Contenido

[1. RESUMEN EJECUTIVO 1](#_Toc511122306)

[2. DESCRIPCIÓN DEL PROBLEMA 1](#_Toc511122307)

[3. REQUERIMIENTOS 1](#_Toc511122308)

[4. ESQUEMA FUNCIONAL 1](#_Toc511122309)

[5. DISEÑO DE LA BASE DE DATOS 1](#_Toc511122310)

[6. DISEÑO FÍSICO DE LA BASE DE DATOS 1](#_Toc511122311)

[7. CASOS DE PRUEBA 1](#_Toc511122312)

[8. MANUAL DE USO 1](#_Toc511122313)

# RESUMEN EJECUTIVO

## TRATAMIENTO DE ENTIDADES

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Create** | **Read** | **Update** | **Delete** |
| **Actividades** | X | X | X |  |
| **Administradores** |  | X |  |  |
| **Alumnos** | X | X | X | X |
| **Profesores** | X | X | X | X |
| **Asignaturas** | X | X | X | X |
| **Aulas** | X | X | X | X |
| **Entrega** | X | X |  |  |
| **Material** | X | X | X | X |
| **Mensaje** | X | X |  |  |

## PUNTOS DONDE SE MANIPULAN ENTIDADES

Toda interaccion con la capa de acceso a datos se realiza a través del servicio web. Únicamente la clase DAOImpl del proyecto web emplea a los Jersey Clients para realizar peticiones al servicio. De esta manera, se consigue un menor acoplamiento entre la aplicación web y el acceso a datos. No obstante, en este apartado se especifica la Action de Struts o página JSP (en caso de llamadas directas desde el cliente a través de JavaScript) donde se realiza la llamada para gestionar las entidades.

1. Actividades:
   1. CREAR:
      1. ActividadActions
   2. LEER:
      1. ActividadActions
      2. AsignaturasActions
   3. ACTUALIZAR:
      1. ActividadActions
2. Administradores:
   1. LEER:
      1. LoginAction
3. Alumnos:
   1. CREAR:
      1. CargarBorrarAlumnosAction
      2. AlumnosCRUDActions
   2. LEER:
      1. AsignaturaActions
      2. MessagesActions
      3. AlumnosCRUDActions
      4. AsignaturasCRUDActions
      5. CargarBorrarAlumnosAction
   3. ACTUALIZAR:
      1. AlumnosCRUDActions
   4. BORRAR:
      1. CargarBorrarAlumnosActions
      2. Admin\_alumnos.jsp
4. Profesores:
   1. CREAR:
      1. ProfesoresCRUDActions
   2. LEER:
      1. ProfesoresCRUDActions
      2. CargarProfesoresActions
      3. AsignaturaActions
   3. ACTUALIZAR
      1. ProfesoresCRUDActions
   4. BORRAR:
      1. Admin\_profesores.jsp
5. Asignaturas:
   1. CREAR:
      1. AsignaturasCRUDActions
   2. LEER:
      1. AsignaturasCRUDActions
      2. CargarAsignaturasActions
      3. AsignaturaActions
      4. CargarBorrarAlumnos
   3. ACTUALIZAR
      1. AsignaturasCRUDActions
   4. BORRAR:
      1. Admin\_asignaturas.jsp
6. Aulas:
   1. CREAR:
      1. AulasCRUDActions
   2. LEER:
      1. AulasCRUDActions
      2. CargarAulasActions
      3. AsignaturasCRUDActions
   3. ACTUALIZAR
      1. AulasCRUDActions
   4. BORRAR:
      1. Admin\_aulas.jsp
7. Entrega:
   1. CREAR:
      1. EntregaActions
   2. LEER:
      1. EntregaActions
   3. ACTUALIZAR
      1. EntregaActions
8. Material:
   1. CREAR:
      1. MaterialActions
   2. LEER:
      1. MaterialActions
      2. AsignaturasCRUDActions
   3. ACTUALIZAR
      1. MaterialActions
   4. BORRAR:
      1. MaterialActions
9. Mensajes:
   1. CREAR:
      1. MessagesActions
   2. LEER:
      1. Messages.jsp

## APORTACIONES EXTRAORDINARIAS

#### DISEÑO RESPONSIVE

Para logar un diseño completamente *responsive*, atractivo y que se adapte a cualquier dispositivo, se ha empleado la plantilla W3.CSS ofrecida por W3School[[1]](#footnote-1). En concreto, la aplicación se mostrará en modo compacto si la anchura de la ventana es inferior a 800 píxeles.

