

## **2695-1: Herramienta de generación de cuestionarios para la enseñanza**

**Memòria del Projecte Fi de**

Carrera

d'Enginyeria en Informàtica

realitzat per

Jordi Martin Gil

i dirigit per

Ramon Musach Pi

Bellaterra, 12 de Setembre de 2011

El sotasignat, Ramon Musach Pi

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

**CERTIFICA:**

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per en Jordi Martin Gil

I per tal que consti firma la present.

Signat: Ramon Musach Pi

Bellaterra, 12 de Setembre de 2011

## CONTENIDO

---

Introducción .....	1
Planificación.....	3
Objetivos.....	3
Estudio de viabilidad del proyecto .....	4
Planificación temporal del proyecto.....	5
Diseño .....	6
Idea principal .....	6
Entorno de la base de datos .....	8
Seguridad.....	8
Estructura página web.....	10
Acceso súperusuario.....	11
Implementación.....	13
Descripción .....	13
Creación de la Base de Datos .....	15
Página Web.....	16
Página de control de acceso .....	16
Página de control de súperusuario.....	17
Modificación de Nivel Y Materia .....	19
Inserción de usuarios.....	22
BORRADO DE usuarios.....	23
Base de Datos .....	23
Página Principal .....	24
Inserción de preguntas tipo test.....	25
Inserción de preguntas descriptivas.....	26
Consulta de preguntas.....	27
Modulo para el formato pdf.....	31
Pruebas y resultados .....	32
Pruebas.....	32
Pruebas del módulo de control de acceso .....	32
Pruebas del módulo de control súperusuario .....	32
Pruebas del módulo de modificación de nivel y materia .....	33
Pruebas del módulo principal.....	33
Pruebas del módulo de inserción de preguntas.....	33

Pruebas del módulo de consulta de preguntas.....	34
Resultados .....	35
Conclusiones.....	36
Objetivos cumplidos .....	36
Conclusiones obtenidas.....	36
Problemas surgidos .....	37
Planificación final.....	37
Posibles ampliaciones.....	38
Bibliografía.....	39
Índice de Imágenes.....	40
TABLAS DE LA BBDD .....	41

## INTRODUCCIÓN

---

---

El proyecto que se describe a continuación se centra en la realización de un nuevo software para la comunidad educativa, en especial a los profesores de diferentes niveles educativos. Este software es un paso más en el desarrollo de herramientas en el ámbito del e-learning, en concreto en la generación de cuestionarios.

Este proyecto tiene como meta resolver la problemática que supone a los diferentes profesores disponer rápidamente de cuestionarios para el trabajo diario en los centros educativos, como son los ejercicios hechos en clase, los deberes o las pruebas de evaluación.

Para esto hemos creado un software capaz de disponer de cuestionarios de forma rápida, a partir de unos criterios de búsqueda deseados (nivel y materia).

En esta memoria se detallaran los siguientes contenidos sobre el proyecto:

**Planificación:** En este capítulo explicamos la planificación temporal realizada para el desarrollo de todo el proyecto (diagramas de GANTT, informe previo, etc...).

**Diseño / Desarrollo:** En este apartado explicamos el diseño del proyecto y como a partir de la idea inicial planteamos el problema a resolver.

**Implementación:** Continuaremos con la implementación del diseño en una página web (PHP + MYSQL).

**Banco de pruebas:** Este capítulo explica el banco de pruebas que se ha realizado para poder dar el visto bueno al proyecto

**Conclusiones:** Comprobaremos los datos obtenidos y los compararemos con los objetivos del proyecto. También se señalan posibles ampliaciones.

**Bibliografía:** Este apartado incluye la bibliografía consultada durante la realización de este proyecto.

# PLANIFICACIÓN

---

## OBJETIVOS

---

El objetivo del proyecto es el desarrollo de una aplicación en el ámbito del e-learning concretamente en la creación automática de cuestionarios a partir de una base de datos previamente fijada.

Principalmente, el usuario entra en la página web donde se encuentra la aplicación y rellenando diferentes campos, el programa devuelve varios archivos con las preguntas y respuestas sobre el tema solicitado. Los campos son:

- **Materia:** son las diferentes tipos de materias existentes según lo que establezca el administrador del sistema.
- **Nivel:** es el nivel de enseñanza o dificultad que tienen los cuestionarios, dependiendo del nivel al que estén destinados.
- **Número de preguntas:** son el número de preguntas y respuestas que retornara el programa. Tiene un tope máximo y mínimo según el tamaño de la base de datos.
- **Tipo de archivo de salida:** indica con qué tipo de formato de salida deseamos que se devuelvan los cuestionarios. En este caso, serán en doc y pdf.

A partir de estos campos la base de datos genera los dos archivos de preguntas y respuestas. El fichero de preguntas solo tiene las preguntas, en el caso de una pregunta simple o múltiple si son de tipo test. Y el de respuestas contiene las preguntas con la respuesta correcta.

La web tendrá una identificación por usuario y contraseña para el acceso controlado de usuarios. También hay un usuario tipo administrador para el control total de la base de datos.

El estudio de viabilidad para este proyecto se ha organizado en dos fases:

La primera fase, es un trabajo de investigación centrado en las necesidades que tiene la enseñanza en relación al e-learning. Descubrimos que había herramientas como exe-Learning o Moodle, pero no tan específicas para un nivel usuario en los cuestionarios.

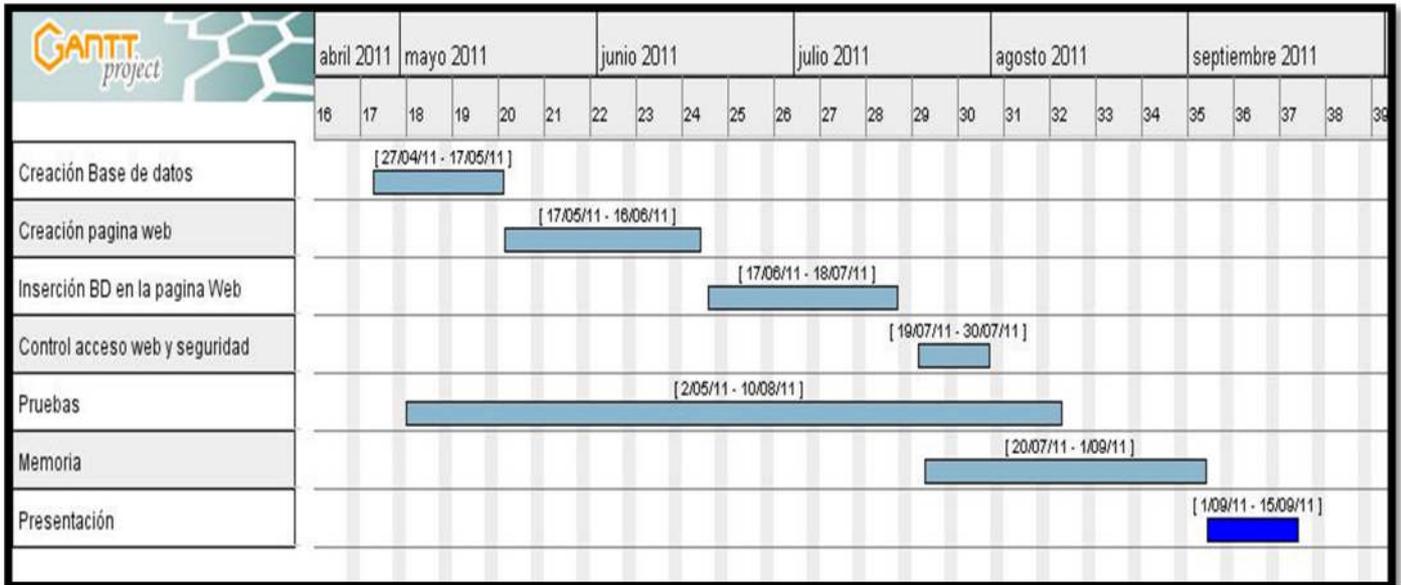
Para valorar la posibilidad de una base de datos de cuestionarios, se consultaron diferentes web y foros para ver la necesidad de esta aplicación.

La segunda fase del estudio fue la de analizar que era más importante, tener una gran cantidad de preguntas en la base de datos o simplemente que la base de datos fuera suficientemente robusta para soportar cualquier posibilidad. Nos decantamos por esta última ya que es la que nos daría mejores resultados.

El programa se ha pensado para hacerse en lenguaje PHP para la generación de la página web, mientras que se usara MYSQL para la gestión de la base de datos.

**En resumen, el proyecto es viable según las necesidades.**

## PLANIFICACIÓN TEMPORAL DEL PROYECTO



**IMAGEN 1. DIAGRAMA DE GANTT**

El diagrama de Gantt muestra 7 tareas claves de este proyecto. Las 4 primeras son tareas de desarrollo relacionadas con el diseño del software de la aplicación. La tarea 5ª está relacionada con el testeo de la aplicación que se va produciendo durante todo el desarrollo, pero especialmente en la primera semana del mes de agosto.

Finalmente las últimas dos tareas están relacionadas con la entrega del proyecto. La memoria se realiza durante el último mes del proyecto. La presentación se prepara 10 días antes de la exposición en el tribunal.

En la fase de desarrollo todo tiene un sentido secuencial y no podremos avanzar una fase sin haber terminado la tarea anterior ya que todo el proyecto está relacionado.

Algunos de los trabajos que se deberían realizar no han sido expuestos dentro del ámbito del proyecto ya que directamente no están relacionados con la idea del proyecto. Es el caso de la investigación para decidir qué proyecto de e-learning íbamos a diseñar o la instalación de la aplicación XAMPP para el desarrollo de la aplicación en Windows y Linux.

## DISEÑO

---

### IDEA PRINCIPAL

---

La idea principal, basada en los objetivos señalados, es la de crear una base de datos en una página web. En ella se insertan y consultan cuestionarios sobre un tema determinado.

Se empieza a partir de una tabla en la base de datos como esta:

Campo
<u>id</u>
pregunta
materia
nivell
n_respostes
resposta_1
resposta_2
resposta_3
resposta_4
resposta_5
correcte
tipus

IMAGEN 2. TABLA DE PREGUNTAS

En esta tabla se inserta una pregunta con su/s respuesta/s según si queremos preguntas descriptivas o tipo test. También se inserta la materia de la pregunta así como el nivel requerido. Finalmente se introduce la respuesta correcta (en el caso de que sean tipo test).

El programa deberá rellenar el identificador de pregunta (**id**), el número de respuestas (**n\_respostes**) y el tipo de pregunta (**tipus: test o descriptiva**) automáticamente y almacenarlo en la base de datos. Y deberá tener un contador para cada vez que se inserta una pregunta y tener controlado el número de preguntas existentes.

Teniendo las preguntas almacenadas podremos solicitar al programa la extracción de preguntas según unos criterios esenciales:

- **Nivel:** Corresponde a la dificultad de la pregunta.
- **Materia:** Será la temática de la pregunta.
- **Tipo:** Si es descriptiva, tipo test o mixta.
- **Numero preguntas:** Será el número de preguntas que se retornaran al hacer la extracción.

