
ShareProject

Gestiona los contenidos de tu proyecto y mantén una comunicación activa con tu tutor desde cualquier navegador (incluidos navegadores de dispositivos móviles).

Acceder

© Aitor Guisado Camacho

ShareProyect

Autor: Aitor Guisado Camacho

martes, 9 de junio de 2015

Desarrollo de Aplicaciones Web

SHAREPROYECT	1
1. Definición del proyecto	4
1.1. Tipos de empresas / Sectores productivos	4
1.2. Descripción del problema.	4
1.3. Objetivos. Especificación de requisitos.	4
1.4. Herramientas de diseño.	5
1.5. Estudio de viabilidad.	5
2. planificación del proyecto	6
2.1. Secuenciación de las fases del proyecto.	6
2.1. Planificación de recursos y tiempo.	6
3. análisis del proyecto	8
3.1. Casos de uso.	8
3.2. Modelo de datos.	9
3.2.1 Esquema Entidad / Relación.	9
3.2.2. Modelo relacional.	10
3.2.3. Diccionario de datos.	11
4. Diseño del proyecto	12
4.1. Diseño físico de la BD. Estructura de los ficheros	12
4.2. Diseño de la estructura de clases y librerías.	16
4.3. Diseño de procedimientos especiales.	17
4.3. Diseño de la interfaz gráfica.	18
5. Desarrollo y realización del proyecto.	24
5.1. Indicadores de calidad.	24
5.2. Batería de pruebas y solución de incidencias.	25
5.4 Evaluación y seguimiento de actividades.	25

6. implantación del proyecto	26
6.1. Plan de implantación.	26
6.2. Manual de instalación.	26
6.2.1. Instalación Servidor en Windows.	26
6.2.2. Instalación Servidor en Linux.	28
6.2.3. Instalación de la aplicación.	29
6.3. Manual de usuario.	30
7. Conclusiones finales	31
7.1. Evaluación del proyecto. Objetivos alcanzados y no alcanzados.	31
7.2. Propuesta de mejora.	31

1. DEFINICIÓN DEL PROYECTO

1.1. Tipos de empresas / Sectores productivos

El proyecto está dirigido a todos los centros de formación que requieran una gestión de proyectos finales de ciclos formativos, donde profesor y alumno podrán seguir el estado y ciclo de vida de los proyectos de fin de ciclo, además de mantener una comunicación y estar informados del estado del proyecto.

También, los alumnos podrán consultar proyectos acabados como ejemplos.

1.2. Descripción del problema.

Al final un ciclo formativo, se debe realizar un proyecto final, con un seguimiento tutor-alumno. Este proyecto se suele realizar durante la formación en centros de trabajo, y nunca se realiza en el centro escolar, por lo que la comunicación entre el alumno y su correspondiente tutor no es fluida.

Esta aplicación permite al tutor y al alumno seguir el ciclo de vida del proyecto. Dentro de la misma, se puede indicar el progreso del proyecto, subir archivos de documentación, mantener una conversación profesor-alumno, e incluso permite a los profesores valorar el proyecto.

Además, contiene un panel de administración para los profesores y para el administrador, de forma que se puedan gestionar los proyectos, alumnos y profesores existentes, así como los cursos y la relación alumno-tutor. Cada profesor tendrá a una vista los proyectos asignados y podrá gestionarlos.

1.3. Objetivos. Especificación de requisitos.

- La aplicación debe registrar los proyectos que se están realizando y los que se han realizado anteriormente.
- El acceso a la aplicación debe tener una seguridad por medio de usuario y contraseña; esta seguridad también diferenciará entre profesor, alumno y administrador. Los alumnos accederán por medio del DNI y una contraseña, mientras que los profesores accederán por medio de un usuario y una contraseña.

- Al ingresar a la aplicación, el interfaz del alumno debe permitir ver el ciclo de vida de su proyecto y la información del mismo, su perfil con sus datos, poder cambiar el progreso del proyecto, ser capaz de subir archivos de documentación al servidor y mantener una comunicación vía mensajes con su tutor correspondiente.
- El profesor, al conectarse a la aplicación, debe tener acceso a las mismas opciones que el alumno. Además, deberá poder visualizar los proyectos de todos los alumnos que tiene asignados, y poder modificar dichos proyectos y añadir más. También puede ver los alumnos que tiene asignados, y añadir alumnos nuevos.
Se puede cambiar la asignación de los proyectos, o asignar un proyecto a un alumno existente.
- El administrador, tendrá las mismas acciones que los dos anteriores, pero además podrá administrar y añadir profesores.
- Los proyectos pueden estar reservados, asignados, en curso, finalizados, sin acabar o sin finalizado. Debe haber un registro de todos los proyectos finalizados para que los alumnos puedan consultarlos.
- Existen dos convocatorias: Junio y Septiembre. La convocatoria se asigna a la hora de asignar un proyecto.

1.4. Herramientas de diseño.

Como herramientas de diseño, se ha usado Adobe Photoshop para retoques de imágenes y de diseño de la web. También se ha usado el software DIA para el diseño del modelo entidad-relación de la base de datos, y de algunos otros diagramas de relaciones.

Como entorno de desarrollo, se ha usado Sublime Text, con licencia gratuita.

La aplicación corre en un servidor Apache, instalado sobre un equipo Macintosh. La aplicación está preparada para funcionar sobre Apache sobre cualquier sistema operativo.

1.5. Estudio de viabilidad.

Para la realización de este proyecto, se han utilizado las tecnologías de Ajax, JQuery (con JQuery UI), JavaScript, HTML, MySQL, CSS3, y en algunos casos el uso del framework BootStrap.

El coste del proyecto es nulo, las tecnologías aplicadas son de libre uso, el entorno de desarrollo también es gratuito. Por tanto, el coste del proyecto estaría ligado a las horas de programación dedicadas al mismo. En caso de otros programas para partes mínimas del proyecto, como puede ser PhotoShop, se ha usado la licencia de prueba de los mismos.

Por tiempo y conocimientos, se ha concluido que el proyecto es viable.

2. PLANIFICACIÓN DEL PROYECTO

2.1. Secuenciación de las fases del proyecto.

La primera fase del proyecto es la obtención de requisitos. Esta fase se ha completado con la obtención del antiguo gestor de proyectos que usaba el IES Arcipreste de Hita, además de las reuniones con el tutor y diversos emails intercambiados para aclarar los cambios y los requisitos. También destaca la aportación de un documento por parte del tutor donde se especifica de forma resumida los requisitos del proyecto.

En la segunda fase, durante el diseño de la solución al problema, se ha usado el programa DIA para el diseño de la base de datos, así como la relación de tablas y otras relaciones entre archivos para hacer un esquema del diseño del proyecto.

En la fase de desarrollo, se ha usado PHP como lenguaje de servidor, y CSS3, HTML, JavaScript (Ajax, JQuery, JQuery UI) y el framework BootStrap como lenguajes del lado del cliente, además de MySQL como sistema gestor de bases de datos.

Por último, durante la fase de puesta en funcionamiento, se ha desarrollado un documento de como exportar la información existente de la base de datos del centro al nuevo gestor.

2.1. Planificación de recursos y tiempo.

El tiempo se ha repartido a lo largo de un mes y medio, dedicando una parte inicial del tiempo a la toma de decisiones sobre el diseño, frameworks, y tecnologías a usar.

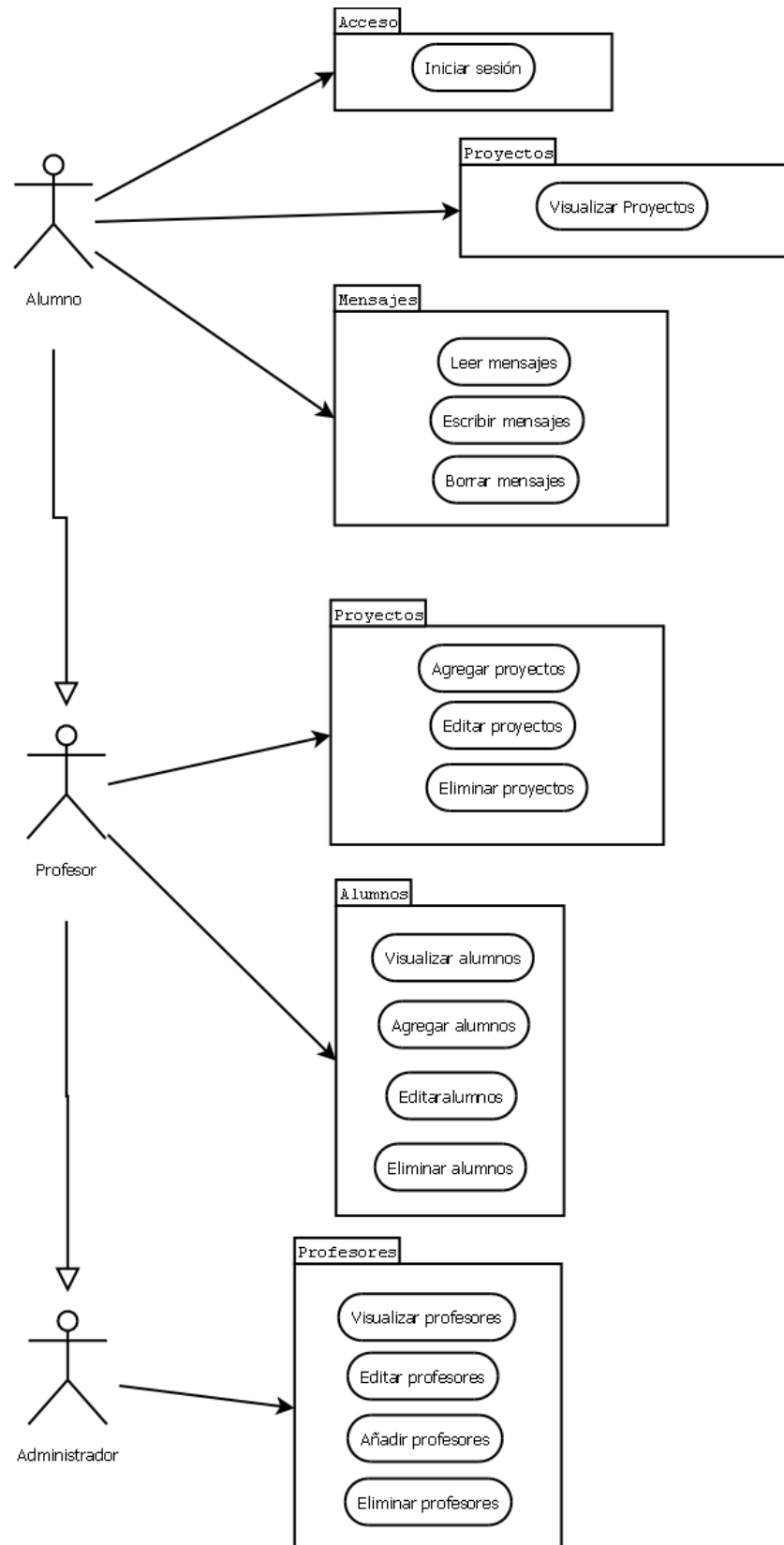
Una vez tomada las decisiones del diseño, que ha tomado aproximadamente unas 5 horas de dedicación de búsqueda y selección, se ha dispuesto a adaptar el diseño al proyecto. Esto ha supuesto alrededor de una cuarta parte del tiempo dedicado al proyecto. (Se han elegido colores