#### CONTROL DE VERSIONES

El equipo ha empleado el sistema de control de versiones Git para controlar y compartir el trabajo, alojando el proyecto en un repositorio remoto de GitHub. El repositorio puede consultarse en: <https://github.com/mridpin/IT_Group07_Project>

# DESCRIPCIÓN DEL PROBLEMA

Para que los alumnos de las universidades públicas cuenten con una educación asequible y de calidad, es esencial que estas ofrezcan una aplicación web para dinamizar y descentralizar la impartición de esta educación. Por este motivo, se va a desarrollar una aplicación web que permita tanto a alumnos como profesores acceder de manera remota desde sus propios dispositivos a todo lo necesario para dar clases, compartir material, y comunicar resultados y actividades.

Con este fin, la aplicación contará con componentes que permita la gestión de alumnos y profesores por parte de usuarios administradores. Los alumnos podrán comunicarse mediante mensajes privados con otros alumnos. Los profesores necesitan subir contenido para sus asignaturas, así como proponer y calificar actividades. Los alumnos podrán descargar este contenido y subir sus respuestas a las actividades, que deberán ser calificadas.

El diseño de la aplicación debe ser minimalista e intuitivo, así como capaz de adaptarse a diferentes dispositivos. No se contempla la posibilidad de modificar el *layout* de los componentes ni la personalización de opciones avanzadas. Finalmente, con el propósito de agilizar sus tareas, los administradores pueden crear grupos completos de alumnos mediante un fichero CSV.

# REQUERIMIENTOS

## REQUISITOS FUNCIONALES

|  |  |  |
| --- | --- | --- |
| **Código** | **Requisito** | **Descripción** |
| **RF01** | Login | Los alumnos y profesores deben poder iniciar sesión y acceder a funcionalidades personalizadas según las asignaturas en las que están matriculados |
| **RF02** | Gestión de alumnos | El sistema permite gestionar los alumnos de la universidad. El administrador podrá darlos de alta y los alumnos podrán cambiar su información personal |
| **RF03** | Gestión de profesores | El sistema permite gestionar los profesores de la universidad. El administrador podrá darlos de alta y los profesores podrán cambiar su información personal |
| **RF04** | Gestión de aulas | El administrador podrá gestionar las aulas disponibles para clases y otras actividades, además asignará aulas a las asignaturas. También los profesores podrán solicitar un traslado de aula para las asignaturas que impartan. |
| **RF05** | Gestión de asignaturas | El administrador podrá gestionar las asignaturas según la oferta de la universidad y gestionar los alumnos matriculados |
| **RF06** | Gestión de actividades | Los profesores podrán ofrecer actividades y los alumnos subir sus trabajos. Los profesores podrán calificar las actividades y los alumnos consultar sus calificaciones |
| **RF07** | Gestión de material | Los profesores podrán subir material para los alumnos de sus asignaturas, y estos descargarlo |
| **RF08** | Envío de mensajes | Los alumnos podrán intercambiar mensajes |

## REQUISITOS NO FUNCIONALES

|  |  |  |
| --- | --- | --- |
| **Código** | **Requisito** | **Descripción** |
| **RNF01** | Diseño *Responsive* | La aplicación debe adaptarse a distintos tamaños de pantalla y dispositivo. Se dará prioridad a los dispositivos con relación de aspecto 16:9 y 9:16, es decir, monitores de ordenador tradicional y dispositivos móviles en vertical. |
| **RNF02** | Seguridad | * El sistema debe estar protegido frente a ataques de inyección SQL. * Los usuarios no deben poder acceder a funcionalidades que no les corresponden * La información sensible debe estar protegida * Algunos elementos, como alumnos y mensajes, no se podrán borrar, solo ocultar |

## ACTORES Y DIAGRAMA DE CASOS DE USO

Los actores de la aplicación quedan recogidos en la siguiente tabla, en orden de aparición en el diagrama posterior:

|  |  |  |
| --- | --- | --- |
| **Código** | **Actor** | **Descripción** |
| **ACT01** | Todos | Representa una combinación de todos los usuarios que van a participar en la aplicación. Solo se usa para representar los casos de uso login y logout. |
| **ACT02** | Administrador | Representa a un empleado de la universidad que tiene acceso al back-end para gestionar los componentes de la aplicación a cuya información los ACT03-Usuario solo tienen acceso de consulta. |
| **ACT03** | Usuario | Representa a alumnos y profesores. Solo tiene acceso al front-end de la aplicación. |
| **ACT04** | Profesor | Representa un usuario profesor de la universidad |
| **ACT05** | Alumno | Representa un usuario alumno de la universidad |