Con estos criterios la aplicación accederá a la base de datos y seleccionara las preguntas de la materia y nivel solicitado. Retornara cuantas preguntas hayamos especificado en el número de preguntas y del tipo deseado.

Al extraer las preguntas deseadas de la base de datos, la aplicación nos da la opción de poder elegir el formato de texto deseado de salida. En nuestro caso hemos decidido tener el formato .doc para facilitare al usuario posteriores modificaciones y el .pdf para generar un archivo inalterable. A partir de aquí, la aplicación entonces generara dos ficheros:

- **Fichero preguntas:** Fichero que contendrá el titulo con el nivel y materia seleccionada, las preguntas extraídas de la página web y las respuestas si es una pregunta multi-respuesta.
- **Fichero respuestas:** Fichero que contendrá lo mismo que el fichero preguntas añadiendo las respuestas.

Antes de generar los archivos, tendremos la opción de seleccionar las preguntas que queremos tener en los archivos. La aplicación nos mostrará por pantalla el número de preguntas pedidas y nosotros tendremos la opción de eliminar o insertar las preguntas que queremos en los ficheros de preguntas y respuestas.

Finalmente obtendremos los dos ficheros con los criterios solicitados. Si volvemos a generar el mismo tipo de búsqueda la aplicación no tiene por qué devolver las mismas preguntas ya que extrae las preguntas aleatoriamente de la base de datos.

## ENTORNO DE LA BASE DE DATOS

---

Para desarrollar la base de datos hemos decidido usar un entorno web. Esto nos permitirá usar la base de datos en un entorno como Internet o en un entorno de una red local (escuela, instituto, universidad, academia, etc...).

Usaremos para ello formularios web para poder insertar y extraer información en la base de datos. Contaremos también con enlaces entre los diferentes formularios para poder navegar entre ellos.

---

## SEGURIDAD

---

El problema que nos genera tener la opción de alojar la web en Internet o en una red local es inequívocamente la seguridad. Podemos tener dos principales problemas de seguridad:

- Usuarios que inserten preguntas basura en la base de datos, que al consultar después preguntas en ella nos den datos incorrectos o no relacionados con los criterios de búsqueda.
- Intentos de acceso al código de la aplicación o a la base de datos, como puede ser eliminar todas las entradas de la base de datos o modificación o supresión de ficheros de la página web.

Para el primer problema, generaremos un sistema de control de acceso a la página web, el usuario que no esté identificado en nuestro sistema no podrá acceder a la base de datos. Lo conseguiremos creando una nueva tabla en la base de datos donde almacenaremos a los usuarios registrados:

<u>id</u>
nom
cognoms
identificador

IMAGEN 3. TABLA DE  
USUARIOS

Tendremos en esta tabla el nombre, los apellidos, el identificador (DNI, NIU, PASSAPORTE, etc...) y el id de cada usuario en la tabla.

En la base de datos tendremos un listado de usuarios con sus en los distintos campos. Cuando el usuario acceda a la página web se encontrará que tiene que identificarse con su nombre apellido y DNI.

Si ese usuario está en la base de datos se le permitirá el acceso a la página web, sino se le rechazara el acceso.

Para el segundo problema planteado necesitaremos una solución más técnica. El problema de evitar el acceso al código y a la base de datos se soluciona en parte con el control de acceso pero puede darse la situación de que un usuario que no tenga acceso, conozca el alojamiento de los ficheros de la página web y pueda acceder. Esto lo solucionaremos de dos maneras:

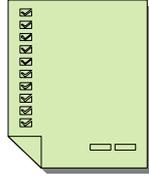
- Alojaremos los ficheros de configuración y creación de la base de datos en una carpeta externa donde los usuarios no puedan acceder.
- Usando el identificador de usuario, para identificar cada vez que cambiemos de fichero (navegar) en la página web. De esta manera evitaremos el acceso a los ficheros y su posterior eliminación o modificación.

---

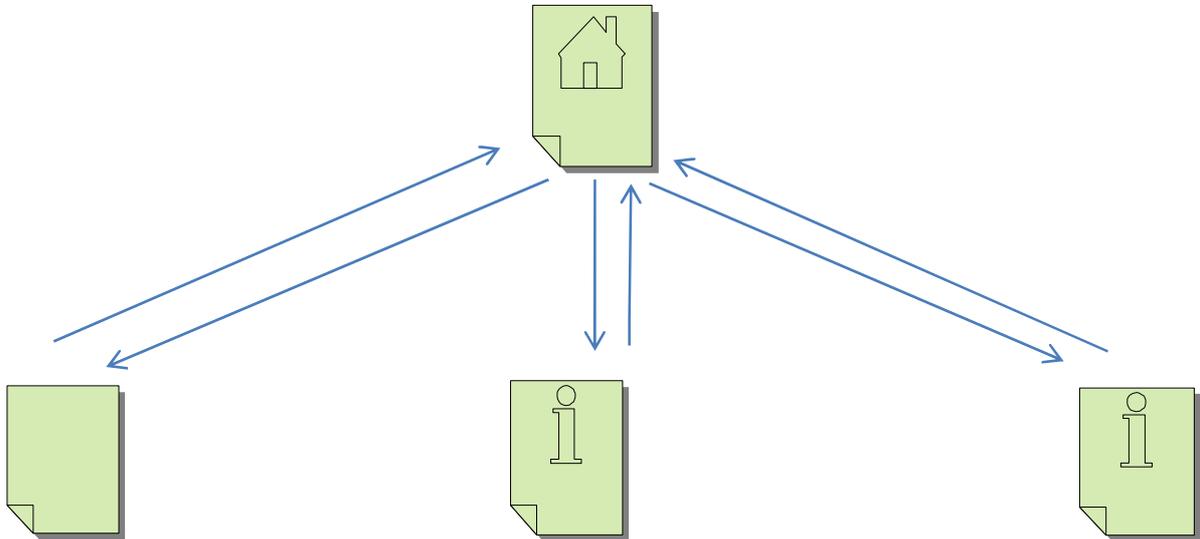
## ESTRUCTURA PÁGINA WEB

---

CONTROL DE ACCESO



PÁGINA PRINCIPAL



CONSULTA DE PREGUNTAS

INSERCIÓN PREGUNTAS TEST

INSERCIÓN PREGUNTAS DESCRIPTIVAS

La estructura de la web sería la del anterior gráfico.

Tenemos la página inicial con el control de acceso a la aplicación. Si tenemos acceso a la web entraremos en la página principal y podremos navegar entre las tres páginas principales. La primera es la de consulta de preguntas, la segunda y tercera son similares, una es para la inserción de preguntas tipo test y la otra para la inserción de preguntas descriptivas.

Desde las tres páginas finales podemos volver a la página principal siempre que queramos.

---

## ACCESO SÚPERUSUARIO

---

En la aplicación web necesitaremos un usuario que tenga control sobre la base de datos y los usuarios. Para eso necesitamos la figura del súperusuario (utilizaremos el nombre de súperusuario para el control de la base de datos y administrador para la persona que instala e inicializa la aplicación). Este usuario tendrá privilegios para poder controlar la web y la base de datos.

La diferencia con otro usuario será tener acceso a una página anterior a la página principal con las siguientes características:

- Base de datos: Acceso normal a la *Página principal*.
- Modificación nivel y usuario: Permite al administrador insertar, modificar o eliminar las diferentes variables de búsqueda de la base de datos. Para ello se crean dos tablas en la bases de datos (una para el nivel y otra para materia) para insertar los parámetros de búsqueda:



Diagrama de la tabla de niveles. Muestra un cuadro con dos filas. La primera fila contiene el texto 'id' con una línea de subrayado debajo. La segunda fila contiene el texto 'nivell'.

IMAGEN 4. TABLA DE NIVELES



Diagrama de la tabla de materias. Muestra un cuadro con dos filas. La primera fila contiene el texto 'id' con una línea de subrayado debajo. La segunda fila contiene el texto 'materia'.

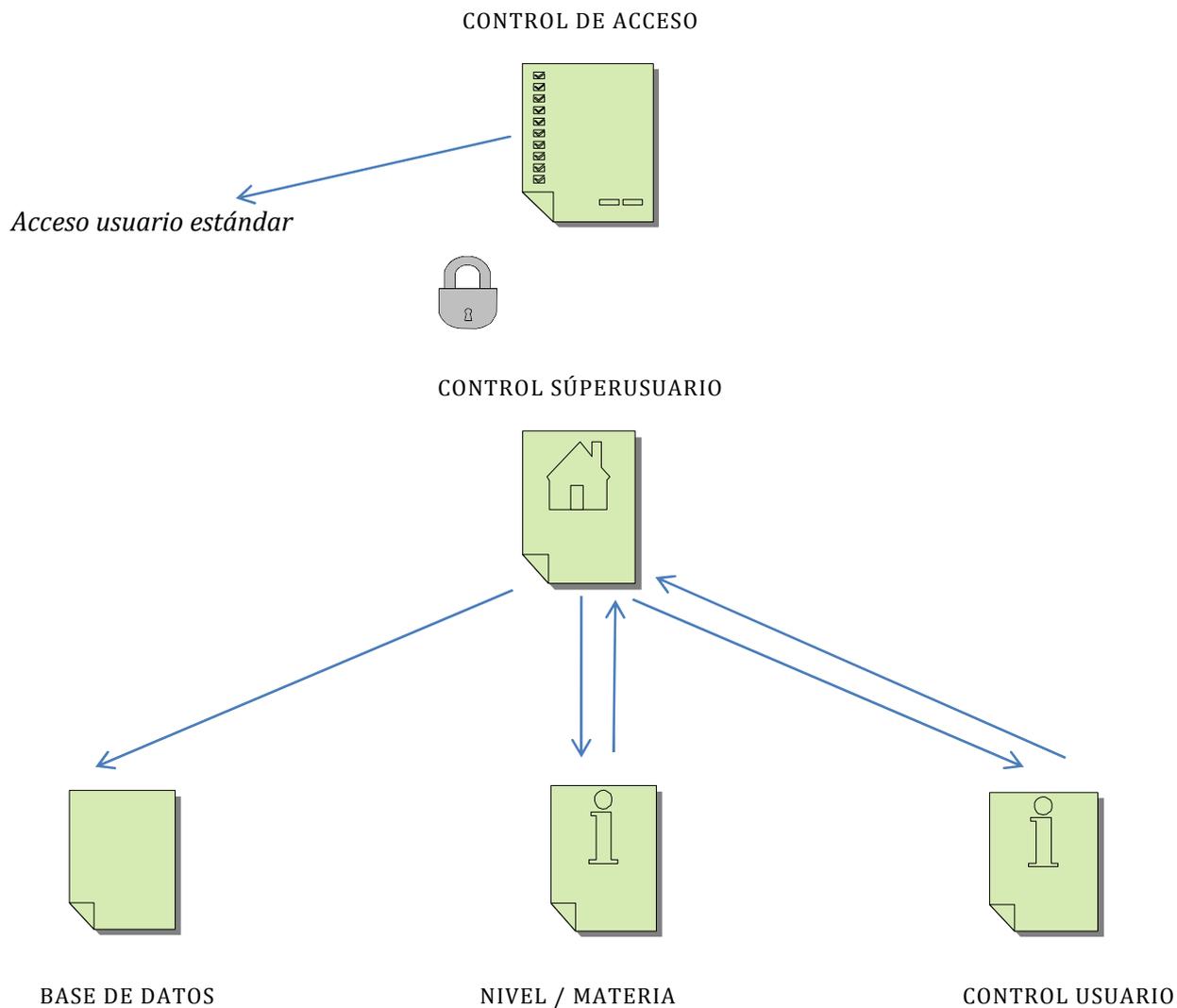
IMAGEN 5. TABLA DE MATERIAS

Posteriormente se usaran estas tablas para extraer y consultar las preguntas a partir de los campos insertados por el súperusuario.

