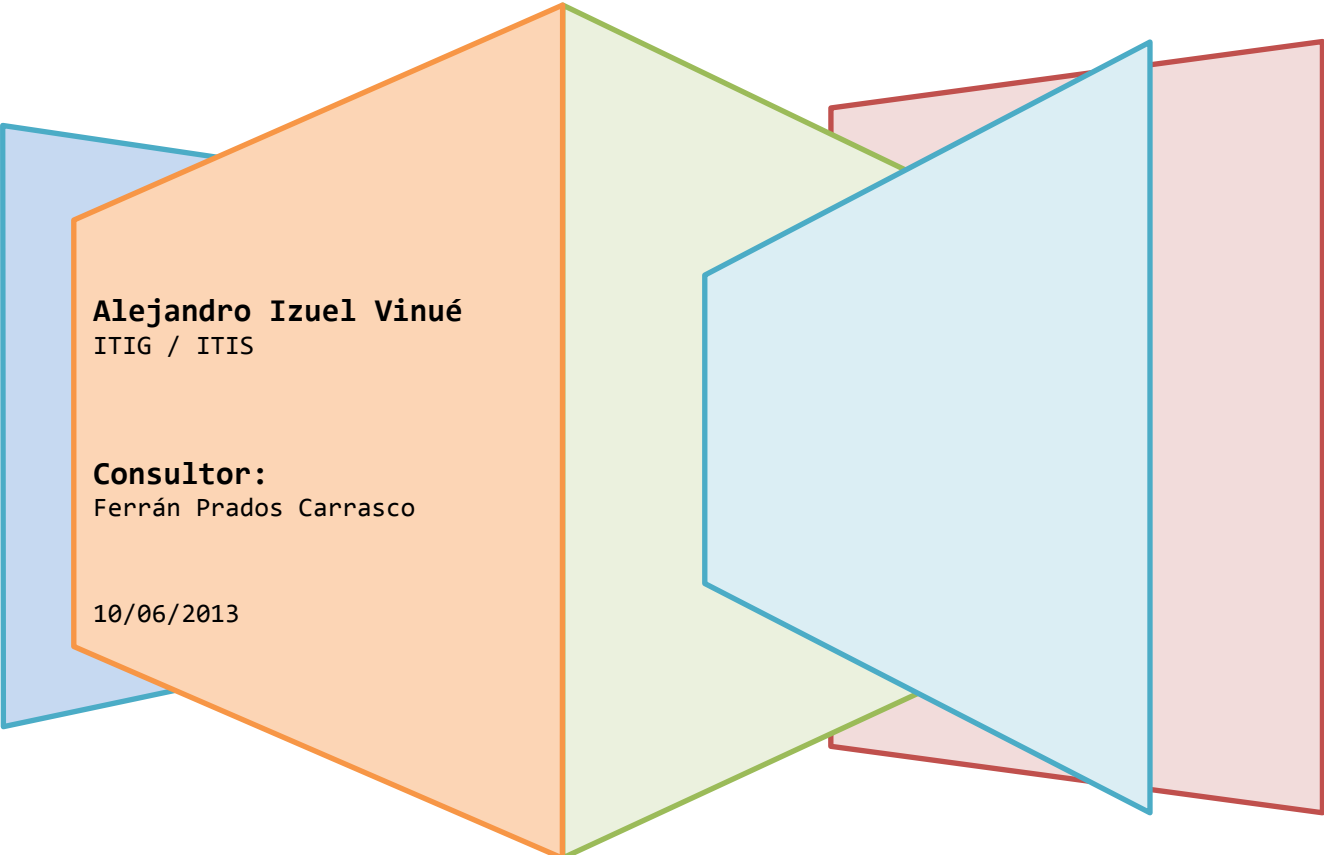


---

# Aplicaciones web para el trabajo colaborativo

---



**Alejandro Izuel Vinué**  
ITIG / ITIS

**Consultor:**  
Ferrán Prados Carrasco

10/06/2013

## Contenido

1	Resumen .....	3
2	Alcance del proyecto y objetivos .....	3
3	Metodología .....	4
4	Planificación .....	4
4.1	Plan de Trabajo .....	4
4.2	Especificación y Análisis .....	4
4.3	Diseño .....	5
4.4	Fase Final.....	5
4.5	Entrega final .....	5
5	Hitos principales .....	5
5.1	Plan de trabajo .....	5
5.2	Especificación y Análisis .....	5
5.3	Diseño .....	5
5.4	Implementación .....	5
5.5	Entrega Final .....	5
6	Planificación .....	6
7	Diagrama de Gantt .....	6
8	Requerimientos del sistema .....	8
8.1	Requerimientos Funcionales.....	8
8.2	Requerimientos no Funcionales.....	9
9	Tareas de los diferentes actores .....	9
9.1	Consultor.....	9
9.2	Alumno.....	10
10	Análisis de los subsistemas que debe tener el sistema.....	10
10.1	Sistema de diseño de formularios.....	10
10.2	Sistema de cumplimentación de formularios .....	11
10.3	Sistema de consulta de estadísticas.....	11
11	Diagrama de Casos de Uso .....	11
12	Diseño de la BBDD.....	12
12.1	Tabla Formularios .....	12
12.2	Tabla EstructuraFormulario .....	13
12.3	Tabla Elementos.....	13
12.4	Tabla Resultados .....	13
12.5	Tabla UsuariosFormularios .....	13
12.6	Tabla Usuarios.....	14
12.7	Tabla Archivos .....	14
12.8	Tabla Listas.....	14
12.9	Funciones SQL .....	14

13	Diseño de la parte Web.....	15
13.1	Páginas de navegación .....	16
13.2	Comunes .....	16
	• index.php .....	16
	• login.php .....	17
13.3	Profesores .....	17
	• index.php .....	17
	• gestion_formulario.php .....	18
	• asignar_alumnos.php.....	18
	• edit.php.....	19
	• edit_lista.php .....	19
	• nuevo_formulario.php.....	20
	• estadisticas.php .....	20
	• ver_formulario.php.....	21
13.4	Alumnos .....	21
	• login.php.....	21
	• registrar.php .....	21
	• index.php.....	22
	• estadisticas.php .....	22
	• resolver_formulario.php.....	22
13.5	Clases .....	22
	• class_usuario.php .....	22
13.6	Utilidades/Funciones .....	23
	• Funciones.php .....	23
	• Funciones.js .....	23
14	Implementación y pruebas.....	24
15	Implantación y resultados .....	24
16	Estimación económica.....	25
17	Conclusiones.....	26
18	Trabajo futuro .....	26
19	Bibliografía .....	27
20	Anexos(Tests) .....	27

## 1 Resumen

La motivación de haber escogido este área se debe a que viendo como está hoy en día el mundo de las aplicaciones web lo veo muy interesante ya que creo que se le pueden dar muchos y distintos usos, por lo que veo que será una aplicación muy útil globalmente hablando. A parte de ello, la programación web junto a las BBDD son las dos temáticas con las cuales he tratado y con las cuales veo muy interesante el desarrollar el proyecto.

También se puede escuchar mucho el tema “Clouding”, o trabajar en la nube, el cual se basa en disponer de un espacio en la red donde se encuentran alojados las aplicaciones a los que accedemos regularmente y de esta manera desde cualquier ordenador conectado a Internet y una vez hayamos pasado la política de seguridad adecuada podríamos ejecutar nuestra aplicación o documento o lo que sea que hayamos almacenado allí. Gracias a esta “revolución” muchas empresas y particulares están dando el salto al uso de internet también como un medio de trabajo el cual se puede compartir con las entidades o personas que se deseen.

## 2 Alcance del proyecto y objetivos

El proyecto final cumplirá con las funcionalidades demandadas por la UOC para gestionar los formularios rellenados por los alumnos y los resultados obtenidos serán procesados estadísticamente. Estos requisitos se entregarán en los plazos establecidos y respetarán las valoraciones realizadas para su desarrollo.

Objetivos del proyecto:

- Control y gestión de usuarios.
- Creación y edición de formularios.
- Accesos a BBDD (lectura y escritura).
- Extracción de resultados y procesado.
- Presentación de resultados.

El sistema se desarrollará para que sea compatible con cualquier Web.

## 3 Metodología

En cuanto a la metodología de desarrollo, he optado por intentar tener lo máximo organizado el código para que la programación aparte de ser más clara, fuese a su vez también más sencilla.

