



Asignatura: Programación Web

Curso 2013/14

Curso: 3º Grado

Grupo de prácticas: _____

Práctica: Trivial Question Manager Ejercicio: Final

Datos de los Alumnos: (ordenar alfabéticamente por apellidos).

Apellidos: El Alaoui

Nombre: Noureddine

D.N.I.: W456114

e-mail ugr: nourdine@correo.ugr.es



Apellidos: Serrano Collado

Nombre: Ernesto

D.N.I.: 74003802Z

e-mail ugr: erseco@correo.ugr.es



ÍNDICE:

1. Introducción

1.1. Propósito

1.1.1. Propósito del documento

1.1.2. Audiencia a la que va dirigido

1.2. Alcance

1.2.1. Identificación del producto mediante un nombre

1.3. Personal involucrado

1.4. Definiciones, acrónimos y abreviaturas

1.4.1. Acrónimos

1.4.2. Definiciones

1.5. Referencias

1.6. Resumen

1.6.1. Descripción del contenido del resto del documento

1.6.2 Organización del documento

2. Descripción general

2.1. Perspectiva del producto

2.2. Funcionalidad del producto

2.3. Características de los usuarios

2.4. Restricciones

2.5. Suposiciones y dependencias

3. Requisitos específicos

3.1. Requisitos comunes de los interfaces

3.1.1. Interfaces de usuario

3.1.1.1. Interfaces de hardware

3.1.1.2. Interfaces de software

3.1.1.3. Interfaces de comunicación

3.2. Requisitos funcionales

3.2.1. Requisito funcional 1: Gestión usuarios

3.2.1.1. Requisito funcional 1.1: Autentificar usuario

3.2.1.2. Requisito funcional 1.2: Gestión sección Profile

3.2.1.3. Requisito funcional 1.3: Actualización datos de usuarios

3.2.1.4. Requisito funcional 1.4: Administrar cierre de sesión “Logout”

3.2.2. Requisito funcional 2: Gestión de pagina principal “Home”

3.2.3. Requisito funcional 3: Gestión Preguntas “Questions”

- [3.2.3.1. Requisito funcional 3.1: Añadir preguntas](#)
- [3.2.3.2. Requisito funcional 3.2: Editar preguntas](#)
- [3.2.3.3. Requisito funcional 3.3: Borrar preguntas](#)
- [3.2.3.4. Requisito funcional 3.4: Buscar preguntas](#)
- [3.2.4. Requisito funcional 4: Administración Categorías “Categories”](#)
 - [3.2.4.1. Requisito funcional 4.1: Añadir categorías](#)
 - [3.2.4.2. Requisito funcional 4.2: Editar categorías](#)
 - [3.2.4.3. Requisito funcional 4.3: Borrar categorías](#)
 - [3.2.4.4. Requisito funcional 4.4: Buscar categorías](#)
- [3.2.5. Requisito funcional 5: Gestión Idiomas “languages”](#)
 - [3.2.5.1. Requisito funcional 5.1: Añadir idiomas](#)
 - [3.2.5.2. Requisito funcional 5.2: Editar Idiomas](#)
 - [3.2.5.3. Requisito funcional 5.3: Borrar idiomas](#)
 - [3.2.5.4. Requisito funcional 5.4: Buscar Idiomas](#)
- [3.2.6. Requisito funcional 6: Administración usuarios “users”](#)
 - [3.2.6.1. Requisito funcional 6.1: Añadir usuarios](#)
 - [3.2.6.2. Requisito funcional 6.2: Editar Usuarios](#)
 - [3.2.6.3. Requisito funcional 6.3: Borrar usuarios](#)
 - [3.2.6.4. Requisito funcional 6.4: Buscar usuarios](#)
- [3.2.7. Requisito funcional 7: Gestión Herramientas “Tools”](#)
 - [3.2.7.1. Requisito funcional 7.1: Gestión API de Sugerencia.
 - \[3.2.7.1.1. Requisito funcional 7.1.1: Sugerir Preguntas “Suggest Question”\]\(#\)
 - \[3.2.7.1.2. Requisito funcional 7.1.2: Sugerir Corrección “Suggest Correction”\]\(#\)](#)
 - [3.2.7.2. Requisito funcional 7.2: Gestión de copias de seguridad.](#)
 - [3.2.7.3. Requisito funcional 7.3: Gestión de la herramienta Exportar
 - \[3.2.7.3.1. Requisito funcional 7.3.1: Exportar en Formato JAVA\]\(#\)
 - \[3.2.7.3.2. Requisito funcional 7.3.2: Exportar en Formato JavaScript\]\(#\)
 - \[3.2.7.3.3. Requisito funcional 7.3.3: Exportar en Formato CSV\]\(#\)](#)
 - [3.2.7.4. Requisito funcional 7.4: Administrar Contact Support](#)
 - [3.2.7.5. Requisito funcional 7.5: Gestión Estadísticas.](#)
- [3.2.8. Requisito funcional 8: Desarrollo](#)
 - [3.2.8.1 Requisito funcional 8.1: Cantidad de usuarios concurrentes.](#)
 - [3.2.8.2 Requisito funcional 8.2: Cantidad de información almacenada](#)
 - [3.2.8.3 Requisito funcional 8.3: Sistema Operativo](#)
 - [3.2.8.4 Requisito funcional 8.4: Control de errores HTTP](#)

3.2.9. Requisito funcional 9: Disponibilidad del sistema

3.3. Requisitos no funcionales

3.3.1. Requisitos de rendimiento

3.3.2. Seguridad

3.3.3. Fiabilidad

3.3.4. Disponibilidad

3.3.5. Mantenibilidad

3.3.6. Portabilidad

3.3.7. Multiplataforma.

3.4. Otros requisitos

3.4.1 Normas de estilo de codificación

3.4.1.1 Base de datos:

3.4.1.2 Consultas SQL

3.4.1.3 Lenguaje PHP

3.4.1.4 Lenguaje HTML

3.4.1.5 Lenguaje CSS

3.4.1.6 Lenguaje JavaScript y Java

3.4.2. Comentarios en el código.

3.5. Apéndices

4. Descripción de Solución

4.1. Estructura de página

4.1.1. Página principal “Home”

4.1.2. Preguntas “Questions”

4.1.3 Categorías “Categories”

4.1.4. Idiomas “Languages”

4.1.5. Usuarios “Users”

4.1.6. Herramientas “Tools”

4.1.7. Perfil “Profile”

4.2. Estructura de carpetas y archivos entregados:

4.3. Descripción detallada:

4.3.1.: Gestión usuarios “Users”

4.3.1.1. Autenticar usuario

4.3.1.2. Gestión del perfil “Profile”

4.3.1.4. Administrar cierre de sesión “Logout”

4.3.2. Gestión de pagina principal “Home”

[4.3.3. Gestión Preguntas “Questions”](#)

[4.3.4. Administración de categorías “Categories”:](#)

[4.3.5. Gestión de idiomas “Languages”](#)

[4.3.6. Gestión de usuarios “Users”:](#)

[4.3.7. Gestión de Herramientas “Tools”](#)

[4.3.7.1 Gestión de la API](#)

[4.3.7.1.1 Sugerir Preguntas “Suggest Question”:](#)

[4.3.7.1.2. Sugerir Correcciones “Suggest Correction”:](#)

[4.3.7.2. Copias de seguridad “Backup”](#)

[4.3.7.3. Gestión de la herramienta Exportar “Export”](#)

[4.3.7.3.1. Exportar JAVA](#)

[4.3.7.3.2. Exportar JS](#)

[4.3.7.3.3. Exportar CSV](#)

[4.3.7.4. Página de soporte “Contact Support”](#)

[4.3.7.5. Gestión Estadísticas](#)

[4.3.8.1. Cantidad de usuarios concurrentes.](#)

[4.3.8.2. Cantidad de información almacenada](#)

[4.3.8.3. Sistema Operativo](#)

[4.3.8.4. Gestión de errores HTTP](#)

[5. Conclusiones](#)

[5.1 Problemas encontrados](#)

[5.2. Soluciones y conclusiones](#)

[6. Bibliografía](#)

[6.1. Páginas web](#)

[6.2. Libros](#)

1. Introducción

A continuación se presenta la documentación de un sistema web de gestión de preguntas trivial en entorno web, para facilitar el mantenimiento del juego trivial mobile, y coordinar el trabajo de varias personas.

Trivial Mobile es un juego de preguntas y respuestas para android con una base de usuarios de más de un millón de jugadores, dichos jugadores pueden sugerir nuevas preguntas desde la aplicación, así como marcar las preguntas como incorrectas, indicando cual es el problema con dicha pregunta.

Hasta ahora el mantenimiento de dicha base de datos de preguntas se hacía de forma manual, copiando desde *phpMyAdmin* hasta una hoja de excel, y luego, una vez corregidas las preguntas se generan unos archivos *.java* y *.js* para las distintas versiones del juego.

Dicho mantenimiento se ha vuelto tedioso dado el volumen de usuarios y preguntas a corregir, además del problema del idioma, ya que no se dispone de ninguna herramienta que facilite el trabajo.

1.1. Propósito

1.1.1. Propósito del documento

Este documento recoge la especificación de requisitos así como toda la documentación correspondiente del análisis a la aplicación. A partir de él estaremos en condiciones de establecer un diseño que se ajuste a los requerimientos aquí expuestos.

El objeto de la especificación es definir de manera clara y precisa todas las funcionalidades y restricciones del sistema que se desea construir.

1.1.2. Audiencia a la que va dirigido

El documento va dirigido en primer término a **D. Waldo Fajardo Contreras**, profesor de Programación Web en la E.T.S.I.I.T de la Universidad de Granada, y a los creadores de Trivial Mobile en segundo término.

1.2. Alcance

1.2.1. Identificación del producto mediante un nombre

Se adjudica al proyecto el nombre de Trivial Question Manager (**TQM**). Cualquier futura referencia a Trivial Question Manager en este documento se hará usando el acrónimo TQM.

1.3. Personal involucrado

Nombre	Ernesto
Rol	Administrador, Programador
Responsabilidades	Programar y administrar el sistema Web.
Información de contacto	erseco@correo.ugr.es

Nombre	Noureddine El Alaoui
Rol	Administrador, Programador
Responsabilidades	Programar y administrar el sistema Web.
Información de contacto	nourdine@correo.ugr.es

1.4. Definiciones, acrónimos y abreviaturas

1.4.1. Acrónimos

Término	Significado
TQM	Trivial Question Manager
ERS	Especificación de requisitos Software
SW	Software
URL	Uniform Resource Locator

HTTP	Protocolo de Transferencia de Hiper-Texto
CSV	Fichero de texto delimitado por comas
JS	Javascript
CSS	Hojas de estilo en cascada
JSON	Notación de objetos en Javascript
AJAX	Xml y Javascript asíncrono
API	Interfaz de programación de aplicaciones
SO	Sistema Operativo
SQL	Lenguaje de consultas estructurado
PHP	Lenguaje de programación OO
OO	Orientado a objetos
HTML	Lenguaje de Marcas de Hiper-Texto
PDO	Objetos de Datos de PHP
BBDD	Base de datos
MD5	Algoritmo de reducción criptográfico de 128 bits
POST	Es un método de petición HTTP que oculta los parámetros
GET	Es un método de petición HTTP que muestra los parámetros
UTF-8	Formato de codificación de caracteres Unicode

1.4.2. Definiciones

Término	Significado
Servidor web	Es un programa que implementa el protocolo HTTP. Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML: textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.
Apache	Uno de los servidores web mas conocidos y utilizados
MySQL	Servidor de BBDD relacionales
Java	Lenguaje de programación OO
Backup	Copia de seguridad
Unicode	Unicode es un estándar de codificación de caracteres diseñado para facilitar el tratamiento informático
Javascript	Lenguaje de programación interpretado en el lado del cliente
sqldump	Función interna de MySQL que genera un volcado completo de la base de datos.

1.5. Referencias

Referencia	Título	Ruta	Fecha	Autor
IEEE	<i>Estándar IEEE Std 830-1998</i>			IEEE

1.6. Resumen

1.6.1. Descripción del contenido del resto del documento

Este documento consta de varias secciones. La sección actual sirve como introducción al problema, explicando los detalles que no tienen que ver directamente con la definición del problema.

En las siguientes secciones se va abordando el problema desde su descripción (con el fin de conocer las principales funciones que debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo) hasta su implementación, pasando por todos los requisitos necesarios para desarrollar dicho sistema.

1.6.2 Organización del documento

Estas especificaciones de requisitos software siguen las recomendaciones del estándar [IEEE98] en cuanto a la organización de los distintos apartados así como el contenido de cada uno de ellos.

2. Descripción general

En esta sección se presenta una descripción a alto nivel del sistema. Se presentarán las principales áreas de negocio a las cuales el sistema debe dar soporte, las funciones que el sistema debe realizar, la información utilizada, las restricciones y otros factores que afectan al desarrollo del mismo.

La práctica aborda la realización un Sistema Web para gestión de preguntas de trivial, para no depender de programas.

Nuestro sistema debe gestionar 2 tipos de usuarios:

✓ **Administrador:** que puede

- Dar de alta y baja usuarios.
- Dar de alta y baja idiomas.
- Editar, bloquear y borrar tanto preguntas.
- Hacer copias de seguridad del sistema y restaurarlas (sqldump).
- Borrar preguntas marcadas para eliminar
- Generar los archivos para el juego (descargar en.zip)

- Hacer todo aquello que puedan hacer los usuarios identificados.
- Mostrar estadísticas de cantidad de preguntas sugeridas y/o corregidas por fecha.

✓ **Usuario:** que puede:

- Editar sus datos.
- Crear, editar preguntas en el idioma que tienen asignado.
- Acceder al resto de idiomas en modo solo lectura, y permitir crear una nueva a partir de otro idioma.
- Marcar preguntas cómo corregida, aceptada o borrada.
- Gestionar todo el material relacionado con el idioma que tienen asignado.
- Gestionar duplicados y parecidos (función similar_text de PHP).
- Visualizar la información oculta de los recursos del resto de los usuarios identificados.
- Hacer todo aquello que puedan hacer los usuarios no identificados.

2.1. Perspectiva del producto

El sistema TQM será un producto diseñado para trabajar en entornos WEB, lo que permitirá su utilización de forma descentralizada, además trabajará de manera independiente por lo tanto no interactuará con otros sistemas.

2.2. Funcionalidad del producto

El sistema TQM permitirá realizar las siguientes funciones:

- a. Administración de Usuarios:** El administrador del sistema podrá gestionar los usuarios (agregar, modificar, eliminar, buscar, listar).
- b. Administración de Roles:** El administrador del sistema gestionará los roles de usuarios del sistema, de igual forma asignará los roles a los usuarios.

c. Coordinación de Preguntas: El coordinador de preguntas planificará los Recursos necesarios para las preguntas, tales como: editar, eliminar, traducir y otros recursos en general.

e. Publicación de Información: El administrador del sistema actualizará la información general del TQM.

f. Inscripción: Proceso que permitirá a una persona inscribirse, para que el administrador lo pueda dar de alta.

g. Navegación: Proceso por el cual una persona podrá utilizar el TQM.

2.3. Características de los usuarios

Cada usuario tendrá un perfil específico para que su interacción con el sistema sea correcto y no conlleve a fallos.

El sistema TQM contendrá 5 tipos de usuarios que interactúan y/o administran:

Tipo de usuario	Administrador
Formación	Administrador de servidores y páginas Web.
Habilidades	No se requiere habilidad específica.
Actividades	<ul style="list-style-type: none">● Dar de alta y baja usuarios.● Dar de alta y baja idiomas.● Editar, bloquear y borrar preguntas.● Hacer copias de seguridad del sistema y restaurarlas (sqldump).● Borrar preguntas marcadas para eliminar.● Generar los archivos para el juego (descargar en.zip).

	<ul style="list-style-type: none"> • Hacer todo aquello que puedan hacer los usuarios identificados. • Mostrar estadísticas de cantidad de preguntas sugeridas/corregidas por fecha.
--	--

Tipo de usuario	Traductor
Formación	No se requiere formación específica.
Habilidades	No se requiere habilidad específica.
Actividades	<ul style="list-style-type: none"> • Puede acceder a todos los idiomas en modo lectura. • Puede acceder al idioma que traduce en modo escritura.

Tipo de usuario	Corrector
Formación	No se requiere formación específica.
Habilidades	No se requiere habilidad específica.
Actividades	<ul style="list-style-type: none"> • Solo puede acceder al idioma que corrige en modo escritura

Tipo de usuario	Usuario Identificado
Formación	No se requiere formación específica.
Habilidades	No se requiere habilidad específica.
Actividades	<ul style="list-style-type: none"> • Editar sus datos. • Crear, editar preguntas en el idioma que tienen asignado. • Acceder al resto de idiomas en modo solo lectura y permitir crear una nueva a partir de otro idioma. • Marcar preguntas como corregida, aceptada o borrada.

	<ul style="list-style-type: none"> • Gestionar todo el material relacionado con el idioma que tienen asignado. • Visualizar la información oculta de los recursos del resto de los usuarios identificados. • Hacer todo aquello que puedan hacer los usuarios no identificados.
--	--

Tipo de usuario	Usuario no Identificado
Formación	No se requiere formación específica
Habilidades	No se requiere habilidad específica
Actividades	<ul style="list-style-type: none"> • Agregar nuevas preguntas en cualquier idioma de los disponibles en la plataforma • Agregar nuevas correcciones de las preguntas

2.4. Restricciones

Para esta aplicación será necesario un computador servidor en el cual se instalará el servidor *WEB apache*, *MySQL*, *PHP* y la aplicación TQM.

2.5. Suposiciones y dependencias

Debido a que el sistema no interactúa con otros sistemas y es autónomo, no se desarrollaron interfaces con otras aplicaciones. Las conexiones necesarias para la utilización del servidor *Web*, *MySQL*, *PHP*, se hará por medio de la configuración de estos programas.

El sistema no necesita un sistema operativo preciso, da igual si es *Linux*, *Windows* o *Mac*...

También, por el tema de navegador, el sistema es Multi-navegador.

3. Requisitos específicos

En este apartado se presentan los requisitos funcionales que deberán ser satisfechos por el sistema. Todos los requisitos aquí expuestos son ESENCIALES, es decir, no sería aceptable un sistema que no satisfaga alguno de los requisitos aquí presentados.

3.1. Requisitos comunes de los interfaces

El sistema TQM no tendrá interconexión con otros sistemas de información, por lo tanto no es necesaria la utilización de interfaz alguna.

La relación con un servidor WEB, y Gestor de Base de Datos se hará a través de los archivos de configuración de éstos.

3.1.1. Interfaces de usuario

La interfaz de usuario debe ser orientada a ventanas, y el manejo del programa se realizará a través del teclado y ratón. Interfaz intuitiva de fácil uso.

La interfaz de los administradores, será la que más privilegios tiene, ya que un administrador de sistema, tendrá acceso a todas las secciones, y podrá usar todas las herramientas proporcionadas de la página.

