar un proyecto



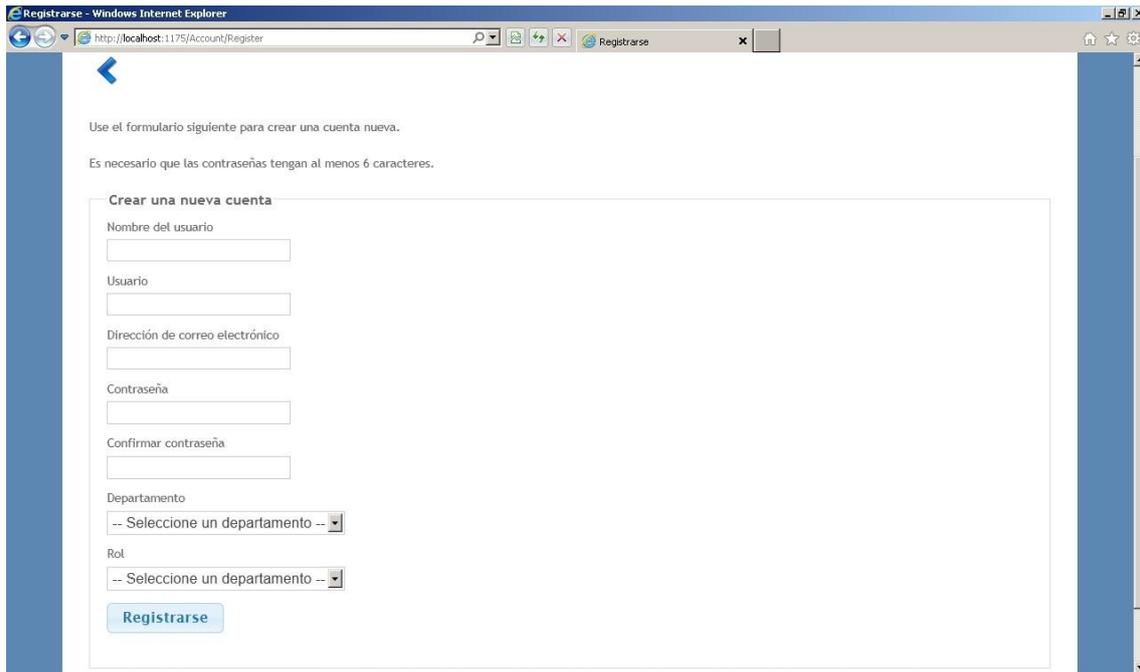
7.5.21 Pantalla de lista de usuarios

En esta pantalla podemos ver la lista de usuarios de la aplicación:



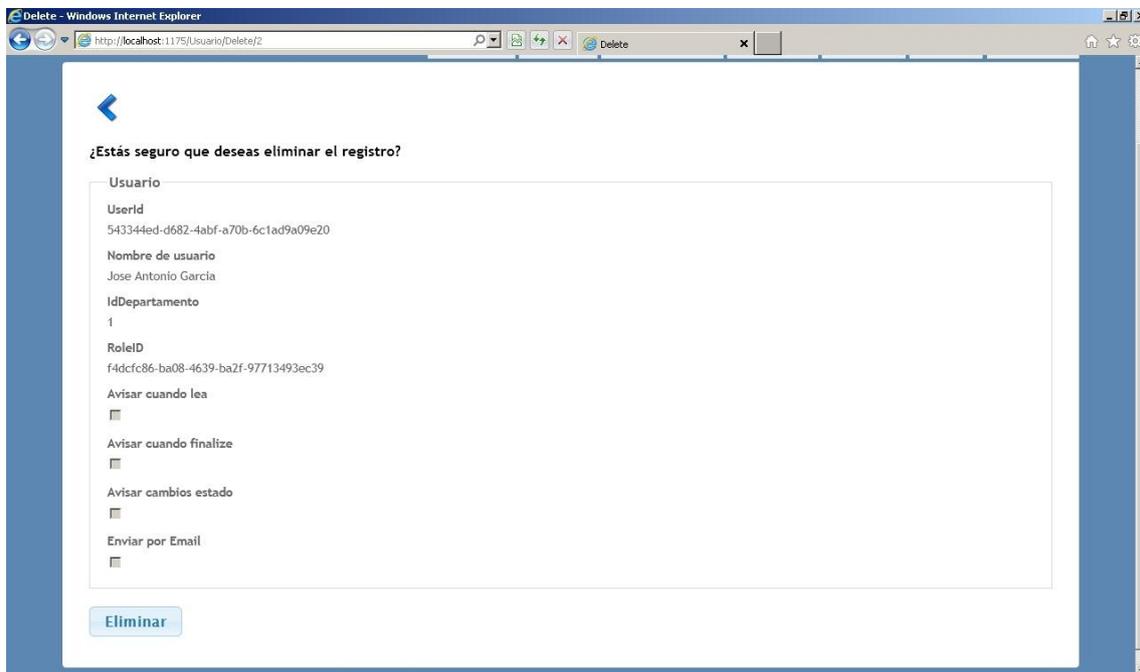
7.5.21 Pantalla de creación de usuarios

Pulsando este icono  vamos a la ventana de creación de usuarios:



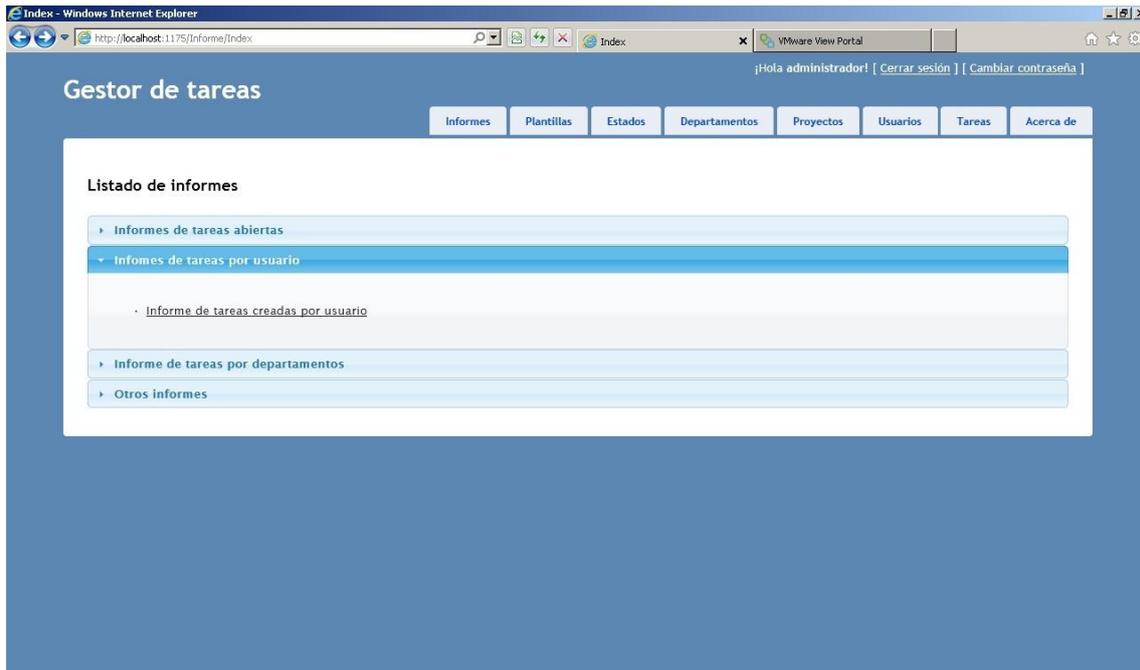
7.5.22 Pantalla de eliminación de usuarios

Pulsando este icono  podemos eliminar un usuario:

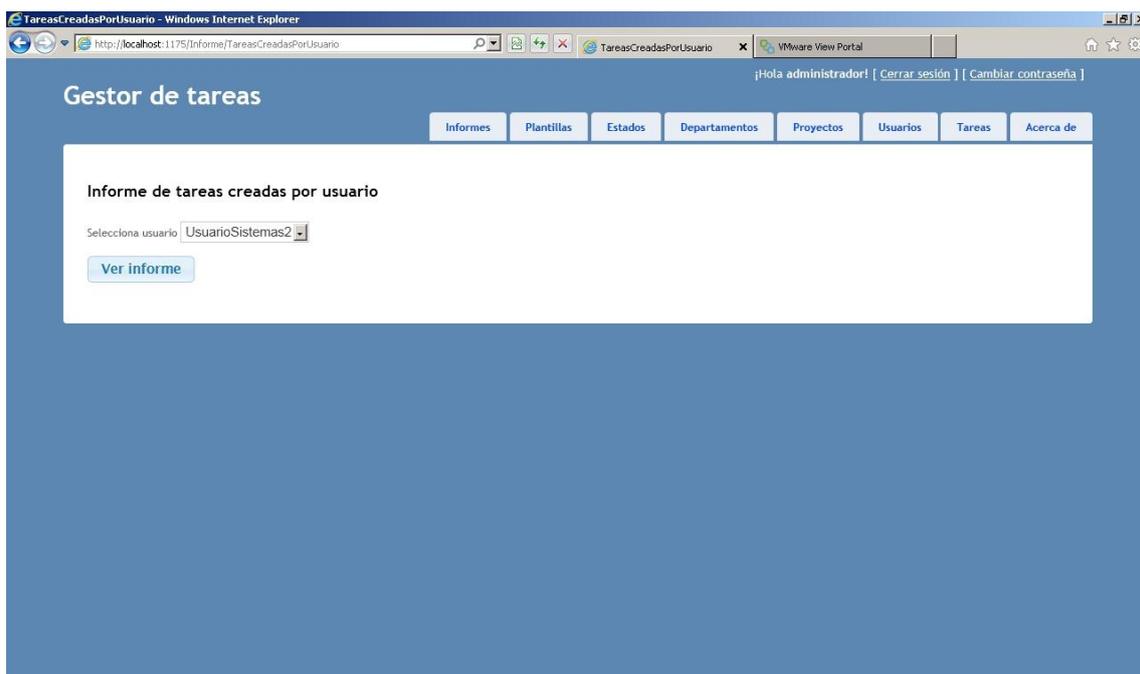


7.5.23 Pantalla de visualización de informes

Desde esta pantalla podemos ver los informes disponibles en la aplicación:



Seleccionando un informe de los disponibles, si necesita parámetros, veremos una pantalla similar a la siguiente:



Y ya por último veremos el informe:

http://localhost:1175/Informe/TareasCreadasPorUsuario - Windows Internet Explorer

http://localhost:1175/Informe/TareasCreadasPorUsuario

TareasCreadasPorUsuario VMware View Portal

INFORME DE TAREAS CREADAS POR USUARIO

UsuarioGenerico3 (Contabilidad)
Sin finalizar

<u>Destinatario</u>	<u>Dep_Sistemas (Sistemas)</u>	<u>Estado tarea</u>	<u>FechaCreacion</u>	<u>Fechafinalizacion</u>	<u>Minutos</u>
Peticion de informe		Sin empezar	29/03/2012		0
Nuevo desarrollo		Sin empezar	29/03/2012		0

<u>Destinatario</u>	<u>UsuarioSistemas2 (Sistemas)</u>	<u>Estado tarea</u>	<u>FechaCreacion</u>	<u>Fechafinalizacion</u>	<u>Minutos</u>
Revisión de la impresora del despacho		Sin empezar	01/04/2012		0
Informe de bultos del almacén		Sin empezar	02/04/2012		0

Imprimir archivo (Ctrl+P)

8.- Instalación y configuración del software

8.1.- Pre-requisitos:

Para el proyecto hemos usado Visual Studio 2010 Ultimate, con Service Pack 1. Por lo que se requerirá como mínimo esta configuración, pudiendo substituirse la versión "Ultimate" por la "Web Developer".

Adicionalmente para poder usar ASP.NET MVC3 tenemos que instalar el siguiente software:

- "AspNetMVC3ToolsUpdateSetup.exe". Descargado desde:
<http://www.microsoft.com/en-us/download/details.aspx?id=1491>
- "AspNetMVC3ToolsUpdateSetup_ESN.exe. Descargado desde:
<http://www.microsoft.com/downloads/es-es/details.aspx?FamilyID=82cbd599-d29a-43e3-b78b-0f863d22811a>

Como gestor de base de datos hemos usado SQL Server 2008 R2, en su versión Express, con Service Pack 3, el cual podemos obtener desde:

<http://www.microsoft.com/downloads/es-es/details.aspx?familyid=8b3695d9-415e-41f0-a079-25ab0412424b>

Aunque también podemos usar cualquier otra versión, como la 2005/2008/2012 en todas las modalidades, como la Express, Enterprise, etc.

Además, como gestor de páginas web debemos tener instalado y configurado un IIS (Internet Information Server)

Para los informes hemos usado Crystal Reports, como no viene instalado por defecto debemos descargarlo de:

<http://www.businessobjects.com/jump/xi/crvs2010/default.asp?ClickID=bygenmgIndv1ns11eznkzusgnl6vknm1dsev>

Una vez descargado debemos ejecutar el setup y seguir los pasos que nos indica.