- Usuarios: Permite el control sobre la gestión de la tabla de usuarios de la base de datos. Permite la consulta, inserción y borrado de cualquier usuario.

El administrador del sistema será quien facilite derechos de súperusuario, y preferiblemente a un único usuario, para poder así asegurar la seguridad del sistema.

D este modo, la estructura de la web para el súperusuario sería esta:



Correspondería a la estructura para el control por parte del súperusuario de la página web y de la base de datos.

## IMPLEMENTACIÓN

---

### DESCRIPCIÓN

---

Para implementar la aplicación de base de datos de cuestionarios hemos usado varios lenguajes relacionados con páginas web y bases de datos. Para la página web hemos usado PHP y HTML5 mientras que en la base de datos hemos usado MySQL, para la creación y gestión de la base de datos.

Todos estos lenguajes permiten crear la aplicación, pero para poder ejecutarlo en un entorno web necesitamos un servidor como XAMPP (**X** para el tipo de sistema operativo y **A**pache, **M**ySQL, **P**HP, **P**erl). Con esta herramienta tenemos todo lo necesario para empezar a desarrollar sin necesidad de instalar y configurar uno a uno los distintos programas que la componen.

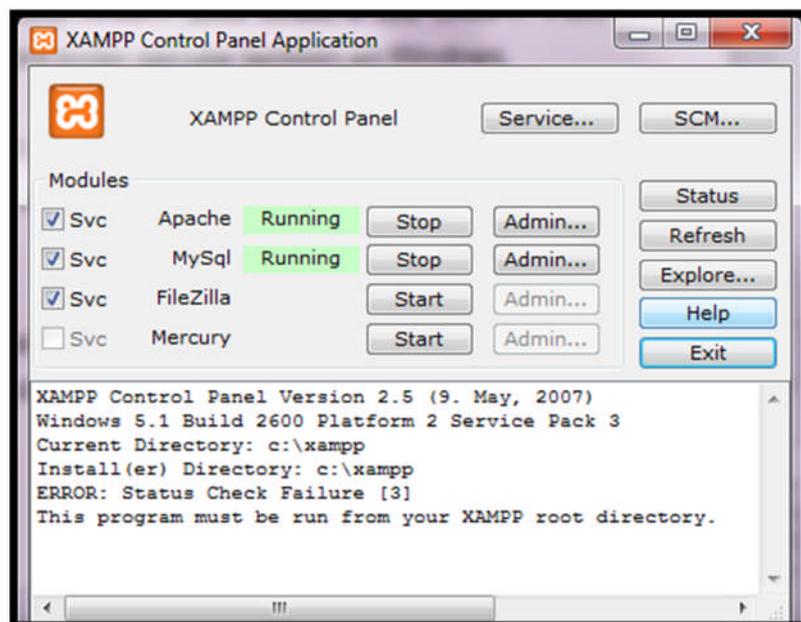


IMAGEN 6. XAMPP

En nuestro caso hemos hecho el desarrollo con XAMPP para Linux, o sea LAMPP. Pero también hemos utilizado la versión WAMPP para poder ejecutar también en Windows. Como navegador hemos utilizado el Mozilla Firefox para el uso en Linux pero en Windows hemos preferido Internet Explorer, no hemos escogido Firefox por problemas de compatibilidad con XAMPP.

Dentro de la aplicación XAMPP hemos utilizado el módulo phpmyadmin que dispone de una interface vía web para el control de base de datos. Desde ella como administrador se puede consultar y gestionar las diferentes tablas de la página web:

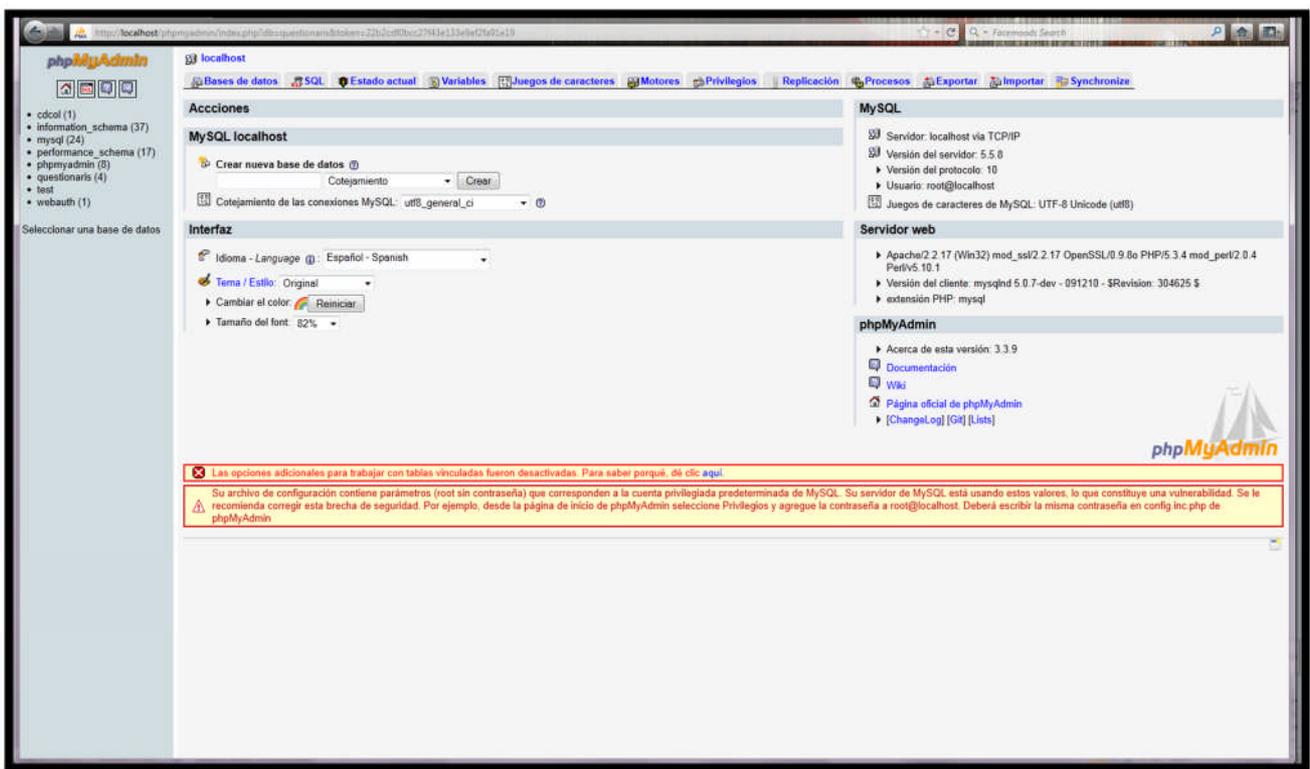


IMAGEN 7. PHPMYADMIN

Por problemas de compatibilidad de la versión de XAMPP que había en Abril de 2011 para Windows 7, tuvimos que implementar el programa en un sistema Ubuntu con LAMPP. Finalmente en Agosto de 2011 XAMPP apareció una versión compatible con Windows 7 y modificamos el programa también para esa versión, que es finalmente la definitiva.

Dentro de la aplicación LAMPP hemos creado dos carpetas: *htdocs* y *configuración*. La primera que contiene los ficheros de cada una de las páginas de la web y la segunda los ficheros de configuración y creación de la web y la base de datos.

## CREACIÓN DE LA BASE DE DATOS

En la creación de la base de datos *cuestionarios* lo que hacemos es crear las cuatro tablas (*preguntas*, *nivel*, *materia* y *usuari*) que la componen. En cada tabla tenemos los diferentes campos. Estas se crearon con sus nombres y campos, sin insertar ningún registro en las tablas:

Tabla ▲	Acción	Registros <sup>1</sup>	Tipo	Cotejamiento	Tamaño
<input type="checkbox"/> materia	      	1	InnoDB	latin1_swedish_ci	16.0 KB
<input type="checkbox"/> nivell	      	1	InnoDB	latin1_swedish_ci	16.0 KB
<input type="checkbox"/> preguntas	      	2	InnoDB	latin1_swedish_ci	16.0 KB
<input type="checkbox"/> usuari	      	1	InnoDB	latin1_swedish_ci	16.0 KB
<b>4 tabla(s)</b>	<b>Número de filas</b>	<b>5</b>	<b>InnoDB</b>	<b>latin1_swedish_ci</b>	<b>64.0 KB</b>

IMAGEN 8. TABLAS BBDD

El archivo de creación de la base de datos y de sus tablas es de un solo uso. Solo se usa una vez para crearlas, luego no se vuelve a usar y se desplaza a la carpeta de *configuración*, donde si se tuviera que volver a crearla base de datos se cogería de allí.

Entonces las tenemos creadas y lo único que tenemos que hacer es conectarlas con la aplicación web para poder trabajar con ellas. Esto lo hacemos con PHP y MySQL que nos permiten acceder a las tablas e insertarle nuevas entradas a partir de formularios HTML.

Una vez que ya está creada la base de datos con sus tablas podemos empezar con la programación de la página web.

## PÁGINA WEB

---

En esta parte explicaremos paso a paso las diferentes páginas que componen nuestra aplicación Web.

---

### PÁGINA DE CONTROL DE ACCESO

---

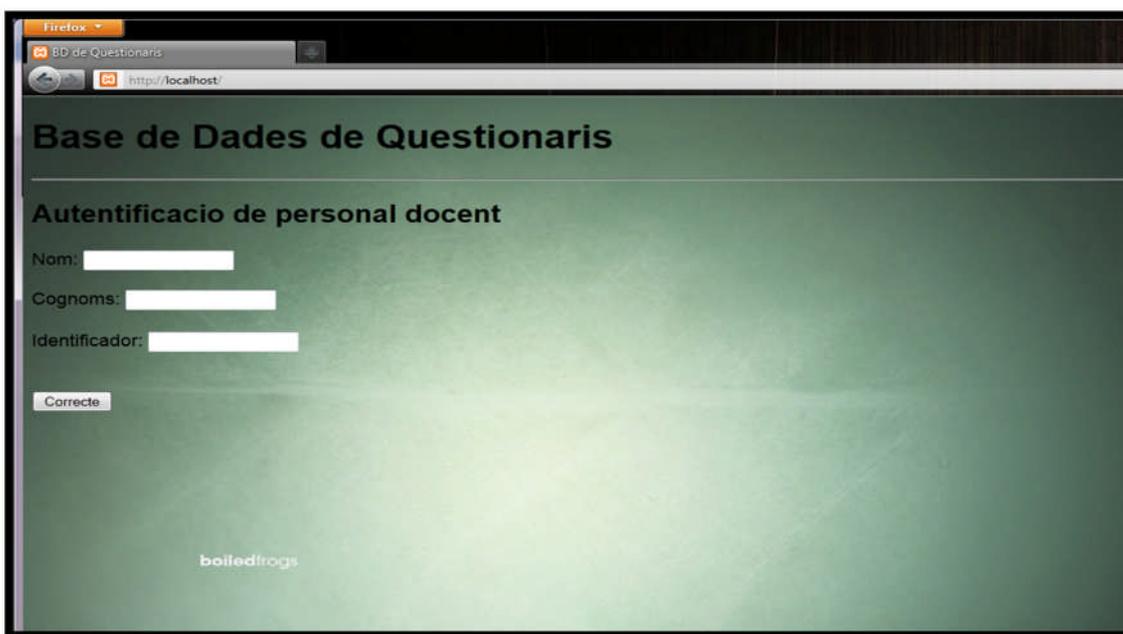


IMAGEN 9. PAGINA DE ACCESO

Esta es la página inicial de nuestra aplicación. Está compuesta por un formulario para el control de acceso a la Web. Cuando un usuario entra en la página, rellena los campos de *nombre*, *apellidos* e *identificador* y hace clic al botón que aparece. Si el usuario está creado dentro en tabla *usuario* de la BD, se le permite el acceso, sino sale un mensaje informando de que no tiene acceso.