3.1.1.1. Interfaces de hardware

No se han definido

3.1.1.2. Interfaces de software

No se han definido

3.1.1.3. Interfaces de comunicación

No se han definido

3.2. Requisitos funcionales

3.2.1. Requisito funcional 1: Gestión usuarios

3.2.1.1. Requisito funcional 1.1: Autentificar usuario

Número	RF 1.1
Nombre	Autentificar usuario “Login”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>El sistema deberá pedir autenticación. Se validará el acceso mediante el correo y contraseña que estará cifrada mediante MD5 en la base de datos.</p> <p>Secuencia exacta de operaciones</p> <p>Si un usuario se autentica correctamente y se comprueba que sus datos son correctos, se guardan las variables de la sesión (<i>name</i>, <i>email</i>, <i>role</i>, <i>id_language</i>, <i>code_language</i>, <i>date</i>) y se abre una sesión con la información correspondiente.</p> <p>A la hora de identificar, el usuario tendrá una casilla “Remember Me” que guarda una cookie para recordar la sesión.</p> <p>Respuesta a situaciones anormales</p> <p>En el caso, que los datos introducidos no sean correctos, saltara un error en rojo informando que el correo o la contraseña no son correctos.</p> <p>Parámetros</p> <p>correo electrónico contraseña</p>

	<p>Generación de salidas</p> <p>Una vez autenticado el sistema generará una pantalla con todas las operaciones disponibles. Dependiendo de los roles que tenga el usuario asignados se mostrarán unas operaciones u otras, si es un:</p> <ul style="list-style-type: none"> - Usuario normal: solo tendrá acceso a la sección <i>questions</i>, donde puede añadir, modificar, o borrar preguntas, y la sección “<i>tools</i>” para sugerir preguntas o correcciones a partir de un API. - Administrador: tendrá acceso a todas las secciones de la página con todas las opciones disponibles. <p>Relaciones entre entradas y salidas:</p> <p>No procede</p> <p>Especificación de los requisitos lógicos:</p> <p>correo electrónico como cadena contraseña como cadena codificada con MD5</p>
--	--

3.2.1.2. Requisito funcional 1.2: Gestión sección Profile

Número	RF 1.2
Nombre	Gestión La sección Profile.
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema debe generar una sección que haga referencia al perfil de los usuarios registrados, esta sección contiene información</p>

	<p>como (nombre, email, rol, fecha de inscripción) que debe ser propia y privada de cada usuario.</p> <p>El sistema debe dar la posibilidad a los usuarios de modificar cualquiera de estos datos, menos la fecha de inscripción que es propia del sistema y se genera de forma automática, en cada perfil.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>Nombre</p> <p>Correo</p> <p>Rol</p> <p>Fecha de inscripción</p> <p>Generación de salidas</p> <p>Una vez que el usuario acceda a su perfil se genera una sección propia “Profile”, que contiene la información anterior, con la posibilidad de modificar cada una de ellas.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>Nombre como cadena</p> <p>correo electrónico como cadena</p> <p>contraseña como cadena codificada con MD5</p> <p>rol como valor booleano</p> <p>fecha de inscripción como fecha</p>
--	---

3.2.1.3. Requisito funcional 1.3: Actualización datos de usuarios

Número	RF 1.3
Nombre	Actualización datos de usuarios.

Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir la actualización de datos de los usuarios</p> <p>Comprobación de validez de las entradas</p> <p>al pinchar el botón editar, donde sale un formulario con todos los campos:</p> <ul style="list-style-type: none"> - Campo nombre: que como mínimo tiene que tener 4 caracteres. - Campo correo: que tiene que ser escrito de la siguiente forma nombre@ejemplo.es. - Campos contraseña: que como mínimo tienen que contener 6 caracteres en cada campo (campo original y campo de repetición), con la obligación que los dos campos sean iguales. <p>Secuencia exacta de operaciones</p> <p>Si un usuario por cualquier motivo, le apetece cambiar sus datos (nombre, correo o contraseña), lo hará desde su perfil con la opción editar, después de la modificación, se comprueba la validez de datos, y se guardan en la base datos.</p> <p>Respuesta a situaciones anormales</p> <p>Si la información introducida no cumple con las restricciones anteriores, el campo que contiene un error saldrá en rojo.</p> <p>Parámetros</p> <p>Nombre Contraseña Correo</p> <p>Generación de salidas</p> <p>Una vez que los nuevos datos introducidos sean correctos, el sistema debe generar un aviso en rojo con la opción del logout, para</p>

	<p>cerrar la sesión y cargar las modificaciones, eso se verá reflejado en la próxima apertura de sesión.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>correo electrónico como cadena.</p> <p>contraseña como cadena codificada con MD5.</p> <p>nombre como cadena.</p>
--	---

3.2.1.4. Requisito funcional 1.4: Administrar cierre de sesión “Logout”

Número	RF 1.4
Nombre	<i>Logout</i>
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe incluir la opción de cerrar la sesión abierta por el usuario.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que un usuario le da a la opción <i>logout</i>, el sistema debe borrar la sesión y limpiar las cookies generadas durante la sesión.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p>

	<p>Generación de salidas</p> <p>El sistema debe generar como salida de un <i>logout</i>, la página de inicio, que contiene el formulario para permitir un nuevo acceso.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.2. Requisito funcional 2: Gestión de pagina principal “Home”

Número	RF 2
Nombre	Gestión Home – Página Principal
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe mostrar tras el inicio de sesión una página principal.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>La página debe tener un pequeño resumen sobre el sistema trivia questions manager, y la colección de juegos que soporta nuestro sistema (<i>Trivial Mobile</i>, <i>Geek Trivia</i>, <i>Rock Trivia</i>),</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p>

	<p>Al pinchar en cada uno de los juegos se generará como salida un enlace para descargar cada uno de ellos en <i>Google Play</i>.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	--

3.2.3. Requisito funcional 3: Gestión Preguntas “Questions”

Número	RF 3
Nombre	Gestión Preguntas “Questions”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema debe gestionar de forma muy ordenada y específica todas las preguntas del sistema. Permitiendo a los usuarios de la página sean administradores o usuarios registrados las siguientes opciones:</p> <ul style="list-style-type: none"> - RF 3.1 Añadir preguntas - RF 3.2 Editar preguntas - RF 3.3 Borrar preguntas - RF 3.4 Buscar preguntas <p>Parámetros</p> <p>No procede</p>

	<p>Generación de salidas La salida generada será dependiendo del uso de cada opción.</p> <p>Relaciones entre entradas y salidas No procede</p> <p>Especificación de los requisitos lógicos No procede</p>
--	--

3.2.3.1. Requisito funcional 3.1: Añadir preguntas

Número	RF 3.1
Nombre	Añadir preguntas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir a los usuarios añadir preguntas de todas las categorías e idiomas que soporte el sistema.</p> <p>Comprobación de validez de las entradas El sistema generará como entrada, a la hora de añadir una nueva pregunta un formulario que contiene los campos siguientes:</p> <ul style="list-style-type: none"> - Campo con un menú desplegable de las categorías de preguntas, definidas en la tabla de categorías. - Campo para introducir el texto de la pregunta. - 3 Campos iguales, para introducir las posibles respuestas de la pregunta, de las cuales la primera será la correcta. - un campo para indicar el idioma en el que está escrita la pregunta. <p>Secuencia exacta de operaciones</p>

	<p>Una vez que todos los campos están rellenos, después de darle a añadir se guarda la pregunta en la base de dato, y se actualiza la lista de preguntas.</p> <p>El sistema debe ser capaz de reflejar los cambios actualizados en la gráfica de estadísticas, relacionada con la sección preguntas.</p> <p>Respuesta a situaciones anormales</p> <p>Si la información introducida no cumple con las restricciones anteriores, el campo que contiene un error saldrá en rojo.</p> <p>Parámetros</p> <p>campo dificultad</p> <p>campo de texto de pregunta</p> <p>los 3 campos de respuesta</p> <p>campo comentarios</p> <p>Generación de salidas</p> <p>El sistema generará una nueva lista de preguntas, donde sale la última pregunta introducida al final de todo.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>identificador</i> de question como entro</p> <p><i>identificador</i> de categoría como entero</p> <p>fecha como variable date</p> <p>dificultad como entero</p> <p>pregunta como cadena</p> <p>respuesta 1 “<i>answer1</i>” como cadena</p> <p>respuesta 2 “<i>answer2</i>” como cadena</p> <p>respuesta 2 “<i>answer3</i>” como cadena</p> <p>Comentario como cadena</p>
--	---

3.2.3.2. Requisito funcional 3.2: Editar preguntas

Número	RF 3.2
Nombre	Editar preguntas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá la modificación de preguntas a partir de la opción editar.</p> <p>El sistema debe permitir las distintas formas de editar una pregunta modificando el texto de la pregunta, idioma, categoría, o una de las 3 respuestas.</p> <p>Comprobación de validez de las entradas</p> <p>A la hora de pinchar en el ícono de editar el sistema debe ser capaz de generar el formulario anterior que contiene la información de la pregunta que queremos modificar.</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que la modificación se ha terminado, y le damos a enviar “submit”, se guarda la nueva versión y se actualiza la lista de preguntas de forma automática.</p> <p>Respuesta a situaciones anormales</p> <p>Si la información introducida no cumple con las restricciones anteriores, el campo que contiene un error saldrá en rojo.</p> <p>Parámetros</p> <p>campo dificultad</p> <p>campo de texto de pregunta</p> <p>los 3 campos de respuesta</p> <p>campo comentarios</p>

	<p>Generación de salidas</p> <p>El sistema generará una nueva lista donde se ven reflejados los cambios que hemos hecho en la pregunta.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>id</i> de question como entro</p> <p><i>id</i> de categoría como entero</p> <p>fecha como variable date</p> <p>dificultad como entero</p> <p>pregunta como cadena</p> <p>respuesta 1 “answer1” como cadena</p> <p>respuesta 2 “answer2” como cadena</p> <p>respuesta 2 “answer3” como cadena</p> <p>Comentario como cadena</p>
--	--

3.2.3.3. Requisito funcional 3.3: Borrar preguntas

Número	RF 3.3
Nombre	Borrar preguntas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir a los usuarios la opción de eliminar una pregunta.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p>

	<p>Cuando le damos a borrar pregunta se generará un aviso "Are you Sure", pidiendo la confirmación, una vez se borra la pregunta, se guardan los cambios, y se actualizará la lista de preguntas.</p> <p>A la hora de borrar una pregunta el sistema debe ser capaz de reflejar los cambios actualizados en la gráfica de estadísticas, relacionada con la sección preguntas</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>El sistema generará una nueva lista de preguntas actualizada al momento después de eliminar una pregunta.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede.</p>
--	--

3.2.3.4. Requisito funcional 3.4: Buscar preguntas

Número	RF 3.4
Nombre	Buscar preguntas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	El sistema debe ofrecer la búsqueda de las distintas preguntas de la página a partir de un filtro de búsqueda.

	<p>El sistema debe permitir la búsqueda de preguntas usando caracteres relacionado con la pregunta en sí misma, la categoría o la lengua.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez pinchamos el icono de buscar “<i>toggle table filter</i>”, se genera el filtro de búsqueda, donde podemos introducir los caracteres relacionados con la pregunta que queremos buscar.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>El sistema debe generar un listado de los resultados relacionado con este carácter.</p> <p>En el caso en los caracteres introducidos no coinciden con ninguna pregunta, se generará una lista vacía.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	--

3.2.4. Requisito funcional 4: Administración Categorías “Categories”

Número	RF 4
Nombre	Administración Categorías “Categories”

Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema debe ser capaz de ordenar las preguntas según las categorías que soporta la página.</p> <p>El sistema ofrecerá una sección propia, de las categorías de preguntas "categories" con acceso exclusivo para los administradores, donde tendrán la posibilidad de:</p> <ul style="list-style-type: none"> RF 4.1 Añadir categorías RF 4.2 Editar categorías RF 4.3 Borrar categorías RF 4.4 Buscar categorías <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>La salida generada será dependiendo del uso de cada opción.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>

3.2.4.1. Requisito funcional 4.1: Añadir categorías

Número	RF 4.1
Nombre	Añadir categorías

Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios que sean administradores, incluir nuevas categorías, a la lista con la opción “Add new”.</p> <p>Comprobación de validez de las entradas</p> <p>A la hora de añadir una nueva categoría el sistema generará como entrada un formulario con un único campo:</p> <ul style="list-style-type: none"> - Nombre "Name", para introducir el nombre de la categoría, que como mínimo tiene que ser 4 caracteres . <p>Secuencia exacta de operaciones</p> <p>Una vez introducimos el nombre, le damos a añadir "create" se guarda la nueva categoría, y se añade de forma automática.</p> <p>El sistema debe actualizar la gráfica “categories”, a la hora de añadir una nueva categoría.</p> <p>Respuesta a situaciones anormales</p> <p>En el caso de que el nombre que introducimos no cumple con la restricción anterior(mínimo 4 caracteres), se genera un error "<i>Name is too short or empty!</i>" informando de que el nombre no esta bien.</p> <p>Parámetros</p> <p>Nombre</p> <p>Generación de salidas</p> <p>El sistema debe ser capaz de generar la nueva lista de categorías actualizada mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>Nombre como cadena</p>

	<i>id</i> categoría como entero
--	---------------------------------

3.2.4.2. Requisito funcional 4.2: Editar categorías

Número	RF 4.2
Nombre	Editar categorías
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios con privilegios de administradores modificar las categorías de la página, con la opción editar.</p> <p>Comprobación de validez de las entradas</p> <p>El sistema debe generar un formulario con el campo “Name” para poder cambiar el nombre de la categoría, que tiene que contener 4 caracteres como mínimo.</p> <p>Secuencia exacta de operaciones</p> <p>Después de pinchar en el ícono de editar, introducimos la modificación que queremos, y le damos a actualizar "update", el sistema se ocupará de guardar los nuevos cambios en la base de datos.</p> <p>Respuesta a situaciones anormales</p> <p>El sistema generará un error "Name is too short or empty! " en el caso de que el nombre no cumple con esta restricción.</p> <p>Parámetros</p> <p>Nombre</p> <p>Generación de salidas</p>

	<p>el sistema genera como salida la nueva lista de categorías actualizada, como todos los cambios guardados mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>Nombre como cadena</p> <p><i>id</i> categoría como entero</p>
--	--

3.2.4.3. Requisito funcional 4.3: Borrar categorías

Número	RF 4.3
Nombre	Borrar categorías
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios que sean administradores borrar categorías de la lista.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez pinchamos en el ícono de borrar, el sistema debe generar un aviso “Are you sure”, pidiendo la confirmación al administrador.</p> <p>Después de confirmarlo, se borra la categoría y el sistema se ocupará de eliminar esta última de la base de datos.</p> <p>El sistema debe actualizar la gráfica “categories”, a la hora de borrar una categoría.</p> <p>Respuesta a situaciones anormales</p>

	<p>No procede</p> <p>Parámetros</p> <p>Nombre</p> <p>Generación de salidas</p> <p>El sistema debe ser capaz de generar, como salida la nueva lista de categorías actualizada mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	--

3.2.4.4. Requisito funcional 4.4: Buscar categorías

Número	RF 4.4
Nombre	Buscar categorías
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe ofrecer la búsqueda de las distintas categorías de la página a partir de un filtro de búsqueda.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez pinchamos el icono de buscar “<i>toggle table filter</i>”, se genera el filtro de búsqueda, donde podemos introducir la categoría que queremos buscar.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p>

	<p>Parámetros</p> <p>Nombre</p> <p>Generación de salidas</p> <p>El sistema debe generar un listado de los resultados relacionados con este carácter.</p> <p>En el caso en los caracteres introducidos no coinciden con ninguna categoría, se generará una lista vacía.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.5. Requisito funcional 5: Gestión Idiomas “languages”

Número	RF 5
Nombre	Gestión Idiomas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema ofrecerá una sección propia, de idiomas “languages” con acceso exclusivo para los administradores, donde tendrán la posibilidad de:</p> <p>RF 5.1 Añadir idiomas</p> <p>RF 5.2 Editar idiomas</p> <p>RF 5.3 Borrar idiomas</p>

	RF 5.4 Buscar idiomas
	Parámetros
	No procede
	Generación de salidas
	La salida generada será dependiendo del uso de cada opción.
	Relaciones entre entradas y salidas
	No procede
	Especificación de los requisitos lógicos
	No procede

3.2.5.1. Requisito funcional 5.1: Añadir idiomas

Número	RF 5.1
Nombre	Añadir idiomas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios que sean administradores, incluir idiomas, a la lista con la opción “Add new”.</p> <p>Comprobación de validez de las entradas</p> <p>A la hora de añadir una nueva categoría el sistema generará, como entrada un formulario con 2 campos</p> <ul style="list-style-type: none"> - Campo Nombre “Name”: para introducir el nombre de la categoría. - Campo Código “Code”: para proponer un código que identifique el idioma. <p>Secuencia exacta de operaciones</p>

	<p>Una vez introducimos los datos de cada campo, y darle a crear “create”, el sistema guardará el nuevo idioma en la base de datos.</p> <p>El sistema debe actualizar la gráfica ”languages”, a la hora de añadir un nuevo idioma.</p> <p>Respuesta a situaciones anormales</p> <p>Si la información introducida no cumple con las restricciones anteriores, el campo que contiene un error saldrá en rojo.</p> <p>Parámetros</p> <p>Nombre</p> <p>Código</p> <p>Generación de salidas</p> <p>El sistema debe ser capaz de generar la nueva lista de idiomas actualizada mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>id</i> idioma como entero</p> <p>Nombre como cadena</p> <p>Código como cadena</p>
--	---

3.2.5.2. Requisito funcional 5.2: Editar Idiomas

Número	RF 5.2
Nombre	Editar idiomas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional

Descripción	<p>El sistema permitirá a los usuarios con privilegios de administrador modificar las idiomas “languages” de la página, con la opción editar.</p> <p>Comprobación de validez de las entradas</p> <p>a la hora de modificar un idioma, el sistema debe generar un formulario con los campos anteriores (<i>Code</i>, <i>Name</i>) para poder cambiar el código y le nombre de la idioma correspondiente.</p> <p>Secuencia exacta de operaciones</p> <p>Una vez pinchamos en el icono de editar, y introducimos el nombre y el código, le damos a actualizar “Submit”, y el sistema se ocupará de guardar los datos modificados, en la base de datos.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>Nombre</p> <p>Código</p> <p>Generación de salidas</p> <p>el sistema debe generar como salida, la nueva lista de categorías actualizada, con los cambios guardados mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>id</i> idioma como entero</p> <p>Nombre como cadena</p> <p>Código como cadena</p>
--------------------	--

3.2.5.3. Requisito funcional 5.3: Borrar idiomas

Número	RF 5.3
Nombre	Borrar idiomas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios que sean administradores, borrar idiomas de la lista.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Cuando pinchamos el icono de borrar, el sistema debe generar un aviso “Are you sure ?” pidiendo la confirmación al administrador, una vez se confirma, se borrara el idioma, y se guardaran los cambios.</p> <p>El sistema actualizará de forma automática la gráfica “languages” después de eliminar cualquier idioma.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>Después de borrar un idioma, el sistema debe ser capaz de generar una lista actualizada, reflejando las modificaciones que hemos hecho.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p>

	Especificación de los requisitos lógicos
	No procede

3.2.5.4. Requisito funcional 5.4: Buscar Idiomas

Número	RF 5.4
Nombre	Buscar Idiomas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe ofrecer la búsqueda de las distintas idiomas de la página a partir de un filtro de búsqueda.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez pinchamos el icono de buscar “<i>toggle table filter</i>”, se genera el filtro de búsqueda, donde podemos introducir los caracteres del idioma que queremos buscar.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>Nombre</p> <p>Código</p> <p>Generación de salidas</p> <p>El sistema debe generar un listado de los resultados relacionados con este carácter.</p> <p>En el caso de que los caracteres introducidos no coinciden con ninguno de los idiomas, se generará una lista vacía.</p>

	<p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>id</i> idioma como entero</p> <p>Nombre como cadena</p> <p>Código como cadena</p>
--	--

3.2.6. Requisito funcional 6: Administración usuarios “users”

Número	RF 6
Nombre	Administración usuarios “users”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema ofrecerá una sección propia para englobar todos los usuarios de la página, con acceso exclusivo para los administradores, donde tendrán la posibilidad de:</p> <ul style="list-style-type: none"> RF 6.1 Añadir usuarios RF 6.2 Editar usuarios RF 6.3 Borrar usuarios RF 6.4 Buscar usuarios <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>La salida generada será dependiendo del uso de cada opción.</p>

	Relaciones entre entradas y salidas No procede
	Especificación de los requisitos lógicos No procede
	Comprobación de validez de las entradas No procede
	Especificación de los requisitos lógicos No procede

3.2.6.1. Requisito funcional 6.1: Añadir usuarios

Número	RF 6.1
Nombre	Añadir usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios que sean administradores, incluir nuevos usuarios, a la lista con la opción añadir nuevo “Add new”.</p> <p>Comprobación de validez de las entradas No procede</p> <p>Secuencia exacta de operaciones A la hora de añadir una nuevo usuario, el sistema generará un formulario con los siguientes campos:</p> <ul style="list-style-type: none"> - Idioma “language”: que contiene un menú desplegable con todas las idiomas que soporta el sistema.