Desde el principio he tenido claras las funcionalidades que la aplicación debía de tener, y a partir de ello y basándome en las distintas PECs entregadas, he comenzado a programar el código necesario.

A medida que desarrollaba la aplicación, he encontrado problemas en los cuales en un inicio no había pensado pero que los he podido solventar mediante pequeñas modificaciones que no han afectado en gran medida al resto de la aplicación. A su vez también se me han ido ocurriendo nuevas ideas de cómo implementar los distintos módulos de la aplicación, como también pequeñas funcionalidades que en un principio no estaban pensadas, como el uso de JQuery a la hora de enviar los alumnos los distintos resultados.

## 4 Planificación

### 4.1 Plan de Trabajo

*Periodo 27 Febrero 10 de Marzo*

- Localización del software GNU necesario para la realización del proyecto.
- Estudio del software. Entender la problemática a desarrollar, contemplando la posterior manipulación de los datos extraídos de los informes realizados por los miembros del grupo.
- Desarrollo del documento Plan de Trabajo.

### 4.2 Especificación y Análisis

*Periodo 11 Marzo 25 de Marzo*

- Estudiar los límites del problema y el marco dónde se encuentra
- Analizar los requerimientos no funcionales: de rendimiento, de distribución, de seguridad y de usabilidad
- Definir los diferentes actores que se ven implicados junto a sus tareas y/o actividades.
- Análisis de los subsistemas que debe tener el sistema
- Análisis de los requerimientos funcionales con los diferentes diagramas y fichas de casos de uso

### 4.3 Diseño

*Periodo 26 Marzo 14 de Abril*

- Los diagramas de actividades y/o de secuencia de los diferentes procesos
- El análisis de la jerarquía de paquetes, diagrama de clases, comunicación entre clases y/o utilización de patrones

### 4.4 Fase Final

*Periodo 15 Abril 6 de Junio*

- Implementación
- Pruebas

### 4.5 Entrega final

*Periodo 7 Junio 10 de Junio*

- Revisión final de la Memoria
- Valoración del trabajo realizado
- Preparación de la presentación virtual.

## 5 Hitos principales

### 5.1 Plan de trabajo

Al estar de vacaciones, el plan de proyecto lo he tenido que hacer con demasiada prisa y por lo tanto sólo he dedicado 7,94 días.

### 5.2 Especificación y Análisis

La fecha de entrega es el 25 de Marzo, Teniendo en cuenta que hay que realizar la PEC2, he estimado que dedicaré cerca de 8,81 días.

### 5.3 Diseño

Es una parte muy importante del proyecto y se debe realizar de forma concienzuda y de la forma más precisa, por lo tanto dedicaré 11,94 días.

### 5.4 Implementación

La implementación también es una parte importante del proyecto junto a la documentación y presentación final, por ello estimo que dedicaré 28,06 días.

### 5.5 Entrega Final

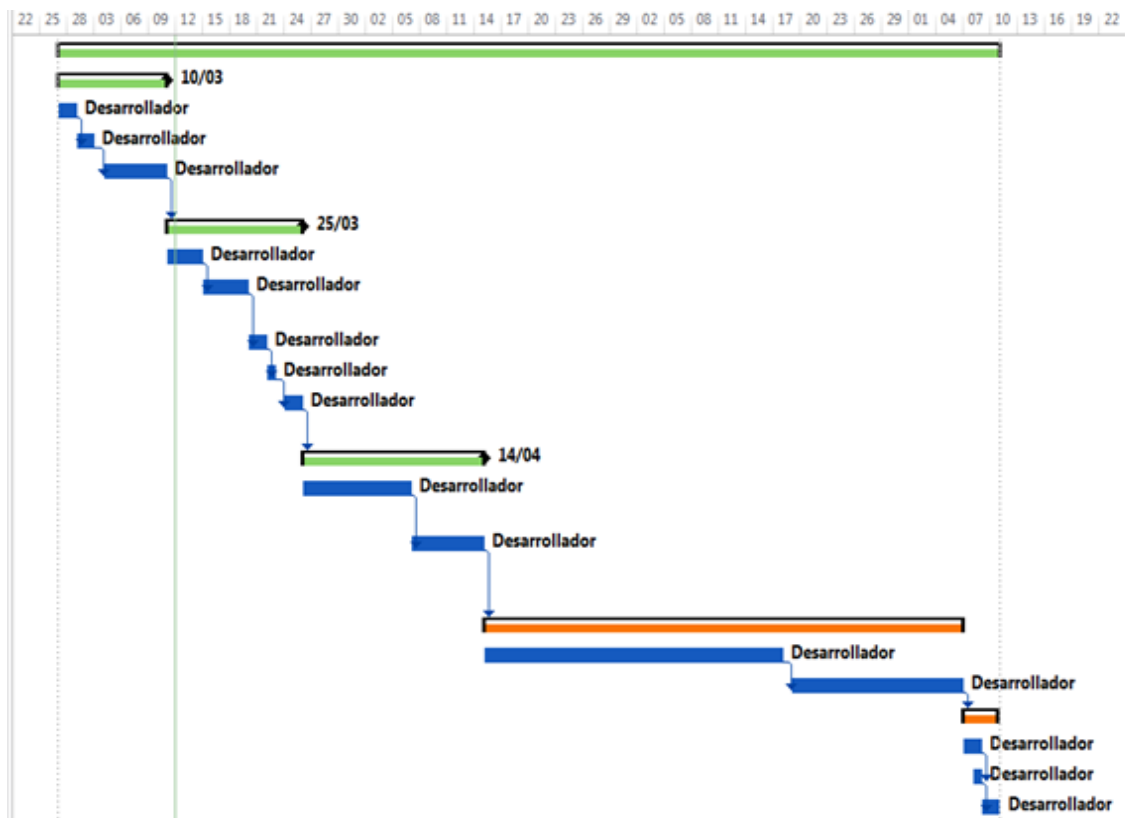
La parte final del proyecto y no por ello la menos importante, deberé terminar la memoria y realizar la grabación de la presentación, al ser una tarea que puede ser compleja, he estimado que dedicaré 2,44 días.

## 6 Planificación

Nombre de tarea	Duración	Comienzo	Fin	Duración	Predecesoras	Nombres de los recursos
<b>4 Construcción del Software</b>	<b>54,06 días</b>	<b>mié 27/02/13</b>	<b>lun 10/06/13</b>	<b>54,06 días</b>		
<b>4 Plan de Trabajo</b>	<b>7,94 días</b>	<b>mié 27/02/13</b>	<b>dom 10/03/13</b>	<b>7,94 días</b>		<b>Desarrollador</b>
Localización del software	1,94 días	mié 27/02/13	jue 28/02/13	1,94 días		Desarrollador
Estudio del software	3 días	vie 01/03/13	dom 03/03/13	3 días	3	Desarrollador
Desarrollo del documento Plan de Trabajo	3,94 días	lun 04/03/13	dom 10/03/13	3,94 días	4	Desarrollador
<b>4 Especificación y Análisis</b>	<b>8,81 días</b>	<b>lun 11/03/13</b>	<b>lun 25/03/13</b>	<b>8,81 días</b>	<b>5</b>	<b>Desarrollador</b>
Estudiar los límites del problema	1,56 días	lun 11/03/13	jue 14/03/13	1,56 días		Desarrollador
Analizar los requerimientos no funcionales	2,94 días	vie 15/03/13	mar 19/03/13	2,94 días	7	Desarrollador
Definir los diferentes actores	0,81 días	mié 20/03/13	jue 21/03/13	0,81 días	8	Desarrollador
Análisis de los subsistemas	0,88 días	vie 22/03/13	sáb 23/03/13	0,88 días	9	Desarrollador
Análisis de los requerimientos funcionales	0,56 días	dom 24/03/13	lun 25/03/13	0,56 días	10	Desarrollador
<b>4 Diseño</b>	<b>11,94 días</b>	<b>mar 26/03/13</b>	<b>dom 14/04/13</b>	<b>11,94 días</b>	<b>11</b>	<b>Desarrollador</b>
Los diagramas de actividades y/o secuencia	7,44 días	mar 26/03/13	sáb 06/04/13	7,44 días		Desarrollador
Análisis de la jerarquía de paquetes, diagrama de clases, comunicación entre clases	4,81 días	dom 07/04/13	dom 14/04/13	4,81 días	13	Desarrollador
<b>4 Implementación</b>	<b>28,06 días</b>	<b>lun 15/04/13</b>	<b>jue 06/06/13</b>	<b>28,06 días</b>	<b>14</b>	<b>Desarrollador</b>
Implementación	16,88 días	lun 15/04/13	sáb 18/05/13	16,88 días		Desarrollador
Pruebas	8,94 días	dom 19/05/13	jue 06/06/13	8,94 días	16	Desarrollador
<b>4 Entrega Final</b>	<b>2,44 días</b>	<b>vie 07/06/13</b>	<b>lun 10/06/13</b>	<b>2,44 días</b>	<b>17</b>	<b>Desarrollador</b>
Revisión de la memoria	12,5 horas	vie 07/06/13	sáb 08/06/13	12,5 horas		Desarrollador
Valoración del trabajo realizado	3,5 horas	sáb 08/06/13	sáb 08/06/13	3,5 horas	19	Desarrollador
Preparación de la presentación virtual	0,56 días	dom 09/06/13	lun 10/06/13	0,56 días	20	Desarrollador