Si el usuario tiene permisos de súperusuario accederá a la página de control de súperusuario, mientras que si solo tiene acceso de usuario accederá a la página principal de la base de datos.

---

## PÁGINA DE CONTROL DE SÚPERUSUARIO

---

Esta es la página de control para súperusuario. Contiene las herramientas necesarias para el control total de la base de datos:

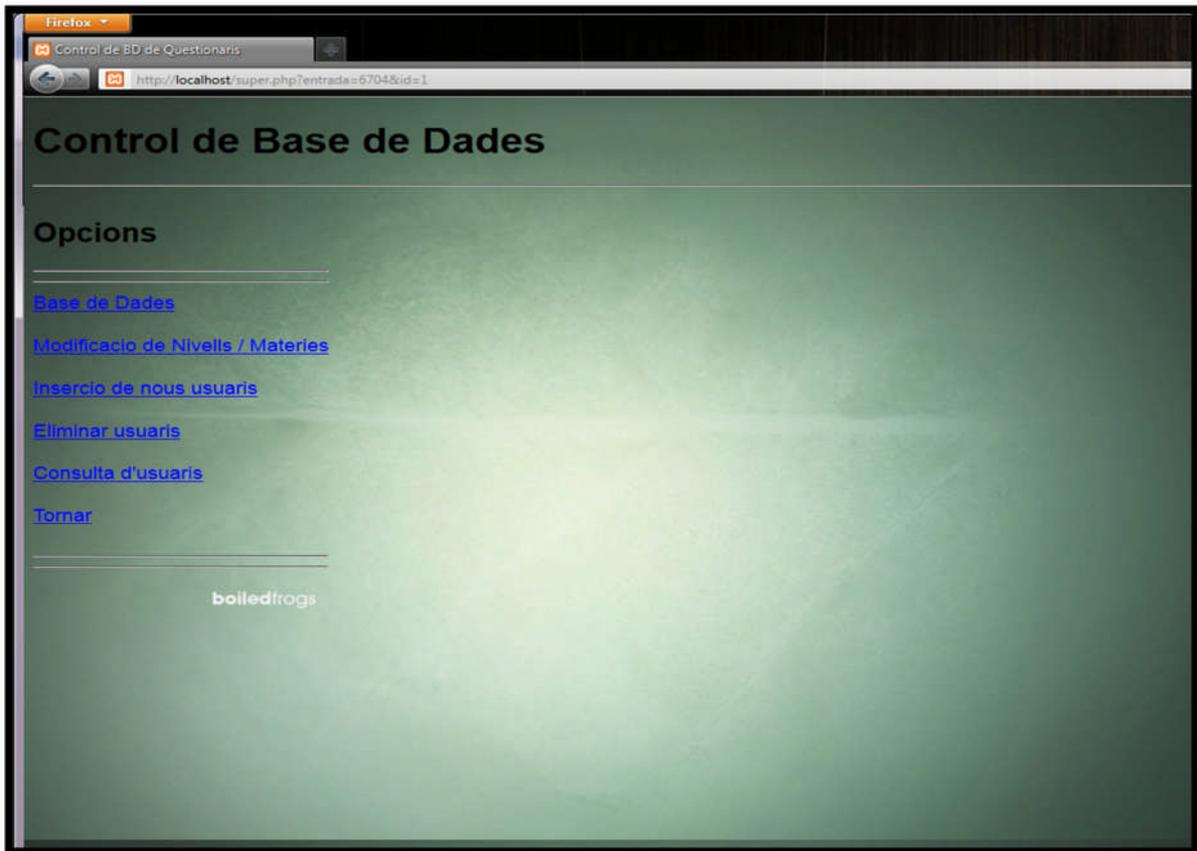
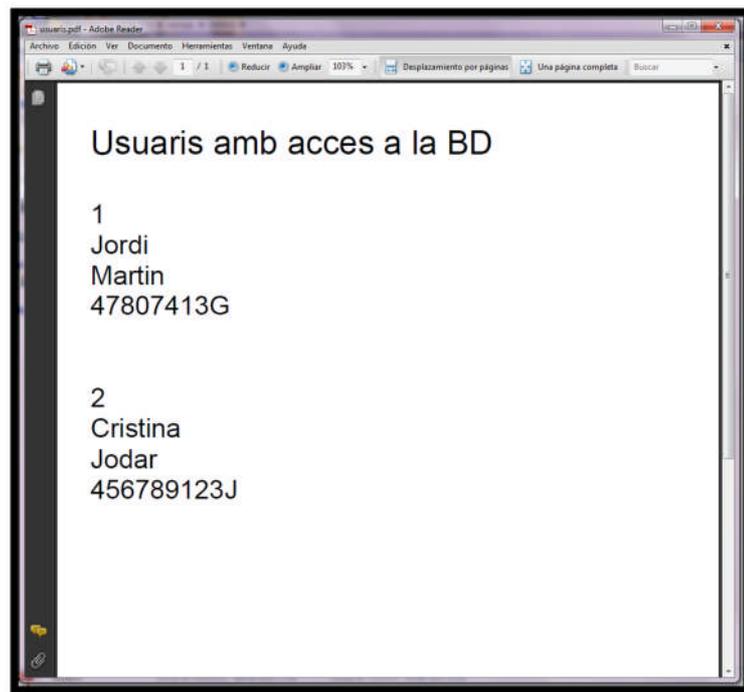


IMAGEN 10. PAGINA DE CONTROL DE USUARIO

Disponemos de diferentes links para navegar, algunos con acceso a otras páginas o a ficheros de consulta (*consulta de usuarios*):

- **Base de datos:** Acceso a la página principal de la aplicación (acceso como si fuéramos usuarios normales).
- **Modificación de Nivel / Materia:** Página para el control de los parámetros de búsqueda de la tabla de *questionarios*.

- **Inserción de usuarios:** Página para la creación de nuevos usuarios.
- **Eliminar usuario:** Página para la eliminación de usuarios.
- **Consulta de usuarios:** Archivo *usuario.pdf* con un listado de los usuarios del sistema, o sea, el contenido de toda la tabla de *usuarios*. El primer usuario que aparece es siempre el súperusuario:



**IMAGEN 11.** ARCHIVO DE USUARIOS

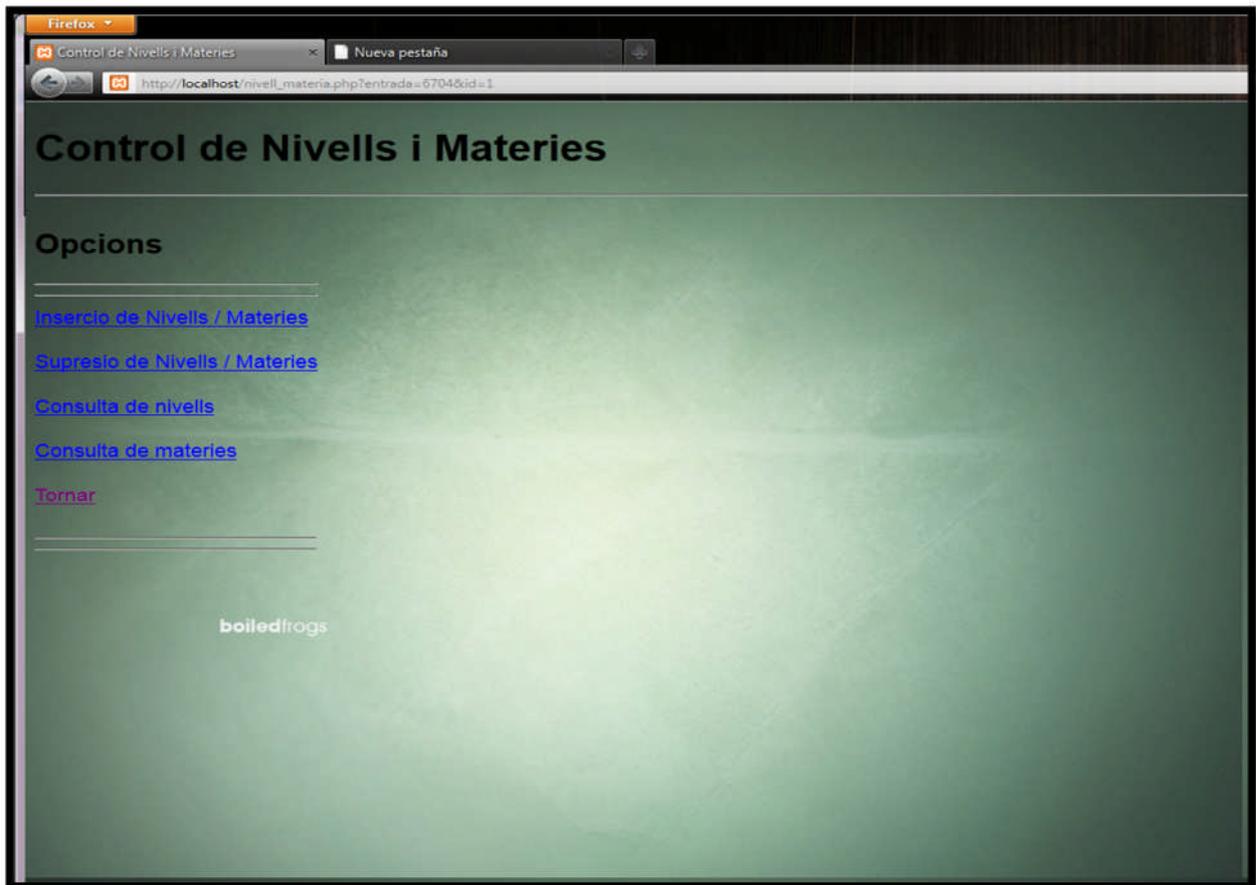
- **Volver:** Volvemos a la página de control de acceso y debemos volver a ingresar los datos.