	<p>-Nombre “<i>Name</i>”: que como mínimo tiene que contener 4 caracteres.</p> <p>-Correo “<i>Email</i>”: que tiene que ser escrito de la siguiente forma usuario@ejemplo.es</p> <p>-Contraseña “<i>Password</i>”: que como mínimo tiene que tener 6 caracteres iguales, en los dos campos(campo original, campo repetición)</p> <p>Además el administrador tendrá una casilla para, para dar permiso de administración a usuarios.</p> <p>Una vez que le damos a crear se guarda el nuevo usuario, y se añade a la lista de forma automática.</p> <p>Respuesta a situaciones anormales</p> <p>En el caso que uno de esos datos no cumple con estas restricciones, el campo que falla saldrá en rojo. para el nombre se generará el error "<i>Name is too short or empty!</i>" informando de que el nombre es corto o no esta bien.</p> <p>Parámetros</p> <p>Nombre correo electrónico contraseña</p> <p>Generación de salidas</p> <p>El sistema debe ser capaz de generar la nueva lista de usuarios actualizada mediante ajax.</p> <p>El sistema debe actualizar la gráfica “<i>users</i>”, a la hora de añadir un nuevo usuario.</p> <p>Especificación de los requisitos lógicos:</p> <p>nombre como cadena. correo electrónico como cadena. contraseña como cadena codificada con MD5.</p>
--	--

3.2.6.2. Requisito funcional 6.2: Editar Usuarios

Número	RF 6.2
Nombre	Editar usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá a los usuarios con privilegios de administradores modificar los datos de usuarios de la página, con la opción editar.</p> <p>Comprobación de validez de las entradas</p> <p>A la hora de editar un usuario el sistema debe generar un formulario con los siguientes campos:</p> <ul style="list-style-type: none">- Idioma “language”: que contiene un menú desplegable con todas las idiomas que soporta el sistema.-Nombre“Name”: que como mínimo tiene que contener 4 caracteres.-Correo “Email”: que tiene que ser escrito de la siguiente forma usuario@ejemplo.es. <p>y un casilla para dar o quitar privilegios a usuarios.</p> <p>En ningún momento los administradores tendrán acceso a las contraseñas de los usuarios.</p> <p>Secuencia exacta de operaciones</p> <p>Los datos se introducen a través del teclado, una vez que se confirma que la forma de introducir estos últimos es correcta, se guardan en la base de datos.</p> <p>Respuesta a situaciones anormales</p>

	<p>En el caso que uno de esos datos no cumple con estas restricciones, el campo que falla saldrá en rojo.</p> <p>para el nombre se generará el error "<i>Name is too short or empty!</i>" informando de que el nombre es corto o no esta bien.</p> <p>Parámetros</p> <p>Nombre.</p> <p>correo electrónico.</p> <p>Generación de salidas</p> <p>Una vez que le damos a actualizar "<i>update</i>", se guardan los cambios y se genera la nueva lista de usuarios actualizada mediante ajax.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede.</p> <p>Especificación de los requisitos lógicos</p> <p>Correo electrónico como cadena.</p> <p>Nombre como cadena de caracteres.</p>
--	--

3.2.6.3. Requisito funcional 6.3: Borrar usuarios

Número	RF 6.3
Nombre	Borrar usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	El sistema debe permitir a los usuarios la opción de eliminar usuarios de forma exclusiva para los administradores.

	<p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>El sistema permitirá a los administradores, por cualquier motivo, borrar usuarios de la lista.</p> <p>A la hora de borrar un usuario, el sistema debe generar un aviso, “Are you sure”, pidiendo la confirmación al administrador, una vez se borra el usuario, se guardan los cambios, y se actualizará la lista de usuarios.</p> <p>A la hora de borrar un usuario el sistema debe ser capaz de reflejar los cambios actualizados en la gráfica de estadísticas, relacionada con la sección usuarios.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>El sistema generará una nueva lista de usuarios actualizada al momento, después de eliminar un usuario.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede.</p>
--	--

3.2.6.4. Requisito funcional 6.4: Buscar usuarios

Número	RF 6.4
Nombre	Buscar usuarios
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe ofrecer la búsqueda de los usuarios de la página a partir de un filtro de búsqueda.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>podemos introducir los caracteres del idioma que queremos buscar.</p> <p>Una vez pinchamos el icono de buscar “<i>toggle table filter</i>”, se genera el filtro de búsqueda, donde podemos introducir los caracteres que queremos buscar.</p> <p>El sistema debe generar un listado de los resultados relacionados con este carácter.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>En el caso en los caracteres introducidos no coinciden con ninguna categoría, se generará una lista vacía.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p>

	Especificación de los requisitos lógicos
	No procede

3.2.7. Requisito funcional 7: Gestión Herramientas “Tools”

Número	RF 7
Nombre	Gestión Herramientas “Tools”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe ofrecer a los usuarios un conjunto de herramientas, incluidas en una sección “Tools”.</p> <p>Comprobación de validez de las entradas</p> <p>El acceso a esas herramientas será según el rol de cada uno:</p> <ul style="list-style-type: none"> - Administrador: tendrá acceso a todas las herramientas del sistema. - Usuario Registrado: solo puede usar la API para sugerir preguntas y correcciones, y acceder a la parte statistics, que contiene las estadísticas de cada sección. <p>Secuencia exacta de operaciones</p> <p>Una vez el usuario accede a la sección de tools, se generará un menú desplegable, con las herramientas correspondiente al rol que tiene.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p>

	<p>A la hora de pinchar en cualquiera de estas últimas, el sistema generará una página propia para cada herramienta.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.7.1. Requisito funcional 7.1: Gestión API de Sugerencia.

Número	RF 7.1
Nombre	Gestión API de sugerencia
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir la opción de sugerir preguntas o corrección mediante una Interfaz de programación de aplicaciones(API).</p> <p>Esta herramienta debe estar incluida con las otras herramientas en la sección tools.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez accedemos a la API, se genera una sección, con las 2 opciones que incluye esta herramienta.</p> <p>El usuario puede usar esta herramienta para sugerir:</p> <p>RF 6.1.1 Sugerir Preguntas “Suggest Question”</p> <p>RF 6.2.2 Sugerir Correcciones “Suggest Correction”</p> <p>Respuesta a situaciones anormales</p>

	<p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>Después de cada sugerencia el sistema mostrará un mensaje d gracias informando que la operación se ha realizado correctamente.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.7.1.1. Requisito funcional 7.1.1: Sugerir Preguntas “Suggest Question”

Número	RF 7.1.1
Nombre	Sugerir Preguntas “Suggest Question”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Esencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir la sugerencia de preguntas de las distintas categorías e idiomas que soporta el sistema mediante una API.</p> <p>Comprobación de validez de las entradas</p> <p>A la hora de sugerir preguntas, el sistema debe facilitar al usuario 2 opciones:</p>

- Un código en HTML para insertar la pregunta al sistema mediante la siguiente dirección: http://[URL]/api/insert_question.php
- o
- Usando un formulario normal con los siguientes campos:
 - Category: para la categoría de la pregunta, que tiene que ser un número.
 - Language: para introducir el idioma que soporta la pregunta, que también tiene que ser un número.
 - Difficulty: para introducir la dificultad de la pregunta.
 - Question: para introducir la pregunta.
 - Answer 1(correct): la primera respuesta que será la correcta.
 - Answer 2: la segunda respuesta.
 - Answer 3: la tercera respuesta.
 - Comments: Un último campo, en el caso de que los usuarios quieren comentar algo.

Secuencia exacta de operaciones

Después de que el usuario accede a la sección tools, y pinchar en *Suggest Question*, se generará el formulario anterior, una vez introducidos los datos, el sistema se ocupará de guardar la nueva pregunta, y insertarla en la lista de preguntas del sistema.

Respuesta a situaciones anormales

No procede

Parámetros

campo dificultad

campo de texto de pregunta

los 3 campos de respuesta

campo comentarios

Generación de salidas

	<p>Una vez que el usuario le da a sugerir, se le enseñara un aviso de gracias, confirmando el éxito de la sugerencia.</p> <p>Al mismo tiempo se añadirá la pregunta surgida al final de la lista de preguntas</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p><i>id</i> de question como entro</p> <p><i>id</i> de categoría como entero</p> <p>fecha como variable date</p> <p>dificultad como entero</p> <p>pregunta como cadena</p> <p>respuesta 1 “<i>answer1</i>” como cadena</p> <p>respuesta 2 “<i>answer2</i>” como cadena</p> <p>respuesta 2 “<i>answer3</i>” como cadena</p> <p>Comentario como cadena</p>
--	--

3.2.7.1.2. Requisito funcional 7.1.2: Sugerir Corrección “Suggest Correction”

Número	RF 7.1.2
Nombre	Sugerir Corrección “Suggest Correction”
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir las correcciones de las preguntas que salen las sección questions de nuestra página, mediante una API.</p> <p>Comprobación de validez de las entradas</p>

	<p>A la hora de introducir la corrección de una pregunta, se generará un formulario de entrada con 2 campos:</p> <p>Question: Para introducir el número de la pregunta que queremos corregir.</p> <p>Comments: Para introducir el comentario que queremos a la pregunta, que será reflejado en la lista de preguntas.</p> <p>Secuencia exacta de operaciones</p> <p>Una vez introducimos los datos de cada campo, y le damos a enviar, se añade el comentario introducido a la lista de preguntas.</p> <p>El usuario tendrá la opción de usar el código HTML ofrecido, o usar el formulario normal.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>pregunta comentario</p> <p>Generación de salidas</p> <p>Una vez que el usuario le da a sugerir, se le enseñara un aviso de gracias, confirmando el éxito de la sugerencia, y la corrección sugerida se verá reflejada en la tabla de preguntas mediante un triángulo informativo.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>id pregunta como entero fecha como variable date comentario como cadena</p>
--	---

3.2.7.2. Requisito funcional 7.2: Gestión de copias de seguridad.

Número	RF 7.2
Nombre	Gestión Backup
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema permitirá guardar copias de seguridad, mediante un Backup, de forma exclusiva para los administradores de la página.</p> <p>La opción Backup debe ser incluida en la sección <i>Tools</i> (Herramientas) de la página.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>A la hora de consultar el Backup el sistema debe generar una copia actualizada, con la posibilidad de restaurar o borrar esta copia.</p> <p>Respuesta a situaciones anormales</p> <p>Después de cada modificación en el Backup, el sistema debe mostrar un aviso informando si la operación se ha realizado con éxito o no.</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>El sistema generará un fichero .sql, que contiene una copia de seguridad de nuestro sistema.</p> <p>Relaciones entre entradas y salidas</p>

	No procede
Especificación de los requisitos lógicos	
	No procede

3.2.7.3. Requisito funcional 7.3: Gestión de la herramienta Exportar

Número	RF 7.3
Nombre	Gestión de la Herramienta Exportar
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema está obligado a ofrecer la opción de exportar las preguntas del sistema para los administradores en diferentes formatos: CSV, Java, y JavaScript.</p> <p>La opción exportar debe ser incluida en la sección Tools (Herramientas) de la página.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>En la sección tools el administrador tendrá 3 formas para exportar:</p> <ul style="list-style-type: none"> RF 7.3.1 Exportar en Formato Java RF 7.3.2 Exportar en Formato JavaScript RF 7.3.3 Exportar en Formato CSV <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p>

	<p>Generación de salidas</p> <p>El sistema debe generar como salida un fichero específico dependiendo de cada formato (.java, .js, .csv).</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.7.3.1. Requisito funcional 7.3.1: Exportar en Formato JAVA

Número	RF 7.3.1
Nombre	Exportar en Formato JAVA
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El administrador puede exportar preguntas de todas las categorías e idiomas que existan en la página en un determinado momento, en formato JAVA</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que estamos en tools, y le damos a exportar, se generará automáticamente el fichero que debe incluir todas las preguntas.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p>

	<p>El Sistema debe generar un fichero específico (.java) programando con lenguaje java, como salida a la hora de pinchar en la opción “<i>Export as Java</i>”.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.7.3.2. Requisito funcional 7.3.2: Exportar en Formato JavaScript

Número	RF 7.3.2
Nombre	Exportar en Formato <i>JavaScript</i>
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Arquitectura del sistema
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El administrador puede exportar preguntas de todas las categorías e idiomas que existan en la página en un determinado momento, en formato <i>JavaScript</i></p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que estamos en tools, y le damos a exportar, se generará automáticamente el fichero que debe incluir todas las preguntas.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p>

	<p>No procede</p> <p>Generación de salidas</p> <p>El Sistema debe generar un fichero específico (.js) programando con lenguaje JavaScript, como salida a la hora de pinchar en la opción “<i>Export as Java Script</i>”.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	--

3.2.7.3.3. Requisito funcional 7.3.3: Exportar en Formato CSV

Número	RF 7.3.3
Nombre	Exportar en Formato CSV
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El administrador puede exportar preguntas de todas las categorías e idiomas que existan en la página en un determinado momento, en formato CSV</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que estamos en tools, y le damos a exportar, se generará automáticamente el fichero que debe incluir todas las preguntas.</p> <p>Respuesta a situaciones anormales</p>

	<p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>El Sistema debe generar La extensión de archivo CSV (Valores separados por comas) es un archivo de texto que contiene una serie de valores separados por comas como salida a la hora de pinchar en la opción “<i>Export as CSV</i>”.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	---

3.2.7.4. Requisito funcional 7.4: Administrar Contact Support

Número	RF 7.4
Nombre	Administrar <i>Contact Support</i>
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>El sistema debe permitir las funciones de contacto y sugerencias de los usuarios, mediante un Contact support, que estará en la sección profile.</p> <p>Comprobación de validez de las entradas</p> <p>La función del contact support será servida mediante un formulario de entrada construido por 2 campos:</p> <ul style="list-style-type: none"> - Campo <i>subject</i>: Para introducir el asunto del mensaje. - Campo <i>Your Message</i>: el texto del mensaje para especificar con más detalles el tipo de problema, queja, o sugerencia.

	<p>Secuencia exacta de operaciones</p> <p>Una vez estamos en la página support contact, después de llenar los dos campos, y darle a enviar “Submit”, el sistema se ocupará de mandar el mensaje generado al administrador de página mediante un correo.</p> <p>Los usuarios pueden mandar el numero de correos que deseen hasta que su problema sea resuelto.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>Después de cada envío de mensajes se generará un aviso de gracias, informando al usuario que el mensaje se ha enviado, y que le contactaremos lo antes posible</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--	--

3.2.7.5. Requisito funcional 7.5: Gestión Estadísticas.

Número	RF 7.5
Nombre	Gestión Estadísticas
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Formulario de ingreso de datos
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional

Descripción	<p>El Sistema debe tener una sección propia "Statistics" para agrupar las estadísticas de todas las secciones del sistema (<i>Questions, Categories, Languages, users</i>).</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>Una vez que estamos en la sección tools, si pinchamos en la herramienta <i>Statistics</i> tendríamos acceso a todas las gráficas de la página.</p> <p>Respuesta a situaciones anormales</p> <p>No procede</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>A la hora de entrar a esta sección, el sistema debe generar como salida, las estadísticas del sistema, de forma ordenada, con el título que identifica cada una de las gráficas.</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--------------------	---

3.2.8. Requisito funcional 8: Desarrollo

3.2.8.1 Requisito funcional 8.1: Cantidad de usuarios concurrentes.

Número	RF 8.1
Nombre	Cantidad de usuarios concurrentes

Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Arquitectura del sistema
Prioridad	<input type="checkbox"/> Alta/Esencial <input checked="" type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas No procede</p> <p>Secuencia exacta de operaciones El Sistema debe soportar un número de usuarios de acuerdo a los recursos de infraestructura. El número máximo dependerá de la capacidad del servidor. El usuario podrá acceder tantas veces al sistema como le sea necesario.</p> <p>Respuesta a situaciones anormales En el caso de que el servidor no puede soportar todas las peticiones de los usuarios, se informará al administrador del sistema, que hará cambios lo antes posible, para solucionar el problema.</p> <p>Parámetros No procede</p> <p>Generación de salidas No procede</p> <p>Relaciones entre entradas y salidas No procede</p> <p>Especificación de los requisitos lógicos No procede</p>

3.2.8.2 Requisito funcional 8.2: Cantidad de información almacenada

Número	RF 8.2
---------------	---------------

Nombre	Cantidad de información almacenada.
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Arquitectura del sistema
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas No procede</p> <p>Secuencia exacta de operaciones El número máximo de información a registrar dependerá de la capacidad del servidor. El sistema debe soportar un número de inscripciones de acuerdo a los recursos de infraestructura.</p> <p>Parámetros No procede</p> <p>Respuesta a situaciones anormales En el caso de que el sistema ya no puede soportar más inscripciones, informará al administrador, para renovar los recursos de infraestructura</p> <p>Generación de salidas No procede</p> <p>Relaciones entre entradas y salidas No procede</p> <p>Especificación de los requisitos lógicos No procede</p>

3.2.8.3 Requisito funcional 8.3: Sistema Operativo

Número	RF 8.3
---------------	---------------

Nombre	Sistema Operativo
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	Arquitectura del sistema
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas No procede</p> <p>Secuencia exacta de operaciones La página debe ser independiente del sistema operativo utilizado . Los usuarios tendrán acceso al sistema desde cualquier SO, sea Windows, Linux, Mac.</p> <p>Respuesta a situaciones anormales No procede</p> <p>Parámetros No procede</p> <p>Generación de salidas A la hora de generar la página, se genera igual para todos los sistemas operativos.</p> <p>Relaciones entre entradas y salidas No procede</p> <p>Especificación de los requisitos lógicos No procede</p>

3.2.8.4 Requisito funcional 8.4: Control de errores HTTP

Número	RF 8.4
Nombre	Control de errores HTTP (403, 404, 500)
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción

Fuente	Arquitectura del sistema
Prioridad	<input type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input checked="" type="checkbox"/> Baja/Opcional
Descripción	<p>Comprobación de validez de las entradas Debe haberse intentado acceder a una página inexistente</p> <p>Secuencia exacta de operaciones Al intentar acceder a una página inexistente el sistema mostrará una página de error personalizada.</p> <p>Respuesta a situaciones anormales No procede</p> <p>Parámetros No procede</p> <p>Generación de salidas Se mostrará una página mostrando el número de error http (403 404, 500).</p> <p>Relaciones entre entradas y salidas No procede</p> <p>Especificación de los requisitos lógicos No procede</p>

3.2.9. Requisito funcional 9: Disponibilidad del sistema

Número	RF 8
Nombre	Disponibilidad del sistema.
Tipo	<input checked="" type="checkbox"/> Requisito <input type="checkbox"/> Restricción
Fuente	
Prioridad	<input checked="" type="checkbox"/> Alta/Essencial <input type="checkbox"/> Media/Deseado <input type="checkbox"/> Baja/Opcional

Descripción	<p>El sistema debe ofrecer una disponibilidad completa a menos que sucedan causas externas como: que el administrador esté actualizando la información.</p> <p>Comprobación de validez de las entradas</p> <p>No procede</p> <p>Secuencia exacta de operaciones</p> <p>No procede</p> <p>Respuesta a situaciones anormales</p> <p>En el caso de que el sistema no esté disponible por cualquier motivo, se generará un aviso informando de la situación.</p> <p>Parámetros</p> <p>No procede</p> <p>Generación de salidas</p> <p>No procede</p> <p>Relaciones entre entradas y salidas</p> <p>No procede</p> <p>Especificación de los requisitos lógicos</p> <p>No procede</p>
--------------------	---

3.3. Requisitos no funcionales

3.3.1. Requisitos de rendimiento

El sistema se debe ajustar a sus expectativas en cuanto a los tiempos de respuesta y ser capaz de prestar servicio adecuadamente de acuerdo al tipo de usuarios y tamaño para el que ha sido concebido.

3.3.2. Seguridad

La seguridad del sistema no sólo tiene en cuenta la seguridad del sistema sino, además, el ambiente en el que se usará el sistema. Por lo que se tiene que contemplar la seguridad física del lugar donde se usa la aplicación, los controles administrativos que se establecen de acceso al sistema y las regulaciones legales

que afecta o determina el uso del sistema y que serán tenidas en cuenta si se incumple. La seguridad de nuestro sistema trata de tres aspectos diferentes:

- **Confidencialidad:** La información manejada por el sistema está protegida de acceso no autorizado y divulgación.
- **Integridad:** La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual a la fuente o autoridad de los datos.
Pueden incluir también mecanismos de chequeo de integridad y realización de auditorías.
- **Disponibilidad:** Significa que los usuarios autorizados se les garantizará el acceso a la información y que los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.