## 7 Diagrama de Gantt

Nombre de tarea	Duración	Comienzo	Fin	Duración	Predecesoras
<b>4 Construcción del Software</b>	<b>54,06 días</b>	<b>mié 27/02/13</b>	<b>lun 10/06/13</b>	<b>54,06 días</b>	
<b>4 Plan de Trabajo</b>	<b>7,94 días</b>	<b>mié 27/02/13</b>	<b>dom 10/03/13</b>	<b>7,94 días</b>	
Localización del software	1,94 días	mié 27/02/13	jue 28/02/13	1,94 días	
Estudio del software	3 días	vie 01/03/13	dom 03/03/13	3 días	3
Desarrollo del documento Plan de Trabajo	3,94 días	lun 04/03/13	dom 10/03/13	3,94 días	4
<b>4 Especificación y Análisis</b>	<b>8,81 días</b>	<b>lun 11/03/13</b>	<b>lun 25/03/13</b>	<b>8,81 días</b>	<b>5</b>
Estudiar los límites del problema	1,56 días	lun 11/03/13	jue 14/03/13	1,56 días	
Analizar los requerimientos no funcionales	2,94 días	vie 15/03/13	mar 19/03/13	2,94 días	7
Definir los diferentes actores	0,81 días	mié 20/03/13	jue 21/03/13	0,81 días	8
Análisis de los subsistemas	0,88 días	vie 22/03/13	sáb 23/03/13	0,88 días	9
Análisis de los requerimientos funcionales	0,56 días	dom 24/03/13	lun 25/03/13	0,56 días	10
<b>4 Diseño</b>	<b>11,94 días</b>	<b>mar 26/03/13</b>	<b>dom 14/04/13</b>	<b>11,94 días</b>	<b>11</b>
Los diagramas de actividades y/o secuencia	7,44 días	mar 26/03/13	sáb 06/04/13	7,44 días	
Análisis de la jerarquía de paquetes, diagrama de clases, comunicación entre clases	4,81 días	dom 07/04/13	dom 14/04/13	4,81 días	13
<b>4 Implementación</b>	<b>28,06 días</b>	<b>lun 15/04/13</b>	<b>jue 06/06/13</b>	<b>28,06 días</b>	<b>14</b>
Implementación	16,88 días	lun 15/04/13	sáb 18/05/13	16,88 días	
Pruebas	8,94 días	dom 19/05/13	jue 06/06/13	8,94 días	16
<b>4 Entrega Final</b>	<b>2,44 días</b>	<b>vie 07/06/13</b>	<b>lun 10/06/13</b>	<b>2,44 días</b>	<b>17</b>
Revisión de la memoria	12,5 horas	vie 07/06/13	sáb 08/06/13	12,5 horas	
Valoración del trabajo realizado	3,5 horas	sáb 08/06/13	sáb 08/06/13	3,5 horas	19
Preparación de la presentación virtual	0,56 días	dom 09/06/13	lun 10/06/13	0,56 días	20



Los errores que se pueden observar de las fechas son debidos a días no laborables. Dado que yo voy a ser el único desarrollador del proyecto, he pasado por alto este detalle y he calculado una media de horas al día de trabajo durante el periodo de desarrollo del proyecto.

Las horas de trabajo en concreto son, de Lunes a Viernes de 18:00 a 21:00 y Sábados y Domingos de 10:00 a 13:00 y de 15:00 a 19:00



## 8 Requerimientos del sistema

### 8.1 Requerimientos Funcionales

- El sistema debe permitir generar formularios desde una página web que será de acceso exclusivo para los profesores.
- Los formularios deberán de poder ser consultados por los alumnos para realizar el envío de los resultados.
- Existirá la opción de modificar los resultados siempre y cuando no se realice el “envío final”.
- Una vez enviados los resultados no habrá opción de poder volver a enviarlos, por lo que una vez que se envíe el formulario relleno no se dispondrá de la posibilidad de modificación.
- Los diseños de los distintos formularios deberán ser guardados en la BBDD para que así puedan ser consultados desde la página web.
- Existirán procedimientos almacenados en la Base de Datos que se encargarán de generar información estadística de los aciertos y/o fallos que han tenido las soluciones enviadas por los alumnos.
- Las estadísticas generadas se guardarán en la Base de Datos y existirán de dos tipos: públicas y privadas.
- A la hora de consultar los informes estadísticos se diferenciarán dos tipos de usuarios: alumnos y consultores.
- Las cuentas de los alumnos se podrán crear desde la página principal.
- Las cuentas de los profesores serán administradas por el administrador de la aplicación y serán creadas manualmente en la BBDD

## 8.2 Requerimientos no Funcionales

- Al ser una herramienta que va a tener un volumen de trabajo no muy grande, el servidor donde deba estar instalado no requerirá ser muy potente. Hay que tenerlo en cuenta a la hora de diseñar la base de datos y la periodificación de procesos.
- El usuario dispondrá de una documentación con todas las funcionalidades implementadas, un correcto manual de usuario
- El servidor necesitará tener instalado:
  - Apache
  - PHP
  - MySql
- El usuario dispondrá de una documentación con todas las funcionalidades implementadas y manual de usuario.
- Base de datos:
  - Se debe optimizar la base de datos para que no se produzcan problemas de rendimiento.

# 9 Tareas de los diferentes actores

## 9.1 Consultor

Los consultores serán los únicos usuarios con acceso a la creación de los formularios que más tarde rellenaran los alumnos.

Estos definirán también a, la hora de crear el formulario, si las estadísticas serán públicas o privadas.

La creación del formulario la realizarán definiendo los distintos campos que este debe tener, de que tipo serán, que “Labels” acompañarán al campo y si debe tener un valor por defecto el campo para poder insertar comentarios.

A la consulta de estadísticas se podrá acceder desde la página principal. En ella se podrán ver los porcentajes de aciertos y fallos de los formularios, así como qué opciones de las listas han sido las más respondidas.

## 9.2 Alumno

Los alumnos accederán con su usuario y contraseña a la aplicación y desde allí podrán ver que formularios pueden rellenar. A su vez, desde esta pantalla podrán ver que estadísticas pueden consultar.

Los alumnos podrán rellenar las soluciones en el formulario y enviarlas/modificarlas tantas veces como quieran siempre y cuando el formulario no haya sido finalizado.

Una vez haya accedido al formulario, tendrá las opciones de guardar, para los distintos campos, o finalizar formulario. Guardar, almacenará los valores rellenados en los campos para poder continuar con el formulario en un futuro. Una vez proceda a finalizar el formulario ya no podrá modificar los resultados enviados en éste.

La consulta de estadísticas se realizará como con los profesores, como he comentado anteriormente, a diferencia de que los alumnos solamente podrán acceder a las públicas.

# 10 Análisis de los subsistemas que debe tener el sistema

La aplicación Web contendrá tres subsistemas definidos,

- Sistema de diseño de formularios
- Sistema de cumplimentación de formularios
- Sistema de consulta de estadísticas

## 10.1 Sistema de diseño de formularios

El sistema de diseño de formularios consistirá en una pantalla a la que solo tendrán acceso los profesores y en la cual se definirán los diferentes campos del formulario, así como qué alumnos pueden acceder a él y qué tipo de estadísticas tendrá, públicas o privadas.