---

## MODIFICACIÓN DE NIVEL Y MATERIA

---

Esta página se encarga de poder insertar, eliminar y consultar los campos Nivel y Materia de la base de datos. Para eso usamos las tablas *materia* y *nivel* que contienen los valores de los campos:



**IMAGEN 12. PAGINA DE NIVEL Y MATERIA**

En esta página disponemos de varios links, unos para acceder a formularios y otros para consulta de archivos:

- **Inserción de Niveles / Materias:** Inserción de niveles y/o materias en las tablas de *nivel* y *materia*:



IMAGEN 13. PAGINA INSERCIÓN DE NIVEL Y MATERIA

- **Eliminar Niveles / Materias:** Borrado de niveles y/o materias en las tablas de *nivel* y *materia*:

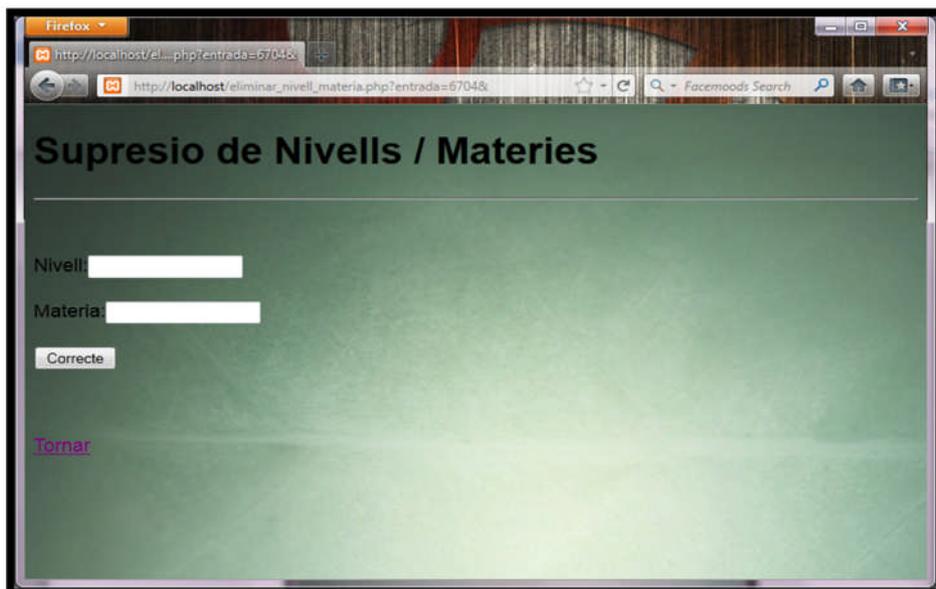
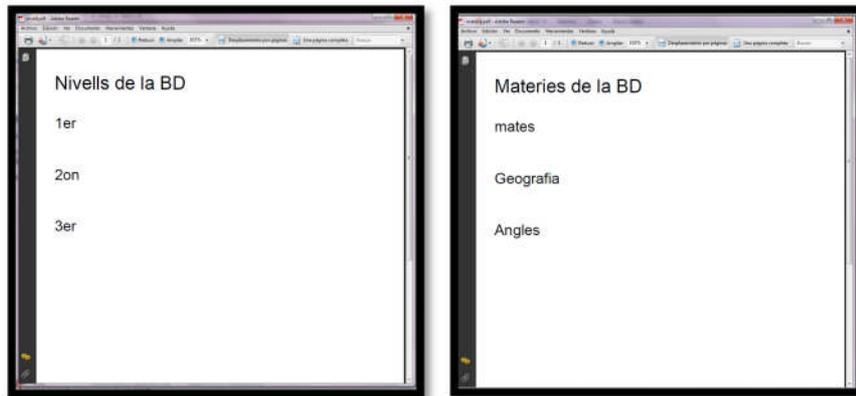


IMAGEN 14. PAGINA DE BORRADO DE NIVEL Y MATERIA

- **Consulta de Niveles / Materias:** Dos archivos *nivel.pdf* y *materia.pdf* con el listado de materias y niveles que hay en las tablas *nivel* y *materia*:



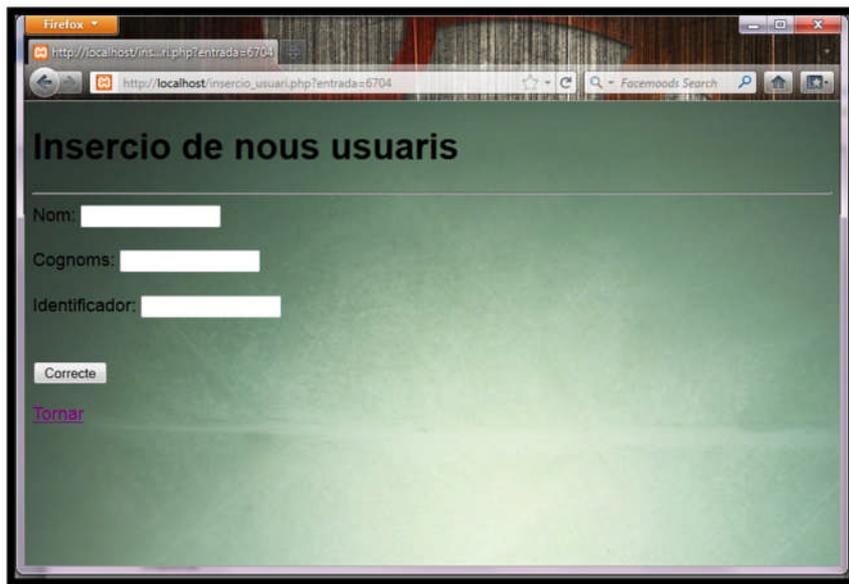
**IMAGEN 15. FICHEROS DE NIVEL Y MATERIA**

---

## INSERCIÓN DE USUARIOS

---

Presionando en el link *de inserción de usuarios* de la página de control de súperusuario, entramos en una página con un formulario para la inserción de un nuevo usuario en la tabla *usuario* de la base de datos. Tenemos que rellenar los 3 campos correctamente para poder agregarlo a la tabla:+



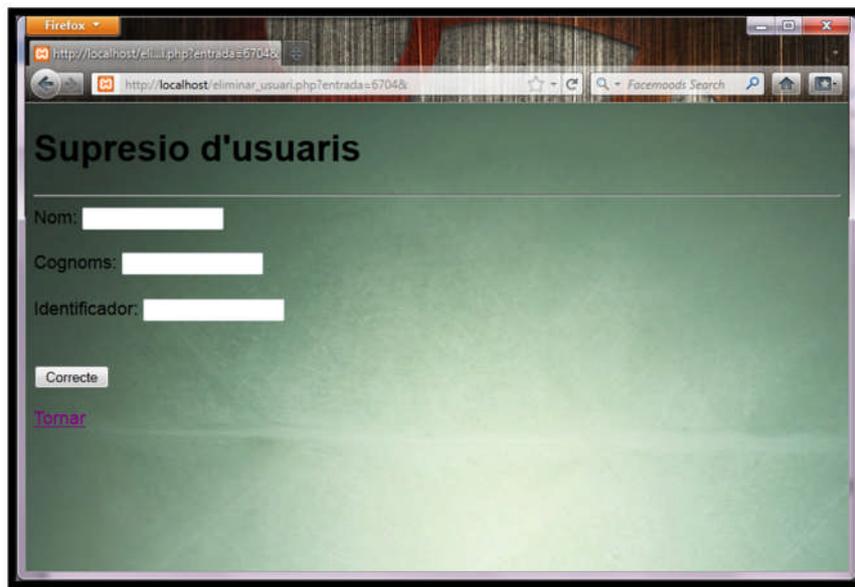
**IMAGEN 16.** PAGINA DE INSERCIÓN DE USUARIOS

---

## BORRADO DE USUARIOS

---

Presionando en el link de *Eliminar usuarios* de la página de control de súperusuario, accedemos a una página con un formulario para eliminar usuarios de la tabla *usuario* de la base de datos. Solo se puede suprimir un usuario si están todos los campos correctamente rellenos y el usuario existe en la tabla:



**IMAGEN 17.** PAGINA DE BORRADO DE USUARIOS

---

## BASE DE DATOS

---

Con el link *Base de Datos* accedemos a la página principal para la inserción y consulta de cuestionarios.

---

## PÁGINA PRINCIPAL

---

Esta es la página principal de la aplicación, donde podemos trabajar con los cuestionarios, se accede a esta página a partir de *la página de control de acceso* (si somos usuarios estándar) o por *la página de control de súperusuario*:

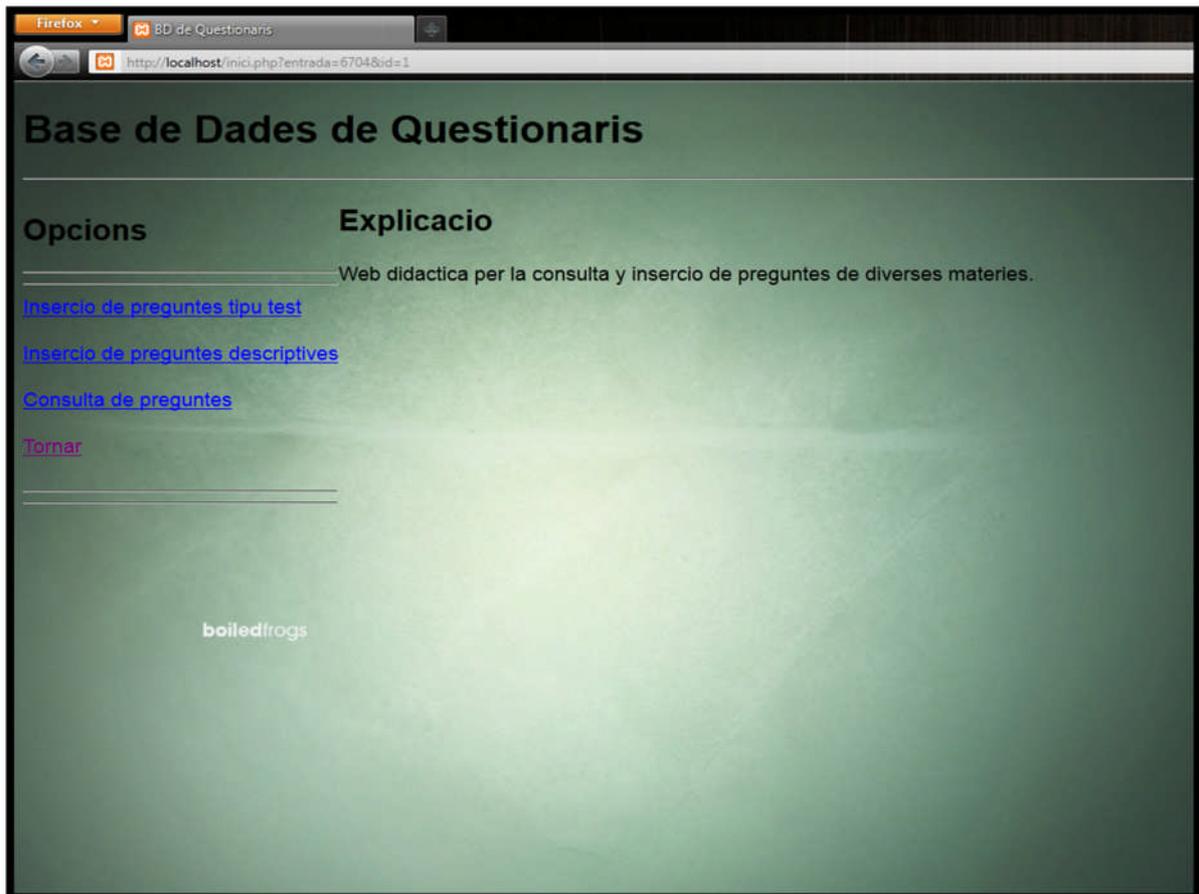


IMAGEN 18. PAGINA DE PRINCIPAL

Se compone básicamente de una explicación de lo que hace la página y diversas opciones o funcionalidades:

- **Inserción de preguntas tipo test:** Pagina con un formulario para la inserción de preguntas con multi-respuesta.