3.3.3. Fiabilidad

Uno de los más importantes por parte del usuario es la fiabilidad debido a que debe de tener en cuenta la recuperación frente a fallos de conexión: asegurar que no se pierdan los datos del perfil definido por el usuario.

Esto incluye no perderlos en el envío al servidor o en el envío a otras máquinas, como no perderlos si hay un fallo de conexión entre el receptor del usuario y el servidor.

- **La recuperación frente a fallos del sistema:** posibilidad de reiniciar el sistema. Verificar la fiabilidad en la autentificación de los usuarios y la posibilidad de dar marcha atrás en la definición del perfil de cada usuario.

3.3.4. Disponibilidad

- La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días x 24 horas.
- Debe contemplar requerimientos de confiabilidad y consistencia de los componentes ante recuperaciones. En caso de fallos de algún componente, no debe haber pérdida de información.

3.3.5. Mantenibilidad

El sistema cuenta con características parametrizables lo que permitirá futuros mantenimientos. Cada tres meses se va a realizar un mantenimiento preventivo, realizado por los desarrolladores.

3.3.6. Portabilidad

Al adecuarse a los estándares web, el sistema puede ser instalado en cualquier servidor HTTP con soporte de PHP, de todas formas el sistema está optimizado para Apache y requiere al menos la versión 5 de PHP por utilizar muchas de las mejoras en orientación a objetos de dicha versión.

Se podría cambiar de motor de BBDD de una manera sencilla al estar esta parte bien localizada gracias al uso del patrón MVC.

3.3.7. Multiplataforma.

El sistema debe ser accesible desde cualquier tipo de navegador (Internet Explorer, Google Chrome, Mozilla Firefox, Opera....) y desde dispositivos móviles (iOS, Android).

El sistema generará una página bien ordena, con todas las funcionalidades y con el tamaño de pantalla variado según con el navegador que estamos usando.

3.4. Otros requisitos

3.4.1 Normas de estilo de codificación

Para el desarrollo del código se seguirán las siguientes normas:

3.4.1.1 Base de datos:

- Nombres siempre en inglés.
- Siempre en minúsculas.
- No usar caracteres especiales.
- Las tablas siempre se pondrán en plural.
- Los campos siempre en singular.

3.4.1.2 Consultas SQL

- Siempre en mayúsculas.
- Siempre usar comillas simples para las cadenas.
- Cada cláusula de la sentencia abrirá una nueva línea.

3.4.1.3 Lenguaje PHP

- Nombres de variables siempre en inglés utilizando [lower Camel Case](#).
- Nombres de funciones siempre en inglés, minúsculas y separación con subguiones, utilizar el estilo verbo_acción (ej: get_question.php).
- Nombres de fichero siempre en inglés, minúsculas y separación con subguiones.
- Utilizar siempre comillas dobles.
- Si hay que concatenar variables dentro de una cadena se utilizará el sistema {\$nombrevariable} para hacer el código más legible.
- El estilo de apertura y cierre de llaves será el [Kernighan & Ritchie / Kernel 1TBS](#).
- La indentación será mediante tabuladores.

3.4.1.4 Lenguaje HTML

- Nombres de clases e identificadores siempre en inglés, minúsculas y separación con subguiones.
- Nombres de fichero siempre en inglés, minúsculas y separación con

subguiones.

- Utilizar siempre comillas dobles.

3.4.1.5 Lenguaje CSS

- Nombres de clases e identificadores siempre en inglés, minúsculas y separación con subguiones.
- El estilo será igual que PHP ajustado a las particularidades del lenguaje

3.4.1.6 Lenguaje JavaScript y Java

- El estilo será igual que PHP ajustado a las particularidades del lenguaje

3.4.2. Comentarios en el código.

- Los comentarios en el código serán siempre en castellano.
- Se intentará hacer el código lo más sencillo posible y que sea autoexplicativo.
- Se agregará un comentario con la explicación encima de la definición de cada función.
- Se agregarán comentarios explicativos en las funciones más extensas o de más difícil comprensión.
- Se evitará el “spaghetti coding” y el síndrome de diógenes del programador (dejar grandes porciones de código comentadas por si hiciera falta en un futuro).

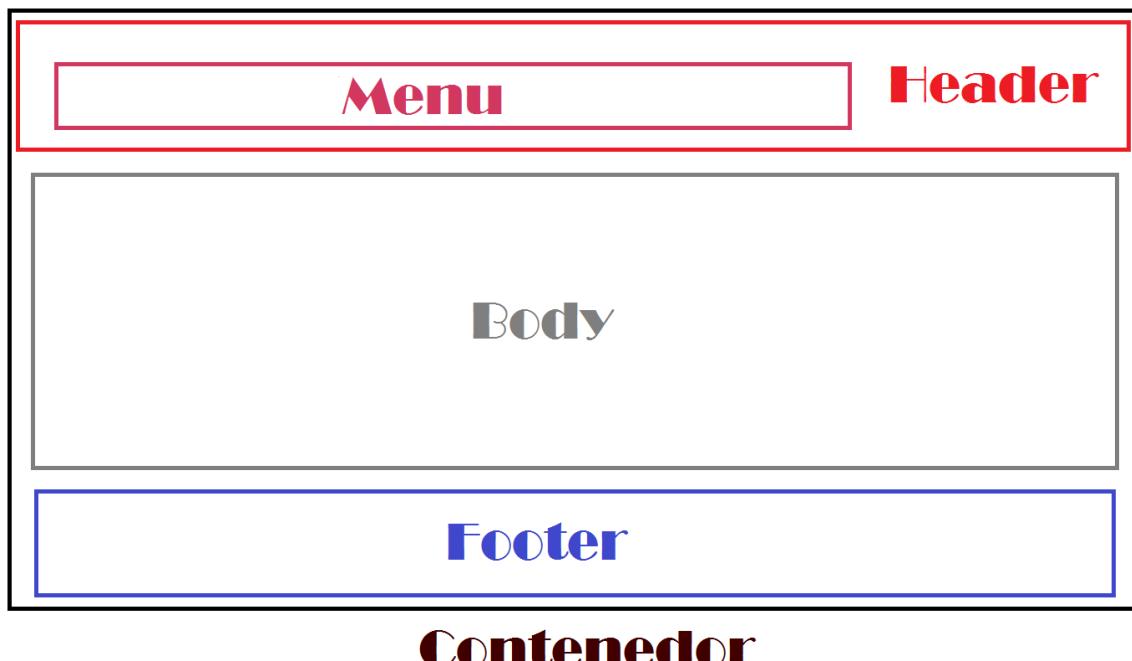
3.5. Apéndices

No se han definido.

4. Descripción de Solución

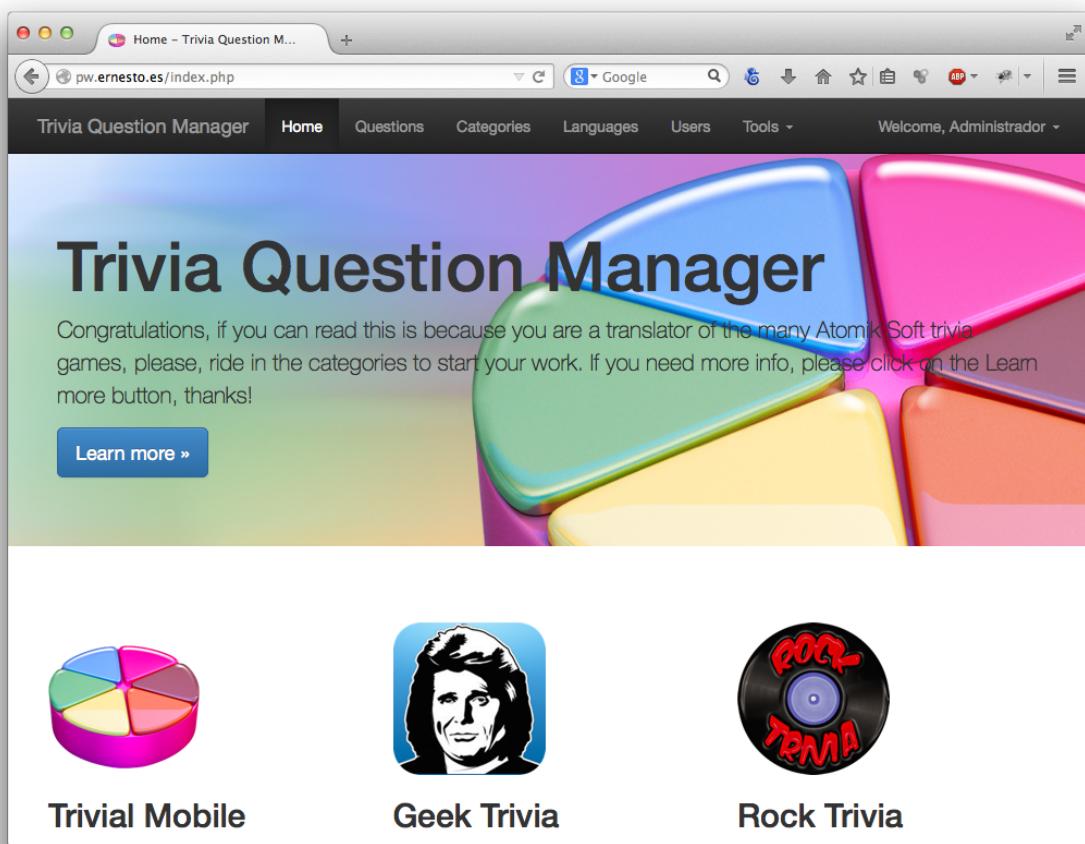
4.1. Estructura de página

Nuestra página se estructura de la siguiente forma:



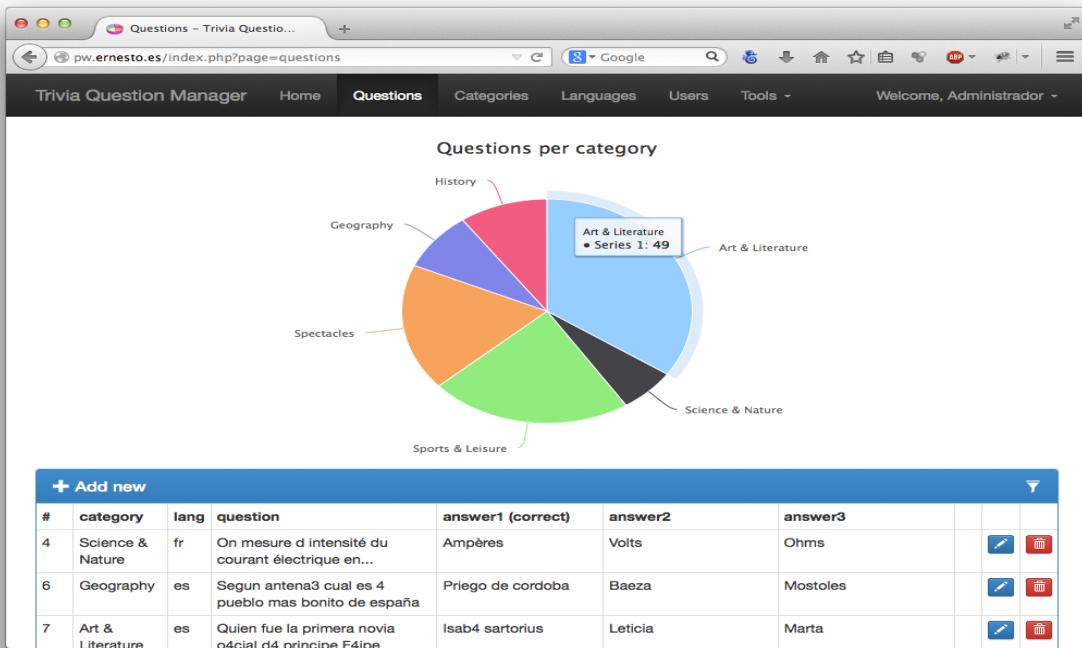
4.1.1. Página principal “Home”

Una vez accedemos a la página en la URL del servidor web donde se haya instalado nos encontramos con la pagina principal que tiene un pequeño resumen sobre el sistema de corrección de preguntas y las diferentes aplicaciones Trivial, con posibilidad de descargar cada una en Google Play.



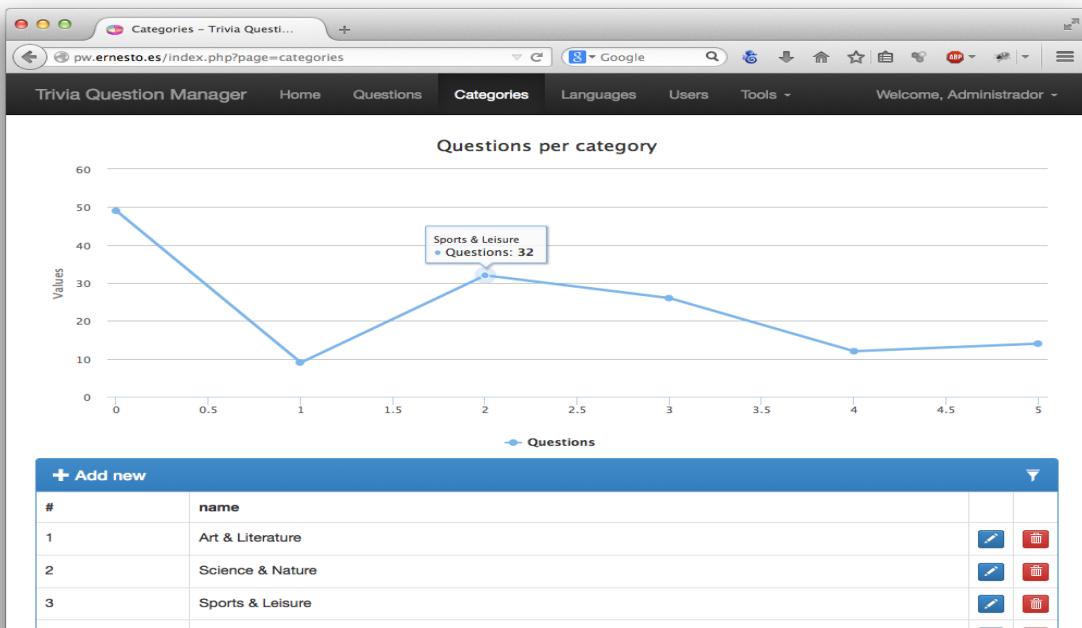
4.1.2. Preguntas “Questions”

Es la parte donde los usuarios registrados tienen la posibilidad de editar, eliminar las preguntas, luego tenemos una gráfica que demuestra cuántas preguntas hay editadas por los usuarios.



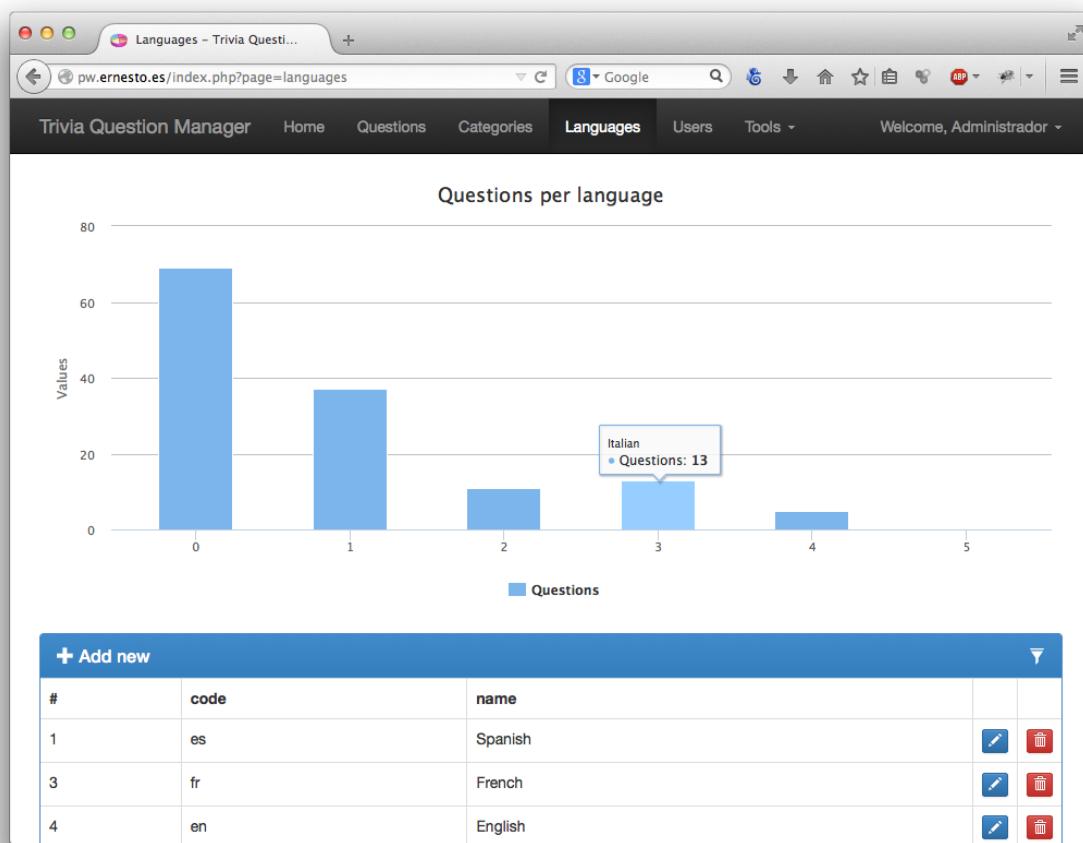
4.1.3 Categorías “Categories”

Donde se muestran las diferentes categorías de las preguntas que tenemos - Arte, ciencia, deporte, historia - con la opción de editar y eliminar, y una gráfica donde salen las estadísticas de cada una de las categorías.



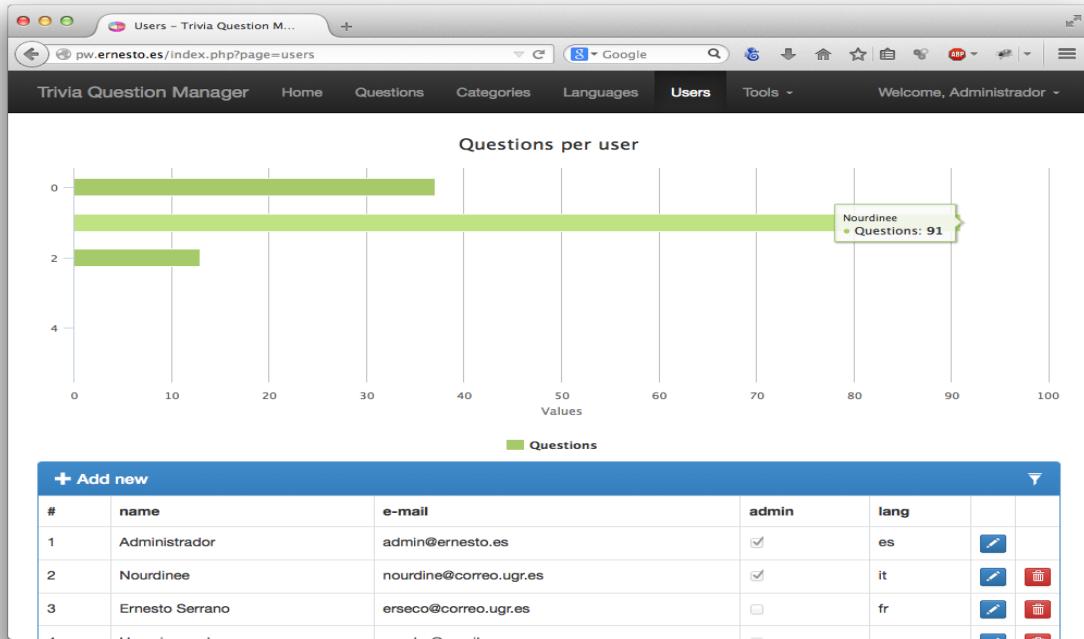
4.1.4. Idiomas “Languages”

Aquí salen los diferentes lenguajes que soporta nuestro sistema, inglés español, italiano, que podemos editar, eliminar o añadir otros nuevos, con una gráfica de las diferentes estadísticas de cada lengua.