corporativos para el proyecto, se ha elegido los componentes de diseño que usará cada parte del proyecto, y se ha desarrollado de forma estática cada página del proyecto).

Por último, se ha creado la base de datos y se ha dispuesto a la programación del proyecto para hacerlo funcional. Tras esta etapa, se ha dedicado un tiempo extra a la reestructuración del código, a la revisión de la documentación del código, y a la refactorización y eficiencia del código (revisión del código con finalidad de hacerlo más eficiente y ordenado).

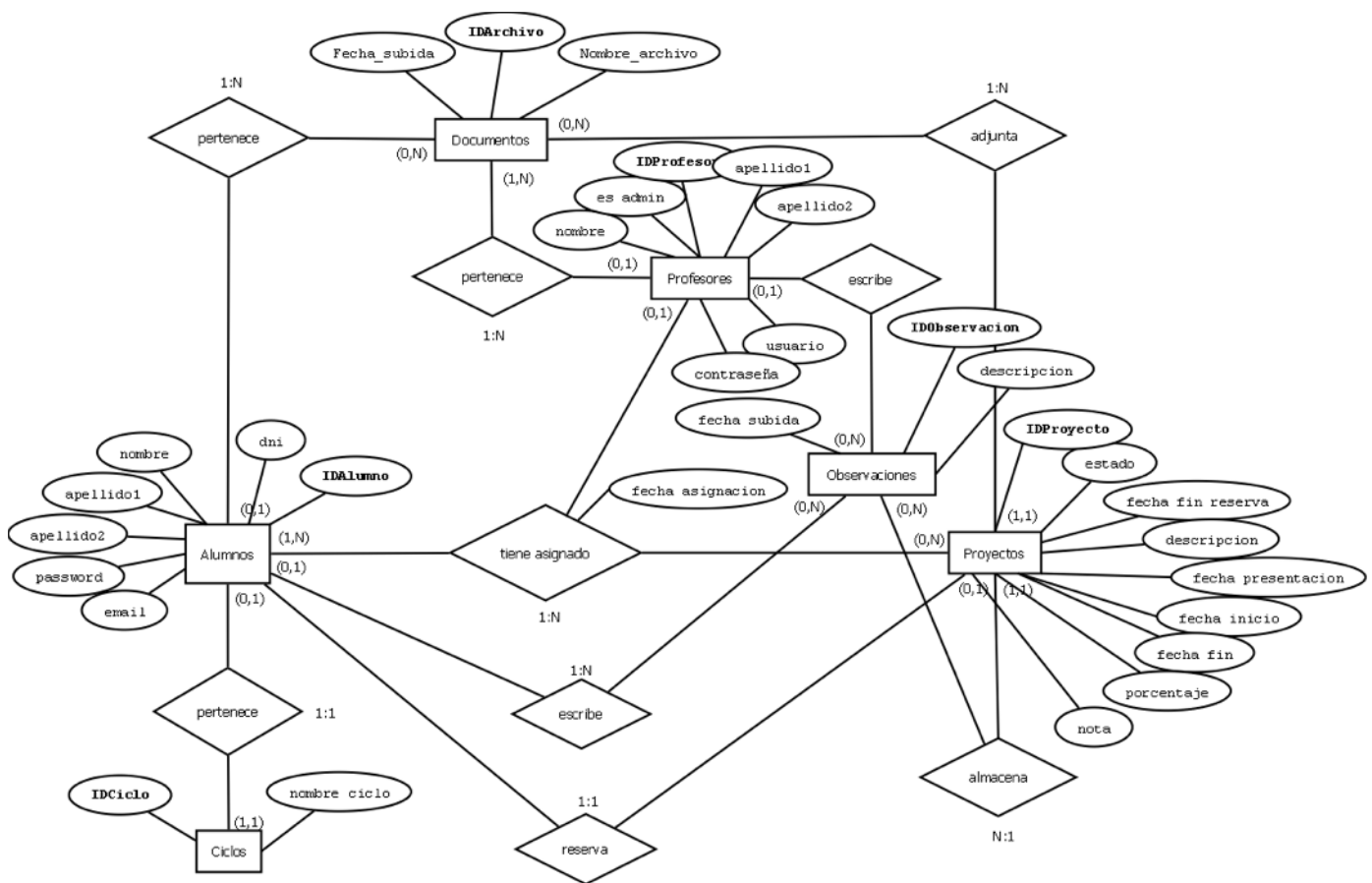
3. ANÁLISIS DEL PROYECTO

3.1. Casos de uso.



3.2. Modelo de datos.

3.2.1 Esquema Entidad / Relación.



3.2.2. Modelo relacional.

Se realiza el modelo relacional:



Como podemos observar, en el paso a tablas se han tomado las siguientes decisiones de diseño:

1. El atributo estado de la tabla proyectos pasa a ser una tabla, con su identificador y su descripción, denominada Estados.
2. Como un documento pertenece a un profesor o a un alumno, para identificar esta funcionalidad, en la tabla documentos además de añadir el id del usuario al que corresponde el documento, se añade un campo llamado "profesor" que tomará el valor 1 si es profesor y 0 si es alumno.
3. Como una observación puede ser realizada por un profesor o por un alumno, en la tabla observaciones además de añadir el id del usuario al que corresponde el documento, se añade un campo llamado "profesor" que tomará el valor 1 si el profesor el que ha añadido la observación y 0 si ha sido el alumno.
4. En las tablas alumno y proyecto, se ha añadido un campo llamado "asignado", que nos permite saber si ese alumno y/o proyecto ya está asignado. El valor del campo "asignado" se pondrá a 1 en el momento en el que un alumno o proyecto tenga una asignación. De esta forma ahorramos

consultas a la BBDD ya que no será necesario acceder continuamente a la tabla Proyecto_alumno_profesor para buscar esta información.

5. La entidad profesor dispone de un campo admin, que nos permite saber si ese usuario es administrador del sistema.

3.2.3. Diccionario de datos.

En el programa se utiliza un fichero que conecta el cliente con el servidor, el cual se encarga de recoger la función solicitada y ejecutarla.

Las funciones ejecutadas obtienen los datos ejecutando sentencias sobre una base de datos MySQL, además también editan dichos datos, los eliminan o agregan nuevos registros.

Los datos que se manejan en el sistema son:

Alumnos: Son los alumnos que tienen que realizar un proyecto, de ellos se almacena el nombre y apellidos, la contraseña que usarán para entrar al sistema, el DNI, el email, el ciclo al que pertenecen y si tiene un proyecto asignado o no.

Proyectos: Son los proyectos almacenados en el sistema, los cuales pueden estar sin asignar, en curso, finalizados, sin acabar o reservados. De cada proyecto se almacena una descripción, el estado del mismo, la fecha de presentación, de inicio y de fin, la nota, el progreso en forma de porcentaje, el alumno al que esta reservado y la fecha de fin de reserva. También se almacena la materia en la que se basa la mayor parte del proyecto.

Profesores: Son los profesores registrados en el sistema, y de ellos se almacena el nombre y apellidos, el usuario y la contraseña, el email y se indica si el profesor es administrador o no.

Ciclos: Se almacena todos los ciclos a los que pueden pertenecer los alumnos registrados.

Estados: Se almacenan los posibles estados de un proyecto indicados anteriormente.

Documentos: Se indica los documentos que se han subido al servidor por parte del profesor o del alumno. De ellos se almacena el nombre, la fecha, el profesor al que pertenecen, el usuario al que pertenecen, y el proyecto al que pertenecen.

Observaciones: Se almacena la comunicación profesor-alumno a través de este campo, indicando el proyecto al que pertenece, su tutor, su alumno y la fecha.