Los casos de uso de la aplicación se recogen en los siguientes diagramas, agrupados por actores:









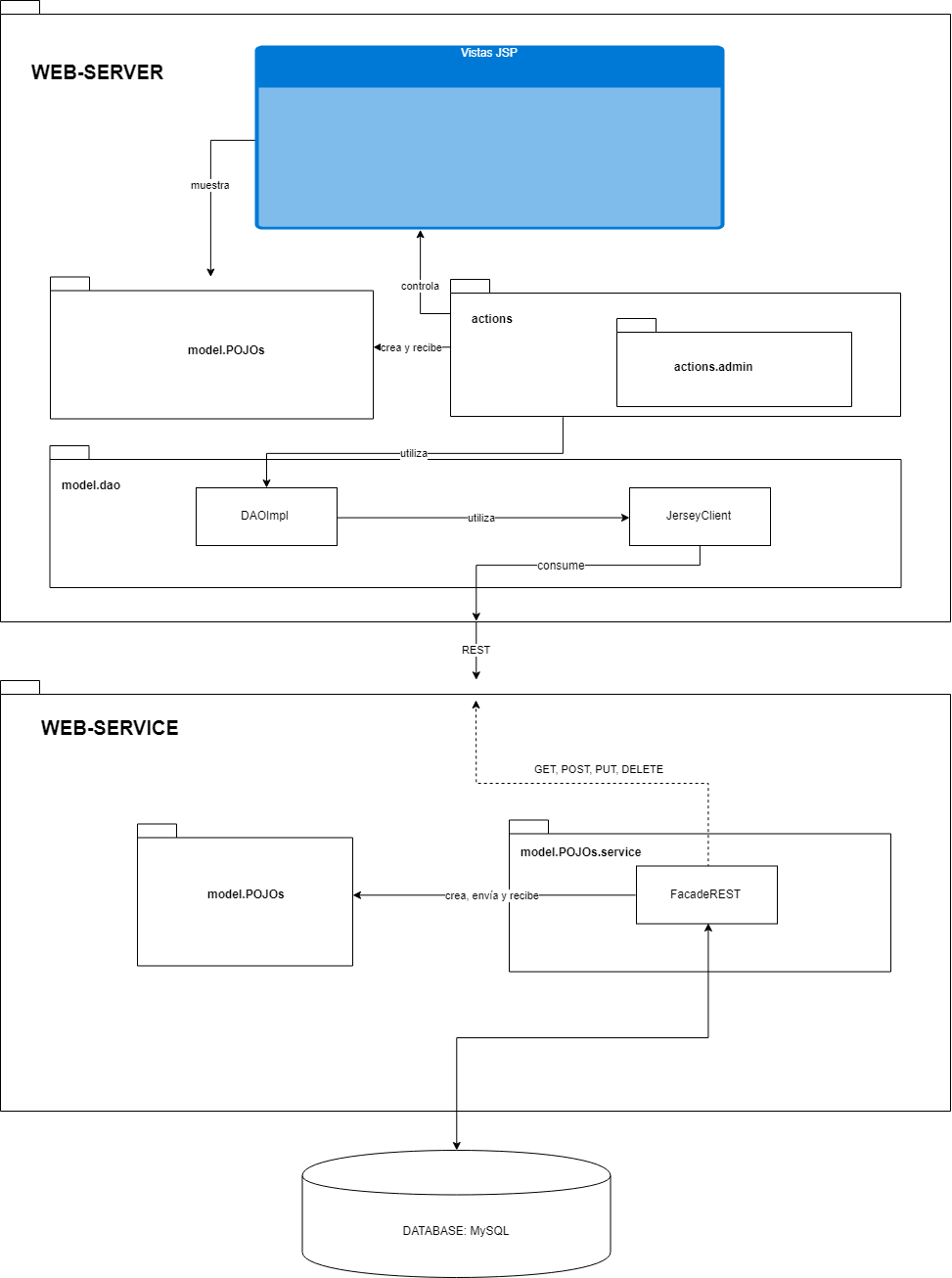
## RESTRICCIONES FUNCIONALES

Se aplican las siguientes restricciones:

* Borrado de información: Para no provocar borrados en cascada que causen situaciones imprevistas, y fallos en la integridad de la aplicación, no se permite el borrado ni la modificación de los mensajes enviados.
* No se contempla la gestión de administradores. Estos obtendrán sus credenciales directamente de la base de datos, y su información no podrá ser modificada mediante el uso de la aplicación.