La interfaz para el diseño del formulario se basará en campos desplegables en los cuales se parametrizará que tipo de campo se quiere insertar, en qué posición debe aparecer, el resultado que los alumnos deberán de enviar, que etiqueta tendrá y que valor por defecto tendrá el campo si se desea.

Existirá también la opción de per visualizar el formulario mediante la cual se podrá visualizar como será visto el formulario por los alumnos.

## 10.2 Sistema de cumplimentación de formularios

Los alumnos accederán a este módulo mediante la pantalla principal en la cual podrán ver los formularios a los que pueden acceder.

Una vez seleccionado el formulario deseado, accederán a la pantalla de cumplimentación, dónde deberán rellenar las distintas preguntas con los resultados que desee. Siempre tendrán la posibilidad de guardar las respuestas antes de realizar el envío final.

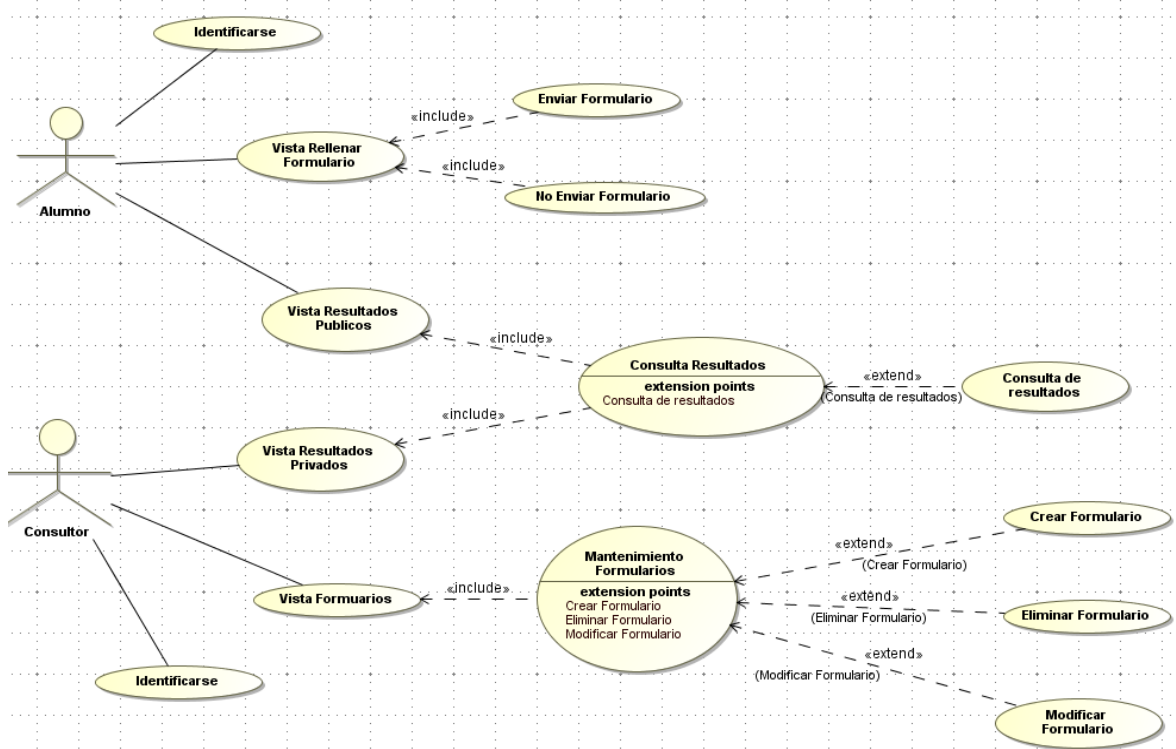
Una vez realizado el envío de las respuestas no podrán reenviar el formulario con respuestas modificadas.

## 10.3 Sistema de consulta de estadísticas

A la consulta de estadísticas podrán acceder los profesores y los alumnos si se ha definido anteriormente el formulario como público.

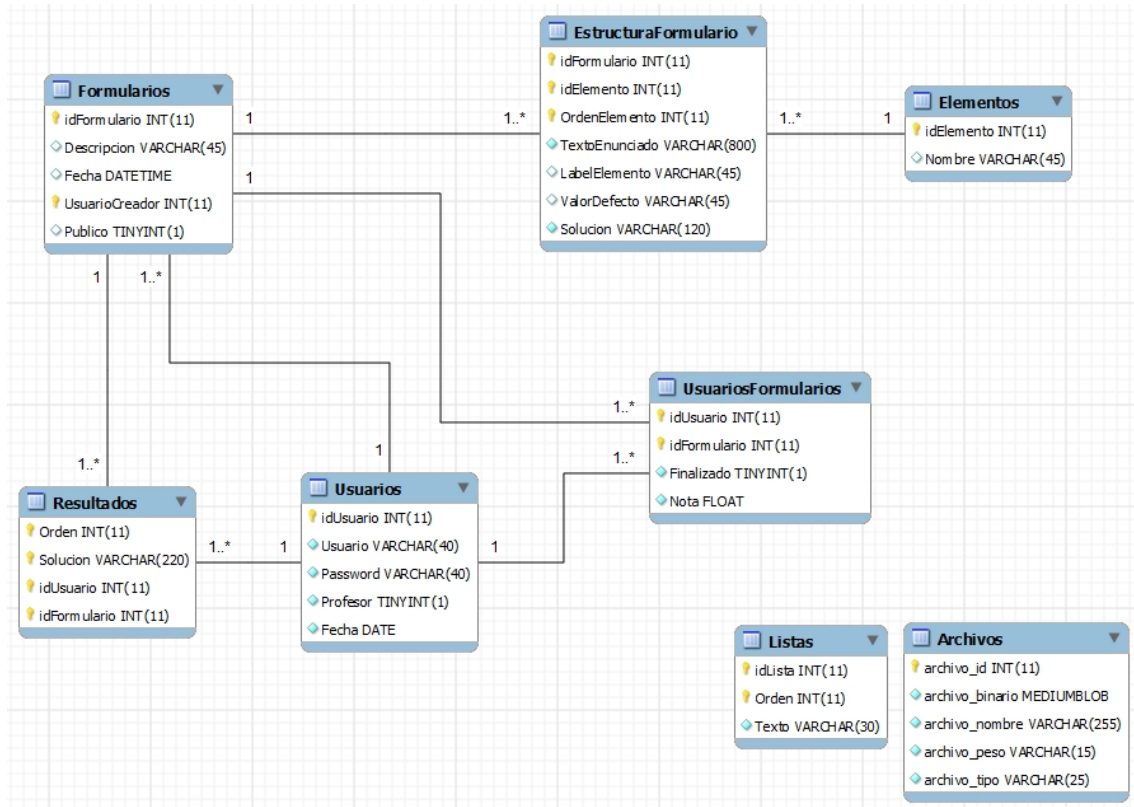
En ella se mostrarán los porcentajes de aprobados y suspendidos, así como también que respuestas han sido las más seleccionadas para los campos que tuviesen distintas opciones a elegir.

# 11 Diagrama de Casos de Uso



## 12 Diseño de la BBDD

Las distintas relaciones establecidas entre las tablas de la BBDD se pueden observar en la siguiente imagen, así como también las claves primarias:



### 12.1 Tabla Formularios

En la tabla Formularios se guardará la información básica de cada distinto formulario que se diseñe.

- **idFormulario INT**, clave primaria numérica del formulario
- **Descripcion VARCHAR(45)**, breve descripción del formulario
- **Fecha DATETIME**, fecha de creación
- **UsuarioCreador TEXT**, usuario creador
- **Publico TINYINT**, booleano para marcar si las estadísticas serán públicas o privadas

## 12.2 Tabla EstructuraFormulario

En la tabla EstructuraFormulario se almacenarán las distintas estructuras de los formularios creados.

- **idFormulario** INT, clave foránea de la tabla Formularios
- **idElemento** INT, clave foránea de la tabla Elementos
- **OrdenElemento** INT, indica en que posición aparecerá la pregunta
- **TextoEnunciado** VARCHAR(800), texto del enunciado de la pregunta
- **LabelElemento** VARCHAR(45), etiqueta de la pregunta
- **ValorDefecto** VARCHAR(45), valor por defecto que aparecerá en dicho campo
- **Solucion** VARCHAR(120), solución de la pregunta

## 12.3 Tabla Elementos

La tabla Elementos contendrá los distintos tipos de campos que se pueden insertar en el formulario.