4. DISEÑO DEL PROYECTO

4.1. Diseño físico de la BD. Estructura de los ficheros

La base de datos se compone de las siguientes tablas:

Tables_in_gestor_proyectos
alumnos
ciclos
documentos
estados
observaciones
profesores
proyecto_alumno_profesor
proyectos

Tabla alumnos:

Field	Type	Null	Key	Default	Extra
IDAlumno	int(11)	NO	PRI	NULL	auto_increment
Nombre	varchar(80)	NO		NULL	
Apellido1	varchar(80)	NO		NULL	
Apellido2	varchar(80)	NO		NULL	
Password	varchar(20)	NO		NULL	
DNI	varchar(80)	YES		NULL	
Email	varchar(100)	NO		NULL	
Ciclo	int(11)	NO		NULL	
Asignado	int(11)	NO		NULL	

Tabla Ciclos:

Field	Type	Null	Key	Default	Extra
IDCiclo	int(11)	NO	PRI	NULL	auto_increment
NombreCiclo	varchar(120)	NO		NULL	

Tabla Documentos:

Field	Type	Null	Key	Default	Extra
IDArchivo	int(11)	NO	PRI	NULL	auto_increment
Nombre	varchar(100)	NO		NULL	
IDProyecto	int(11)	NO		NULL	
IDUsuario	int(11)	NO		NULL	
Profesor	int(11)	NO		NULL	
Fecha_subida	datetime	NO		NULL	

Tabla estados:

Field	Type	Null	Key	Default	Extra
IDEstado	int(11)	NO	PRI	NULL	auto_increment
Descripcion	varchar(50)	NO		NULL	

Tabla observaciones:

Field	Type	Null	Key	Default	Extra
IDObservacion	int(11)	NO	PRI	NULL	auto_increment
Descripcion	varchar(255)	NO		NULL	
IDProyecto	int(11)	NO		NULL	
IDUsuario	int(11)	NO		NULL	
Profesor	int(11)	NO		NULL	
Fecha	datetime	NO		NULL	

Tabla profesores:

Field	Type	Null	Key	Default	Extra
IDProfesor	int(11)	NO	PRI	NULL	auto_increment
Nombre	varchar(80)	NO		NULL	
Apellido1	varchar(80)	NO		NULL	
Apellido2	varchar(80)	NO		NULL	
Usuario	varchar(100)	NO		NULL	
Password	varchar(20)	NO		NULL	
Email	varchar(100)	NO		NULL	
Admin	int(11)	NO		NULL	

Tabla proyecto_alumno_profesor:

Field	Type	Null	Key	Default	Extra
IDProyecto	int(11)	NO	PRI	NULL	
IDAlumno	int(11)	NO	PRI	NULL	
IDProfesor	int(11)	NO	PRI	NULL	
Fecha_asig	date	NO	PRI	NULL	
Convocatoria	varchar(100)	NO		NULL	

Tabla proyectos:

Field	Type	Null	Key	Default	Extra
IDProyecto	int(11)	NO	PRI	NULL	auto_increment
Descripcion	varchar(80)	NO		NULL	
Estado	int(11)	NO		NULL	
Fecha_presentacion	date	NO		NULL	
Fecha_inicio	date	NO		NULL	
Fecha_fin	date	NO		NULL	
Nota	float	NO		NULL	
Porcentaje	int(11)	NO		NULL	
Asignado	int(11)	NO		NULL	
IDAlumno_reservado	int(11)	NO		NULL	
Fecha_fin_reserva	date	NO		NULL	

Por parte de la estructura de ficheros, el proyecto se dispone de 7 carpetas, a saber:

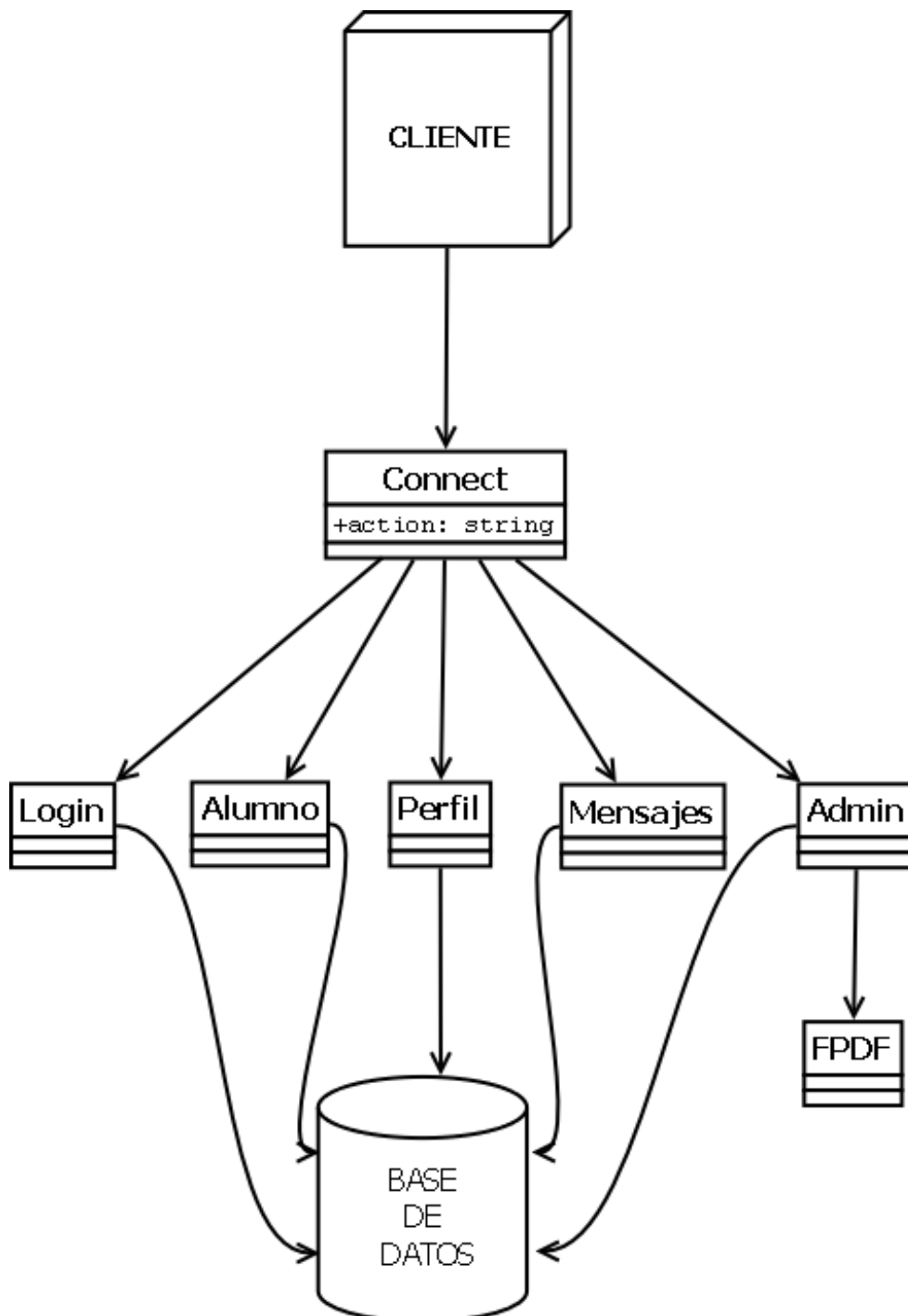
1. CSS: En esta carpeta se disponen los diversos archivos de estilos, que se encargan del diseño de la web. El diseño de la web es responsive, por lo que se adaptará perfectamente a cualquier dispositivo. En esta carpeta también está el archivo CSS correspondiente al framework Bootstrap.
2. font-awesome: En esta carpeta se disponen los tipos de letra utilizados, así como varios iconos como los del panel de administración. Estos iconos son extraídos a través de CSS y archivos de fuentes, que agiliza bastante la carga de dichos iconos.
3. html: Se distribuyen los diferentes ficheros HTML. En esta carpeta se encuentran dos ficheros principales: alumno.html y profesor.html, que son los que muestran la pagina de ambos

interfaces, el resto de ficheros corresponden a código html que se va recargando sobre ciertas capas a través de AJAX con el fin de no recargar la página completa y ordenar mejor el código (no embeberlo en javascript).

4. `img`: En esta carpeta se colocan las imágenes que se han ido usando a lo largo de proyecto, como por ejemplo en la gestión de proyectos, la papelera para eliminar, la flecha para retroceder, etc.
5. `javascript`: Se colocan varios ficheros de programación en JavaScript. Los ficheros `bootstrap.min.js`, `jquery.min.js`, `jquery-ui.js` y `bootbox.min.js` son ficheros de frameworks. `admin.js` corresponde al javascript empleado en el panel de administración. `index.js` corresponde al código de la página index, `perfil.js` corresponde al manejo del perfil de usuario, `mensajes.js` corresponde al código empleado para el sistema de mensajes, y, por último, `alumno.js` corresponde al javascript necesario para el funcionamiento de la interfaz del alumno.
6. `php`: La programación del lado del servidor se coloca en este directorio, y se compone de varios archivos PHP. El archivo principal es `connect.php`, el cual recibe la función que solicita el cliente, y ejecuta el módulo correspondiente que contenga dicha función. `admin.php` se encarga de la gestión del panel de administración. `login.php` se encarga del sistema de inicio de sesión de usuario. `alumno.php` se encarga del tratamiento de datos para la interfaz del alumno. `mensajes.php` se encarga de la gestión de los mensajes. Y `perfil.php` se encarga de la gestión de datos de perfiles.
7. `fpdf`: Es una librería escrita en php, encargada de generar archivos PDF. Se usa para generar los informes.

4.2. Diseño de la estructura de clases y librerías.

Se define el siguiente diagrama de clases, que indica la jerarquía en los archivos que realizan las operaciones del lado del servidor.



4.3. Diseño de procedimientos especiales.

Se han diseñado todos los procedimientos en modo de aplicación web. Por tanto, todas las recargas y solicitudes al servidor se hacen por medio de AJAX.

La ventaja de usar este método es que sólo se recargan aquellos elementos que requieren ser actualizados.