# ESQUEMA FUNCIONAL

El esquema funcional de la aplicación es el siguiente:



# DISEÑO DE LA BASE DE DATOS



Usuario: Representa a una persona que hace uso de la aplicación. Existen dos tipos, alumnos y profesores. Además, este es gestionado por los administradores.

Alumno: Representa a un alumno de la universidad que utiliza la aplicación y estará matriculado en una serie de asignatura. El alumno puede enviar y recibir mensajes, realizar entregas de las actividades que han creado los profesores.

Profesor: Representa a un profesor de la universidad que hace uso de la aplicación. El profesor imparte una serie de asignaturas y gestiona una serie de actividades.

Administrador: Representa a un administrador que gestiona los usuarios, las aulas y las asignaturas.

Mensaje: Representa los mensajes que envía un alumno que van dirigidos a otros alumnos.

Entrega: Los alumnos realizan una entrega por cada una de las actividades de la asignatura.

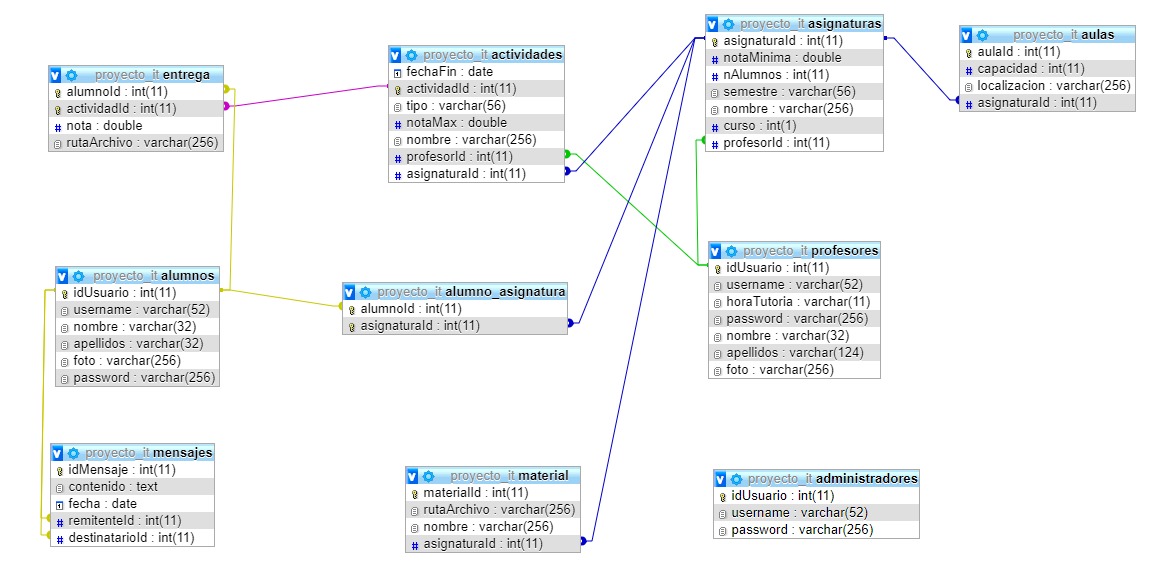
Aula: Las asignaturas se imparten en un aula y son gestionadas por los administradores.

Asignatura: Representa la clase que imparte un profesor en la que están matriculados una serie de alumnos. Además, tiene una serie de actividades y una serie de material.

Actividad: Las asignaturas pueden tener una serie de actividades gestionadas por un profesor, donde estas son evaluadas mediante una entrega realizada por un alumno.

Material: Representa el material perteneciente a una asignatura que se pueden descargar los alumnos matriculados.