- **idElemento** INT, clave primaria
- **Nombre** VARCHAR(45), texto descriptivo del elemento.

## 12.4 Tabla Resultados

En la tabla Resultados quedarán reflejados los resultados que definan los profesores para los determinados campos.

- **Orden** INT, posición del campo al que pertenece el resultado
- **idUsuario** INT, id del usuario que ha mandado esa solución
- **idFormulario** INT, id del formulario al que pertenece el resultado
- **Solucion** VARCHAR(220), resultado enviado por el usuario

Mediante los campos Orden, idUsuario e idFormulario se forma la clave primaria que nos permitirá distinguir cada una de las distintas soluciones enviadas.

## 12.5 Tabla UsuariosFormularios

La tabla usuarios almacenará los distintos usuarios, alumnos, asignados a los distintos formularios.

- **idUsuario** INT, id del usuario estudiante al que se le asigna el formulario
- **idFormulario** INT, id del formulario que se asigna al estudiante
- **Finalizado** TINYINT, indicador de si está finalizado el formulario o no
- **Nota** FLOAT, nota que tiene el usuario en el formulario

## 12.6 Tabla Usuarios

La tabla usuarios almacenará los distintos usuarios.

- **idUsuario INT**, clave primaria del usuario
- **Usuario VARCHAR(40)**, nombre del usuario
- **Password VARCHAR(40)**, contraseña del usuario cifrada en **SHA1**
- **Profesor TINYINT**, indicador de si es profesor o no
- **Fecha DATE**, fecha de creación

A la hora de las contraseñas he optado por el cifrado SHA1 ya que el MD5 hoy en día ya no es tan seguro.

## 12.7 Tabla Archivos

La tabla usuarios almacenará las imágenes relacionadas con los campos.

- **archivo\_id INT**, clave primaria formada por los valores *idformulario*, *idElemento* y *Orden*
- **archivo\_binario MEDIUMBLOB**, información binaria de la imagen
- **archivo\_nombre VARCHAR(225)**, nombre de la imagen
- **archivo\_peso VARCHAR(15)**, tamaño de la imagen
- **archivo\_tipo VARCHAR(25)**, formato de imagen

En esta tabla decidí formar manualmente la clave primaria y no crear tres campos por facilidad a la hora de programar y referenciar a esta clave. Lo óptimo para tener las relaciones de la BBDD perfectas habría sido crear campos individuales, pero por motivo de tiempo no se ha podido realizar la modificación ya que estaba el proyecto muy avanzado.

## 12.8 Tabla Listas

La tabla usuarios almacenará los distintos usuarios.

- **idLista INT**, clave primaria formada por los valores *idformulario*, *idElemento* y *Orden*
- **Orden INT**, orden del valor en la lista
- **Texto VARCHAR(30)**, texto del valor

Por la misma razón que en la tabla Archivos decidí formar la clave primaria manualmente.

## 12.9 Funciones SQL

Para calcular si un alumno ha enviado una respuesta correcta o no, en los casos en los que se trata de un campo de texto, se utiliza la función **levenshtein** y **levenshtein\_ratio**.

Esta función calcula la “distancia” entre dos cadenas, lo cual se traduce en que devuelve la cantidad de modificaciones que es necesario hacer en una cadena origen para que sea igual a la cadena establecida como base.

La función de ayuda usada `levenshtein_ratio` devuelve en porcentaje el grado de aproximación a la respuesta correcta que ha enviado el usuario, comparando la respuesta del usuario con la solución oficial.

## 13 Diseño de la parte Web

Las distintas páginas del proyecto se pueden separar entre páginas de “navegación” y páginas “llamada”. Las páginas de navegación son aquellas a las que el usuario, ya sea profesor o alumno, accede y trabaja con la aplicación. Las páginas *llamada* son aquellas que son llamadas desde otra web únicamente para recibir parámetros y ejecutar una funcionalidad.

En el caso de navegación tendríamos las siguientes:

- [index.php](#)
- [login.php](#)
- [gestion\\_formulario.php](#)
- [nuevo\\_formulario.php](#)
- [edit.php](#)
- [edit\\_lista.php](#)
- [estadisticas.php](#)
- [asignar\\_alumnos.php](#)
- [ver\\_formulario.php](#)
- [registrar.php](#)
- [resolver\\_formulario.php](#)

Las páginas de llamada serían las siguientes:

- `class_usuario.php`
- `delete_alumnoformulario.php`
- `delete_elemento.php`
- `delete_formulario.php`
- `delete_lista.php`
- `enviar_resultado.php`
- `limpiar_resultado.php`
- `finalizar_formulario.php`
- `logout.php`
- `session_control.php`
- `subirfoto.php`
- `validar_usuario.php`
- `ver.php`
- `ver_thumbnail.php`

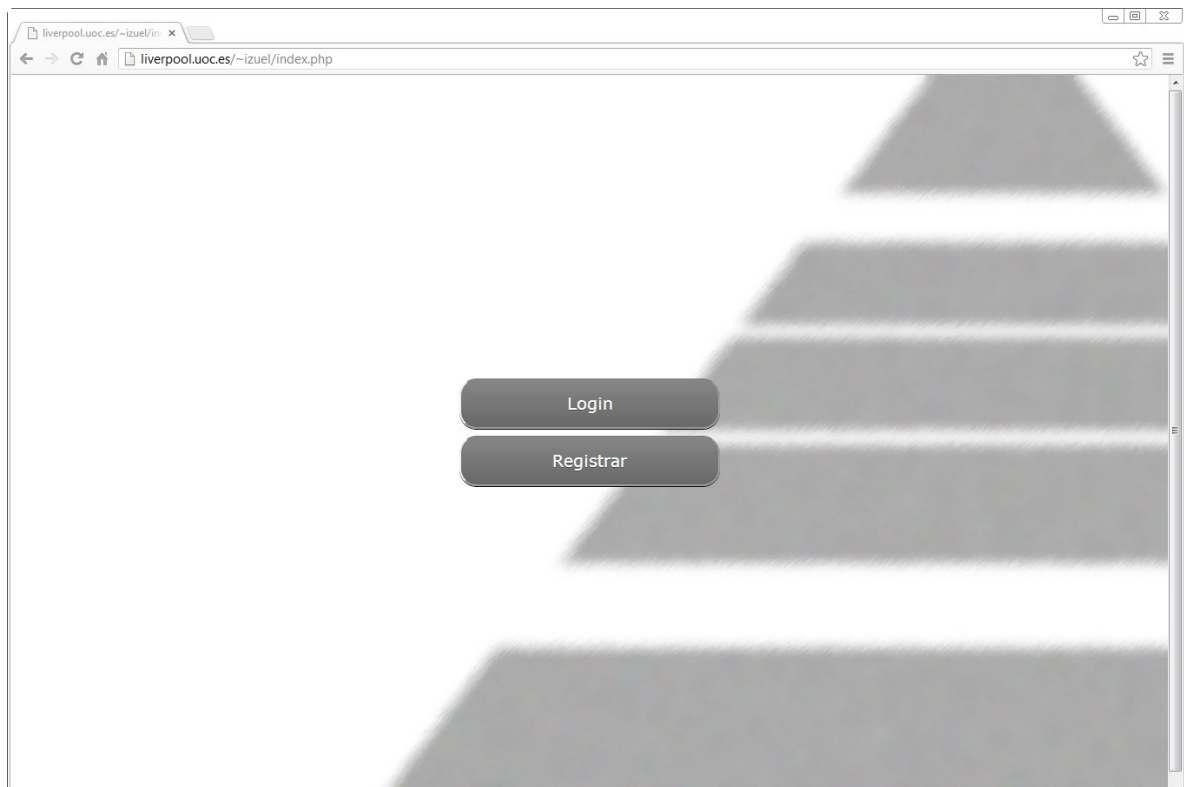


## 13.1 Páginas de navegación

A continuación se muestran las distintas páginas web que los usuarios utilizarán en la aplicación, separadas por la visualización/uso que tendrán profesores y alumnos.