Se invoca una llamada a una función javascript a través de un evento, como puede ser un botón. En la función, se hace una solicitud al servidor por medio de AJAX:

- Se crea el objeto AJAX: `var ajax = new XMLHttpRequest();`
- Se solicita un archivo del servidor, especificando método get o post: `ajax.open("post",SERVER);`
- Se envía una solicitud al servidor, la solicitud: `ajax.send("action=valor");` En este caso se envía una variable "action" con un valor que indicará la función que el servidor debe realizar.
- La solicitud es recibida por el servidor, resuelta, y el mismo servidor devolverá una respuesta.
- Como protocolo de comunicaciones se utiliza JSON.
- Desde AJAX recoges la respuesta: `var respuesta = ajax.responseText;`
- La respuesta está en formato JSON, para decodificarla se utiliza: `JSON.parse(respuesta);`
- Ahora, dentro de la función de AJAX, se realizan las acciones necesarias una vez recibidos los datos con éxito

Todo este proceso es asíncrono, por lo tanto la página no se queda inservible esperando los datos para recargar, sino que puedes realizar varias acciones simultáneamente, lo que aporta una mayor fluidez y la ventaja de multitarea.

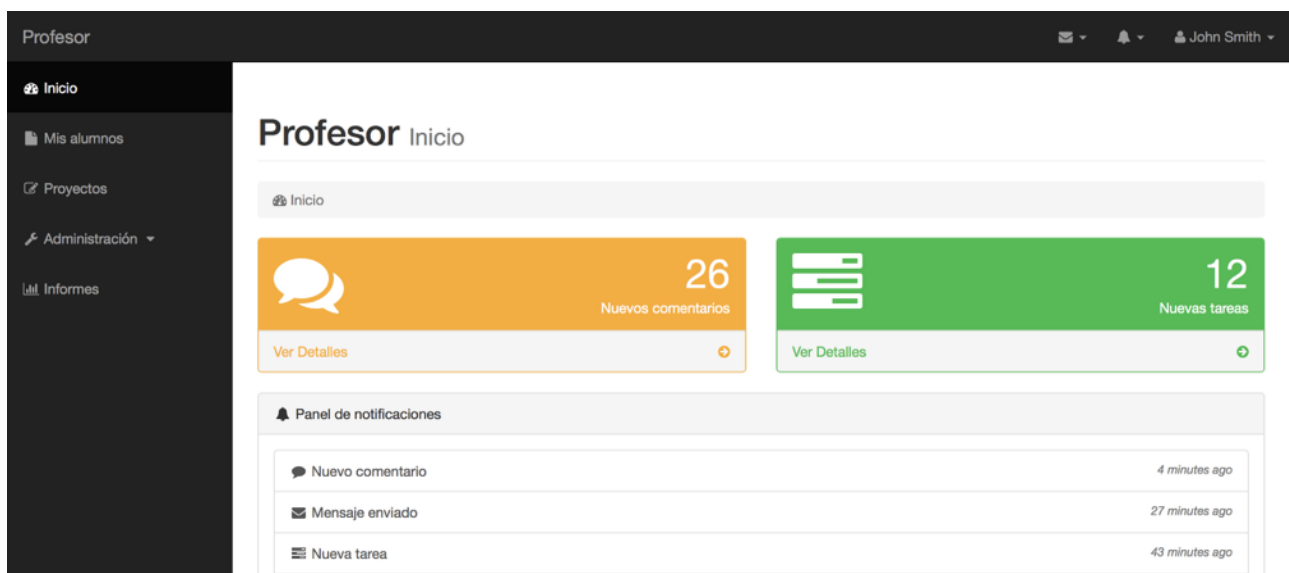
Además, otra ventaja de esta tecnología es la ocultación de errores del lado del servidor. Si en el servidor se produjese algún error a la hora de resolver la petición, devolvería un código de error a AJAX, y por tanto, no ocurriría nada en la aplicación web.

4.3. Diseño de la interfaz gráfica.

A la página se accede mediante un index, que ya diferencia entre la conexión mediante una cuenta de profesor / administrador o mediante una cuenta de alumno:



Para el acceso de profesores y de administrador, se abre una interfaz que permite, a simple vista, todas las nuevas tareas que se han producido



Dentro del panel izquierdo, se puede acceder a la pestaña Mis alumnos, que muestra una tabla con los alumnos que el profesor tiene asignados, y el progreso de su proyecto:

Profesor

Inicio Mis alumnos Proyectos Administración Informes

←

Título: Creacion de super lobezno

Alumno: Aitor Guisado camacho

Tutor: Administrador

Fecha de Asignación: 2015-06-06

Fecha de Presentación: 0000-00-00

Convocatoria: Junio-2015

Ciclo: Desarrollo de Aplicaciones Web

Progreso: 25 %

Estado: Asignado

Nota: 10

En proyectos, se muestra cada uno de los proyectos registrados en el sistema, con la posibilidad de ordenarlos por alumno, tutor, ciclo, convocatoria y materia, y filtrarlos por todos, en curso, finalizados, sin acabar o sin asignar:

Profesor

Inicio Mis alumnos Proyectos Administración Informes

Ordenar por: Alumno

Todos En curso Finalizados Sin acabar Sin asignar

Creacion de super lobezno

Alumno: Guisado camacho, Aitor

Tutor: Administrador

Ciclo: Desarrollo de Aplicaciones Web

25%

Ver proyecto

Crear un lobo para una loba

Alumno: lobezno lobeznoo, lobo

Tutor: Guisado Camacho, Aitor

Ciclo: Administracion de Sistemas Informaticos en Red

55%

Ver proyecto

Comparador de tablas de SkateBoard

Alumno: Zam Lobo, Sergio

Tutor: Guisado Camacho, Aitor

Ciclo: Administracion de Sistemas Informaticos en Red

Completado

Nota: 9

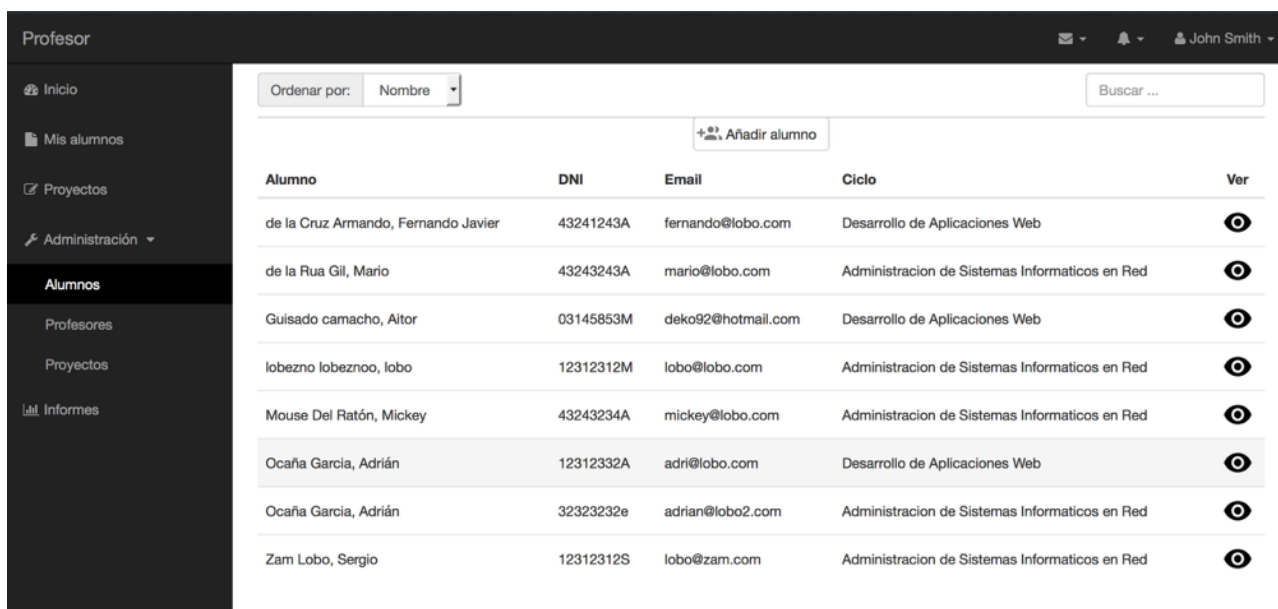
Ver proyecto

Pulsando sobre Ver proyecto, te lleva a la página individual de ese proyecto, para poder visualizar su información completa:

Esta información se puede editar, pulsando sobre el botón con forma de lápiz. En ese caso, los campos se volverán editables y aparecerán dos botones, uno para cancelar y otro para aplicar cambios.

La sección de administración abre un submenú, para administrar alumnos, profesores o proyectos. El botón para administrar profesores solo existe en caso de ser administrador:

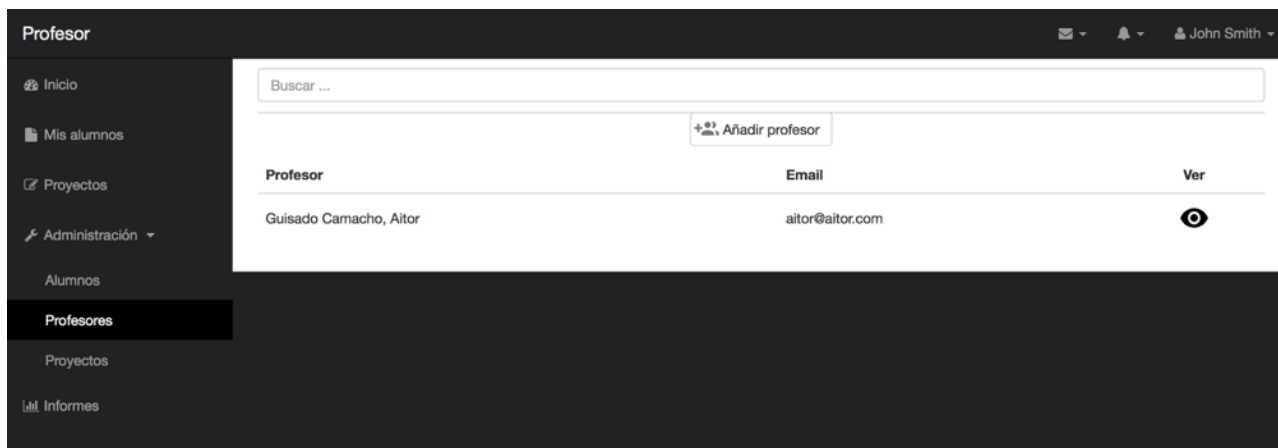
En el caso de los alumnos, se pueden ordenar o incluso buscar en el cuadro de búsqueda.