- **Inserción de preguntas descriptivas:** Pagina con un formulario para la inserción de preguntas con una sola respuesta.
- **Consulta de preguntas:** Pagina más importante de la aplicación con la búsqueda de preguntas en la base de datos.
- **Volver:** Volvemos a la página inicial de control de acceso de usuario.

---

## INSERCIÓN DE PREGUNTAS TIPO TEST

---

En esta página lo que hacemos es insertar preguntas que son tipo test a la base de datos, es decir, insertamos preguntas con más de una posibilidad de respuesta.

La inserción se hace en la tabla de *preguntas* y se guardan los diferentes campos en ella para una posterior consulta. La diferencia de este formulario con el de *inserción descriptiva* es el campo de respuesta correcta, que nos dice cuál de las respuestas insertadas es la respuesta válida.

Se ha decidió poner como máximo 5 respuestas y como mínimo 2. No podemos poner más de 5 respuestas, y si ponemos solo 1 el programa genera un error. También controlamos que se rellenen los campos mínimos (pregunta y dos respuestas), sino generamos también un error.

Cada vez que se inserta una pregunta se suma una a un contador para poder saber el número de preguntas en la base de datos.

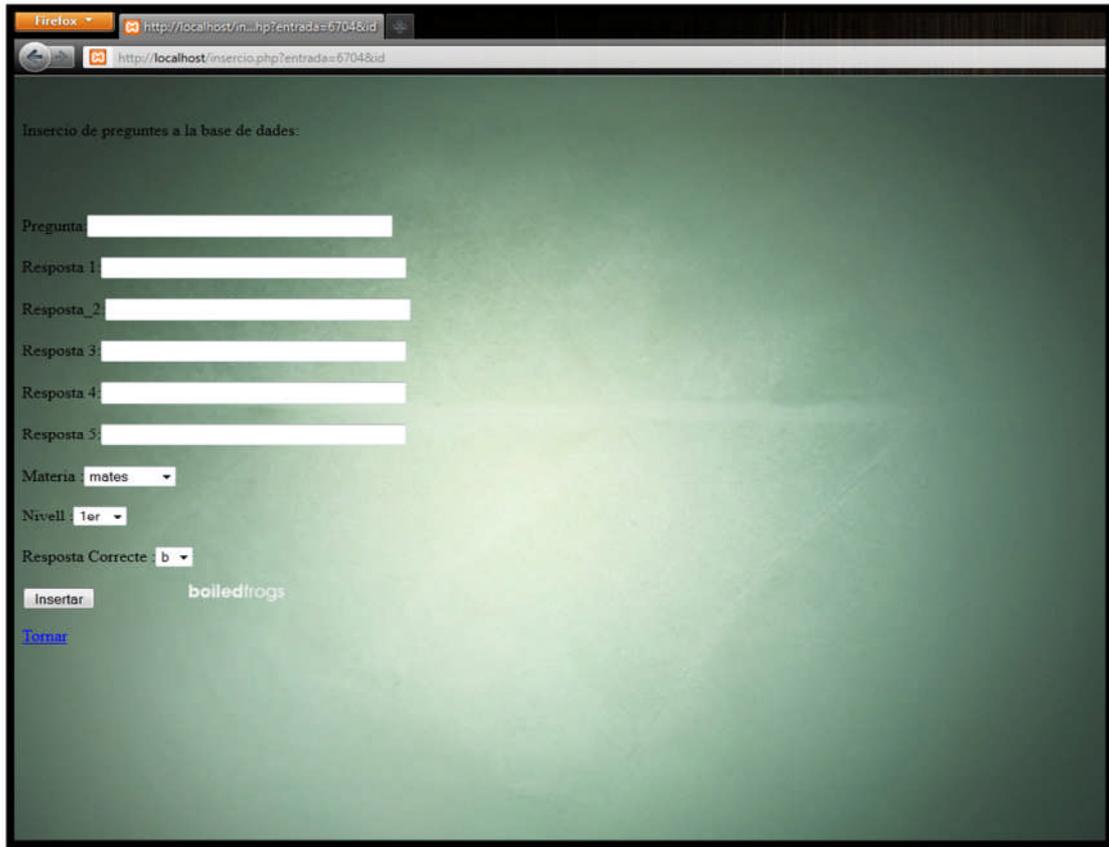


IMAGEN 19. PAGINA DE INSERCIÓN DE PREGUNTAS TEST

---

## INSERCIÓN DE PREGUNTAS DESCRIPTIVAS

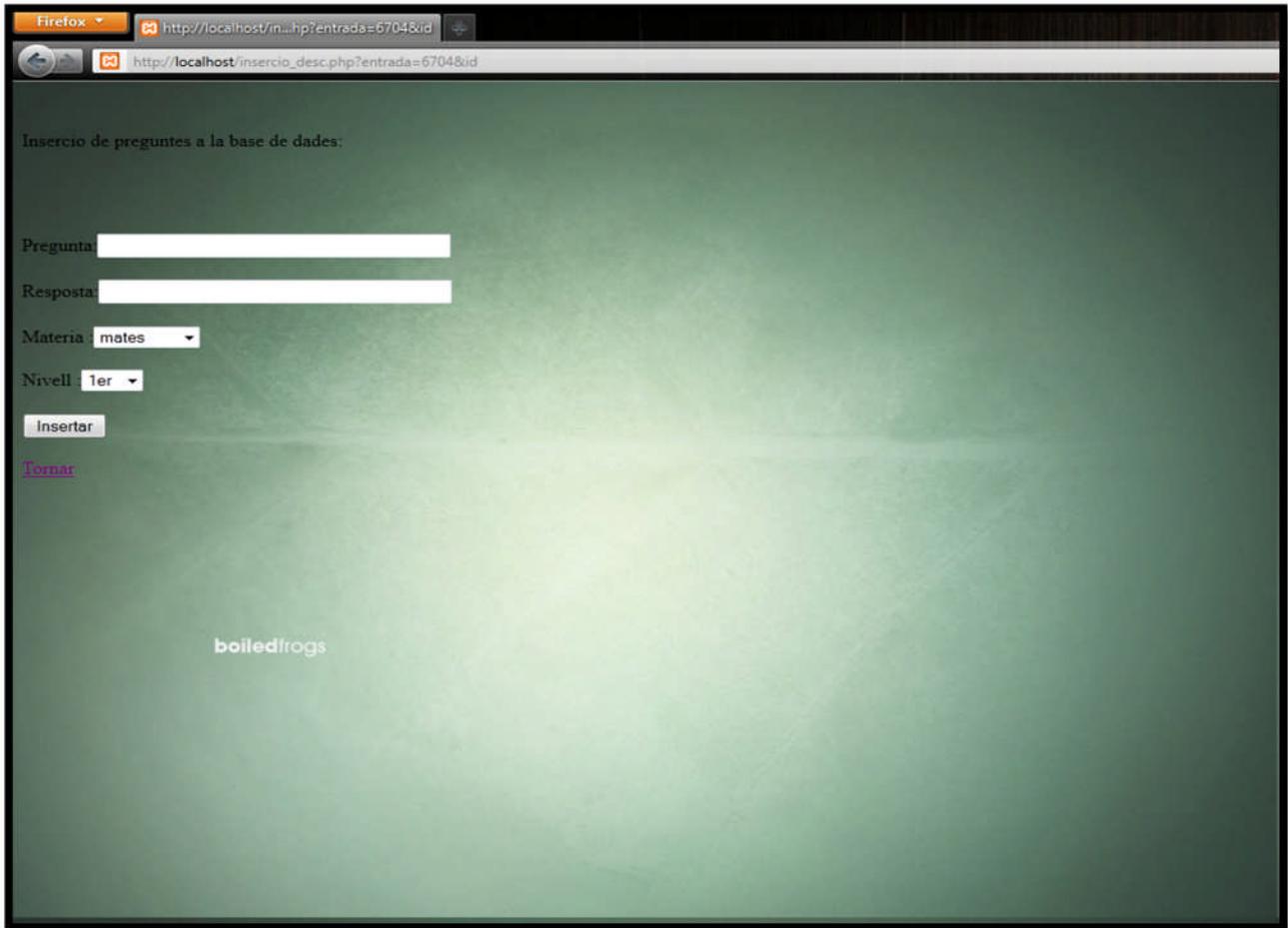
---

En la siguiente página hemos creado un formulario para preguntas descriptivas o de respuesta larga. Este tipo de preguntas solo contiene una respuesta y suelen ser más largas que las de tipo test.

La inserción también se hace en la tabla de *preguntas* pero solo se rellenan los campos de *respuesta 1* y se deja en blanco el de *respuesta correcta*.

En este caso solo son obligatorios los campos *pregunta* y *respuesta*.