# DISEÑO FÍSICO DE LA BASE DE DATOS



Cada entidad del diseño entidad-relación tiene su tabla, y cada atributo una columna. Se han relacionado de la siguiente manera:

1. Relaciones muchos-a-muchos: Se relacionan creando una tabla intermedia, las entidades con este tipo de relación son las siguientes:
2. alumnos y asignaturas se relacionan mediante la tabla alumno\_asignatura, que guarda un id de cada una de las dos entidades.
3. Relaciones uno-a-muchos y muchos-a-uno:
4. alumnos y mensajes: Un alumno puede enviar muchos mensajes y muchos mensajes van dirigidos a un alumno, por lo que cada mensaje tiene unas claves foráneas que representan los identificadores del remitente y el destinatario.
5. profesores y actividades: Un profesor puede crear varias actividades, por lo que cada actividad tiene una clave foránea que representa el id del profesor que la creó.
6. alumnos y entrega: Un alumno puede realizar varias entregas de varias actividades, por lo que entrega tiene como clave foránea el identificador del alumno que realizó la entrega.
7. actividades y entrega: Cada actividad tiene muchas entregas asociadas, por lo que cada entrega tiene como clave foránea el identificador de la actividad asociada.
8. asignaturas y material: Cada asignatura tiene una serie de materiales asociados, así que cada material tiene una clave foránea que representa el identificador de la asignatura del material.
9. asignaturas y actividades: Las asignaturas tienen una serie de actividades asociadas, por lo que cada actividad tiene una clave foránea que indica la asignatura asociada.

# CASOS DE PRUEBA

# MANUAL DE USO

La aplicación incorpora funcionalidades completamente distintas según si se trata de un usuario docente (alumnos y profesores) o administradores. En cualquier caso, el primer paso será loguearse con las credenciales, ya que la aplicación no ofrece ninguna funcionalidad si no se está logueado.

## ADMINISTRADORES

Es posible loguearse como administrador mediante las credenciales “admin”, “admin”. Los administradores son los encargados de registrar las entidades permanentes, es decir, aquellas que se crean al inicio del curso y no se borran ni modifican hasta el final.



Para gestionar las distintas entidades se navega empleando el menú lateral izquierdo. Podemos gestionar Aulas, Asignaturas y Profesores mediante el mismo procedimiento:

1. Hacer clic en el menú lateral para acceder a la gestión de entidades. Se cargarán todas las entidades guardadas y un formulario para su gestión.



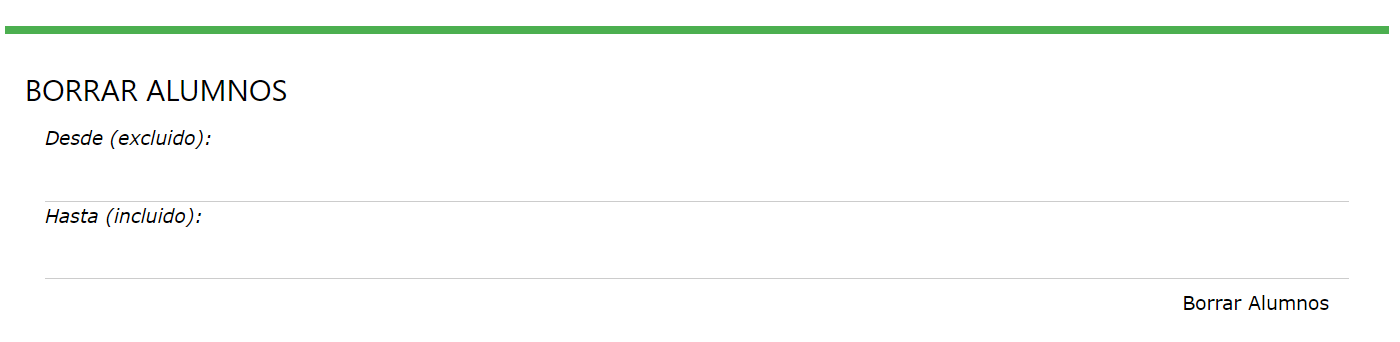
1. Para crear una nueva entidad, rellenar el formulario con los datos y hacer clic en “Crear”.
2. Para editar una entidad, hacer clic en la misma desde la tabla, se cargará su información en el formulario. Modificar los datos deseados y presionar “Editar”, o “Crear” para añadirla como nueva.