The screenshot shows the 'Profesor' management interface. On the left is a sidebar with a menu: Inicio, Mis alumnos, Proyectos, Administración (expanded), Alumnos (selected), Profesores, Proyectos, and Informes. The main area has a header with 'Ordenar por: Nombre' and a search bar 'Buscar ...'. Below the header is a table with columns: Alumno, DNI, Email, Ciclo, and Ver. The table contains 9 rows of student data. A '+ Añadir alumno' button is located above the table.

Alumno	DNI	Email	Ciclo	Ver
de la Cruz Armando, Fernando Javier	43241243A	fernando@lobo.com	Desarrollo de Aplicaciones Web	
de la Rua Gil, Mario	43243243A	mario@lobo.com	Administracion de Sistemas Informaticos en Red	
Guisado camacho, Aitor	03145853M	deko92@hotmail.com	Desarrollo de Aplicaciones Web	
lobezno lobeznoo, lobo	12312312M	lobo@lobo.com	Administracion de Sistemas Informaticos en Red	
Mouse Del Ratón, Mickey	43243234A	mickey@lobo.com	Administracion de Sistemas Informaticos en Red	
Ocaña Garcia, Adrián	12312332A	adri@lobo.com	Desarrollo de Aplicaciones Web	
Ocaña Garcia, Adrián	32323232e	adrian@lobo2.com	Administracion de Sistemas Informaticos en Red	
Zam Lobo, Sergio	12312312S	lobo@zam.com	Administracion de Sistemas Informaticos en Red	

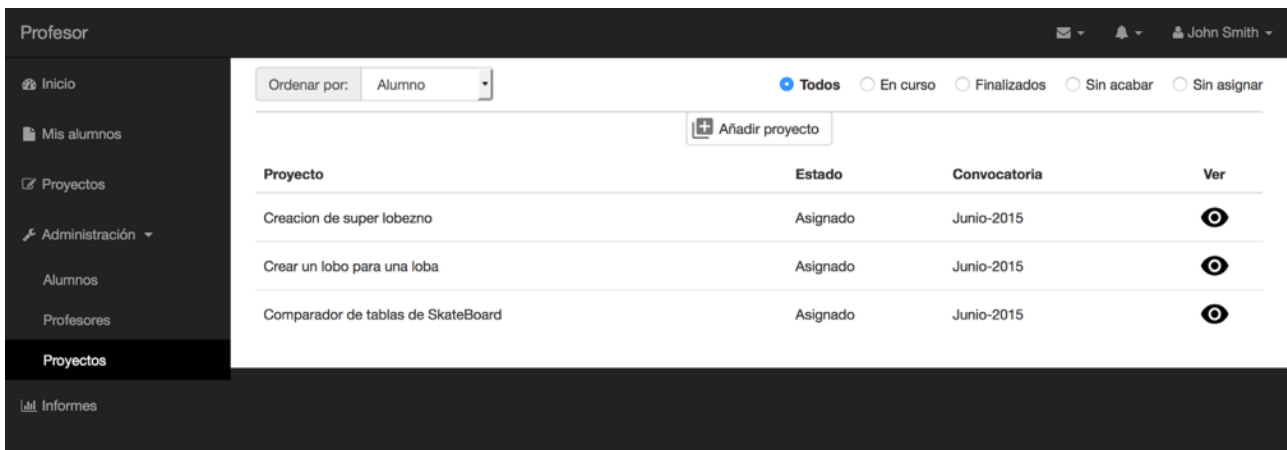
Para los profesores, se puede buscar un profesor para administrarlo.



The screenshot shows the 'Profesor' management interface with the 'Profesores' sub-menu selected. The sidebar menu is the same as the previous screenshot. The main area has a search bar 'Buscar ...' and a '+ Añadir profesor' button. Below the header is a table with columns: Profesor, Email, and Ver. The table contains 1 row of teacher data.

Profesor	Email	Ver
Guisado Camacho, Aitor	aitor@aitor.com	

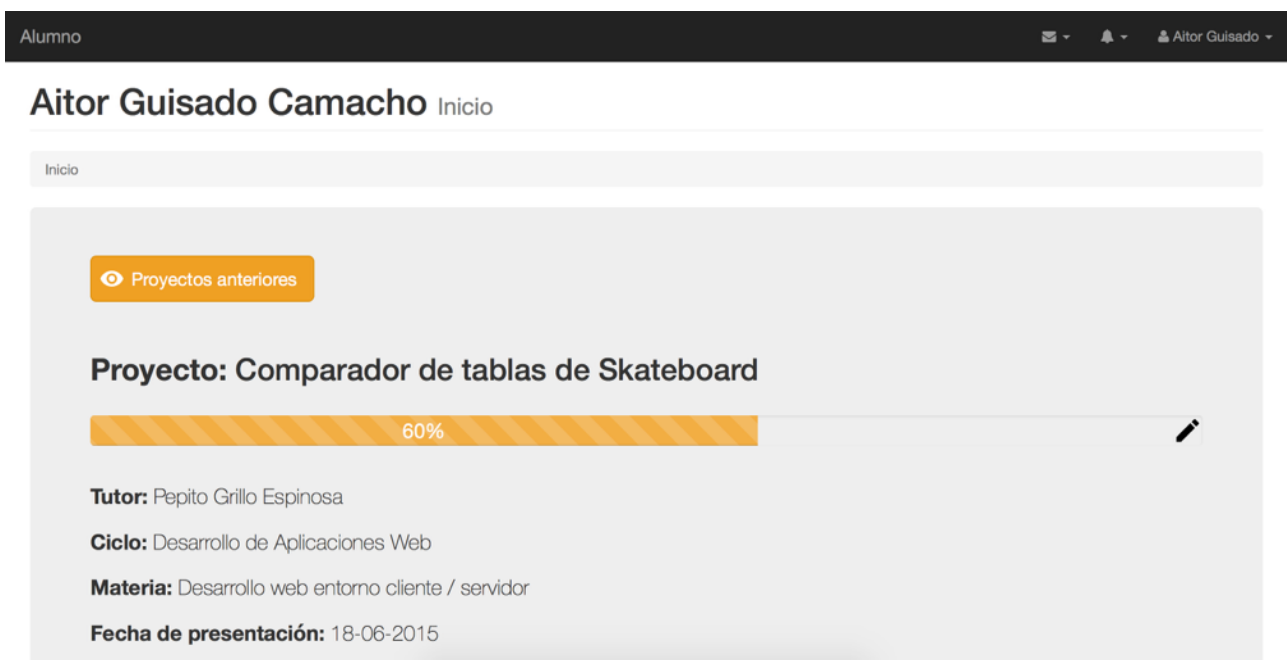
Por último, para los proyectos, se puede usar los mismos filtros y órdenes que en el apartado de proyectos general.



The screenshot shows the 'Profesor' (Professor) interface. On the left is a sidebar with navigation links: Inicio, Mis alumnos, Proyectos, Administración, Alumnos, Profesores, Proyectos (highlighted), and Informes. The main area has a header with 'Ordenar por: Alumno' and filter buttons: Todos (selected), En curso, Finalizados, Sin acabar, and Sin asignar. Below the filters is a table of projects.

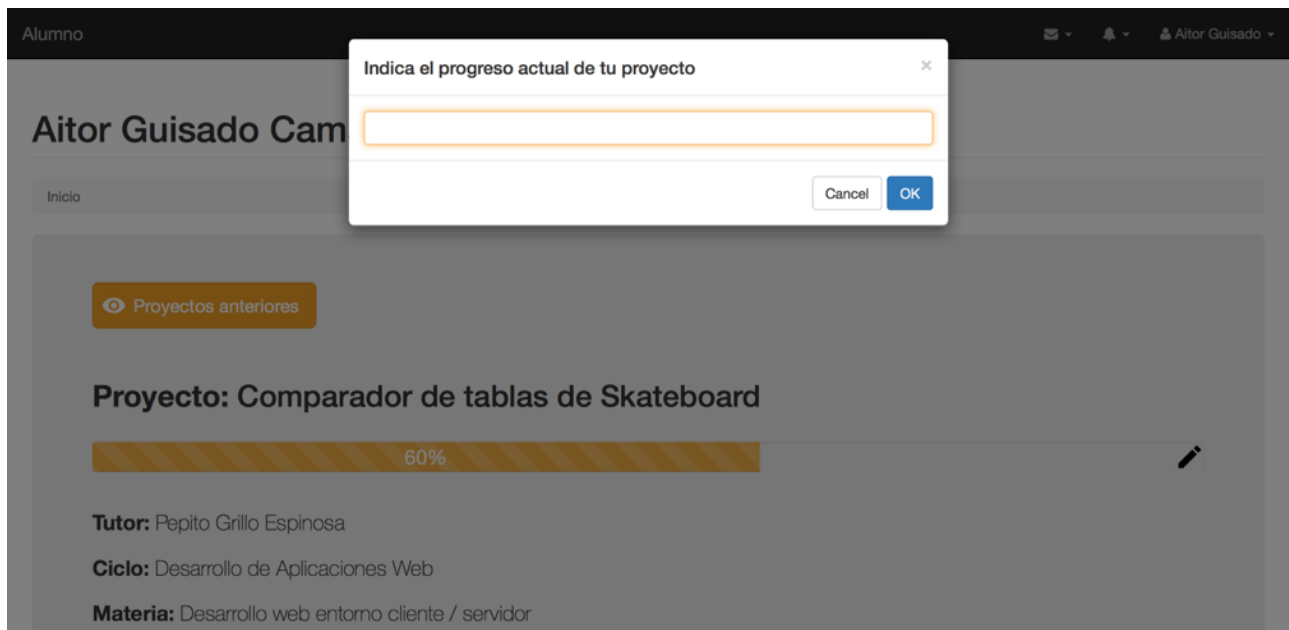
Proyecto	Estado	Convocatoria	Ver
Creacion de super lobezo	Asignado	Junio-2015	
Crear un lobo para una loba	Asignado	Junio-2015	
Comparador de tablas de SkateBoard	Asignado	Junio-2015	