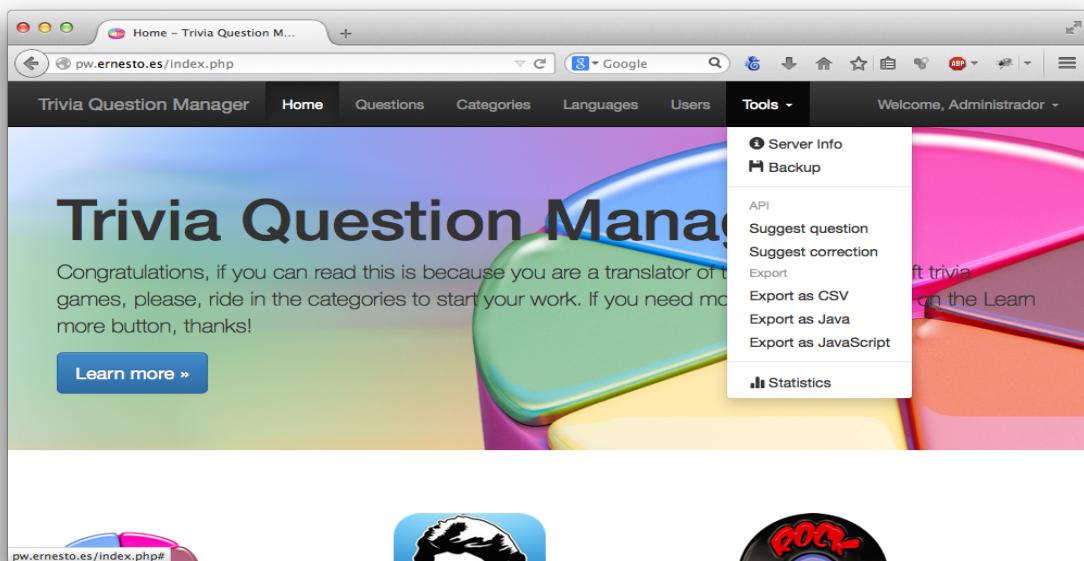
4.1.5. Usuarios “Users”

Sale una lista de todos los usuarios del sistema, que solo el administrador la puede modificar añadiendo o eliminando un usuario, y una gráfica con las estadísticas de cada uno de ellos.



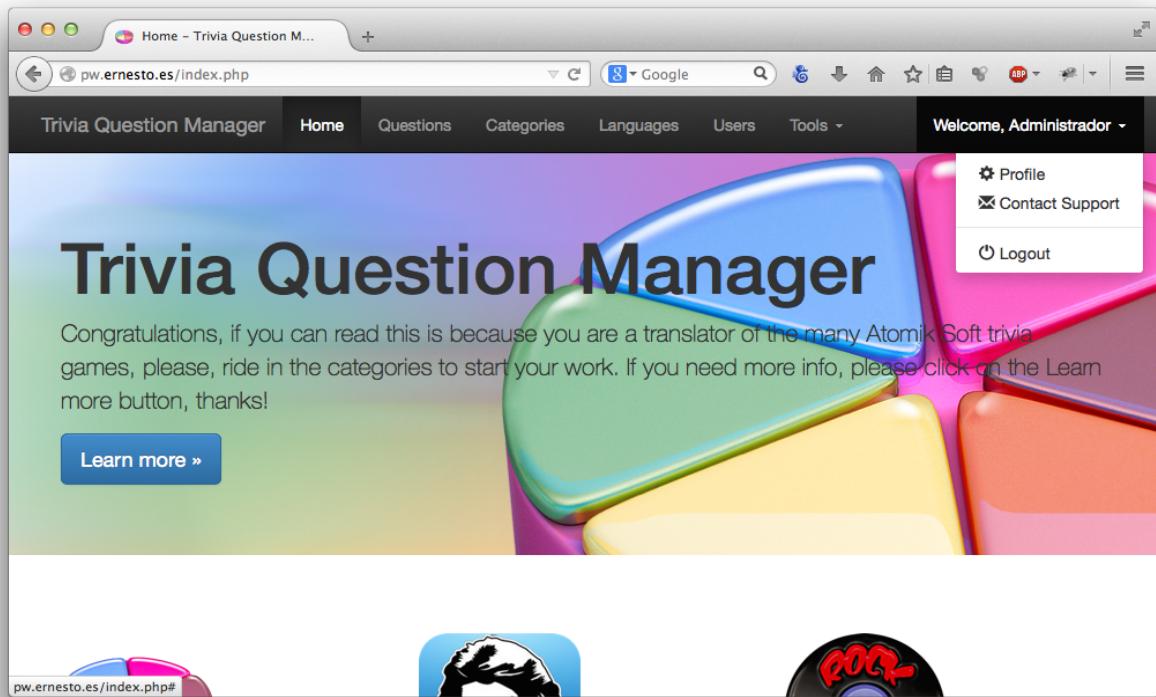
4.1.6. Herramientas “Tools”

Una ventana que nos da la opción de exportar las preguntas, en formato CSV, .JAVA O.JAVASCRIPT, y donde los usuarios pueden sugerir preguntas o correcciones a partir de un API (Interfaz de programación de aplicaciones), además de una ventana de todas las estadísticas del sistema.



4.1.7. Perfil “Profile”

Una pequeña ventana, a la derecha que contiene los datos del usuario (foto, nombre, correo...), con una opción de contacto, y otra para salir de la página.



Además de la estructura de página hemos puesto una tabla explicando de forma detallada las carpetas y archivos que hemos entregado con la documentación:

4.2. Estructura de carpetas y archivos entregados:

/	Carpeta raíz del sitio
.htaccess	Archivo de configuración de apache, para activar el 404.htm personalizado y el uso de memoria de php.
404.html	Página de error personalizada.
config.php	Archivo de configuración del sitio.
favicon.ico	Icono del sitio
humans.txt	Archivo donde se indican los creadores de la página, según lo indicado en http://humanstxt.org/

index.php	Página de entrada al sitio, carga las distintas páginas (cabecera y pie) y carga una vista según el parámetro indicado por GET
login.php	Muestra una página para el inicio de sesión
logout.php	Cierra la sesión actual y redirige a login.php
robots.txt	Archivo de configuración de motores de búsqueda, hace que la pagina no se indexe, ya que es para uso privado
session.php	Archivo de chequeo de sesión, si no está iniciada redirige a login.php
model	Carpeta donde van los archivos del Modelo (acceso a datos)
categories.php	Clase para interactuar con la tabla categorías de la BBDD, hereda de model.php
corrections.php	Clase para interactuar con la tabla corrections de la BBDD, hereda de model.php
db.php	Clase que gestiona la conexión a la BBDD, se usa pdo-mysql
export.php	Clase base para generar archivos de exportación.
export_csv.php	Clase que genera archivos CSV, hereda de export.php
export_java.php	Clase que genera archivos Java, hereda de export.php
export_js.php	Clase que genera archivos JS, hereda de export.php
languages.php	Clase para interactuar con la tabla languages de la BBDD, hereda de model.php
model.php	Clase base para generar consultas a la BBDD.
mysqldump.php	Clase utilizada para hacer un dump de una base de datos MySQL
questions.php	Clase para interactuar con la tabla questions de la BBDD, hereda de model.php
users.php	Clase para interactuar con la tabla users de la BBDD, hereda de model.php
view	Carpeta donde van los archivos de la Vista (interfaz). Nótese que se usa la extensión .phtml para indicar que contienen código html mezclado con php
backup.phtml	Interfaz para realizar y restaurar backups, depende de mysqldump.php
categories.phtml	Página para visualizar las categorías
credits.phtml	Página para visualizar los créditos

footer.phtml	Pie común de todas las páginas
header.phtml	Cabecera común de todas las páginas, contiene el menú superior
home.phtml	Página principal, es la que se ve después de hacer login
info.phtml	Muestra información de PHP llamando a la función phpinfo();
insert_correction_de_mo.phtml	Demostración de uso de la API
insert_question_demo.phtml	Demostración de uso de la API
languages.phtml	Página para visualizar los idiomas
login.phtml	Página de inicio de sesión
profile.phtml	Página para visualizar el perfil del usuario
questions.phtml	Página para visualizar las preguntas
statistics	Página para visualizar las gráficas de estadísticas
support	Página para mostrar el formulario de soporte
users	Página para visualizar los usuarios
controller	Carpeta donde van los archivos del Controlador (comunicación entre vista y modelo)
backup_create.php	Crea un backup llamando a mysqldump.php
backup_restore.php	Funciones para restaurar un backup
chart_data_[*].php	Obtiene los datos JSON utilizados en las gráficas, para ser llamado mediante AJAX
delete.php	Controlador común para borrar registros en cualquier tabla
get_chart_[*].php	Obtiene una gráfica de la tabla indicada por el nombre del archivo
get_combo_[*].php	Genera un combo de la tabla indicada en el nombre del archivo
get_rows_[*].php	Obtiene las filas de la tabla indicada en el nombre del archivo, para ser llamado mediante AJAX
insert.php	Controlador común para insertar registros en cualquier tabla
login.php	Controlador de login, comunica con el modelo y crea las cookies y sesiones pertinentes
modal_[*].php	Obtiene una ventana modal de Bootstrap para inserción o modificación de la tabla indicada en el nombre del archivo.

support.php	Envía un email al soporte técnico, es llamado desde la página de formulario de soporte
update.php	Controlador común para actualizar registros en cualquier tabla
export	Aquí se ubican las páginas que exportan archivos
export_csv.php	Llama al modelo para exportar un archivo CSV
export_java.php	Llama al modelo para exportar un archivo Java
export_js.php	Llama al modelo para exportar un archivo JavaScript
api	API que expone unas páginas que reciben una serie de parámetros por POST e insertan un registro en la BBDD
insert_correction.php	Inserta una corrección
insert_question.php	Inserta una pregunta
img	Carpeta de imágenes, no es necesaria su explicación
js	Carpeta que contiene las funciones javascript, las funciones de la carpeta vendor son funciones de terceros y remitimos a su correspondiente documentación.
main.js	Funciones propias de Inserción, Actualización y Borrado mediante ajax, así como los distintos refreshes y validaciones.
tablefilter.js	Funciones para el filtrado de las tablas usando jQuery
tablesort.js	Funciones para el orden de las tablas usando jQuery
resizable-tables.js	Funciones para cambiar el ancho de las columnas
css	Archivos de definición de CSS, la carpeta vendor contiene hojas de estilos de terceros y remitimos a su correspondiente documentación.
main.css	Archivo que contiene los estilos CSS personalizados de la página heredan de los propios de Bootstrap
ie.css	Archivo con ajustes para la correcta visualización de la página en Internet Explorer

4.3. Descripción detallada:

A la hora de describir los requisitos de nuestro sistema, hemos seguido una estructura fija para todos, que es;empezar describiendo la base de datos, luego detallar la la parte de html y css, y por último, todo lo que tiene que ver con la parte de php.

Para la realización de este proyecto, nos basamos en la idea de una página web con una base de datos.Por lo tanto las herramientas más adecuadas para su desarrollo serán el lenguaje de marcado hipertextual o *HTML* y hojas de estilo en cascada o *CSS*.además de los lenguajes de programación *css* y *html*, hemos utilizado herramientas como:

Twitter Bootstrap: que es un conjunto de herramientas de software libre para diseño de sitios y aplicaciones web, basado en *html* y *css*, así como, extensiones de *JavaScript*.

y

jQuery: que es una biblioteca de javascript, que permite simplificar la manera de interactuar con los documentos *html*, manipular el árbol *dom*, manejar eventos, desarrollar animaciones y agregar interacción con la técnica *ajax* a páginas web.

Como es un sistema web de gestión de preguntas necesitaremos de alguna herramienta que nos permita la comprobación de datos. Nos valdremos del lenguaje de programación *Javascript*. Para manipular la base de datos y hacer peticiones al servidor, nos valdremos de la funcionalidad del lenguaje *php*.

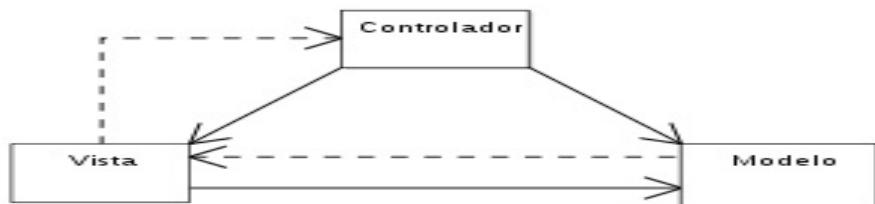
Para crear, y desarrollar nuestra base de datos, hemos utilizado un programa llamado ***MySQL Workbench*** que te permite crear las tablas de forma ordenada, y muy profesional.

Hemos usando la pagina <http://www.highcharts.com/> como referencia, para implementar y diseñar las graficas.

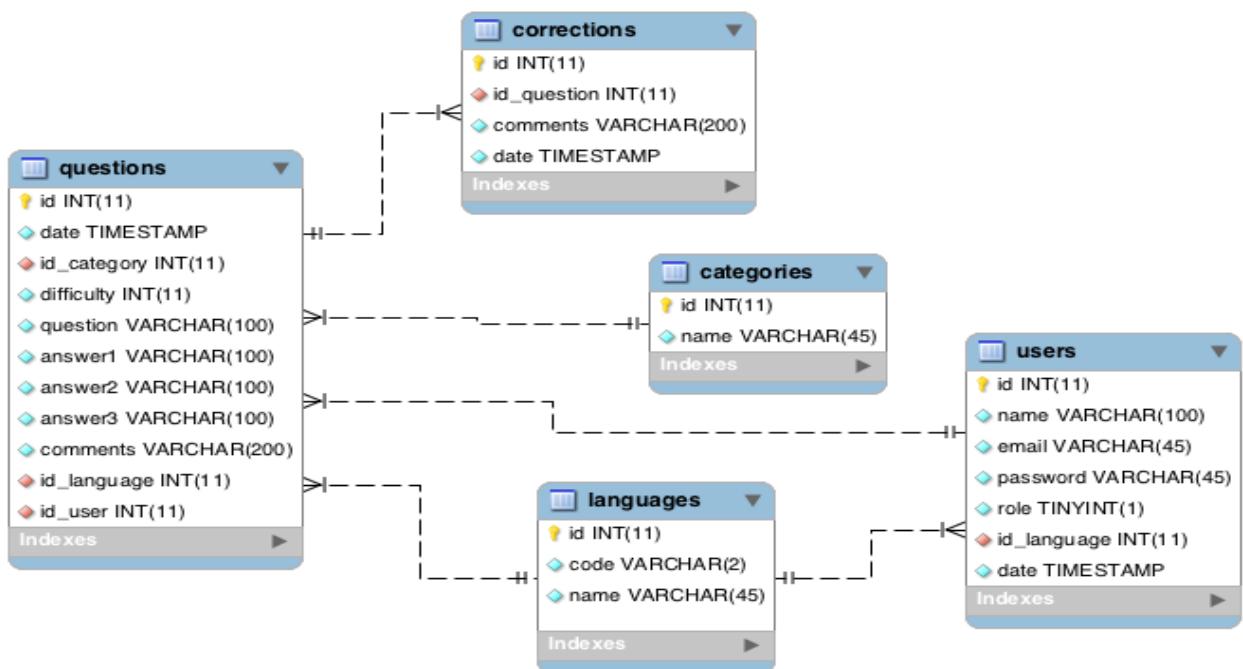
Hemos seguido una guía a la hora de de implementar nuestro código, usando **tabulaciones, formato de llaves Kernel y Thompson, CamelCase** (es un estilo de escritura que se aplica a frases o palabras compuestas)

Todas las variables que vienen en el código están en inglés, y los comentarios en español.

A la hora de desarrollar el código hemos utilizado la estructura modelo vista controlador(model view controller)que funciona de la siguiente forma:



En seguida viene una descripción detallada de cada parte de nuestro proyecto:

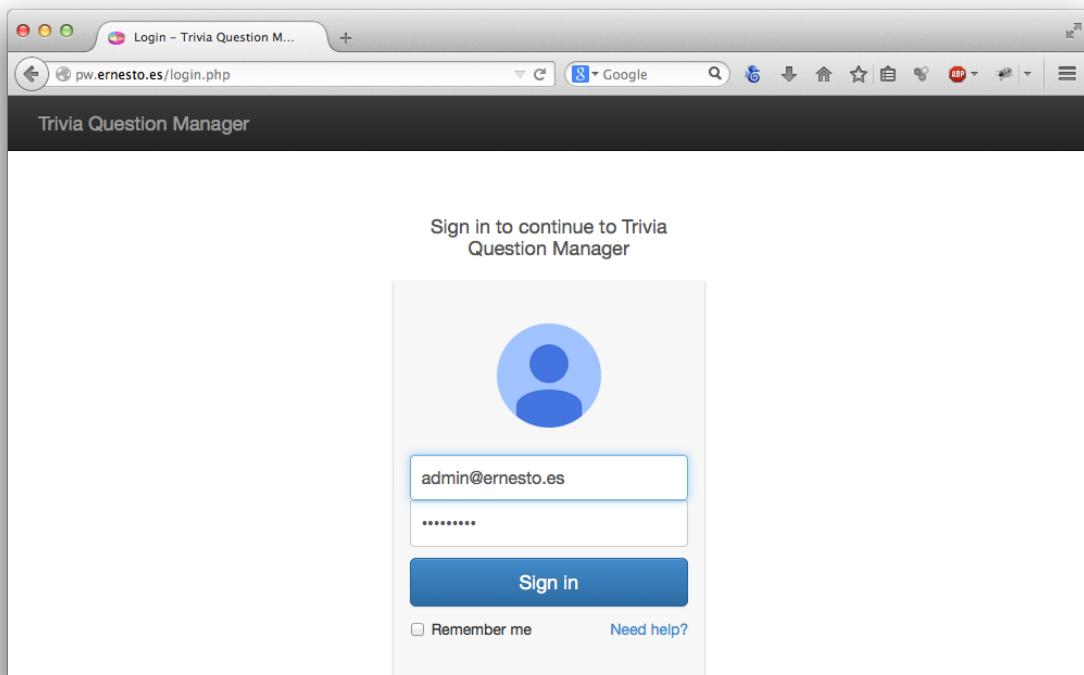


4.3.1.: Gestión usuarios “Users”

4.3.1.1.Autentificar usuario

Este Apartado refiere al requisito funcional:

RF 1.1 Autentificar usuario.



La ventana de acceso para los usuarios, de aqui si definirá el rol de cada uno de ellos, y se llamará a todos las demás páginas.

Empezamos hablando de cómo hemos desarrollado la parte *html* y *css* que se considera como *vista* en el patrón de arquitectura que hemos usado (modelo–vista–controlador):

CSS y HTML: Primero hemos tenido que declarar una etiqueta div que se refiere al contenedor, y dentro de este div metemos un conjunto de etiquetas div, una para contener la imagen del perfil, otro para indicar que hay registrar para entrar mediante un aviso.

Una vez tenemos esta parte hecha, queríamos mostrar un mensaje de error, en el caso de que fallan los datos de entrada, y eso lo hemos hecho mediante otra etiqueta div, con una función condicional, de que si los datos introducidos nos son correctos, que salte ese error.

Por último creamos los dos campos del formulario de entrada que son *email* y *password*, y una etiqueta para el botón de entrar, y para terminar la opción recuérdame que genera el cookie, que viene detallada seguida.

Terminada la parte de *html* y *css*, pasamos a detallar de cómo hemos gestionado el login en la parte de *php*

PHP: Mediante el modelo, cargamos primero la vista creada anteriormente, después la configuración de *php*, y incluimos el controlador que nos gestiona el *login*, para eso hemos tenido que usar un controlador de *login* que chequea si los valores establecidos son correctos y guarda los valores en la sesión y en *cookies*.

Empezamos con la bandera de error para mostrar una advertencia en la vista, luego creamos una función de control de *Email Password*, que obtiene esos dos campos anteriores, e inicia un nuevo objeto de la clase *Users*.

Si todo esta en condiciones iniciamos la sesión *php*, obtenemos la información de usuario, y guardamos todas las variables variables de sesión.