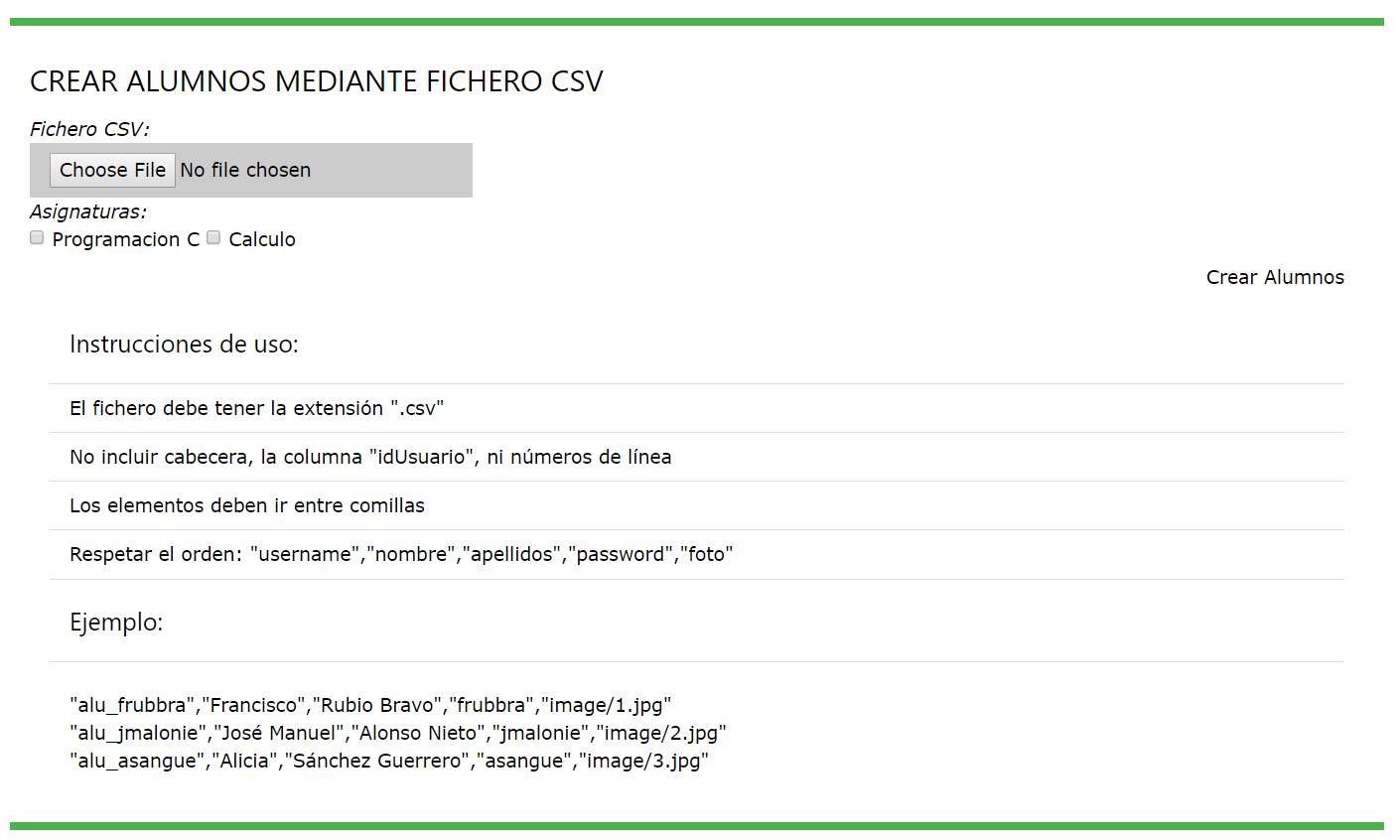
1. Para borrarla, presionar en el botón “x” a la derecha de la tabla en cada fila. Se borrará dicha entidad.

Para gestionar alumnos, se incluyen las funcionalidades anteriormente mencionadas y dos extras:

1. Para borrar varios alumnos a la vez, indicar en el formulario los ID desde y hasta donde se desea borrar y hacer clic en “Borrar Alumnos”



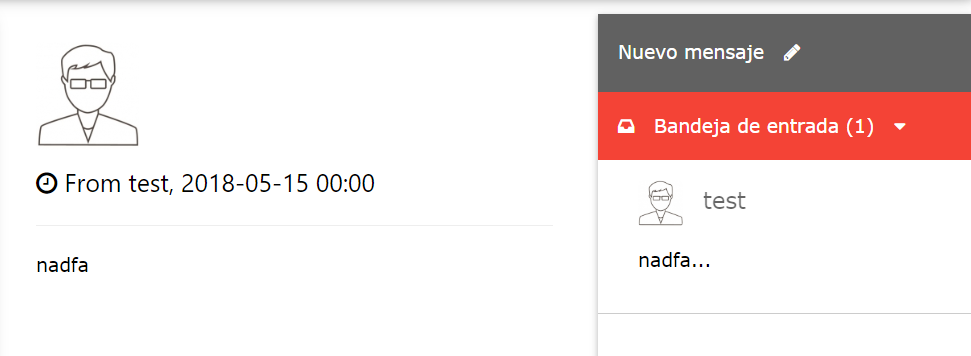
1. Se pueden añadir varios alumnos a la vez subiendo un fichero CSV que siga las instrucciones de formato indicadas en la página, e indicando las asignaturas en las que se desea matricularles.