Como en el otro caso, cada vez que se inserta una pregunta se suma 1 al contador que cuenta el número de preguntas existentes en la base de datos



**IMAGEN 20. PAGINA DE INSERCION DE PREGUNTAS DESCRIPTIVAS**

---

## CONSULTA DE PREGUNTAS

---

Esta última página es la más importante de la aplicación. Consiste en generar las preguntas que quiere el usuario a partir de diferentes parámetros. Por lo tanto, si los parámetros de búsqueda son más específicos y el número de entradas en la base de datos es alto, encontraremos preguntas más específicas y más acorde con lo especificado:

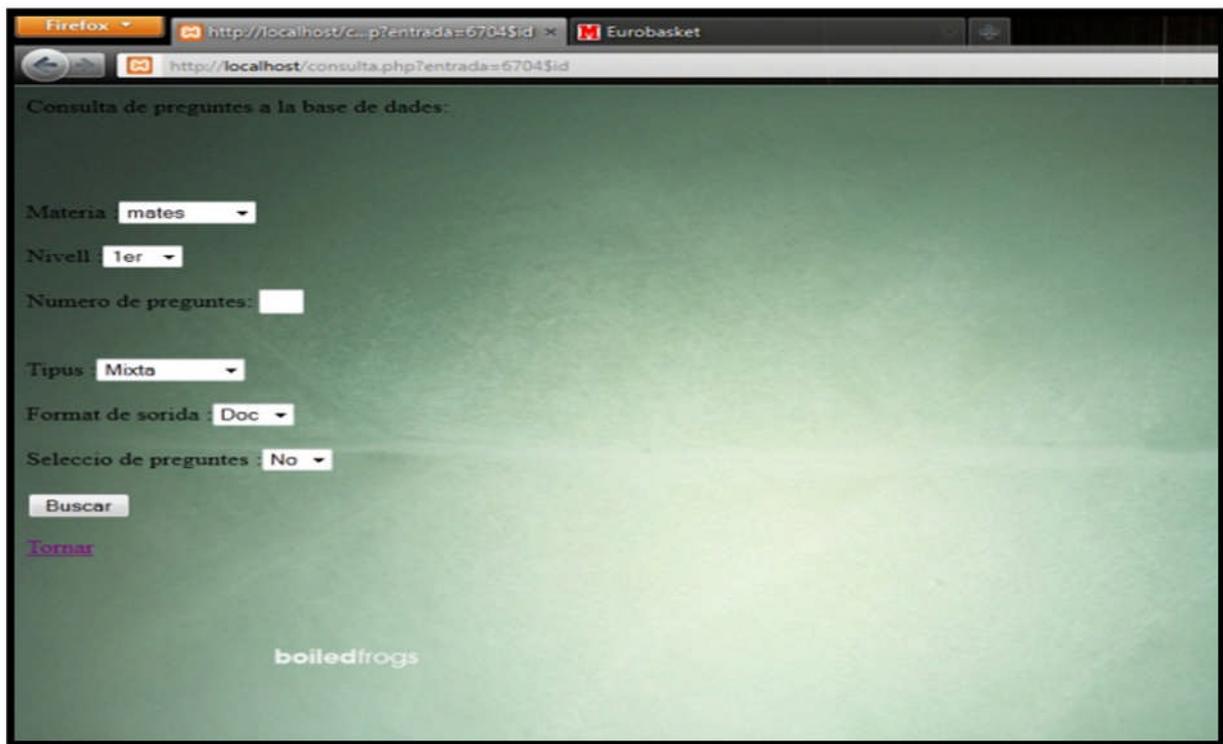


IMAGEN 21. PAGINA DE CONSULTA DE PREGUNTAS

Los parámetros de búsqueda que permiten la extracción de preguntas son los siguientes:

- **Nivel:** Especifica el campo *nivel* de las preguntas deseadas, a partir de una lista desplegable ya fijada (generada por el súperusuario y siendo un listado de las que hay en la base de datos).
- **Materia:** Especifica el campo *materia* de las preguntas deseadas, a partir de una lista previamente fijada (generada también por el súperusuario y que es un listado de las que hay en la base de datos).
- **Numero de preguntas:** Especificamos cuantas preguntas queremos extraer de la base de datos. Tenemos que coger como mínimo una y máximo el tope de preguntas de ese tipo en la tabla *preguntas*.
- **Tipo:** Nos permite escoger si queremos que las preguntas que extraemos sean tipo test, descriptivas o mixtas (que el programa no haga distinciones entre test o descriptiva).

- **Formato de salida:** Elegimos que formato de salida queremos que tengas los ficheros *preguntas* y *respuestas*. En nuestro caso tenemos un menú desplegable donde nos da la opción de elegir como formato de salida el pdf y el doc.
- **Selección de preguntas:** Este campo lo utilizamos para que una vez que tengamos nuestros criterios de búsqueda, el programa nos permita elegir que preguntas queremos que nos inserte en los ficheros de salida. Para esto tenemos un menú desplegable donde nos pregunta si queremos seleccionar las preguntas o no.

Si decimos que no, el programa generara los ficheros automáticamente. Pero si contestamos que sí, la aplicación nos envía a una página como la siguiente:

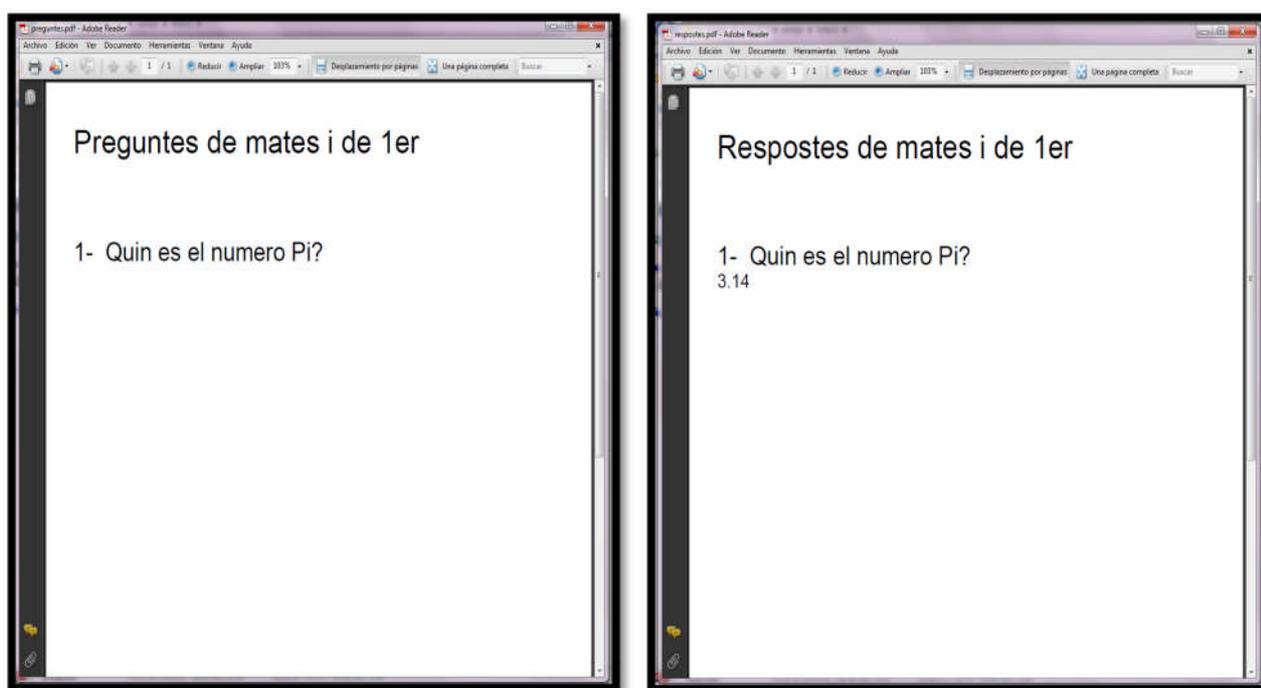


**IMAGEN 22. PAGINA DE SELECCIÓN DE PREGUNTAS**

En esta página nos aparecerá el listado con nuestra búsqueda solicitada. Por cada pregunta tendremos una cuadrado donde podremos marcar o desmarcar la pregunta. Cuando hayamos seleccionado o deseleccionado las preguntas que queremos, le daremos al boto *Generar* y el programa nos generara los ficheros con la selección que hemos hecho.

Aunque hagamos la misma búsqueda varias veces, las preguntas en la selección de preguntas pueden no ser las mismas o estar desordenadas, esto es porque se hace siempre una búsqueda aleatoria de preguntas en la tabla, nunca de manera secuencial.

En esta parte de la aplicación es donde se generan realmente los cuestionarios. Cuando generamos la búsqueda, ya sea directamente o a través de la selección previa, la aplicación encuentra solución en la tabla *preguntas* y genera dos links con los ficheros de *preguntas* y *respuestas*. Estos dos ficheros son similares, el fichero *preguntas* contiene el nivel y la materia pedido junto con las preguntas (y las respuestas si son de tipo test), mientras que el fichero de *respuestas* contiene algo similar al fichero de preguntas pero además vienen indicadas las respuestas correctas:



**IMAGEN 23.** FICHEROS DE PREGUNTAS Y RESPUESTAS

Cuando pensemos en generar los archivos de salida de cuestionarios, tuvimos la idea de que aparte de genéralos en formato doc, sería más elegante poder hacerlo en formato pdf.

El problema fue que en PHP no existe ningún modulo ni función que nos permita hacerlo. Realizar este módulo quedaba ya fuera de los objetivos iniciales del proyecto.

Investigando sobre el tema, encontramos un módulo de código libre en PHP para la creación de ficheros de salida en formato pdf. Este módulo se denomina **ezpdf** y es un módulo que nos permite generar los pdf y hacerlo de una forma rápida, fácil y estética.

El modulo consta de tres ficheros, un fichero con las fuentes de letras, otro con las funciones que llamamos para la creación de pdfs y otro de librerías.

Solo tenemos que usar la función **ezText** e ir insertando líneas al fichero. Cuando hayamos acabado cerramos el archivo .pdf igual que si fuera de otro formato (*fopen, fclose, etc...*).

## PRUEBAS Y RESULTADOS

---

### PRUEBAS

---

A lo largo del desarrollo de la aplicación se han ido realizando pruebas. Como la aplicación se ha creado de formula modular, hemos ido testeando estos módulos a medida que estaban acabados. Por este motivo describiremos las pruebas que hemos realizado en cada módulo.

---

#### PRUEBAS DEL MÓDULO DE CONTROL DE ACCESO

---

En la página inicial para el control de usuario hemos testado el acceso de usuarios autorizados. Para ello hemos probado con usuarios autorizados, usuario no autorizados y con súperusuarios.

También hemos testado que se rellene todo el formulario correctamente, si el formulario no está correctamente rellenado, la aplicación devuelve un error.

---

#### PRUEBAS DEL MÓDULO DE CONTROL SÚPERUSUARIO

---

En esta página hemos testado básicamente que la persona que entre este autorizada, o sea, que se súperusuario. Se ha comprobado que si se entra directamente desde la barra de dirección, el programa te direcciona a la página de control de acceso.

Otro test que se ha realizado ha sido comprobar que todos los links funcionen correctamente y que los ficheros de listado de usuarios se generen correctamente (en Windows 7 para poder verlos hay que ir a la carpeta *htdocs*, en Ubuntu funcionan correctamente).

Como en el módulo de control de acceso, hemos testeado que se rellenen correctamente los formularios de inserción y borrado de usuarios. Obviamente se ha comprobado que acceden correctamente a la base de datos para hacer su petición.

---

### PRUEBAS DEL MÓDULO DE MODIFICACIÓN DE NIVEL Y MATERIA

---

Este módulo al ser gemelo con el de control de súperusuario, se le hacen las mismas pruebas. Se comprueba el funcionamiento de los links, el control desde la barra de dirección, los ficheros de consulta de nivel y materia y los formularios de inserción y borrado.

---

### PRUEBAS DEL MÓDULO PRINCIPAL

---

En la página principal del sistema, hemos hecho la prueba del control de acceso a partir de la barra de dirección y el correcto funcionamiento de los links.

En este caso particular, se ha hecho una prueba que aun siendo súperusuario, al dar al link de volver, te dirija a la página inicial y tengas que volver a ingresar los datos.

---

### PRUEBAS DEL MÓDULO DE INSERCIÓN DE PREGUNTAS

---

En las páginas con los formularios de inserción de preguntas (sean tipo test o descriptivas) hemos hecho pruebas similares. La diferencia entre los dos formularios recae en los campos de respuesta. Por eso, en el formulario de preguntas descriptivas solo tenemos que comprobar un campo, mientras que en el formulario de preguntas tipo test hay que comprobar cada respuesta insertada. Se ha hecho entonces test para ver si los campos de respuestas y preguntas estaban rellenos.

