

Agradecimientos:

Gracias a mis compañeros de Trabajo por los consejos.

A Josep M^a, por todas las ayudas y consejos a lo largo de la carrera.

*A los dos equipos directivos de los campeonatos 9SotaPar y SotaPar,
por las facilidades dadas para poder realizar la web.*

Resum

L'objectiu d'aquest projecte és crear una aplicació web per realitzar la gestió de diferents campionats pels esports de golf y Pitch and Putt, si bé el client desenvolupat està principalment enfocat al P&P. La principal idea per el que va sorgir el projecte va ser la de poder tindre de forma més ràpida les classificacions i els resultats de les diferents proves.

Per el desenvolupament de la web, s'han utilitzat HTML, PHP i JavaScript; i SQL per a les consultes a la base de dades MySQL

Resumen

El objetivo de este proyecto es crear una aplicación web para la gestión de diferentes campeonatos para los deportes de golf y pitch and putt, principalmente dedicada al P&P. La principal idea por la que surgió el proyecto fue la de poder tener de una forma más rápida las clasificaciones y los resultados de las diferentes pruebas.

Para el desarrollo de la web, se ha utilizado HTML, PHP y JavaScript; y SQL para las consultas a la base de datos MySQL.

Abstract

The objective of this project is to create a web application for managing different championships for golf and pitch and putt, although the client is mainly devoted to P&P. The main idea for the project emerged was to have a quicker rankings and results of the different games.

For the web development, it has been used HTML, PHP and JavaScript; and SQL for the management of the MySQL database.

Índice

Índice de figuras.....	III
Índice de tablas.....	VII
Glosario de términos	IX
1. Objetivos	1
1.1. Propósito	1
1.2. Finalidad	1
1.3. Objetivo	1
1.4. Alcance	1
2. Introducción	3
2.1. Descripción del servicio.....	3
2.2. Situación inicial	3
2.3. Situación actual.....	4
2.4. Trabajo a realizar	6
2.5. Planificación	6
3. Propuesta formal	7
3.1. Resumen de las tablas	9
4. Diseño de la aplicación	11
4.1. La base de datos.....	11
4.1.1. Camp.....	11
4.1.2. Circuit	12
4.1.3. Classif.....	12
4.1.4. Convocatoria.....	14
4.1.5. Distancia	14
4.1.6. Equip.....	15
4.1.7. Jugadors	15
4.1.8. Modalitat.....	16
4.1.9. Prova.....	16
4.1.10. Resultats.....	17
4.1.11. Users	18
4.2. Diseño de la aplicación	19

4.2.1. Zona pública.....	19
4.2.1.1. Clasificación.....	23
4.2.1.2. Resultados	29
4.2.1.3. Calendario	32
4.2.1.4. Campos.....	33
4.2.1.5. Idioma	34
4.2.1.6. Login.....	35
4.2.2. Zona administración.....	36
4.2.2.1. Circuito.....	38
4.2.2.2. Prueba.....	43
4.2.2.3. Campo	44
4.2.2.4. Equipo	45
4.2.2.5. Jugador	46
4.2.2.6. Modalidad	47
4.2.2.7. Resultado.....	48
5. Valoración Económica	57
5.1. Costes directos	57
5.2. Amortización Software.....	57
5.3. Amortización Hardware.....	58
5.4. Resumen	58
6. Posibles mejoras.....	59
7. Conclusiones	61
8. Bibliografía y ayuda durante el proyecto	63