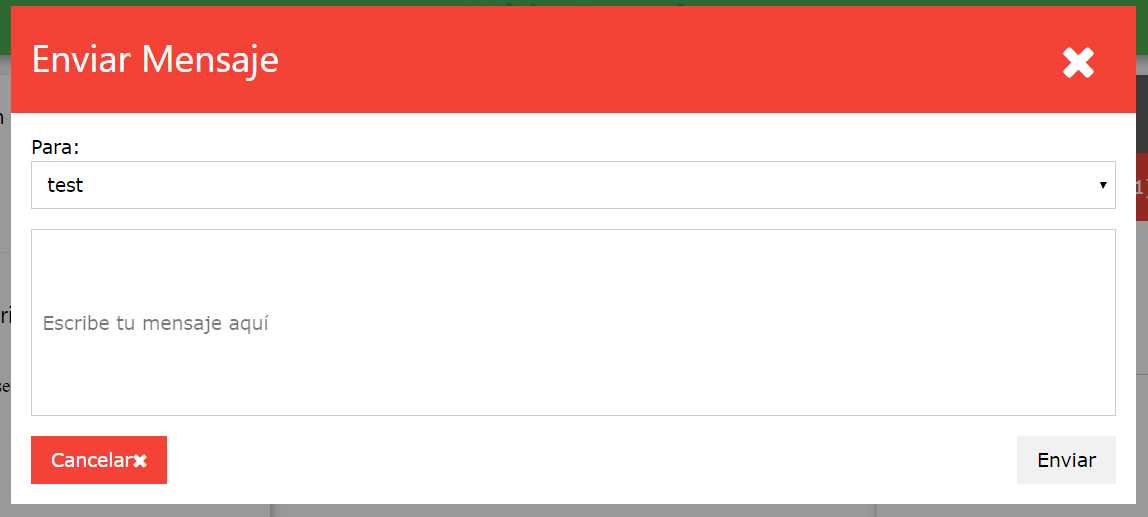
## USUARIOS DOCENTES

#### ALUMNOS

Se puede acceder al sistema de mensajería haciendo clic en “Mensajes” en la barra lateral izquierda. Aparecerá una barra lateral derecha con los últimos mensajes, que se pueden ampliar clicando en ellos.



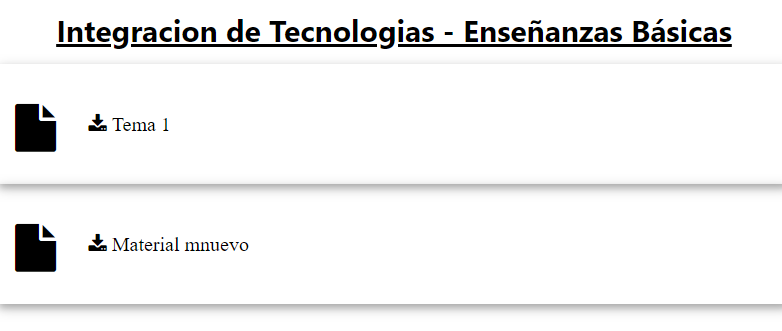
Para crear un mensaje nuevo, se hace clic en “Nuevo Mensaje”, se rellenan los campos y se hace clic en “Enviar”, mediante la ventana emergente.



Los alumnos pueden también acceder a las funcionalidades relacionadas con las asignaturas, haciendo clic en la asignatura deseada en la barra de navegación lateral.



Una vez en el menú de la asignatura, pueden acceder al material disponible haciendo clic en “Enseñanzas Básicas” o “Enseñanzas Prácticas”, y haciendo clic en el material deseado para descargarlo.