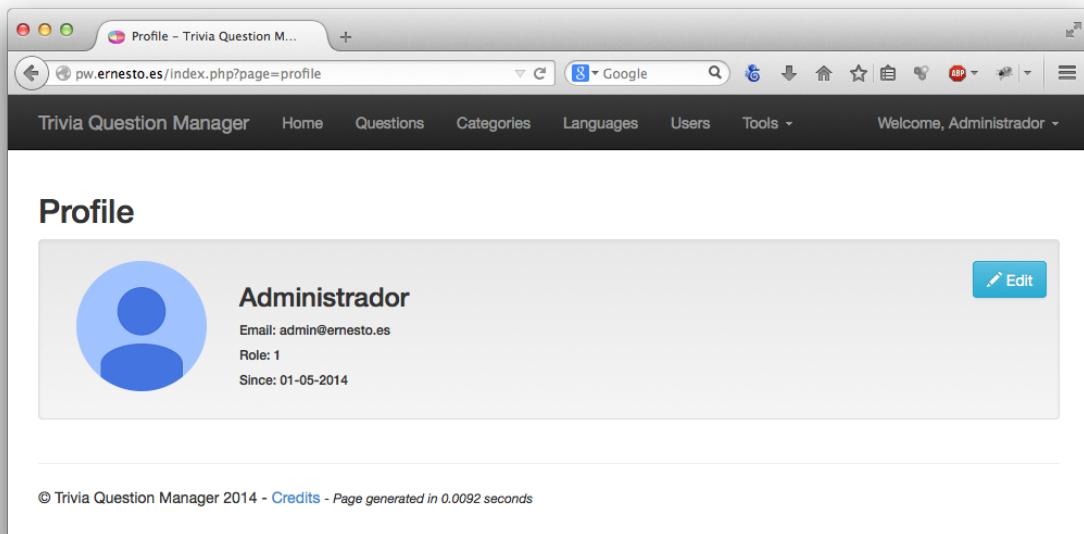
En el caso que tenemos marcada la opción de Recordar, establecemos la duración de una semana mediante una función, luego establecemos las *cookies* de las variables declaradas anteriormente, y terminamos redirigiendo a la página principal.

4.3.1.2. Gestión del perfil “Profile”

Este Apartado refiere al requisito funcional:

RF 1.2 Gestión sección perfil “Profile”.

RF 1.3: Actualización datos de usuarios.



The screenshot shows a web browser window for the 'Trivia Question Manager' application. The URL in the address bar is 'pw.ernesto.es/index.php?page=profile'. The page title is 'Profile - Trivia Question M...'. The main content area is titled 'Profile' and displays a user's profile picture (a blue placeholder icon), the role 'Administrador', and the email 'admin@ernesto.es'. Below this, it shows the role 'Role: 1' and the creation date 'Since: 01-05-2014'. A blue 'Edit' button is located in the top right corner of the profile box. At the bottom of the page, there is a footer with the text '© Trivia Question Manager 2014 - Credits - Page generated in 0.0092 seconds'.

Es la sección donde se muestra el perfil del los usuarios logueados y permite la opción de editarlo.

Empezamos creando la vista, para eso declaramos la primera etiqueta div para contener el título de la sección, después declaramos otra etiqueta div, para incluir las etiquetas de información del perfil.

Hemos tenido que crear otra etiqueta div, que contiene el formulario con los datos del perfil, que se genera una vez el usuario pincha el botón de editar

Otra etiqueta div la hemos definido para contener la ventana de alerta que muestra una alerta al finalizar la modificación, informando de que los datos del perfil se han cambiado correctamente y hay que cerrar la sesión y volver a entrar otra vez.

4.3.1.4. Administrar cierre de sesión “Logout”

Este Apartado refiere al requisito funcional:

RF 1.4: Administrar Logout

Es la parte de la página que permite a los usuarios cerrar la sesión abierta, y limpiar las cookies.

Una vez que un usuario accede a la opción logout, se borrará de forma automática toda la sesión, y mediante una función se establece la duración en ayer.

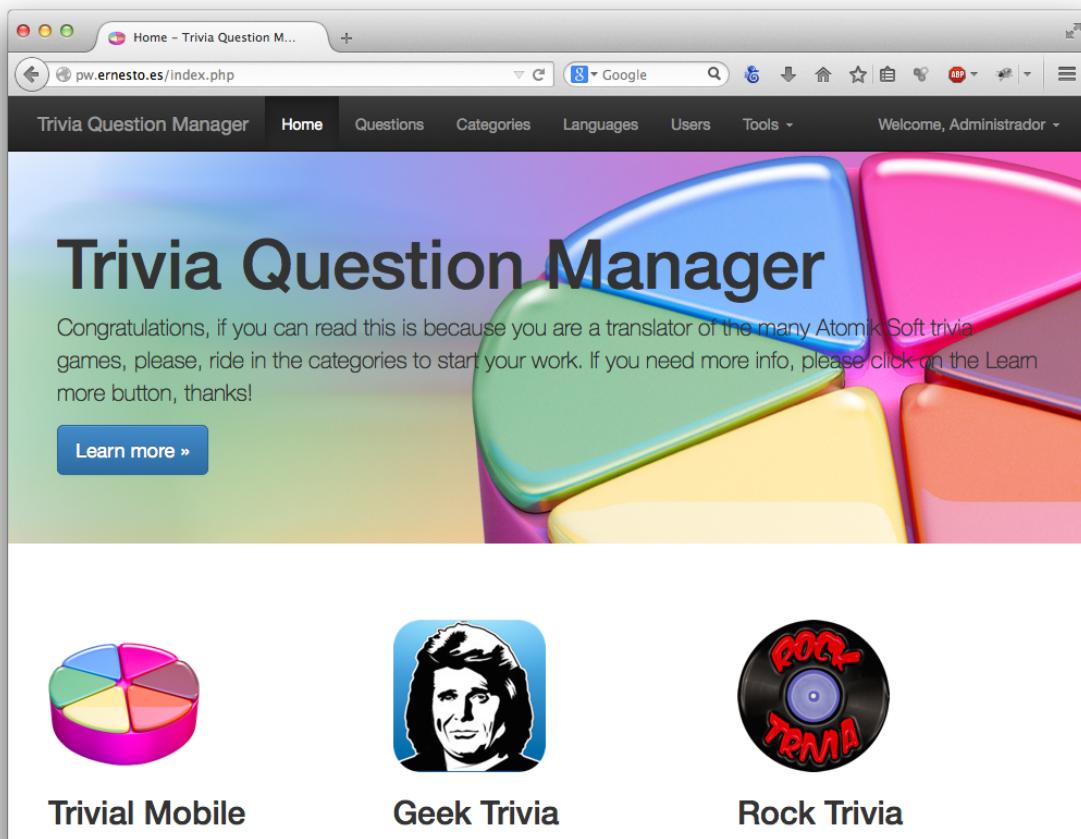
Por último borramos las Cookies correspondientes a la sesión(*name*, *email*, *role*...)

una vez esta hecho todo eso, nos redirigimos a la página principal.

4.3.2. Gestión de pagina principal “Home”

Este Apartado refiere al requisito funcional:

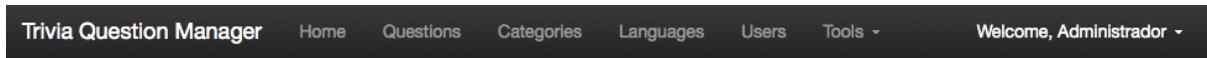
RF 2: Gestión de pagina principal “Home”



Es la pagina principal, que da la bienvenida al usuario, con una pequeño parágrafo, en la parte superior, sobre el sistema *Trivia Question Manger*, justo debajo de ese resumen salen las tres enlaces directos, para facilitar a los usuarios las descargas de los juegos a partir de la pagina de *Play Google*.

Para eso empezamos creando primero el menú declarando un etiqueta div, para englobar un conjunto de etiquetas div, La primera para incluir las diferentes áreas de nuestra página, declarando etiquetas de listas no ordenadas para sección *Questions, Categories, Languages, Users*.

Y luego para *tools* declaramos otra etiqueta para la parte desplegable que incluye todas las opciones de *tools*



Una vez terminado el menú, pasamos a la parte del medio, creamos otra etiqueta div que contiene las etiquetas de título, y la etiqueta que hace referencia a la breve introducción de la página.

Después pasamos a la parte baja de la página, que incluye otra etiqueta div para garantizar el orden de las columnas de forma horizontal, dentro de esa misma etiqueta metemos otras etiquetas para cada uno de los juegos, con la etiqueta que hace la referencia a la foto, otra para el título de los juegos, con un pequeño resumen para cada uno de ellos.

Y por último definimos una etiqueta para el pie de página, que incluye un enlace, donde viene todas las *Tecnologías, Librerías, y Frameworks* que hemos usado.

4.3.3. Gestión Preguntas “Questions”

Este Apartado refiere al requisito funcional:

RF 3 Gestión Preguntas “Questions”

RF 3.1 Añadir preguntas

RF 3.2 Editar preguntas

RF 3.3 Borrar preguntas

RF 3.4 Buscar preguntas

La sección más importante y extensa de nuestro sistema, que contiene todas las preguntas de la página con opciones de añadir, editar y borrar, sea usuario registrado o administrador.

Empezamos por la parte de la base de datos, creando la tabla *Questions* que vamos a usar en desarrollar las preguntas del sistema, que tiene:

- un identificador como clave primaria.
- Variables de tipo *char* como *answer1*, *comments*.
- Variables de tipo *entero* como *category*, *difficulty* y otras de tipo *date*.

questions	
id	INT(11)
date	TIMESTAMP
id_category	INT(11)
difficulty	INT(11)
question	VARCHAR(100)
answer1	VARCHAR(100)
answer2	VARCHAR(100)
answer3	VARCHAR(100)
comments	VARCHAR(200)
id_language	INT(11)
id_user	INT(11)
Indexes	

Después creamos la tabla *corrections*, para incluir correcciones a la sección preguntas que tiene:

- Un identificador como clave primaria.
- Variables de tipo cómo: *language*, *versión*, *comments*... etc
- Variables de tipo entero cómo: *id_question*, *id_reason*.

corrections	
id	INT(11)
id_question	INT(11)
comments	VARCHAR(200)
date	TIMESTAMP
Indexes	

Una vez tenemos la base de datos en condiciones, con todas las tablas pasamos a crear nuestro vista con

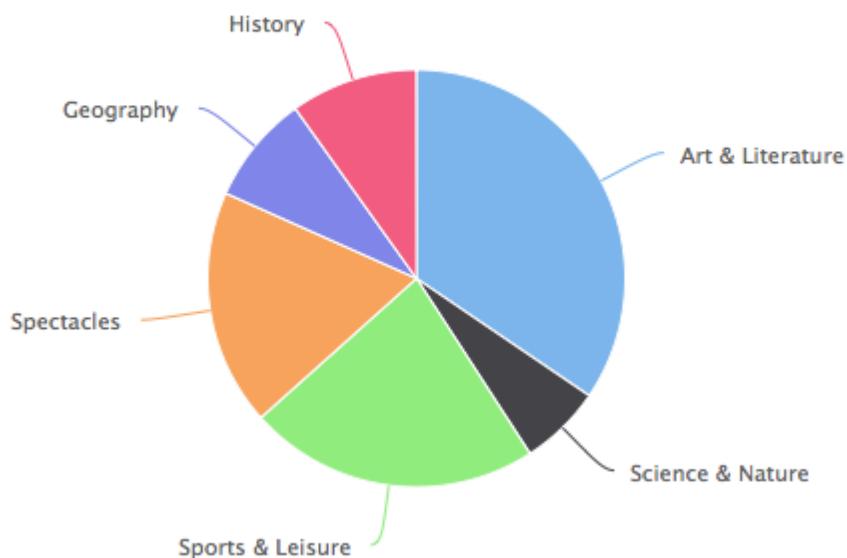
HTML y CSS: usaremos *jQuery* para crear un filtro de las preguntas de nuestro sistema, luego definimos una etiqueta div para contener, la etiqueta que refiere al título de la sección, otra etiqueta para la tabla de preguntas con el uso de otras etiquetas, como la de definir cada fila y columna de la tabla, otra que nos permite agrupar varios elementos en línea seguidos dentro de un mismo bloque.

Una vez vez tenemos la vista creada, de nuestra sección preguntas, pasamos a desarrollar el modelo con el uso de *php*.

PHP: primero incluimos todos los controladores que hacen referencia a esta área como:

a) - el controlador “*get_chart_questions*” hace referencia a la gráfica *questions* que nos representa el total de preguntas en cada una de las categorías.

Questions per category



Declarando primero una variable *chart* (para el área de gráfico), y para construir resto de la gráfica necesitamos una serie de atributos como:

- *Chart*: que contiene el color del fondo, y el tamaño del borde.
- *Title*: contiene el título que vamos a poner a la gráfica.
- *Tooltip*: se refiere a las herramientas que hemos usado en diseñar la gráfica.
- *PlotOptions*: es un objeto contenedor para los otros objetos de configuración para cada tipo de series.
- *Series*: Que contiene un array (tipo, nombre, dato) de los diferentes categorías

b) - el controlador “*modal_questions.php*” que nos genera el código de una cuadro modal para inserción y actualización.

Para eso Inicializamos primero la sesión *php*, y luego incluimos nuestro fichero de configuración que nos guarda los datos de acceso a la base de datos, así como la función común de clases.

Una vez tenemos el modelo configurado, definimos una serie de etiquetas div, la primera que contiene el botón y otra div para definir los campos del formulario (*Category* y *language*) donde incluimos los dos controladores que hacen referencia a estas dos áreas, para obtener las filas y las columnas de la tabla y cargarlas mediante Ajax.

Seguidamente, definimos otra etiqueta div para la dificultad de preguntas, y para terminar creamos un script para obtener los valores *from* y *to* de los idiomas.

Después de incluir todos los controladores, pasamos a desarrollar la clase del modelo que nos permite gestionar la tabla de preguntas, empezamos con declarar una clase que hereda de la clase base *modelo*, definiendo como accesibilidad privada todos los atributos de la clase, luego declaramos la función de tipo *Array* para establecer los atributos.

Necesitaremos otra función que nos ayuda a obtener los registros de la tabla y agregalos los valores relacionados (*idioma* y *Categoría*), así como las correcciones de cada fila concatenadas.

Y para terminar añadimos una serie de funciones, cada una con un funcionalidad propia, la primera para obtener la fila del identificador indicado como parámetro, la segunda para insertar un registro en la base de datos, y las dos últimas para actualizar y borrar el registro pasado como parámetro.

4.3.4. Administración de categorías “Categories”:

Este Apartado refiere al requisito funcional:

RF 4: Administración Categorías “Categories”

RF 4.1: Añadir categorías

RF 4.2: Editar categorías

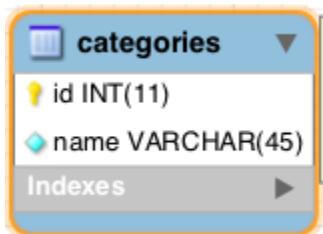
RF 4.3: Borrar categorías

RF4.4: Buscar categorías

Esta sección engloba todas las categorías de las diferentes preguntas de nuestro sistema, como acceso exclusivo para los administradores.

Para eso empezamos con la base de datos, creando la tabla *Categories* que tiene:

- Un identificador como clave primaria
- Una variable nombre de tipo char

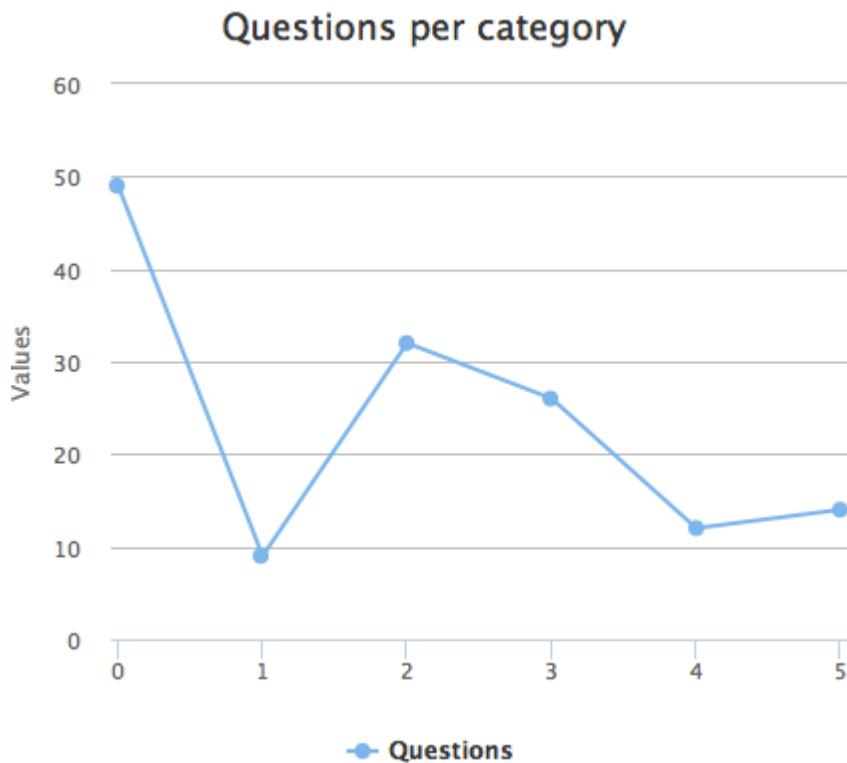


teniendo la tabla creada, pasamos a desarrollar la vista con

CSS y HTML: Primero creamos la tabla de las categorías de preguntas, declarando una etiqueta div, para la opción de añadir una nueva categoría, y otra etiqueta div, dónde metemos las etiquetas necesarias para diseñar la tabla con sus filas y columnas, y en cada fila metemos etiquetas para incluir los dos botones editar y eliminar, y así en cada una de las categorías.

PHP: Terminando esta parte hemos tenido que incluir un conjunto de controladores para la sección categorías primero:

a) - “get_chart_categories” que es el controlador de la gráfica categorías que representa las preguntas añadidas por cada usuario en cada categoría.



igual que la gráfica anterior se crea de la misma forma, las mismas variables, los mismos atributos.

Para configurar este controlador, primero hemos tenido que crear un objeto de la clase, obtenemos todas la preguntas del sistema por idioma, luego recorremos las filas para evitar comprobar si viene vacío. al final unimos todos los valores con un separador “,”.

seguidamente incluimos el controlador que igual en *questions* nos genera un cuadro modal para inserción/actualización.

b) - “modal_categories.php”: Después de inicializar las variables necesarias (*Nombre*, *titulo...*), incluimos nuestro controlador, que nos controla la inserción de registros.

Luego incluimos el fichero de configuración, y declaramos una etiqueta div de cabecera para meter un botón dentro.

Seguidamente, creamos nuestro formulario, y definimos una etiqueta div que contiene el campo *Name*, y otra para llevar el botón dentro.

Terminando la vista y la inclusión de controladores, pasamos a la parte modelo para la gestión de la tabla categorias.

Empezamos definiendo la clase “*categories*” que hereda de la clase base modelo, luego definimos las funciones que vamos a necesitar, una para establecer el nombre, otra para establecer los atributos de la clase, luego otra función para obtener los registros.

También definimos una función que nos facilita la inserción de un registro en la base de datos, y por último definimos las dos funciones, para actualizar y borrar el registro pasado como parámetro.