8.2.- Restauración de la base de datos:

Para empezar a usar el software, necesitamos restaurar el fichero "TareasTFC.bak" en la base de datos. Se recomienda llamar a la base de datos "TareasTFC", para así ahorrarnos algunos pasos luego, al configurar el proyecto de Visual Studio.

8.3.- Uso de los fuentes:

Una vez que hemos terminado con el paso anterior, importamos el proyecto en Visual Studio y nos dirigimos al “**web.config**”, lo abrimos y debemos modificar los siguientes campos para adaptarlos a la máquina donde vamos a ejecutarlo:

- Buscamos el apartado de configuración “**connectionStrings**”, y modificamos:
 - o El **connectionstring** de “**ApplicationServices**”
 - o El **connectionstring** de “**TareasTFCEntities**”

- Buscamos el apartado “**AppSettings**” y modificamos los campos necesarios:
 - o **UsuarioBBDD**: usuario que utilizamos para la conexión para la base de datos. En nuestro caso “**sa**”
 - o **PassBBDD**: password de la base de datos. En nuestro caso “**egacbo**”
 - o **ServidorBBDD**: nombre del pc donde está instalado el servidor de base de datos, no se debe poner “**localhost**” por que sino los informes pueden no funcionar (me ha ocurrido que hay Windows 7 que tienen en el fichero “**hosts**” comentada la línea del “**localhost**”). En mi caso “**Portatill7**”
 - o **BaseDeDatosPredeterminada**: nombre de la base de datos. En nuestro caso “**TareasTFC**”

8.4.- Ponerlo en producción:

Para poner la aplicación en producción debemos irnos al IIS, y con el botón derecho sobre el “Default Web Site” seleccionamos “Crear aplicación”. Una vez que la hemos creado, pulsamos otra vez con el botón derecho sobre la aplicación recién creada y seleccionamos propiedades, y en “Grupo de aplicaciones” seleccionamos “ASP.NET v4.0 Classic”.

Ahora nos dirigimos a la carpeta de los fuentes, seleccionamos y copiamos a la aplicación creada las siguientes carpetas:

- Bin
- Content
- Informes
- Scripts
- Views

Y los siguientes ficheros:

- Global.asax
- Packages.config
- Web.config

Una vez que hemos realizado todos los pasos, editamos el fichero “web.config” y lo modificamos tal y como lo hemos hecho en el punto anterior.

8.5.- Usuarios de prueba

Aunque se pueden crear más usuarios, adjunto un usuario de cada tipo para hacer pruebas:

Rol	Usuario	Password
Administrador	Administrador	administrador
Dep. Sistemas	usuariosistemas2	usuario2
Usuario genérico	usuariogenerico3	usuario3

9.- Posibles mejoras

Dado que del tiempo disponible para la realización del TFC, una gran parte ha estado dedicado a la creación de la documentación y otra parte considerable ha sido destinado al aprendizaje de las tecnologías usadas, nos hemos encontrado con muy poco tiempo para dedicarle realmente a la tarea de generar la aplicación. Todo esto ha hecho que se hayan quedado fuera muchas de las funcionalidades que podíamos haber implementado, como son:

- Mejorar el apartado de las resoluciones de las tareas en cuanto a la variable tiempo, ya que me hubiera gustado poder implementar una asignación de tiempo a una tarea cada vez que le has dedicado tiempo y no has terminado con ella. Con esto conseguiríamos que al finalizar la tarea el sistema automáticamente computara todo el tiempo que le has ido dedicando y no tendríamos que escribirlo nosotros.
- En una empresa no solo el departamento de sistemas es el que provee servicios, por lo que me gustaría que todos los departamentos estuvieran implementados como el de sistemas para poder recibir peticiones.
- El administrador, que es el responsable de sistemas pueda ver todos los calendarios de los usuarios del departamento de sistemas mediante un desplegable en la vista de los calendarios.
- Habilitar un campo de urgencia cuando se crea la tarea.
- Habilitar un sistema de alertas que nos fuera informando de las tareas que no hemos realizado en un intervalo de tiempo (este sería parametrizable por el administrador).
- Crear la aplicación para dispositivos móviles.
- En la vista de tareas podríamos habilitar algunas funcionalidades extra usando la tecnología Ajax, por ejemplo si quisiéramos enviarle una notificación al creador de la tarea usaríamos un botón y no tendríamos que marcar la opción y darle a grabar.
- En el detalle de las tareas podríamos habilitar los componentes de editor de textos web, para hacerlos mas completos.