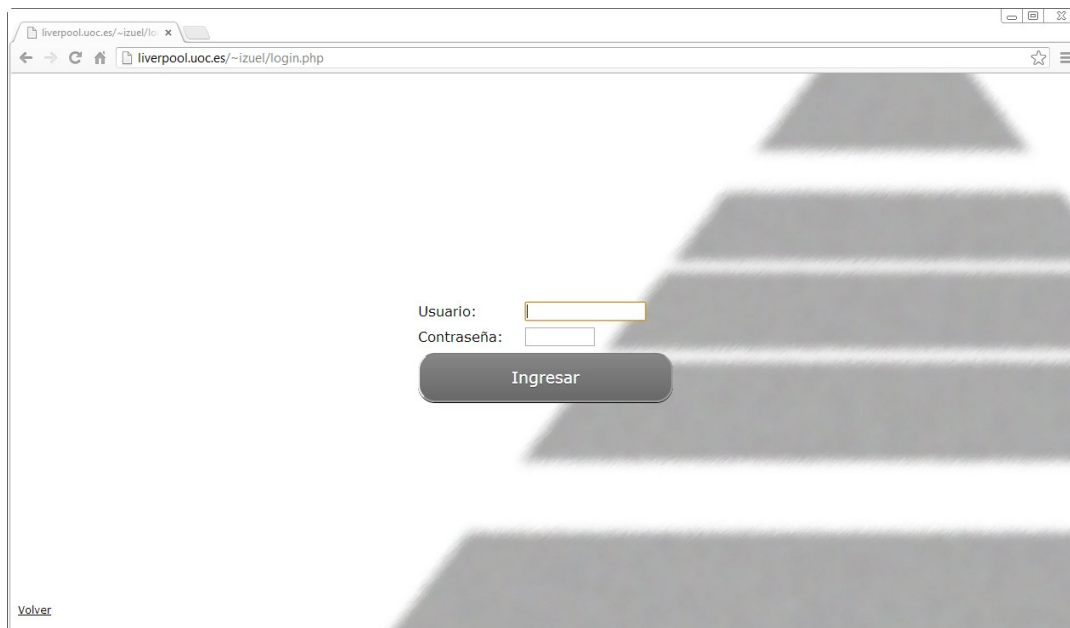
## 13.2 Comunes

- [index.php](#)



Página desde la que ingresarán los distintos usuarios y desde la cual se realizarán los nuevos registros de los alumnos.

- [login.php](#)



Página en la que se verifica el usuario y la contraseña introducidos.

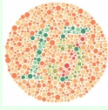
### 13.3 [Profesores](#)

- [index.php](#)



Una vez logueados los profesores tendrán una lista con los distintos formularios que hayan creado y la opción de crear uno nuevo mediante el botón Añadir Formulario. Así mismo esta es la única pantalla desde la que podrán realizar el logout de su cuenta.

- [gestion\\_formulario.php](#)

Pos.	Tipo de Campo	Enunciado	Etiqueta	Valor por defecto	Solucion	Imagen
1	Cuadro de Texto grande	Enuncia el teorema de Pitágoras	El teorema de Pitágoras dice que:	En todo ...	No hay imagen asociada	<a href="#">Editar</a> <a href="#">Eliminar</a>
2	Diversas opciones	Cuanto es 2*2	Solución: 1	4...	No hay imagen asociada	<a href="#">Editar</a> <a href="#">Eliminar</a>
3	Texto	Qué numero hay en la siguiente imagen?	El número es:	15...		<a href="#">Editar</a> <a href="#">Eliminar</a>

Una vez seleccionado un formulario, accederán a una lista de todos los distintos elementos que lo componen y donde podrán ver una pequeña descripción de dichos campos.

Desde esta página tendrán acceso a Añadir Campo, Asignar Alumnos, Pervisualizar diseño, Ver Estadísticas y Eliminar Formulario

- [asignar\\_alumnos.php](#)

Nombre
Laura <a href="#">Eliminar</a>
Marta <a href="#">Eliminar</a>
Juan <a href="#">Eliminar</a>
Pepe <a href="#">Eliminar</a>

Añadir alumnos:  [Añadir](#)

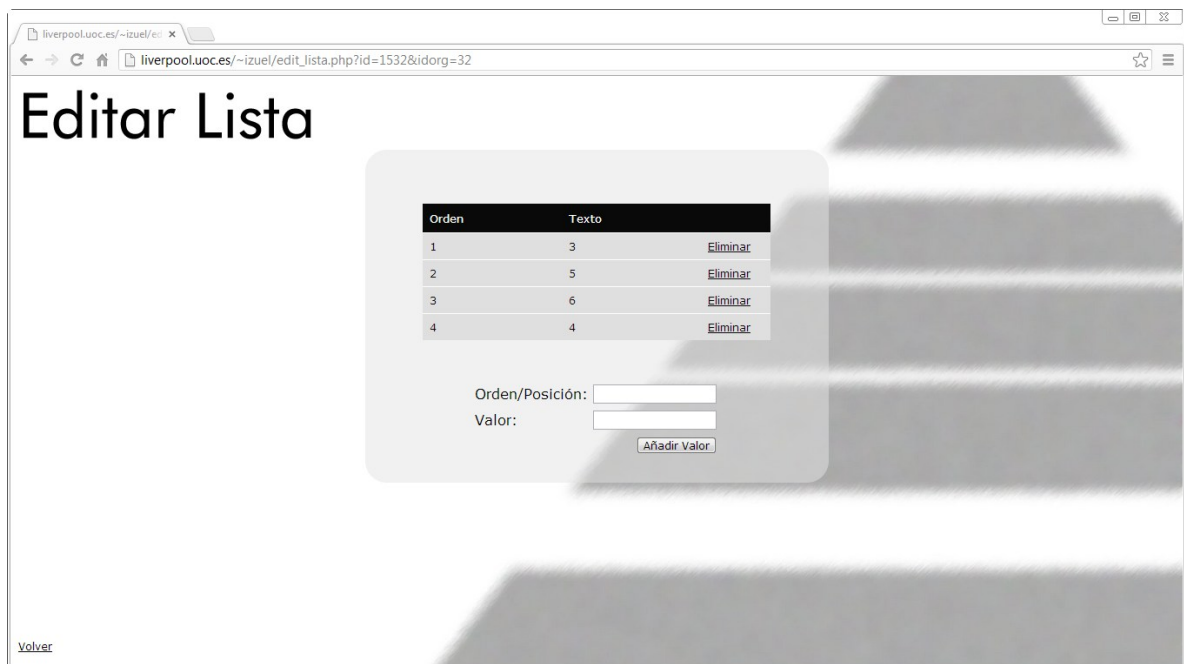
Desde esta Web se definirán los alumnos que van a tener acceso a realizar dicho formulario.

- [edit.php](#)



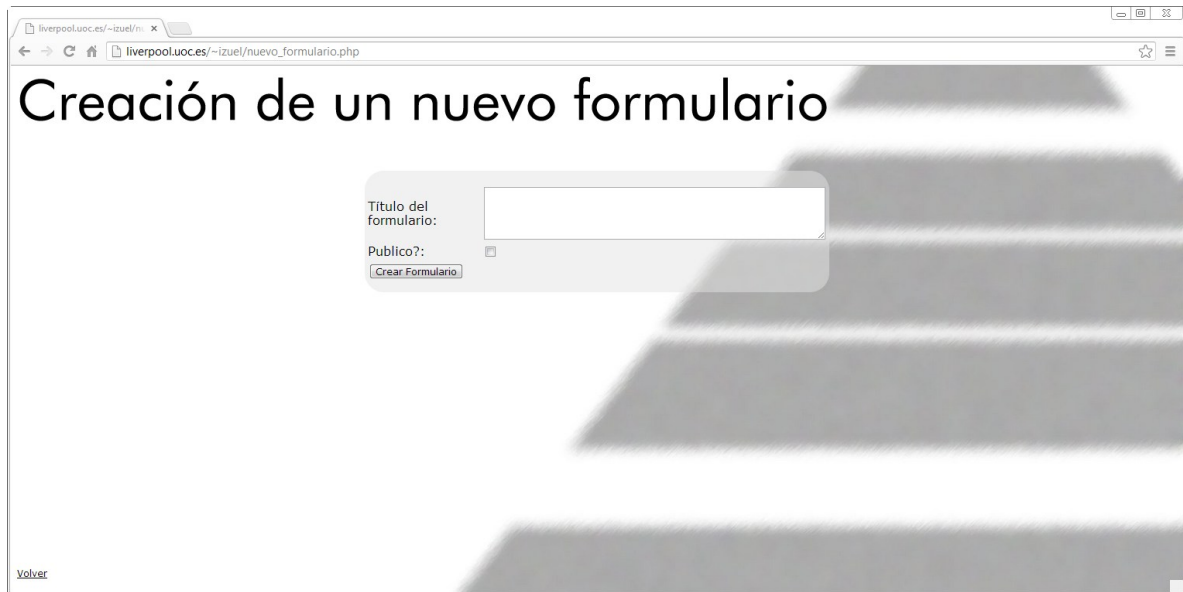
En esta página se editarán las distintas propiedades del campo seleccionado del formulario. Se podrá definir la posición, el tipo de campo que se quiere añadir, su enunciado, etiqueta que aparecerá delante del input, un valor por defecto (opcional), la solución a dicho campo y también se podrá asignar una imagen a dicho campo.