4.3.5. Gestión de idiomas “*Languages*”

Este Apartado refiere al requisito funcional:

RF 5: Gestión Idiomas “languages”

RF 5.1: Añadir idiomas

RF 5.2: Editar idiomas

RF 5.3: Borrar idiomas

RF 5.4: Buscar idiomas

Es la sección de página que incluye todas las idiomas que soporta el sistema, que solo los usuarios tendrán acceso, y podrán añadir modificar o borrar un idioma, para eso empezamos creando la tabla *corrections* en la base de datos, que tiene:

- un Identificador como clave primaria.
- Variables de tipo *char* como *code* y *name*



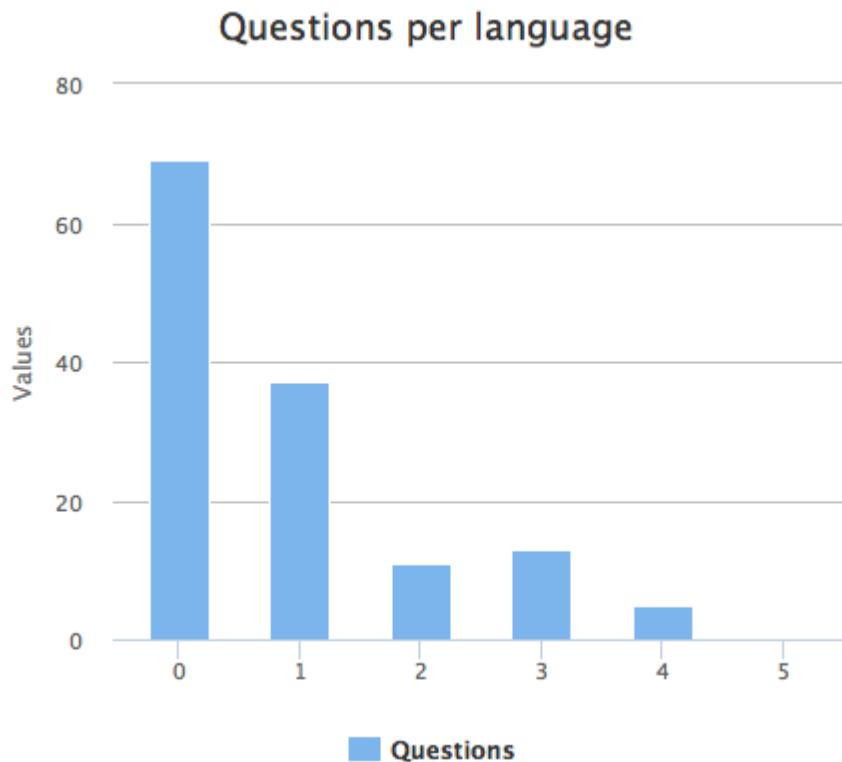
Una vez tenemos la tabla creada en la base de datos, necesitamos crear la vista con

HTML y CSS: Usando etiquetas de tablas, y otras para cada fila y columna, creamos nuestra tabla de idiomas, metiendo un identificador que nos ayuda a diferenciar cada fila, y por último incluimos los botones que hacen referencia a las opciones de editar y eliminar, y así igual en todas las idiomas.

Después de crear la vista de idiomas con *html* seguidamente pasamos a la parte

PHP: incluyendo el primer controlador:

a) - “*get_chart_languages.php*”, que hace referencia a la gráfica *languages*, esta última, representa el total de preguntas según las diferentes idiomas que tenemos en el sistema.



Como las gráficas anteriores, creamos las variables necesarias, y a la hora de configurar lo, creamos un objeto de la clase, obtenemos las preguntas por idioma, luego recorremos las filas y unimos todos los valores usando “,” como separador.

Y como las secciones anteriores incluimos el controlador:

b) - “*modal_languages.php*”: que nos genera el cuadro modal de inserción y actualización.

Empezamos inicializando las variables necesarias (*Name*, *Code*...), Y incluyendo nuestro controlador “*insert.php*”, que nos controla la inserción de registros.

En seguida incluimos el fichero de configuración y definimos una serie de etiquetas div, la primera para el formulario de la ventana, otra para contener el botón y el título del modal, y por último las dos etiquetas al final, para generar cada uno de los campos del modal (*Name*, *Code*), e incluir el botón de crear “*Create*”.

Una vez tenemos la vista creada, con los controladores incluidos, creamos el modelo “languages” para gestionar la tabla de idiomas.

como las tablas anteriores, definimos una clase que hereda de modelo, con los atributos (*Name*, *code*) y las dos funciones para establecer el nombre y el código.

Para terminar definimos una serie de funciones que nos gestiona la operaciones de esta sección, como la de actualizar y borrar los registros pasados como parámetro, y otra que obtienen las idiomas con una columna adicional indicando el número de preguntas por idioma, que lo utilizamos para las estadísticas.

4.3.6. Gestión de usuarios “Users”:

Este Apartado refiere al requisito funcional:

RF 6: Gestión de usuarios “Users”

RF 6.1 Añadir usuarios

RF 6.2 Editar usuarios

RF 6.3 Borrar usuarios

RF 6.4 Buscar usuarios

Es la parte de página que engloba todos los usuarios registrados del sistema, que tendrá acceso exclusivo para los administradores, que pueden añadir, buscar, borrar o editar usuarios. en cada operación se generará una salida propia, como en el caso de añadir un nuevo usuario, se genera un formulario para introducir todos los datos necesarios, ese mismo formulario se usará a la hora de editar datos de usuarios, y para la opción de borrar, se genera siempre antes eliminar un usuario un aviso pidiendo la confirmación.

Para conseguir eso empezamos creando en la base de datos la tabla *users* que hace referencia a los usuarios del sistema, y tiene:

- un identificador como clave primaria, que no puede ser nulo.

- otros atributos (*name*, *password*, *email...etc*) como variables de tipo Char
- por último ponemos el motor de almacenamiento *InnoDB*

users	
!	id INT(11)
◆	name VARCHAR(100)
◆	email VARCHAR(45)
◆	password VARCHAR(45)
◆	role TINYINT(1)
◆	id_language INT(11)
◆	date TIMESTAMP
Indexes	

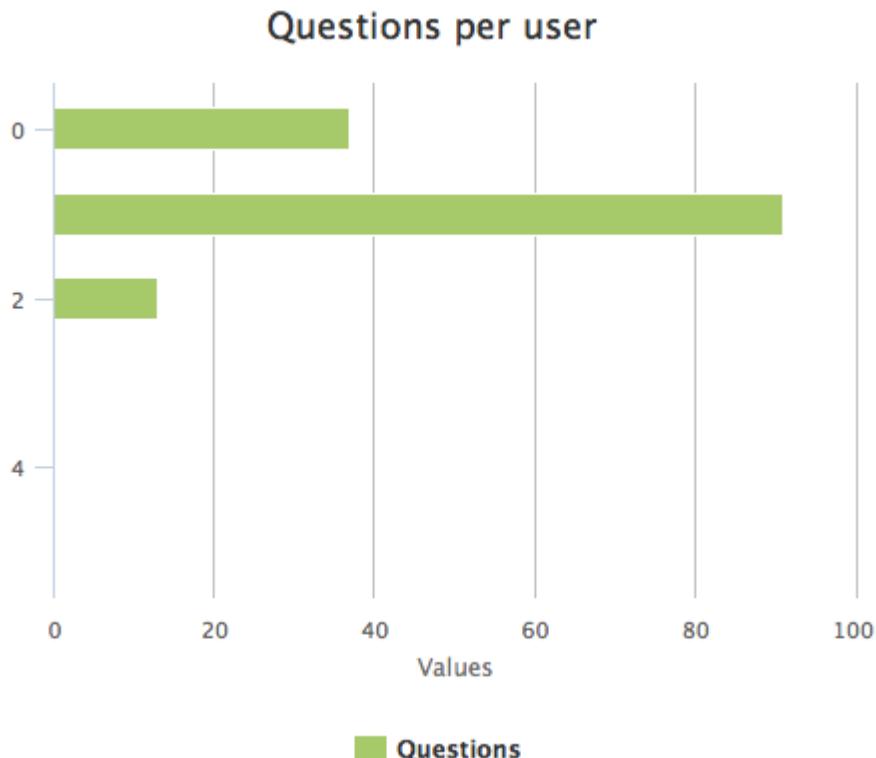
Una vez tenemos la tabla creada, pasamos a implementar la parte de

HTML y CSS: Para empezar definimos una etiqueta div, que incluye la opción de añadir un nuevo usuario, y otra etiqueta para crear la tabla de usuarios agrupando todos los elementos, y usando etiquetas especiales para cada fila y columna. dentro de cada fila definimos los dos botones que hacen referencia a las opciones de eliminar o editar, menos en la fila del administrador, que solo tendrá la opción de editar.

Para gestionar esta tabla con las opciones que lleva, pasamos a la parte

PHP: que como la secciones interiores incluimos primero los controladores:

a) “*get_chart_users.php*”, que es el responsable de controlar la gráfica *Users* que representa el total de preguntas añadidas por cada usuario del sistema, en cada una de las categorías.



Empezamos por definir una variable *chart* (para el área de gráfico), y para construir la gráfica definimos una serie de atributos:

- *chart*: que contiene el color del fondo, y el tamaño del borde.
- *title*: contiene el título que vamos a poner a la gráfica
- *tooltip*: se refiere a las herramientas que hemos usado en diseñar la gráfica.
- *plotOptions*: es un objeto contenedor para los otros objetos de configuración para cada tipo de series.
- *series*: Que contiene un array (tipo, nombre, dato) de los diferentes usuarios (en nuestro caso, son: *Admin*, *Ernesto*, *Nourdine*)

b) modal_users.php: este controlador para generar un cuadro modal, para la inserción y la actualización.

Primero inicializamos la sesión y las variables que vamos a usar (*name*, *email*, *role*), y agregamos nuestro controlador *insert* que controla la inserción de registros.

Luego pasamos a crear el modal, poniendo una etiqueta `div` para cada uno de los campos del formulario generado (`name`, `email`, `password`, `Re-type password`). Por último metemos un condicional, para que solo los administradores pueden cambiar de rol.

Terminada la parte del modal, creamos el modelo “`Users`”, definiendo una clase que nos gestiona la tabla de usuarios y que hereda de la clase base `model`, luego declaramos las variables que vamos a necesitar (`name`, `email`, `password`, `role`). También hemos definido una serie de funciones, para establecer los atributos, obtener todos los usuarios de la base de datos, y otra para obtener la fila del usuario indicado como parámetro.

Como las tablas anteriores definimos las dos funciones que nos asegura la actualización y eliminación de registros pasados como parámetro.

Y para terminar creamos la función especial que chekea `email` y `password` para comprobar si el usuario existe en la base de datos o no.

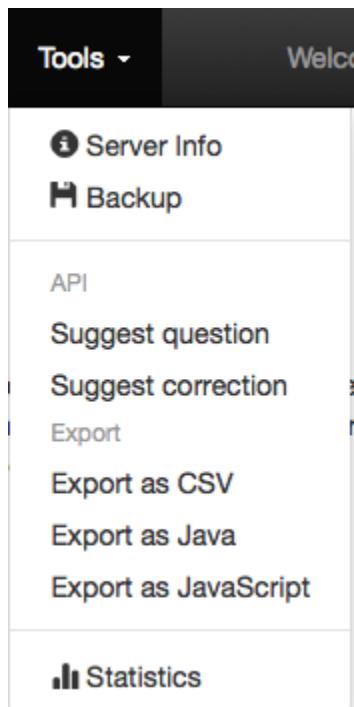
4.3.7. Gestión de Herramientas “Tools”

Este Apartado refiere al requisito funcional:

RF 7: Gestión Herramientas “Tools”

La sección, que contiene todas las herramientas de nuestro sistema, y que serán servidas según el rol que tenga el usuario: si es

- Administrador: tendrá acceso a todas las herramientas del sistema.
- Usuario Registrado: solo puede usar la API para sugerir preguntas y correcciones, y acceder a la parte `statistics`, que contiene las estadísticas de cada sección.



4.3.7.1 Gestión de la API

Este Apartado refiere al requisito funcional:

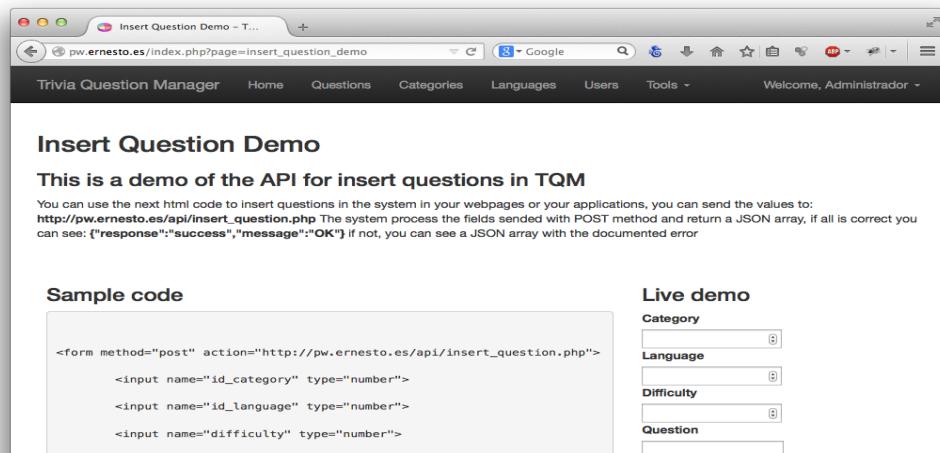
RF 7.1: Gestión API de Sugerencia.

En esta sección, que está incluida en *tools*, los usuarios pueden usar esta herramienta para sugerir preguntas de cualquier categoría e idioma, además pueden sugerir correcciones, que se verán reflejado mediante unos triángulos informativos en la lista de preguntas, a partir de una Interfaz de programación de aplicaciones.

4.3.7.1.1 Sugerir Preguntas “Suggest Question”:

Este Apartado refiere al requisito funcional:

RF 7.1.1: Sugerir Preguntas “Suggest Question”



Se lanza una petición a la página web a la que se le pasan los valores: categoría, idioma, dificultad, pregunta, etc..., mediante *POST*, el sistema procesará los valores y si son correctos los insertará en la tabla *questions* de la base de datos creando un nuevo registro para ello. Si diera un error se notificará al usuario en la respuesta y si todo fuera correcto mostraría un mensaje indicando que la sugerencia se ha realizado con éxito.

Para eso empezamos estableciendo el nombre de la entidad para poder instanciar su clase, después establecemos el identificador del usuario a 0, ya que las sugerencias son accesible a todos los usuarios registrados, y para terminar llamamos al fichero de *insert*, que se encarga de insertar las preguntas en la base de datos.

4.3.7.1.2. Sugerir Correcciones “Suggest Correction”:

Este Apartado refiere al requisito funcional:

RF 7.1.2: Sugerir Corrección “Suggest Correction”

The screenshot shows a web browser window with the title "Insert Correction Demo". The URL in the address bar is "pw.ernesto.es/index.php?page=insert_correction_demo". The page content includes a heading "Insert Correction Demo" and a sub-section "This is a demo of the API for insert corrections in TQM". It provides instructions on how to use the API via a POST request to "http://pw.ernesto.es/api/insert_correction.php". Below this, there are two sections: "Sample code" containing an HTML form snippet, and "Live demo" which shows a form with fields for "Question" and "Comments", and a "Enviar consulta" button.

Siguiendo la misma forma de cómo se insertan las preguntas anteriormente, lanzamos una petición a la página web, a la que se le pasan los valores, que son esta vez: preguntas (refiriendo al número de la pregunta en la tabla), y comentario que contendrá el texto del error que queremos informar, mediante *POST*, una vez que se verifica que los valores introducidos son correctos, el sistema se ocupará de insertar esta corrección en la tabla *corrections* de la base de datos, creando un nuevo registro.

Después de insertar cada corrección se genera un error para notificará al usuario, si todo esta en condiciones, se le informará que la sugerencia se ha realizado correctamente.

Igual que para *questions* establecimos, el nombre de la entidad para poder instanciar su clase, con el identificador del usuario a 0, para dar acceso a todos los usuarios de la página sin excepción, y terminamos llamando al fichero de *insert* que se encargará de insertar esta corrección en la tabla *corrections* de la base de datos.

4.3.7.2. Copias de seguridad “Backup”

Este Apartado refiere al requisito funcional:

RF 7.2: Gestión de copias de seguridad.

The screenshot shows a web browser window for the 'Trivia Question Manager' backup page. The URL is pw.ernesto.es/index.php?page=backup. The page title is 'Backup'. It displays a table of existing database backups:

Backup file	Date	Actions
tqm_backup_20140519.sql.gz	May 19 2014 16:47	restore delete
tqm_backup_20140520.sql.gz	May 20 2014 08:59	restore delete

A blue button labeled 'Make Backup Now!' is visible above the table. A yellow warning box contains the text: 'Warning! Be careful with this options, you can break the system or lose questions if restore the wrong backup!!'. At the bottom of the page, there is a footer note: '© Trivia Question Manager 2014 - [Credits](#) - Page generated in 0.0004 seconds'.

Es la sección de página, donde podemos generar un copia de seguridad de la base de datos, como garantía de recuperación de datos, fácil y rápida, para los administradores de forma exclusiva.

Una vez accedemos a esta pagina tendremos varias opciones, como restaurar una copia que ya la teníamos guardada con otras fechas, o generar un nueva, con la fecha actual.

En cada copia generada, se informará al usuario mediante un aviso que la operación se ha realizado correctamente, y igual en el caso de restaurar otra copia.

Para eso empezamos estableciendo un nombre que tendrá nuestro archivo, en base a la fecha actual y luego establecemos las opciones de una clase que genera un *dump* de la base de datos y la guarda en un archivo con formato *.sql* con un conjunto de objetos *Array*, y por último Instanciamos esta misma clase y llamamos al método que comienza a generar el archivo de texto que contiene la serie de sentencias *SQL* que permiten crear la estructura de nuestra base de datos y cargarla.

4.3.7.3. Gestión de la herramienta Exportar “Export”

Este Apartado refiere al requisito funcional:

RF 7.3: Gestión de la herramienta Exportar

Es la sección de la página donde se puede exportar las preguntas de todo el sistema en diferentes formatos (*CSV*, *JAVA*, *JAVASCRIPT*) de forma exclusiva para los usuarios con privilegios de administración:

Para eso empezamos declarando una clase base desde la que se generan los distintos ficheros de exportación, donde declaramos las dos variables que hacen referencia al nombre del archivo y su extensión, estas variables tendrán visibilidad *protected*, para que no sean accesible desde cualquier lado, sólo de la misma clase o desde las clases que hereden de ella.

Luego declaramos las dos funciones la primera, de tipo de público, que nos genere el fichero, escribiendo las cabeceras del archivo generado, y obteniendo la lista de preguntas de la base datos, luego recoge todas las filas, y por último escribiendo cada línea y el pie del archivo antes de terminar la ejecución.

La segunda función la implementamos con el objetivo de desactivar la *cache*, y forzar la descarga del archivo, y por último establecimos el formato de codificación de caracteres unicode *UTF-8*, y el modo de transferencia.

Una vez tenemos la clase base hecha, pasamos a crear las clases de las diferentes extensiones:

4.3.7.3.1. Exportar JAVA

Este Apartado refiere al requisito funcional:

RF 7.3.1: Exportar en Formato JAVA

Con esta opción se genera un archivo específico en formato *.java* programado con *java* como salida a la hora de acceder a esta opción, y que incluye todas las preguntas de la base de datos.

Teniendo la clase base implementada solo nos queda implementar la clase que hereda, lo hacemos de la siguiente forma:

Declaramos una clase que hereda de la clase base *Export*, y definimos un constructor con los atributos del nombre de archivo y su extensión, después reemplazamos la función que nos escribe la cabecera y la función que escribe el cuerpo, declarando un conjunto de objetos *Array* con los valores, y lo unimos mediante una función que nos permite convertir el arreglo en cadena definiendo un delimitador, y por último reemplazamos la función que nos escribe el pie del archivo.

4.3.7.3.2. Exportar JS

Este Apartado refiere al requisito funcional:

RF 7.3.2: Exportar en Formato JavaScript

Con esta opción se genera un archivo específico en formato *.js* programado con *JavaScript* como salida a la hora de acceder a esta opción, y que incluye todas las preguntas de la base de datos.

Como la extensión anterior declaramos una clase que hereda de la clase base *Export*, definiendo un constructor con los dos atributos (nombre del fichero y su extensión), luego reemplazamos la función que escribe la cabecera y la función que nos escribe el cuerpo declarando un *array* con los valores (identificador de idiomas, categorías, y la dificultad), y los unimos mediante la función para el arreglo en cadena y por último reemplazamos la función que genera el pie del archivo.

4.3.7.3. Exportar CSV

Este Apartado refiere al requisito funcional:

RF 7.3.3: Exportar en Formato CSV

Con esta opción se genera la extensión del archivo CSV (valores separados por comas), como salida a la hora de acceder a esta opción, y que incluye todas las preguntas de la base de datos.