10.- Evaluación de costes

Esta parte es un poco complicada de evaluar, ya que siempre he trabajado por cuenta ajena y nunca como autónomo. Pero si tuviera que poner un precio al desarrollo de esta aplicación sería el siguiente:

Tarea	Horas
Análisis y diseño	30
Desarrollo del software	75
Pruebas	15
Creación de manuales y formación	12
Creación de la documentación	15
TOTAL	147

En total tenemos que podríamos realizar el proyecto en 147 horas, cobrando cada una a 40 €, tendríamos que nos sale por 5.880 €.

Contando con que hacemos una media de 8 horas diarias, nos salen un poco menos de 19 días de trabajo, al que podemos añadir alguno mas por las visitas que deberíamos realizar al cliente, lo que hace que se acerque a los 22 días de media que se trabaja al mes.

A este montante debemos descontarle el autónomo, gastos de luz, de alquiler, teléfono, iva, etc. Por lo que al hacer los cálculos del beneficio neto que hemos obtenido con el proyecto sea mas o menos de 4.300 €

11.- Conclusiones

Quería aprovechar este apartado para comentar que siempre me ha gustado la programación, por lo que he enfocado mi actividad profesional a esta tarea. Creo que es innato en la profesión el deseo y la necesidad de aprender cada día cosas nuevas, por el deseo de conocimiento y la necesidad de no estancarse ante un mundo que está en continua evolución.

Este trabajo de fin de carrera me ha dado la oportunidad de introducirme en un mundo totalmente nuevo para mí, ya que a nivel de tecnologías he visto algunas que en mi trabajo diario no había tenido oportunidad de usar, y que a partir de ahora podré poner en funcionamiento.

Ha sido un trabajo muy duro, por que he tenido que aprender toda la tecnología ASP.NET MVC3, el lenguaje C#, el acceso a datos LINQ, y todo lo que está relacionado con la web, como ha sido el uso de hojas de estilo, usar Ajax, JavaScript, JQuery, etc, a la vez que íbamos compaginando la creación de la documentación en un tiempo muy ajustado.

Ahora que veo el final de este proyecto creo que ha sido un acierto total la elección del mismo, por que me ha servido para poner en práctica muchísimos de los conocimientos que hemos ido adquiriendo en las distintas asignaturas que están relacionadas con este desarrollo.

12.- Bibliografía

MVC

- <http://www.asp.net/mvc>
- <http://msdn.microsoft.com/es-es/magazine/cc337884.aspx>
- <http://pensaenbinario.blogspot.com.es/2011/10...>
- <http://speakingin.net/tutorial-asp-net-mvc-3/>
- <http://ruuddottech.blogspot.com/2009/11/learn-...>
- <http://www.desarrolloweb.com/manuales/fr...>
- <http://sebys.com.ar/tag/asp-net-mvc-3-0/>
- <http://social.msdn.microsoft.com/Forums/es-...>

Linq

- <http://www.asp.net/web-forms/videos/data-acce...>
- <http://speakingin.net/tutorial-de-linq-to-sql/>
- <http://msdn.microsoft.com/es-es/library/bb397...>
- <http://code.msdn.microsoft.com/101-LINQ-Sampl...>
- <http://geeks.ms/blogs/jorge/archive/2007/12/0...>

Ajax,JavaScript,Jquery,Json

- <http://www.jquery.com>
- <http://msprogrammer.serviciipeweb.ro/2011/12/...>
- <http://msprogrammer.serviciipeweb.ro/2011/02/...>
- <http://ricardocovo.com/2012/04/06/asp-mvc3-ed...>
- <http://msprogrammer.serviciipeweb.ro/2011/02/...>
- <http://msprogrammer.serviciipeweb.ro/2011/12/...>
- <http://ricardocovo.com/2012/04/06/asp-mvc3-ed...>
- <http://msprogrammer.serviciipeweb.ro/2011/02/...>
- <http://ricardocovo.com/2012/04/06/asp-mvc3-ed...>
- <http://cup-coffe.blogspot.com.es/2011/11/como-...>

Hojas de estilo

- <http://www.lawebera.es/recursos/videotutoriales/hojas-estilos-css.php>
- <http://www.comocreartuweb.com/ejemplo-con-divs/la-plant...>
- <http://www.comocreartuweb.com/ejemplo-con-divs....>
- <http://www.noupe.com/css/using-css....>
- <http://www.csszengarden.com/tr/espanol/>