Índice de figuras

Fig. 3.1. Mapa conceptual de la base de datos.....	8
Fig. 4.1. Portada aplicación.....	19
Fig. 4.2. HTML menú desplegable.....	20
Fig. 4.3 CSS menú desplegable.....	20
Fig. 4.4. Clasificación - Menú Clasificación.....	23
Fig. 4.5. Clasificación - Selección campeonato.....	23
Fig. 4.6. Clasificación - Scratch.....	24
Fig. 4.7. Clasificación - Hcp Positivo.....	24
Fig. 4.8. Clasificación - Código Ajax.....	25
Fig. 4.9. Clasificación - HTML principal Clasificación.....	25
Fig. 4.10. Clasificación - CSS principal Clasificación.....	28
Fig. 4.11. Resultados - Scratch.....	29
Fig. 4.12. Resultados - Código Ajax.....	29
Fig. 4.13. Calendario.....	32
Fig. 4.14. Calendario - Código HTML.....	32
Fig. 4.15. Campos.....	33
Fig. 4.16. Idioma.....	34
Fig. 4.17. Idioma – idiomas.php.....	34
Fig. 4.18. Login – Identificación.....	35

Fig. 4.19. Login – Error login.....	35
Fig. 4.20. Menú Administración.....	36
Fig. 4.21. Menú Administración.....	36
Fig. 4.22. Menú Administración – Verificación de la sesión.....	37
Fig. 4.23. Menú Administración – Codificación apartado.....	37
Fig. 4.24. Circuito – Nuevo Circuito 01.....	38
Fig. 4.25. Circuito – Nuevo Circuito 02.....	38
Fig. 4.26. Circuito – Función CrearCircuit.....	38
Fig. 4.27. Circuito – Nuevo Circuito 03.....	39
Fig. 4.28. Circuito – Buscar Circuito 01.....	40
Fig. 4.29. Circuito – Buscar Circuito 02.....	40
Fig. 4.30. Circuito – Buscar Circuito 03.....	41
Fig. 4.31. Circuito – Buscar Circuito 04.....	41
Fig. 4.32. Circuito – Consultar Circuito.....	42
Fig. 4.33. Circuito – Editar Circuito 01.....	42
Fig. 4.34. Circuito – Editar Circuito 02.....	42
Fig. 4.35. Prueba – Crear Prueba.....	43
Fig. 4.36. Campo – Crear Campo.....	44
Fig. 4.37. Campo – Buscar Campo.....	44
Fig. 4.38. Equipo – Crear Equipo.....	45
Fig. 4.39. Equipo – Modificar Equipo.....	45

Fig. 4.40. Equipo – Fichar Jugador	45
Fig. 4.41. Jugador – Dar de alta nuevo Jugador	46
Fig. 4.42. Jugador – Modificar Jugador	46
Fig. 4.43. Modalidad – Crear nueva Modalidad de juego.....	47
Fig. 4.44. Introducir Resultado 01	48
Fig. 4.45. Introducir Resultado 02.....	48
Fig. 4.46. Introducir Resultado 03 – Archivo correcto	49
Fig. 4.47. Introducir Resultado 04 – Archivo erróneo	49
Fig. 4.48. Cargar Resultado	50
Fig. 4.49. Cargar Resultado – Carga sin registros.....	51
Fig. 4.50. Cargar Resultado – Carga con registros 01.....	51
Fig. 4.51. Cargar Resultado – Carga con registros 02.....	52
Fig. 4.52. Cargar Resultado – Librería Spreadsheet.....	53
Fig. 4.53. Cargar Resultado – Código PHP	53

Índice de tablas

Tabla 2.1. Fases del proyecto, primera iteración.....	6
Tabla 5.1. Costes directos.....	58
Tabla 5.2. Amortización de Software.....	58
Tabla 5.3. Amortización de Hardware.....	58
Tabla 5.4. Resumen datos.....	58

Glosario de términos

AJAX	JavaScript asíncrono i XML.
BBDD o BD	Abreviatura de base de datos.
CSS	Hoja de estilos.
HTML	Lenguaje de marcas de hipertexto.
JavaScript	Lenguaje de programación interpretado por el cliente (navegador).
MAMP	Sistema que utiliza Macintosh, Apache, MySQL i PHP.
PHP	Lenguaje de programación interpretado por el servidor.
SQL	Lenguaje declarativo de acceso a bases de datos relacionadas
XML	Lenguaje de marcas extensible.