- [edit\\_lista.php](#)



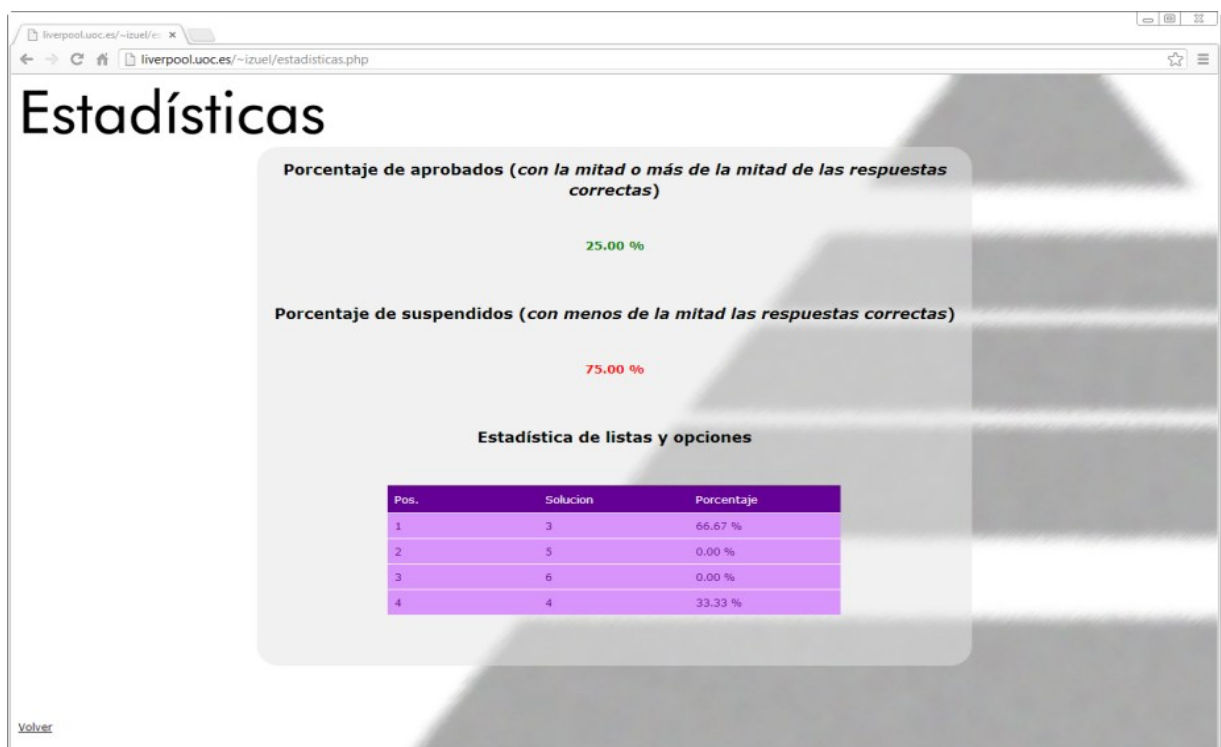
Aquí es donde se definirán los distintos valores que tendrán los campos de tipo lista o radiobuttons.

- [nuevo\\_formulario.php](#)



A la hora de definición de un nuevo formulario se podrá asignar una breve descripción y si las estadísticas de éste serán públicas o privadas.

- [estadisticas.php](#)



En esta web se consultarán las distintas estadísticas del formulario seleccionado, como el porcentaje de suspensos y de aprobados.

- [ver\\_formulario.php](#)

liverpool.uoc.es/~izuel/ver\_formulario.php

## Pervisualización

**Pregunta 1**  
Enuncia el teorema de Pitágoras

El teorema de Pitágoras dice que:

**Pregunta 2**  
Cuanto es  $2 \cdot 2$   
Solución: 3 5 6 4

**Pregunta 3**  
Qué número hay en la siguiente imagen?  
El número es:

Volver

Esta Web simulará cómo los alumnos van a visualizar el formulario a la hora de resolverlo.

### 13.4 [Alumnos](#)

- [login.php](#)  
(Misma función que con los profesores)
- [registrar.php](#)

liverpool.uoc.es/~izuel/registrar.php

## Alta Nuevo Usuario

Usuario (max 20):

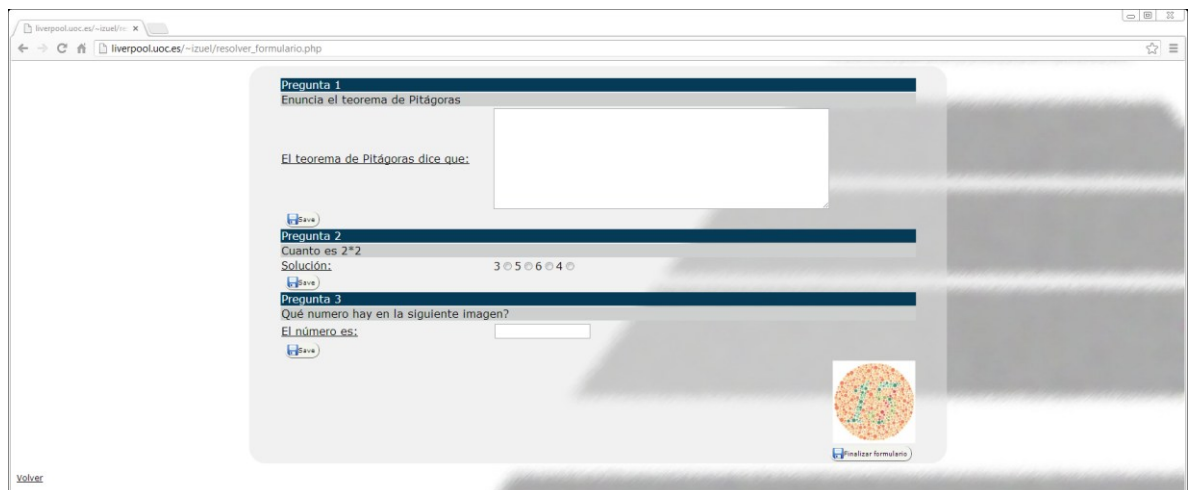
Password (max 10):

Confirmar:

Volver

Página donde se realiza el registro de los nuevos alumnos mediante el nombre de usuario y la contraseña.

- [index.php](#)  
(Misma función que con los profesores)
- [estadisticas.php](#)  
(Misma función que con los profesores)
- [resolver\\_formulario.php](#)



Web en la que los distintos alumnos deberán responder a las preguntas del formulario. Las soluciones se podrán enviar siempre y cuando no se finalice el formulario.

### 13.5 Clases

A continuación la clase creada para validación y registro de los usuarios,

- [class\\_usuario.php](#)  
Esta clase será utilizada a la hora de crear y/o registrar usuarios en la aplicación. Contiene los métodos
  - `_construct($nombre_usuario,$password)`, método constructor.
  - `validar_usuario($usuario, $password)`, realiza la comprobación de si existe el usuario en la BBDD
  - `registrar_usuario($usuario, $password)`, registra el nuevo usuario en la BBDD

## 13.6 [Utilidades/Funciones](#)

Todas las funciones, tanto PHP como JavaScript, que son utilizadas por las distintas páginas, han sido agrupadas en dos distintos archivos. Se localizan dentro de la carpeta `/utils` del servidor Web, y sus nombres son `Funciones.js` y `Funciones.php`.