Como en el formulario de preguntas test tiene un campo de respuesta correcta, se ha comprobado que la respuesta que indica el usuario exista en el formulario, no puede estar la respuesta vacía y que sea la correcta.

Como son dos formularios de inserción a la base de datos, se ha comprobado la inserción correcta en la tabla de preguntas.

---

## PRUEBAS DEL MÓDULO DE CONSULTA DE PREGUNTAS

---

Este es un módulo diferente del resto. Al tener que acceder a la base de datos y extraer la información, hay que testear que la información que insertamos en el formulario sea correcta.

Hemos comprobado que el contenido de los campos de nivel y materia, concuerden con los que hay en la tabla de preguntas, sino debe mostrar el error. Y también concuerde con los datos de nivel y materia de sus respectivas tablas.

Al completar todo el formulario, se ha probado que si no hay ninguna entrada con esos parámetros de búsqueda, devuelva una notificación indicándolo.

El otro test a realizar, es que el número de preguntas que quiere que retorne la aplicación, sea mayor que cero y menor que el número de entradas de esa búsqueda en la tabla preguntas.

Cuando le damos a la opción de seleccionar preguntas, aparecen las preguntas que queremos que aparezcan después en los ficheros. La prueba en este caso ha sido comprobar que al menos se seleccione alguna de las preguntas de la lista, en caso contrario produce un error.

Todas estas pruebas se deben hacer para que los ficheros que genera la página sean correctos, si alguno de estos test fallara tiene que salir una notificación de error.

## RESULTADOS

---

Los resultados obtenidos después de la implementación y las pruebas pertinentes han sido satisfactorios. La aplicación cumple con las pruebas que se han planteado. Además se ha comprobado que con una gran cantidad de datos a buscar en la base de datos, retorna los datos con rapidez.

A parte de las diferentes pruebas que se han hecho debemos comentar los problemas que hemos tenido a la hora de adaptar la aplicación a Windows. Independientemente de los problemas de la aplicación XAMPP, no hemos conseguido que en los archivos .doc se vean las líneas espaciadas, y en el caso de los .pdf ver los archivos por el explorador. Hemos considerado comentarlo ya que es un problema de compatibilidad con el sistema operativo y ajeno a la aplicación.

Tenemos la versión para Linux que funciona correctamente. Hemos hecho la versión Windows porque es más fácil de instalar el servidor XAMPP, creemos que era mejor así ya que en Ubuntu era complicada la instalación y configuración. Por eso decidimos configurar la versión Windows para el uso del profesorado que esta menos familiarizado con el entorno GNU/Linux.

## CONCLUSIONES

---

### OBJETIVOS CUMPLIDOS

---

Este proyecto tenía como principal objetivo solucionar la generación rápida y automática de cuestionarios para personal docente. Este objetivo se ha cumplido satisfactoriamente. En primer lugar se estudiaron los requerimientos y necesidades del personal docente, después se examinaron las soluciones que ofrecía Internet para el los cuestionarios automáticos y finalmente se decidió por la solución que mejor se adaptaba a las necesidades y requerimientos a satisfacer. Indicar que la selección de la solución fue realizada de forma consensuada con el director de proyecto, remarcar también que durante todo el estudio se han tenido muy en cuenta todas las indicaciones y sugerencias realizada por el mismo, todo ello con el fin de seleccionar la mejor solución posible.

En resumen, los objetivos planteados al iniciar el proyecto se han visto cumplidos.

### CONCLUSIONES OBTENIDAS

---

Las aplicaciones de enseñanza (e-learning) ayudan al personal docente de escuelas, academias y universidades. Permite que los profesores se puedan dedicar más a la enseñanza en sí, facilitando su labor docente. Por eso creemos que la aplicación es útil, cada vez que un profesor crea un cuestionario lo inserta en la base de datos, si esto lo hacen varios profesores se consigue disponer de una gran base de datos para que a otros profesores les resulte útil. O simplemente colgar la aplicación en Internet y que cualquiera que esté autorizado pueda acceder y así crear una gran base de datos de cuestionarios.

Nosotros hemos creado la herramienta pero deben ser los usuarios de la aplicación los que consigan sacar el potencial, insertando sus preguntas/respuestas para crear una gran base de datos.

## PROBLEMAS SURGIDOS

---

Hemos comentado durante esta memoria varios problemas que nos han surgido. Principalmente el alto tiempo dedicado a la configuración de la aplicación para Ubuntu y Windows. Resultaba difícil configurar el servidor para un PC de 64 bits ya que XAMPP no estaba preparado para trabajar en esta estructura. Ha sido el peor de los problemas sobre todo para conseguir presentarlo al tribunal. Finalmente, nos dimos cuenta que ha sido problema de Windows 7, las versiones anteriores de Windows funcionan perfectamente.

Tuvimos también problemas al principio de la implementación de la aplicación por el poco conocimiento de programación en PHP / MYSQL. Lo que se tardaría en hacer en una hora duraba el doble al no saber ciertos parámetros del lenguaje.

Comentar también que hubo dificultades para poder generar los archivos pdf. Pero como ya hemos comentado encontramos un módulo libre que nos solucionó el problema.

## PLANIFICACIÓN FINAL

---

La planificación previa que se planteó al principio del proyecto ha sufrido algún cambio, debido a que algunas fases han precisado más tiempo del que se había pensado en un principio. También se ha visto alterada por los exámenes, vacaciones y trabajo. Primeramente se tenía pensado acabarlo en junio pero se vio a mediados de mayo de que si queríamos que el proyecto fuera satisfactorio necesitábamos algún mes más. Por eso, el planning del informe previo presentado en abril cambia completamente con el que se ha presentado, que realmente es un cronograma del trabajo realizado estos últimos 6 meses.

## POSIBLES AMPLIACIONES

---

Como todo proyecto hay posibilidades de mejora y ampliación. En nuestro caso las ampliaciones vendrían más enfocados en los caso de estética y diseño. Se podría mejorar la estética de la página y la organización de los menús pero creímos que era más importante la funcionalidad.

Otras ampliaciones que no se han podido implementar y que se comentaron en su día con el director de proyecto, fueron las de poder insertar imágenes en las preguntas y la de tener la posibilidad de cambiar los formatos (tamaño de letra, color, etc...) de los ficheros de salida.

En general, como hemos dicho, son ampliaciones a nivel estético que no mejoran la funcionalidad de la aplicación.

## BIBLIOGRAFÍA

---

### Manual PHP:

- Spona, Helma. “*Programación de bases de datos con MySQL y PHP*”. Barcelona: Ediciones Marcombo, 2010. 210 p. ISBN: 978-84-267-1468-8.

-

### Manual HTML5:

- McCracken, Scott. “*HTML 5: Curso de Iniciación*”. Barcelona: Ediciones INFORBOOK’S S.L, 2011. 440p. ISBN: 978-84-15033-26-4.

### Páginas Web:

- Página oficial de Moodle. [Ultimo acceso el 8 de abril de 2011]  
<<http://moodle.org/>>.
- Página en español de exelearning. [Ultimo acceso el 12 de abril de 2011]  
<<http://wiki.exe-spain.es/doku.php>>.
- Manual de PHP / MySQL. [Ultimo acceso 30 de julio de 2011]  
<<http://www.desarrolloweb.com/php>>.

## ÍNDICE DE IMÁGENES

---

<b>IMAGEN 1.</b> DIAGRAMA DE GANTT .....	5
<b>IMAGEN 2.</b> TABLA DE PREGUNTAS.....	6
<b>IMAGEN 3.</b> TABLA DE USUARIOS.....	9
<b>IMAGEN 4.</b> TABLA DE NIVELES.....	11
<b>IMAGEN 5.</b> TABLA DE MATERIAS.....	11
<b>IMAGEN 6.</b> XAMPP.....	13
<b>IMAGEN 7.</b> PHPMYADMIN.....	14
<b>IMAGEN 8.</b> TABLAS BBDD.....	15
<b>IMAGEN 9.</b> PAGINA DE ACCESO.....	16
<b>IMAGEN 10.</b> PAGINA DE CONTROL DE USUARIOS .....	17
<b>IMAGEN 11.</b> ARCHIVO DE USUARIOS .....	18
<b>IMAGEN 12.</b> PAGINA DE NIVEL Y MATERIA.....	19
<b>IMAGEN 13.</b> PAGINA DE INSERCIÓN DE NIVEL Y MATERIA .....	20
<b>IMAGEN 14.</b> PAGINA DE BORRADO DE NIVEL Y MATERIA.....	20
<b>IMAGEN 15.</b> FICHEROS DE NIVEL Y MATERIA.....	21
<b>IMAGEN 16.</b> PAGINA DE INSERCIÓN DE USUARIOS.....	22
<b>IMAGEN 17.</b> PAGINA DE BORRADO DE USUARIOS.....	23
<b>IMAGEN 18.</b> PAGINA DE PRINCIPAL.....	24
<b>IMAGEN 19.</b> PAGINA DE INSERCIÓN DE PREGUNTAS TEST.....	26
<b>IMAGEN 20.</b> PAGINA DE INSERCIÓN DE PREGUNTAS DESCRIPTIVAS.....	27
<b>IMAGEN 21.</b> PAGINA DE CONSULTA DE PREGUNTAS.....	28
<b>IMAGEN 22</b> PAGINA DE SELECCIÓN DE PREGUNTAS.....	29
<b>IMAGEN 18.</b> FICHEROS DE PREGUNTAS Y RESPUESTAS.....	30

## TABLAS DE LA BBDD

---

questionaris.preguntas	
id	int(7)
pregunta	char(200)
materia	char(100)
nivell	char(100)
n_respostes	int(7)
resposta_1	char(200)
resposta_2	char(200)
resposta_3	char(200)
resposta_4	char(200)
resposta_5	char(200)
correcte	char(200)
tipus	char(200)

questionaris.materia	
id	int(7)
materia	char(50)

questionaris.usuari	
id	int(7)
nom	char(50)
cognoms	char(50)
identificador	char(50)

questionaris.nivell	
id	int(7)
nivell	char(50)



Memoria realizada por Jordi Martin Gil:

Bellaterra, 12 de septiembre de 2011

## **Resumen**

Este proyecto trata de dar solución a la generación de cuestionarios para la enseñanza. Se usa una aplicación web con cuestionarios almacenados en una base de datos, pudiendo insertar cuestionarios y buscar después a partir de ciertos campos de búsqueda.

Este proyecto pretende ser usado por personal docente que quiera tener un acceso rápido a cuestionarios para su uso académico.

## **Resum**

Aquest projecte tracta de donar solució a la generació de qüestionaris per a l'ensenyament. S'utilitza una aplicació web amb qüestionaris emmagatzemats en una base de dades, podent inserir qüestionaris i cercar posteriorment a partir de determinats camps.

Aquest projecte pretén ser utilitzat pel personal docent que vulgui tenir un accés ràpid a qüestionaris per un ús acadèmic.

## **Abstract**

This project tries to give solution to the generation of questionnaires for the education. A web application is used by questionnaires stored in a database, being able to insert questionnaires and search later from certain fields of search.

This project tries to be used by educational personnel that want to have a rapid access to questionnaires for his academic use.