Catalanes	Medidas mínimas y máximas de la longitud de los hoyos (40-120m)
EPPA	Medidas mínimas y máximas de la longitud de los hoyos (35-90m)
Fourball	Modalidad de juego por parejas en el que se juegan las 4 bolas.
Foursome	Modalidad de juego por parejas en la que se sale con 4 bolas y después se decide con que bola jugar y se sigue alternando los golpes los jugadores de la pareja.
Greensome	Modalidad de juego por parejas en la que se decide que jugador inicia los hoyos impares/pares y se juega alternando los golpes.
HCP	Hándicap.
HPE	Hándicap Exacto.
Individual	Modalidad de juego en el que juegas en solitario.
Medal	Modalidad de juego por golpes.
Positivo	Jugador cuyo hándicap es inferior a 0.
SCH	Scratch: Modalidad de juego, en la que se juega sin hándicap.
Stableford	Modalidad de juego por puntos.

1. Objetivos

1.1. Propósito

Este proyecto describe una aplicación web desarrollada en PHP, interactuando con una base de datos de tipo MySQL alojada en un servidor remoto y perfeccionando la utilización de los dos lenguajes aprendidos durante la formación académica.

1.2. Finalidad

El propósito de este proyecto es ampliar los conocimientos sobre la creación de páginas web, su mantenimiento y la gestión de datos con un volumen pequeño y muy estables.

1.3. Objetivo

El objetivo de este trabajo es generar una aplicación web para gestionar los campeonatos de golf, los diferentes circuitos y la publicación de los resultados (para la revista 9SotaPar), esta aplicación estará realizada con lenguaje PHP sobre una base de datos relacionales en MySQL. Esta web ha de ser accedida desde una url publica.

1.4. Alcance

La aplicación web ha de poder ser accesible desde cualquier parte del mundo mediante una url pública de un servidor de almacenamiento en la red. Esta url pública permitirá el acceso a una zona privada de la aplicación, con acceso restringido mediante una clave de acceso

2. Introducción

En este apartado se explican los objetivos que se quiere obtener en la web, su situación actual al inicio del proyecto, una descripción general del servicio actual y el trabajo realizado para conseguir los actuales servicios.

2.1. Descripción del servicio

Se quiere dar solución al problema de publicación de resultados y tratamiento de las diferentes clasificaciones y agilizar el proceso de publicación de los datos. Actualmente todo esto se hace de forma manual lo que provoca que hayan largas esperas entre la finalización de las pruebas y la publicación de los resultados, especialmente en las clasificaciones ya que son datos acumulados que hay que ir recalculando.

2.2. Situación inicial

La publicación de resultados al inicio del proyecto se hacia de forma manual, extrayendo los datos que interesaban de los ficheros Excel que entregaban los diferentes campos al finalizar las pruebas y creando un nuevo Excel del que se harían capturas de imagen para subir las clasificaciones a la web. También se había intentado realizar la publicación mediante un servicio de terceros pero los constantes cambios de estructura, publicación, privilegios de lectura de los resultados y pérdidas de datos hicieron que se descartara esta opción.

No había ningún tipo de gestión de los jugadores por lo que los jugadores que a medio campeonato cambian de categoría y empiezan a tener puntuaciones en las dos categorías en las que han jugado generaban reclamaciones y había que rehacer el trabajo sobre las clasificaciones anteriormente publicadas.

Los jugadores también se quejaban de que no podían ver los resultados hasta pasados dos o tres días de la finalización de la prueba, ya que el trabajo se hacía manualmente y requería dedicar un tiempo que la organización no dispone al no ser su trabajo principal.

2.3. Situación actual

Con la nueva página web, se pretende mejorar el servicio de tratamiento de resultados de los campeonatos, mejorando los tiempos de publicación y eliminando las cargas manuales de los resultados y el tratamiento de los ficheros Excel, dejando una mínima tarea de preparación para que el servicio de carga haga todo el trabajo de back-end.