Siguiendo la misma manera con las extensiones anteriores, declarando una clase que hereda de la clase base *Export*, y empezamos definiendo un constructor con los atributos del nombre del archivo y la extensión que tiene, luego reemplazamos la función que nos escriba el cuerpo, declarando un *array* con el (identificador, categoría, preguntas...), y lo unimos mediante una función que convierte el *array* que hemos declarado anteriormente en una cadena con el separador (;).

4.3.7.4. Página de soporte “Contact Support”

Este Apartado refiere al requisito funcional:

RF 7.4: Administrar Contact Support

The screenshot shows a web browser window with the title 'Support - Trivia Question ...' and the URL 'pw.ernesto.es/index.php?page=support'. The page is part of the 'Trivia Question Manager' system, with a navigation bar at the top including links for Home, Questions, Categories, Languages, Users, Tools, and a welcome message for 'Administrador'. The main content area is titled 'Contact us' and contains a form with two input fields: 'Subject' and 'Your message'. The 'Subject' field has a placeholder 'Enter subject' and the 'Your message' field has a placeholder 'Please enter your message here...'. A blue 'Submit' button is located at the bottom right of the form. At the bottom of the page, there is a footer with copyright information: '© Trivia Question Manager 2014 - Credits - Page generated in 0.0143 seconds'.

Se lanza una petición a la página al que le pasamos mediante un formulario los dos campos, el primero para introducir el asunto del correo, y otro para expresar el contenido de este último, que puede ser un informe de una incidencia, una queja o sugerencia, una vez que los campos se han rellenado, y se manda el correo, primero se generará un mensaje de agradecimiento, informando a los usuarios que el mensaje se ha enviado correctamente, y que tendrá una respuesta lo antes posible, después el sistema se ocupará de mandar este mensaje al administrador de la página, mediante un correo.

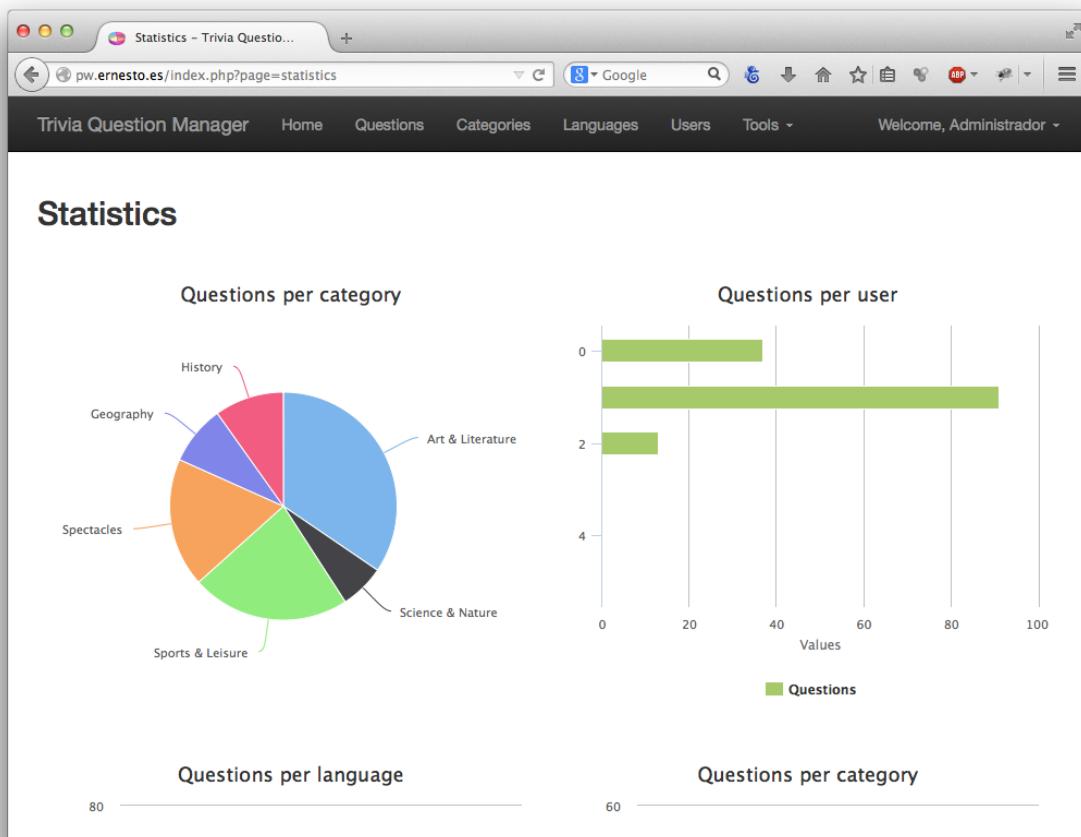
Para eso empezamos creando un controlador, que encarga de enviar el email al administrador, y mostrar el mensaje de gracias, comprobando primero que hayamos enviado el formulario, y comprueba los datos (email, el asunto, el texto del mensaje), después establece las cabeceras del email, y termina enviandolo.

Luego creamos los campos de los formularios (Asunto, el cuerpo del mensaje) poniendo una etiqueta div para cada campo y otra que contiene nuestro botón de enviar.

4.3.7.5. Gestión Estadísticas

Este Apartado refiere al requisito funcional:

RF 7.5: Gestión Estadísticas.



La sección que engloba todas las gráficas de nuestra página, para facilitar al usuario sea administrador o registrado, las estadísticas de las diferentes secciones de nuestra página.

Para ese nos basta con crear una vista que contiene una serie de etiquetas div, cada una incluye los controladores que hacen referencia a la gráfica de cada sección.

4.3.8.1. Cantidad de usuarios concurrentes.

Este Apartado refiere al requisito funcional:

RF 8.1: Cantidad de usuarios concurrentes.

Este requisito se resuelve configurando apache y mysql para que tenga el máximo rendimiento, aunque el número de usuarios concurrentes dependerá en gran medida de la capacidad hardware del servidor.

4.3.8.2. Cantidad de información almacenada

Este Apartado refiere al requisito funcional:

RF 8.2: Cantidad de información almacenada.

La cantidad de información almacenada depende de la cantidad de preguntas que tenga la base de datos, pero al guardarse como texto y no contener imágenes ni nada pesado cualquier servidor medio superará con creces la cantidad de información que se pueda generar en un periodo medio-largo.

4.3.8.3. Sistema Operativo

Este Apartado refiere al requisito funcional:

RF 8.3: Sistema Operativo.

Al ser una página web (html) basada en los estándares del World Wide Web Consortium (W3C) y haber utilizado un diseño “responsivo” que se adapta a cualquier pantalla el sistema es accesible desde prácticamente cualquier sistema operativo que posea un navegador web.

4.3.8.4. Gestión de errores HTTP

Este Apartado refiere al requisito funcional:

RF 8.4: Gestión de errores HTTP.

Si algún usuario acceder a una web inexistente, o acceder a un recurso restringido el sistema mostrará una página html personalizada, para activar esto se debe activar la ruta al archivo 404.html desde el archivo .htaccess de configuración de apache, además se incluye una protección adicional por si se intenta acceder a una vista no existente, entonces se carga la vista 404.phtml con la cabecera y pié general de la pagina.

5. Conclusiones

5.1 Problemas encontrados

A la hora de desarrollar el sistema, hemos encontrado con varios problemas, entre ellos:

Primeramente, en el diseño de la base de datos, nos encontramos con el problema de las tablas, ya que no sabíamos, el conjunto necesario de tablas y atributos (las claves primarias, los tipos de variables, el motor de almacenamiento,) que metemos en cada una, luego hemos tenido que importar la base datos y conectarla desde php (Localhost, conexión con el servidor ...).

Seguidamente, nos abordó problemas en la parte de css, html, a la hora de estructurar la página (body, pie, tabulaciones, ...) y con los div de cada parte, luego hemos tenido unas pequeñas dificultades en crear las tablas de cuestiones (ordenar las columnas de forma horizontal), y separarlas del filtro de preguntas que hemos puesto.

Y por último hemos tenido diversas problemas, con las expresiones regulares, y el uso de php en la inserción de registros y la administración de usuarios.

5.2. Soluciones y conclusiones

Tras la realización de este proyecto, hemos llegado a las siguientes conclusiones:

- ⦿ Que el bootstrap además de ser un Framework que trae varios elementos con estilos predefinidos fáciles de configurar (Botones, Menús desplegables, Formularios incluyendo todos sus elementos e integración jQuery para ofrecer ventanas y tooltips dinámicos), te hace la vida más sencilla al proveer muchas de las reglas CSS que necesitas para hacer que tu sitio se adapte automáticamente a la gran mayoría de pantallas y resoluciones existentes en el mercado, y eso para nosotros fue de gran ayuda.
- ⦿ jQuery es muy útil a la hora de agregar efectos y funcionalidades complejas a nuestro sitio web, como por ejemplo: galerías de fotos dinámicas y elegantes, validación de formularios, hacer aparecer y desaparecer elementos en nuestra página y muchas otras poderosas opciones.

Otra ventaja que nos benefició, sin duda es la posibilidad que nos brinda de trabajar con AJAX, sin preocuparnos de los detalles complejos de la programación.

En otras palabras, podremos lograr los mismos resultados, en mucho menos tiempo y con un diseño profesional y responsive.

➔ También hemos podido comprobar que haciendo uso de una combinación entre html y css se pueden llegar a conseguir páginas muy vistosas y dinámicas ya que aunque es bastante engorroso de programar css da una gran cantidad de posibilidades para dar estilos y formatos.

➔ Hemos comprobado la potencia que tiene Javascript, a la hora de la comprobación de datos, hemos visto también que con javascript, se puede dar funcionalidad a las páginas que por sí solas no tendrían. recoger datos desde un input y tratarlos, dar formato a datos, abrir ventanas nuevas y enviar formularios.

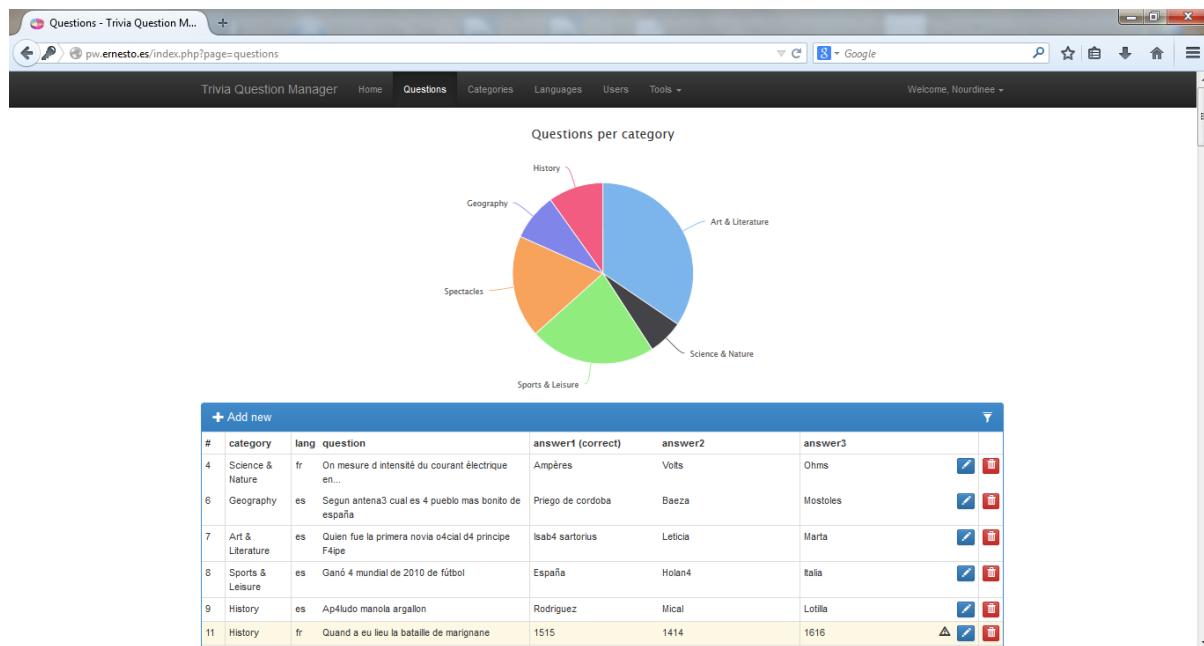
Es una parte que toma bastante tiempo en lo que se refiere a las expresiones regulares dado que son un mecanismo muy potente aunque complejo de entender.

➔ Hemos aprendido como mandar peticiones con php, obtener datos de una base de datos y borrar de ésta.

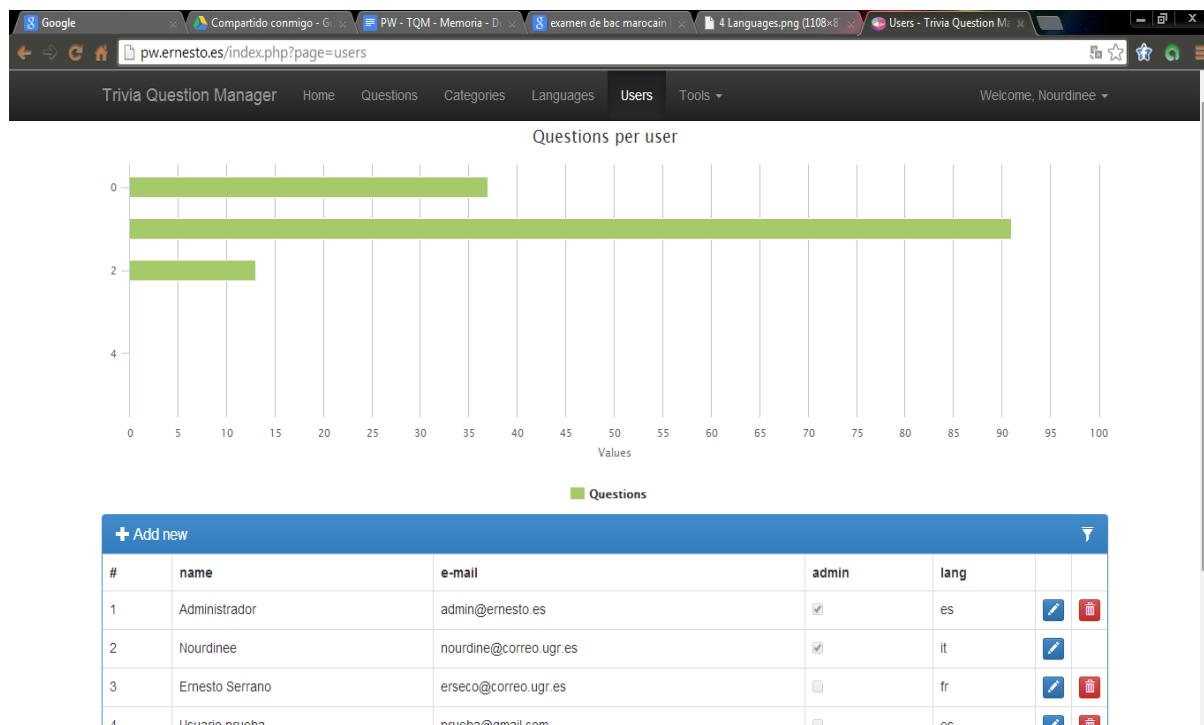
Hemos podido apreciar la verdadera potencia de los formularios, la cual se aprovecha en su totalidad con php, que es un lenguaje simple y pueda trabajar en conjunto a otros componentes.

➔ Y por último, hemos comprobado que la página, se ve perfectamente en todo los navegadores y dispositivos, por ejemplo:

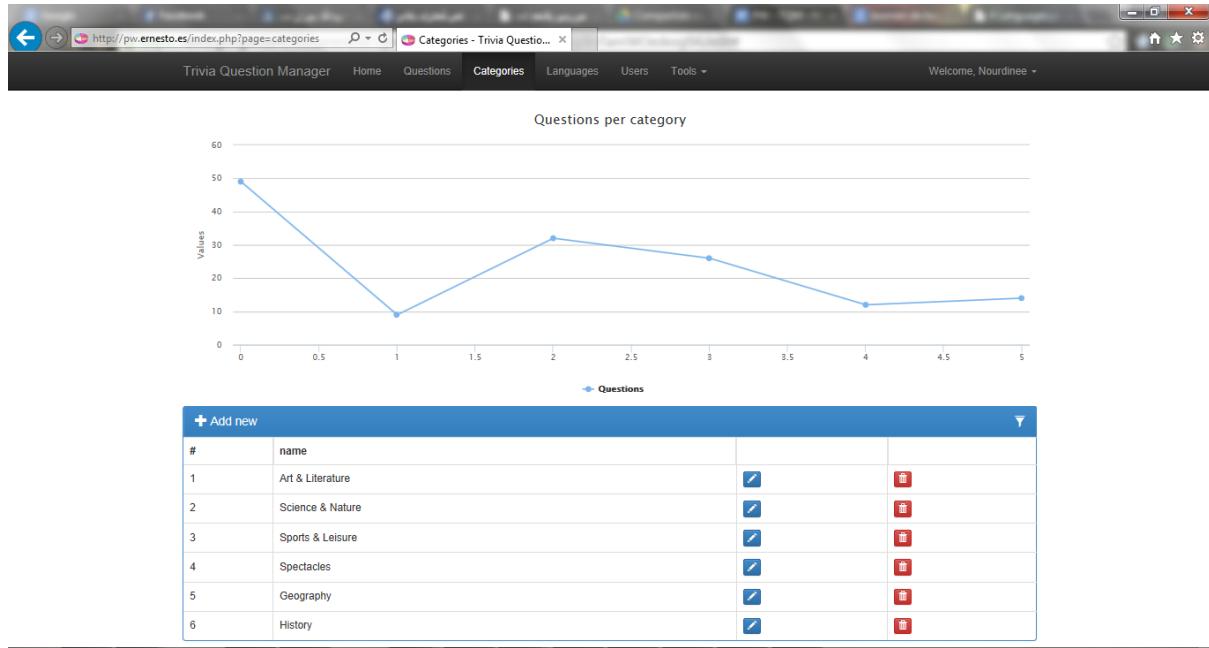
➡ Mozilla Firefox:



➡ Google Chrome:



● Internet Explorer:



● En dispositivos móviles:

The comparison shows the mobile application interface for both Android and iPhone. Both versions have a header bar with the URL "pw.ernesto.es/index.php?page=home", signal strength, battery level (82% for Android, 11% for iPhone), and time (13:42 for Android, 00:00 for iPhone). The main content area displays the "Trivia Question Manager" logo and a welcome message: "Congratulations, if you can read this is because you are a translator of the many Atomik Soft trivia games, please, ride in the categories to start your work. If you need more info, please click on the Learn more button, thanks!". A "Learn more »" button is visible at the bottom. Below this, there is a table titled "+ Add new" with columns for "#" and "name". The table lists the same six categories as the desktop version, each with edit and delete icons. The Android version has a navigation menu icon on the right, while the iPhone version has a user status indicator ("Ernesto", "Nourdin", "Admin") and a navigation menu icon.

#	name		
1	Art & Literature		
2	Science & Nature		
3	Sports & Leisure		
4	Spectacles		
5	Geography		
6	History		

6. Bibliografía

6.1. Páginas web

- <http://www.w3schools.com/>
- <https://community.oracle.com/community/developer/english>
- <http://es.html.net/>
- <http://forums.phpfreaks.com/>
- <http://board.phpbuilder.com/forum.php?s=ba89cabbe41ef3e8e70deb2eec83b925>
- <http://forums.devshed.com/php-development-5/>
- <http://forums.mysql.com/>
- <http://stackoverflow.com/>
- <http://www.codingforums.com/>
- <http://www.librosweb.es>
- <http://www.cssblog.es>
- <http://www.comocreartuweb.com>
- <http://www.htmldog.com/>

6.2. Libros

- David Sawyer Mcfarland – CSS
- Eric A. Meyer - CSS: The Definitive Guide 3rd edition.
- Jeffrey Zeldman - Diseño Con Estándares Web.
- PHP 5.5 Desarrollar un sitio Web dinámico e interactivo.
- Sklar, David - Introducción a PHP5
- Luke Welling y Laura Thomson Desarrollo Web con PHP5 y Mysql 4.1 y 5