La interfaz del alumno se vería de la siguiente forma:

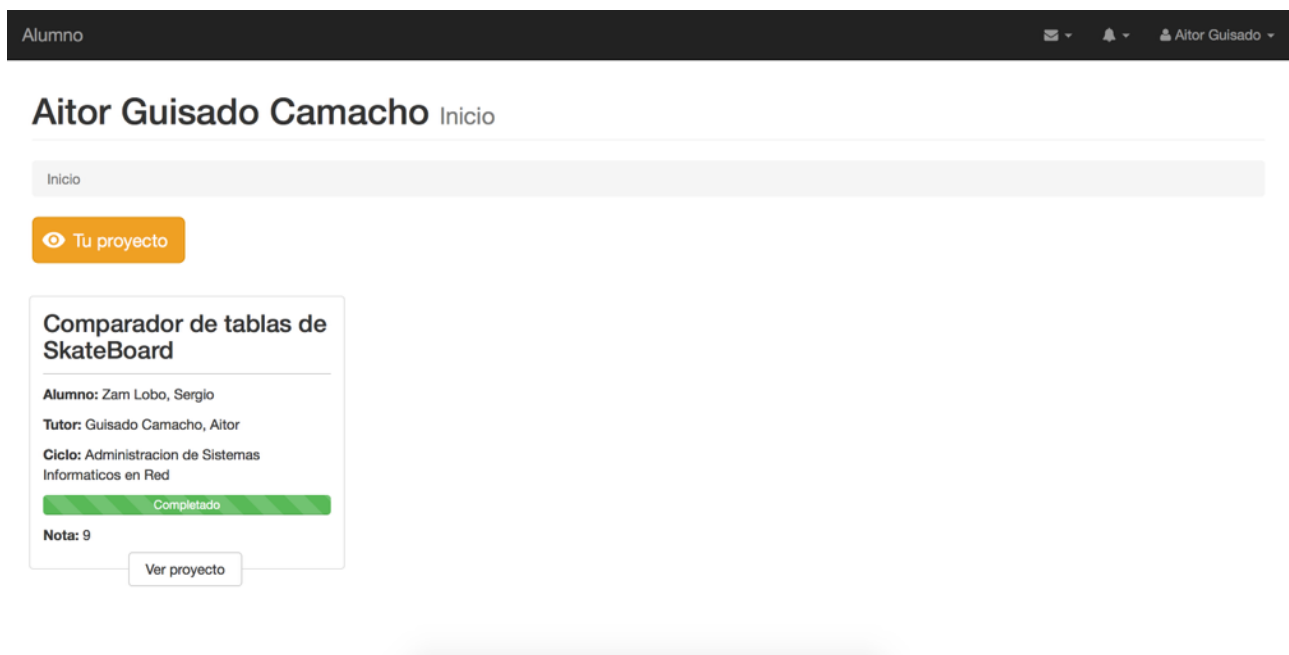


The screenshot shows the 'Alumno' (Student) interface. The header displays 'Aitor Guisado Camacho' and 'Inicio'. Below the header is a button 'Proyectos anteriores'. The main content area shows details for a project: 'Proyecto: Comparador de tablas de Skateboard'. Below the title is a progress bar at 60%. Further down are the following details: Tutor: Pepito Grillo Espinosa, Ciclo: Desarrollo de Aplicaciones Web, Materia: Desarrollo web entorno cliente / servidor, and Fecha de presentación: 18-06-2015.

El alumno, nada más ingresar a la aplicación, tiene la opción de actualizar el progreso que

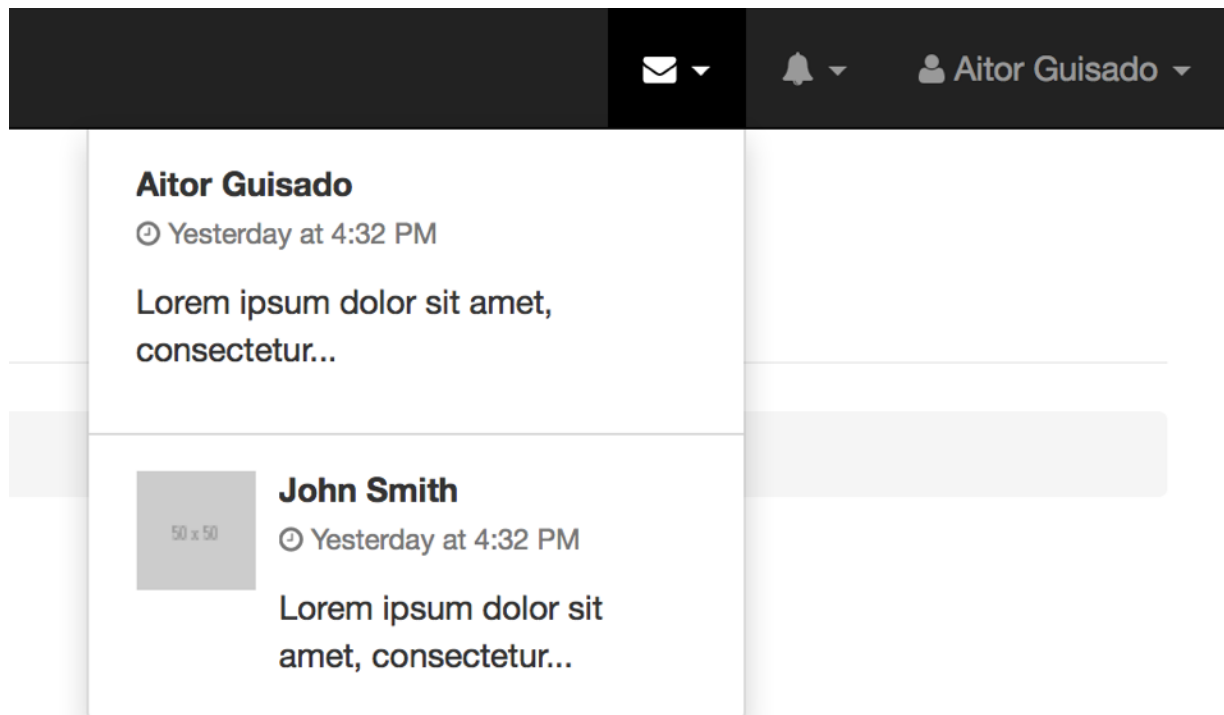


El alumno también puede ver proyectos finalizados en años anteriores, para que le puedan servir de referencia, o de alguna idea.

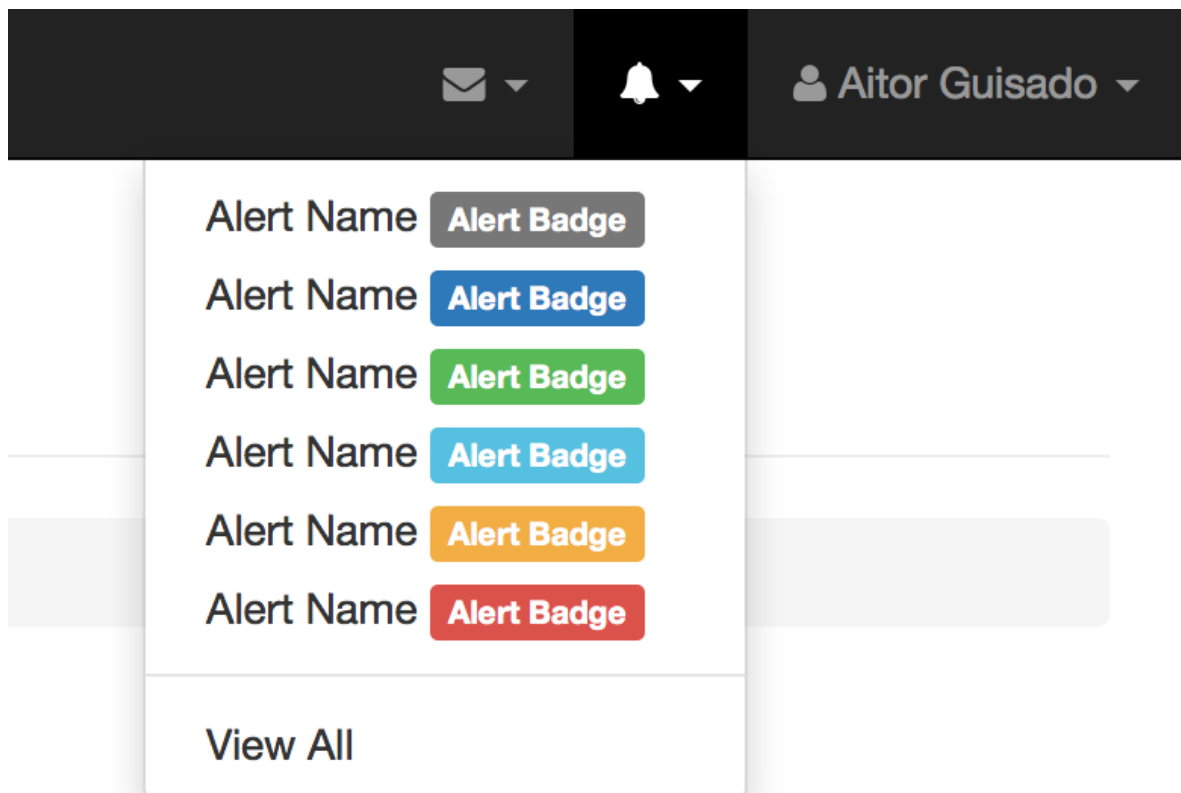


Por último, tanto los profesores como los alumnos tienen un menú superior, que les alerta de las notificaciones o de los mensajes nuevos recibidos, además de poder editar su perfil. Cada elemento superior se visualizaría de la siguiente forma:

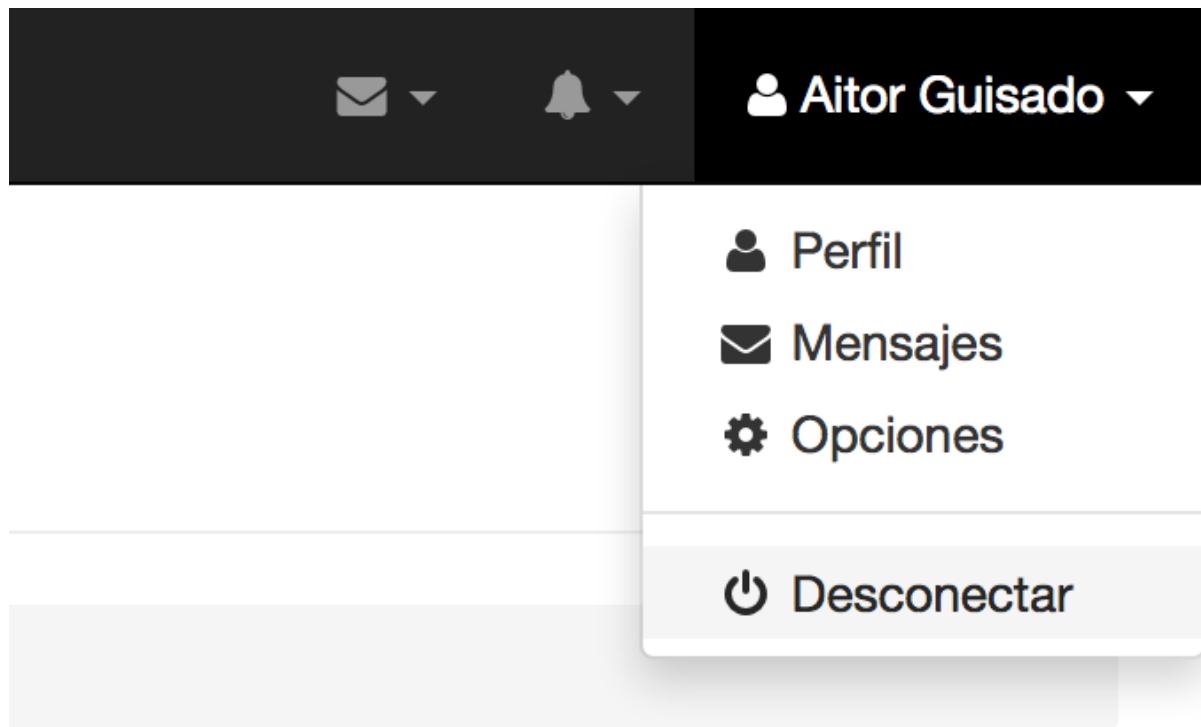
Mensajes:



Noitificaciones:



Perfil:



5. DESARROLLO Y REALIZACIÓN DEL PROYECTO.

5.1. Indicadores de calidad.

- El uso de un framework de estilos valorado muy positivamente de forma internacional, así como un diseño responsive para el buen uso de la aplicación en cualquier dispositivo.
- Se ha realizado una batería de pruebas sobre la aplicación, y ha sido probada durante un largo periodo de tiempo antes de su puesta en marcha.
- Reuniones y conversaciones con el cliente directo de la aplicación, asegurando la mejor adaptación posible a sus necesidades.
- Se ha realizado un diseño simple e intuitivo.
- Las tecnologías usadas son tecnologías modernas usadas actualmente, que asegura un correcto mantenimiento, así como el uso del Modelo Vista-Controlador, asegurando una calidad de código actualizable y sin dificultades de mantenimiento.

5.2. Batería de pruebas y solución de incidencias.

Se ha realizado una serie de pruebas para detectar posibles errores en la aplicación.

Inicialmente, se han comprobado todos los formularios de la aplicación, de forma que se han validado a través de HTML5 y de JavaScript para evitar los posibles errores. Además, se ha modificado la forma en la que se inserta la fecha, para evitar posibles errores en el campo “date”, cogido erróneamente por Firefox.

Tras la segunda batería de pruebas, se ha corregido el campo de subida de documentación de forma que no acepte archivos “.exe” que pueden ser malignos para el servidor que contiene la aplicación.

Se ha agregado el campo de sesión “rol” para evitar la entrada de los alumnos en un panel de administración cambiando funciones del script.

Tras la última fase de pruebas, la aplicación muestra un correcto funcionamiento.

5.4 Evaluación y seguimiento de actividades.

Se han ido evaluando las actividades y se ha producido un seguimiento con el demandante de la aplicación de forma que:

- En las primeras entrevistas se ha fijado el objetivo del proyecto, y se han decidido las funcionalidades del mismo, elaborando un documento de objetivos.
- Durante la elaboración del proyecto, se ha hecho varias visitas al cliente con presentaciones del estado actual, y se ha probado el funcionamiento y llegado a acuerdos de la interfaz de usuario, para alcanzar un desarrollo intuitivo y sencillo.
- Al finalizar el diseño y el funcionamiento, se vuelve a citar al cliente para comprobar y validar el diseño y la usabilidad de la aplicación, realizando pequeños cambios y validando la aplicación.

6. IMPLANTACIÓN DEL PROYECTO

6.1. Plan de implantación.

Para los centros que ya tengan un gestor de proyectos, y deba ser sustituido por este, se pasara un archivo excel modelo con las tablas que deben tener en el sistema de bases de datos para funcionar con este gestor.

Se pedirá al centro que devuelva el mismo excel con sus datos actuales, y el equipo de ShareProject se encargará de migrar sus datos al nuevo sistema, para entregarles la aplicación completamente funcional y con sus datos actuales.

En los centros en los que la base de datos tenga la misma estructura, se deberá acceder a un archivo que se facilitará al administrador, y se indicarán en el manual de instalación, para que cambie las variables globales de conexión a la base de datos y funcione automáticamente toda la aplicación.

6.2. Manual de instalación.

Esta aplicación corre sobre un servidor Apache, y una base de datos MySQL vamos a ver como instalar este paquete instalando XAMP:

6.2.1. Instalación Servidor en Windows.

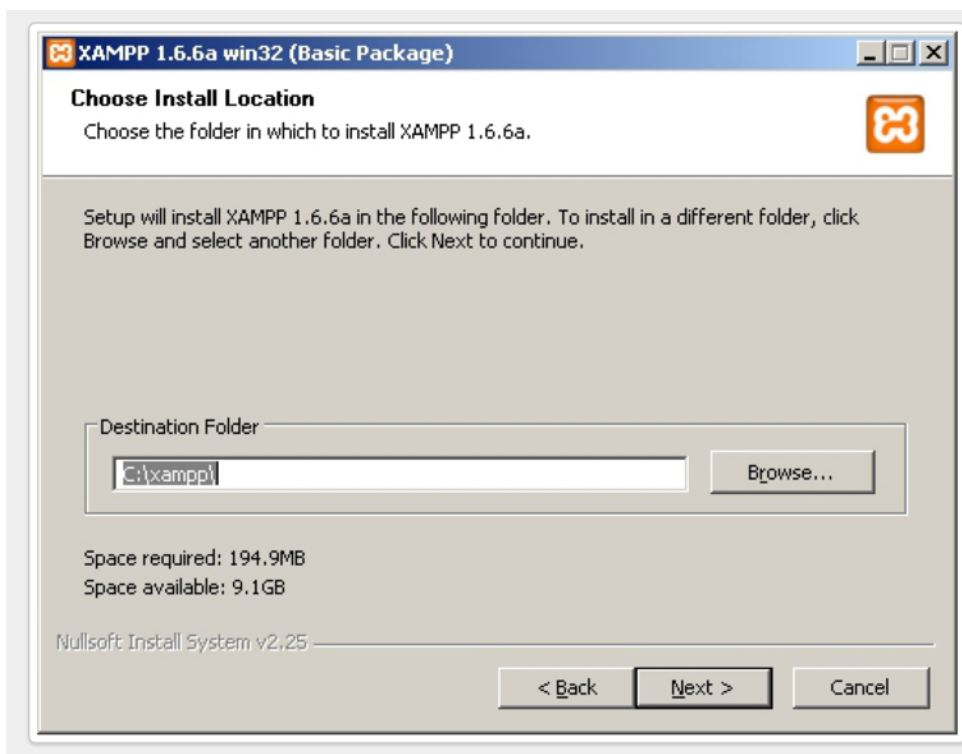
Descargamos el XAMP y lo ejecutamos



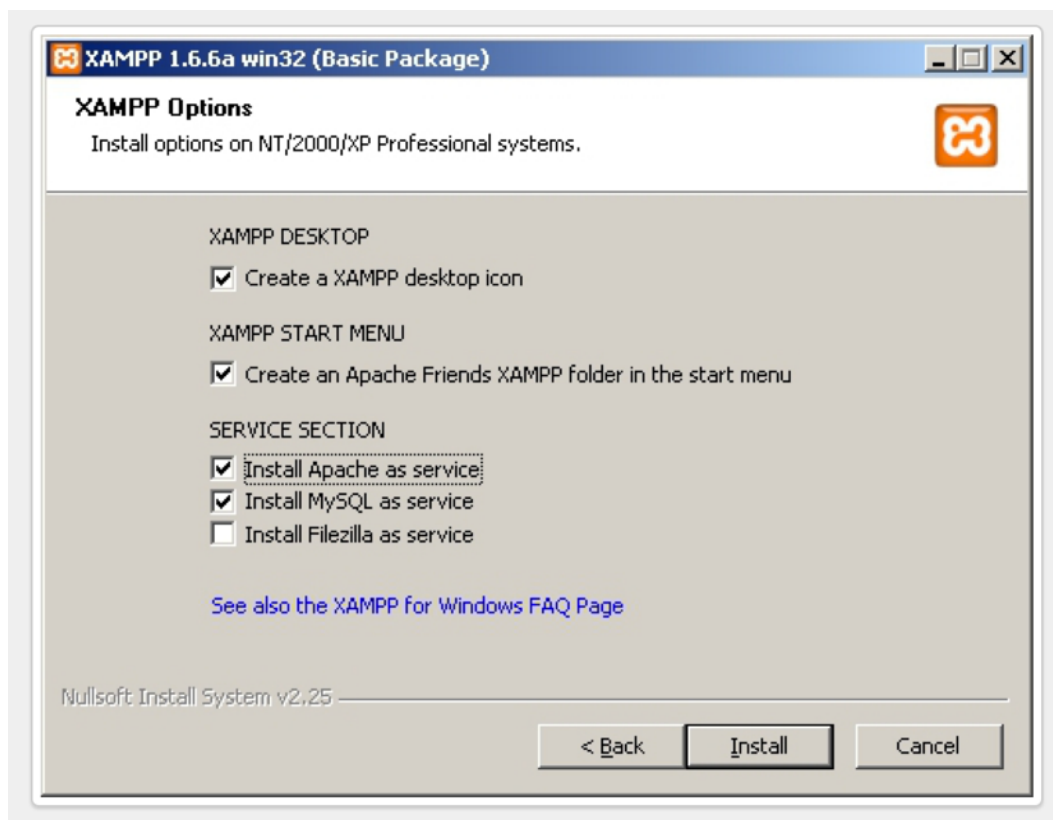
Seleccionamos Español, y damos a OK. Comenzará la instalación y pulsamos en Next:



En la ruta de instalación, ponemos C:/xampp:



En el siguiente paso nos aseguramos de que estén seleccionados Install apache as service y Install MySQL as service.



Pinchamos en instalar, y esperamos a que finalice la instalación.

6.2.2. Instalación Servidor en Linux.

Para instalar Apache y MySQL, utilizamos el paquete LAMP de linux:

Primero, hay que instalar Apache:

```
sudo apt-get install apache2
```

Ahora se debe instalar PHP:

```
sudo apt-get install php5 libapache2-mod-php5 php5-cli php5-mysql
```

Y posterior a la instalación de PHP, se debe instalar MySQL:

```
sudo apt-get install mysql-server mysql-client libmysqlclient-dev
```

Durante la instalación, solicitará una contraseña para el usuario root.

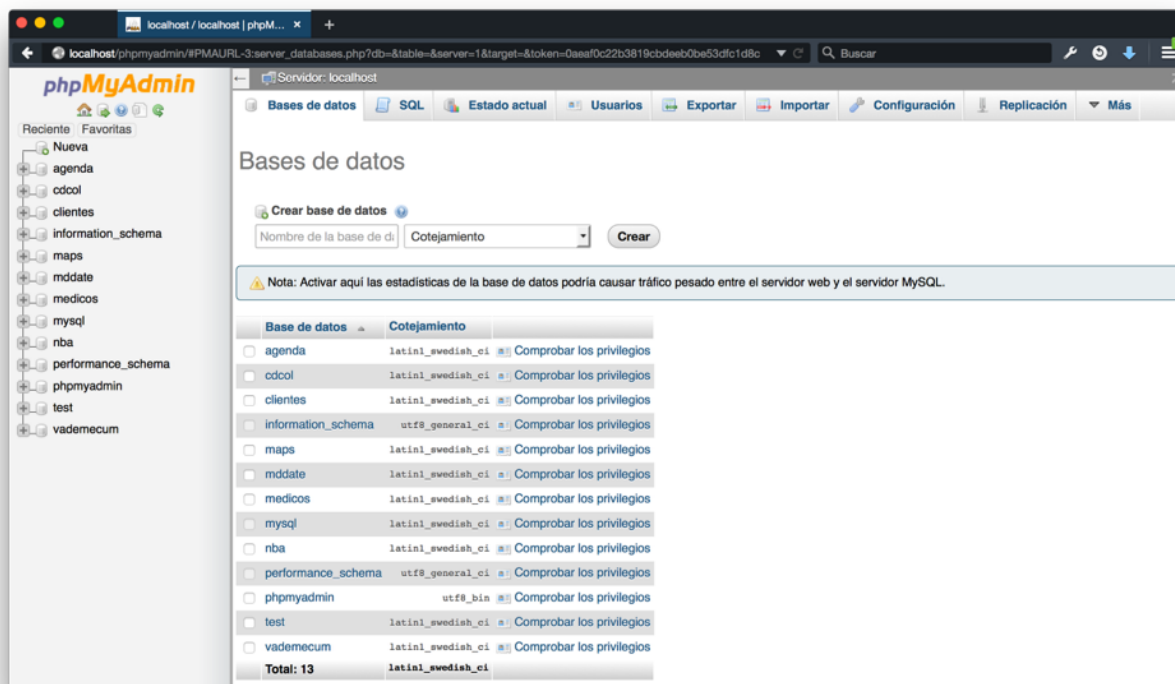
Finalmente, se instala phpMyAdmin

```
sudo apt-get install phpmyadmin
```

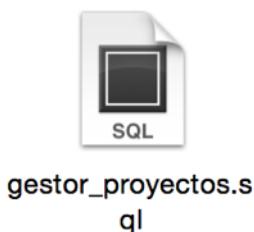
Para que todos los cambios resulten, reinicia la máquina.

6.2.3. Instalación de la aplicación.

El primer paso es ingresar en phpMyAdmin, y crear una nueva base de datos llamada gestor_proyectos (en caso de tener ya una base de datos completa con la estructura de la aplicación, este paso no sería necesarios).



Una vez creada la base de datos, se selecciona. Haz click en importar, y busca dentro de la aplicación, en la carpeta sql, el fichero gestor_proyectos.sql



Y pinchamos en importar.

El siguiente paso es mover la carpeta raíz de la aplicación al servidor.

- En Windows, debemos irnos a C:/xamp/htdocs y mover ahí la carpeta raíz de la aplicación web.
- En linux, el directorio es /var/www. Primero se debe proporcionar a la carpeta los privilegios adecuados:

```
sudo chmod -R 775 /var/www
```

6.3. Manual de usuario.

Para acceder a la aplicación, el primer acceso debe ser con el usuario admin, y la contraseña admin.

Desde el menú superior derecho, se puede acceder para cambiar la contraseña del usuario admin.

Para empezar a añadir profesores, se accede al menú de profesores, y se pulsa sobre el botón Añadir profesor.

Cada profesor añadido, se indica un usuario y contraseña que más tarde el propio profesor puede cambiar para su comodidad.

Al añadir alumnos, el usuario del alumno será el DNI y la contraseña será la que el tutor elija. El propio alumno también podrá cambiar su contraseña una vez accedido a la aplicación.

Para añadir proyectos, se debe acceder al menú de proyectos dentro de administración, y hacer click sobre el botón Añadir proyectos. Dentro del menú de añadir proyectos se puede seleccionar el alumno al que está asignado y el tutor al que está asignado, si se quiere asignar directamente. En caso de no asignar, se pueden asignar proyectos posteriormente, dentro de la administración de proyectos, haciendo click sobre el botón Asignar proyecto.

La asignación de dicho proyecto puede ser modificada en cualquier momento accediendo a la edición de dicho proyecto, por medio del panel de administración, pulsando en el botón Ver del proyecto que se quiere modificar, pulsando en el botón de editar con forma de lápiz, y modificando el alumno.

El ciclo del proyecto cambiará automáticamente dependiendo del alumno.

La fecha de asignación puede ser por defecto la fecha actual, pulsando en el checkbox “fecha actual”.

7. CONCLUSIONES FINALES

7.1. Evaluación del proyecto. Objetivos alcanzados y no alcanzados.

Se han logrado todos los objetivos propuestos para el proyecto, y además se han añadido mejoras significativas, que aumentan la sencillez de la aplicación y además agregan funciones extras que en un principio no se habían contemplado.

Se ha logrado una administración completa de proyectos, alumnos y profesores, así como una limpia y fluida comunicación entre ellos.

7.2. Propuesta de mejora.

Se ha propuesto como posible mejora la implementación de una aplicación para iOS y Android que contemple la posibilidad de acceder a la misma base de datos y gestión que la aplicación web, con la mejora de un sistema de notificaciones cada vez que se produce una nueva tarea como:

- Un alumno ha subido un nuevo archivo.
- El profesor ha contestado a tu mensaje.

- Se ha actualizado el progreso del proyecto.
- Notificación para el alumno: ¡Mañana expones el proyecto!.
- Etc..