Estos dos archivos son llamados desde las distintas páginas, cuando son utilizados, mediante los siguientes includes:

```
<script src="utils/Funciones.js"></script>  
  
include("utils/Funciones.php");
```

A continuación la descripción de ambos archivos y la explicación de sus distintas funciones:

- **Funciones.php**
  - **function formRegistro()**, función encargada de pintar el formulario de registro en la web [registrar.php](#)
  - **js\_str(\$s)** y **js\_array(\$array)**, funciones encargadas de traducir un array de PHP a JavaScript, usadas en [edit.php](#)
- **Funciones.js**
  - **function enviar(tipo,pregunta,solucion)**, función encargada de insertar la información en la BBDD mediante JQuery. [resolver\\_formulario.php](#)
  - **function finalizarFormulario()**,Función que realiza la ejecución de finalización del formulario. [resolver\\_formulario.php](#)
  - **Validar\_alumnos()**,función encargada de comprobar si se ha seleccionado un alumno de la lista en [asignar\\_alumnos.php](#)
  - **function subm(formulario)**, realizar submit de formulario sin mostrar mensaje de aviso en [edit.php](#)
  - **function hacer\_visible(elemento)**, función encargada de mostrar o no mostrar el acceso a editar valores de las listas en la web [edit.php](#)
  - **function Actualizar\_tipo(elemento)**, función para mostrar aviso antes del cambio del tipo de campo. [edit.php](#)
  - **function Set\_Prev(elemento)**, función para resetear el valor del desplegable si no se confirma la modificación. [edit.php](#)



- **function validateForm()**, función que se encarga de verificar que los campos del formulario estén validados. Si existe algún campo sin informar, se realiza un focus sobre éste y se cambia el color del fondo. Una vez comprobados realiza el submit. [edit.php](#)
- **function eliminarFormulario()**, función encargada de mostrar una ventana de confirmación antes de eliminar el formulario. Tras confirmar se procederá a la eliminación de éste. [gestion\\_formulario.php](#)

## 14 Implementación y pruebas

El desarrollo se ha llevado a cabo sin grandes problemas, aunque por ejemplo una idea que rechacé debido a complejidad y tiempo fue el crear preguntas tipo Checkbox, en la que las soluciones podían ser varias. Exceptuando cosas puntuales el tiempo ha sido en su mayoría utilizado a programar en función de lo diseñado.

Otra parte de la implementación fue la creación de usuarios “base” para la realización de las distintas pruebas sobre la aplicación, para ver que su funcionamiento era el correcto. También se han introducido manualmente los valores de la tabla Listas.

## 15 Implantación y resultados

No se ha considerado necesario un desarrollo total orientado a objetos debido a que se trata de una aplicación sencilla y a que los lenguajes de programación utilizados tampoco son estrictamente orientados a objetos.

Se ha optado por realizar distintas páginas .php y no intentar agrupar todas en un index.php ya que así el código está mejor estructuradas y es más sencillo de entender a la hora de realizar cambios o implementar nuevas funcionalidades. A la hora de las funciones, ya sean JavaScript o PHP, se ha optado por crear dos páginas las cuales agrupan las funciones utilizadas por las distintas páginas web.

La función SQL **levenshtein** y **levenshtein\_ratio** proporcionan unos resultados adecuados a la hora de reportar cuánto correcta es una respuesta cuando se trata de un texto, en el cual el usuario puede cometer algún error ortográfico que no le debería de privar de obtener puntuación en dicha pregunta.

## 16 Estimación económica

La estimación económica la he realizado teniendo en cuenta que sólo se da un tipo de nivel “desarrollador” y con que el coste de trabajo por hora es de 30€/h.

En la siguiente tabla se pueden observar los costes que tendrían las distintas partes del Proyecto

### FLUJO DE CAJA



Nombre	Costo	Trabajo
<b>Construcción del Software</b>	<b>27.705,00 €</b>	<b>923,5 horas</b>
<b>Plan de Trabajo</b>	<b>4.035,00 €</b>	<b>134,5 horas</b>
Localización del software	465,00 €	15,5 horas
Estudio del software	720,00 €	24 horas
Desarrollo del documento Plan de Trabajo	945,00 €	31,5 horas
<b>Especificación y Análisis</b>	<b>3.735,00 €</b>	<b>124,5 horas</b>
Estudiar los límites del problema	375,00 €	12,5 horas
Analizar los requerimientos no funcionales	705,00 €	23,5 horas

Definir los diferentes actores	195,00 €	6,5 horas
Análisis de los subsistemas	210,00 €	7 horas
Análisis de los requerimientos funcionales	135,00 €	4,5 horas
<b>Diseño</b>	<b>5.805,00 €</b>	<b>193,5 horas</b>
Los diagramas de actividades y/o secuencia	1.785,00 €	59,5 horas
Análisis de la jerarquía de paquetes, diagrama de clases, comunicación entre clases	1.155,00 €	38,5 horas
<b>Implementación</b>	<b>12.930,00 €</b>	<b>431 horas</b>
Implementación	4.050,00 €	135 horas
Pruebas	2.145,00 €	71,5 horas
<b>Entrega Final</b>	<b>1.200,00 €</b>	<b>40 horas</b>
Revisión de la memoria	375,00 €	12,5 horas
Valoración del trabajo realizado	105,00 €	3,5 horas
Preparación de la presentación virtual	135,00 €	4,5 horas

## 17 Conclusiones

La aplicación cumple los objetivos definidos para el proyecto aunque si es verdad que por falta de tiempo, entre trabajo y viajes, puede parecer que el diseño de la misma podría ser mucho mejor. Pese a esto se puede ver que la aplicación cumple las funcionalidades definidas y se puede utilizar sin problemas “graves”.

Me gustaría comentar que las decisiones tomadas en cuanto a las claves primarias de las tablas Listas y Archivos ha sido en su mayor parte al tiempo y también al no haber visionado antes las relaciones que debería establecer con estas tablas.

También quería mencionar que gracias a la experiencia con el lenguaje SQL he podido prescindir de una tabla Estadísticas, ya que he podido realizar los cálculos de éstas mediante consultas SQL.

El problema de tiempo y de “estabilidad” en el día a día que ahora mismo tengo habría podido echar al traste el desarrollo de la aplicación, y es una de las causas de que existan cosas mejorables, pero me siento satisfecho de haber conseguido finalizar la aplicación con las funcionalidades requeridas y un funcionamiento bueno.

## 18 Trabajo futuro

Como trabajo futuro veo la implementación de nuevas funcionalidades y módulos como los siguientes:

- Mejorar las estadísticas para poder consultar también información conjunta de los distintos formularios.
- Implementar nuevos tipos de preguntas para poder definir en los formularios.
- Añadir un sistema de revisión de resultados mediante el cual los profesores pudiesen consultar las soluciones de los distintos alumnos y a su vez poder dejar comentarios a éstos.
- Implementar un módulo de mensajes para que los profesores se pudieran comunicar con los estudiantes de los distintos formularios.
- Añadir una fecha límite a los formularios, antes de la cual deberían ser finalizados.
- Añadir estadísticas históricas de los distintos formularios resueltos por un alumno.
- Mejorar el registro en la web añadiendo más información personal sobre los alumnos.
- Mejorar la navegación entre las páginas para que utilizar el Volver del navegador no cause problemas.
- Mejorar la visualización de la aplicación en cuanto a distintas resoluciones de pantalla se refiere.

## 19 Bibliografía

- PHP <http://www.php.net>
- MySQL <http://www.mysql.com>
- Stack Overflow <http://www.stackoverflow.com> Ayuda para resolver distintos problemas
- ColorHexa <http://www.colorhexa.com/> Ayuda para definición de colores
- W3Schools <http://www.w3schools.com/> Ayuda para dudas HTML
- JSFIDDLE <http://jsfiddle.net/> Ayuda para realizar test de código
- CSS-TRICKS <http://css-tricks.com> Ayuda para diseño de botones
- <http://coding.smashingmagazine.com/> Ayuda con el estilo de las tablas
- <http://www.md5decrypter.co.uk/sha1-decrypt.aspx/> Web para comprobar la seguridad de las contraseñas

## 20 Anexos (Tests)

### 20.1 Acceso a la aplicación

URL: <http://liverpool.uoc.es/~izuel/>

### 20.2 Usuarios para Pruebas

Usuario	Password	Rol
Profesor1	profesor1	Profesor
Alumno1	alumno1	Alumno
Alumno2	alumno2	Alumno
Pepe	pepe	Alumno
Juan	juan	Alumno
Laura	laura	Alumno
Marta	marta	Alumno