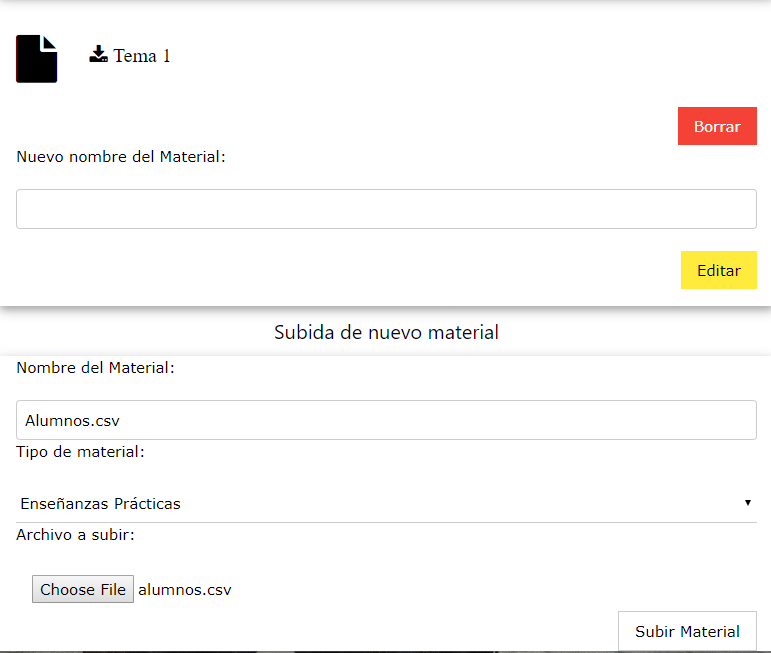
Finalmente, en el apartado “Próximas Tareas”, pueden consultar las tareas pendientes de la asignatura y subir ficheros para completarlas.

#### PROFESORES

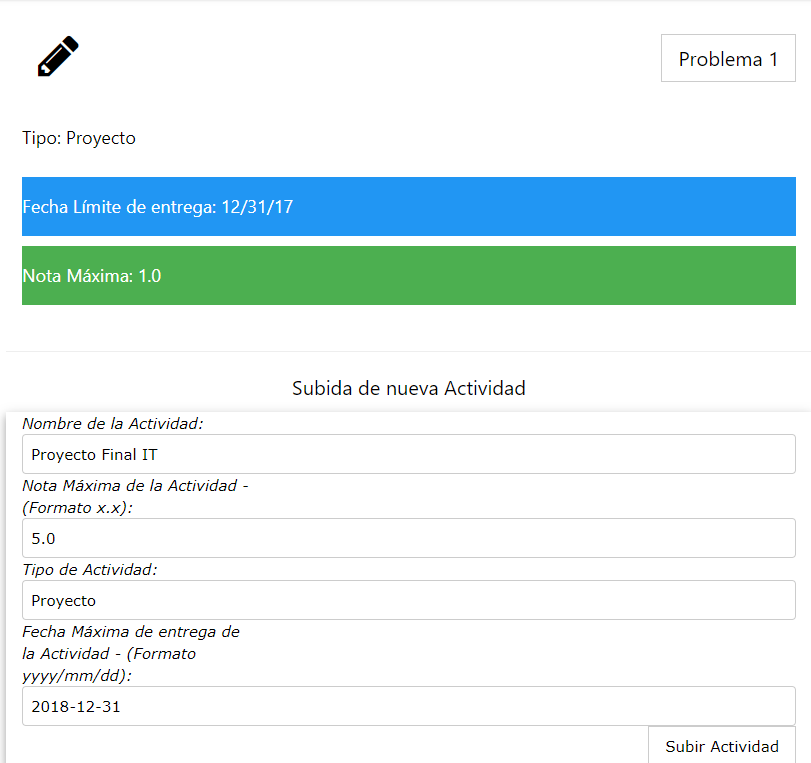
Al igual que los alumnos, los profesores pueden también acceder a las funcionalidades relacionadas con las asignaturas, haciendo clic en la asignatura deseada en la barra de navegación lateral. Se abrirá la vista de asignaturas para profesores, donde disponen de tres funcionalidades, además de consultar las próximas tareas igual que los alumnos.



1) Gestión de Material: Al acceder a esta ventana se ofrecen funcionalidades para borrar o editar material subido, así como para subir nuevo material mediante un formulario:



2) Gestionar tareas: Permite a los profesores consultar, crear, modificar y borrar tareas para la asignatura. Para modificar una actividad, se hace clic en el botón con su nombre a la derecha, lo que carga su información en el formulario. Para subir una actividad, se emplea el formulario etiquetado con tal fin.



3) Ver lista de alumnos: Se consulta un listado de los alumnos matriculados en la asignatura.

# APÉNDICE: APORTACIONES EXTRAORDINARIAS

1. <https://www.w3schools.com/w3css/default.asp> [↑](#footnote-ref-1)