En la nueva web se han dispuesto dos apartados para publicar los resultados. En, el primero de ellos, se podrán encontrar las diferentes clasificaciones generales de los campeonatos, y estará dividido en las 5 principales categorías (scratch, hándicap positivo, hándicap medio y hándicap alto). En el segundo apartado se podrán consultar los resultados individuales de cada prueba en las categorías scratch y hándicap.

La publicación de los resultados se hará de forma dinámica e inmediata a partir de los datos que contenga la base de datos alimentada por los diferentes Excels de los campeonatos.

La aplicación web estará dividida en dos partes diferenciadas. En la zona pública, de acceso para cualquier persona, se publicaran los resultados y estará el calendario del torneo en curso así como el listado de campos con sus direcciones, teléfonos, ... y la opción de cambiar de idioma (actualmente entre catalán y castellano). La segunda parte será la zona privada o de administración, donde los usuarios con clave de administrador podrán realizar toda la gestión de los campeonatos.

En la zona de administración, accesible a través de la última opción del menú público, se encontrarán las opciones para crear y modificar circuitos y añadir campeonatos/pruebas a estos circuitos. También se podrá gestionar la información de los campos, jugadores, equipos y modalidades de juego y se podrán tratar los usuarios, pudiendo crear de nuevos, modificar los existentes o eliminarlos. El apartado de resultados permitirá importar datos de los ficheros Excel al sistema y realizar su tratamiento.

2.4. Trabajo a realizar

- Se ha de crear una cuenta en un servidor de hosting que permita el alojamiento de la web, la base de datos y disponga de una aplicación para poder administrar la base de datos, así como el contenido que se irá subiendo al servidor o que se pueda tener durante la vida de la web.
- Recogida de requerimientos que ha de tener la nueva aplicación de la organización de los campeonatos.
- Recogida de mejoras o sugerencias de los jugadores.
- Crear un procedimiento de carga y publicación de los resultados, extrayendo los datos del archivo Excel.
- Crear una base de datos de los campos activos actualmente en Catalunya para poder usarlos como base para la gestión de los campeonatos.
- Crear un procedimiento de creación de circuitos y publicación de los campeonatos para poder hacer un seguimiento del circuito.

2.5. Planificación

A continuación se muestra una tabla con las tareas generales a realizar para la creación de la página web, objeto del proyecto, juntamente con los tiempos estimados de duración de cada una de las fases.

Tareas	Duración	Fecha Inicio	Fecha Fin
Planificación y selección de herramientas	1 semana	24/03/2014	31/04/2014
Análisis técnico	2 semanas	01/04/2014	14/04/2014
Diseño web	2 semanas	15/04/2014	30/04/2014
Implementación casos de uso	5 semanas	01/05/2014	31/05/2014
Revisión final	2 semanas	01/06/2014	10/06/2014
Documentación	2 semanas	09/06/2014	18/06/2014

Tabla 2.1. Fases del proyecto, primera iteración

Total: 560 horas de dedicación.

3. Propuesta formal

La propuesta inicial consistía en mejorar la web actual haciendo una nueva web desde cero.

Se ha creado un entorno nuevo, mucho más adaptado que la plantilla básica con la que se había creado, para hacer la primera web de 9SotaPar. Puesto que la directiva de 9SotaPar desestimó continuar con su proyecto, no continuó con la realización de más circuitos y dio por finalizada su aventura en la organización de mas eventos, se ha acabado la colaboración con la anterior directiva de 9SotaPar, colaborando con la nueva, SotaPar, actual responsable de la revista y de la realización de los torneos “SotaPar”, con la que se esta realizando una colaboración para poder colaborar en su pagina web o sustituirla por la generada en este proyecto.

La idea inicial era muy ambiciosa para poder terminarla dentro del proyecto por lo que el desarrollo de la aplicación se seguirá realizando después de la finalización del proyecto, ya que las críticas han sido bastante satisfactorias y los conocimientos adquiridos están siendo los esperados.

Actualmente están en funcionamiento los requisitos mínimos que se solicitaban que básicamente consistían en eliminar la gestión manual de los archivos Excel y hacer las publicaciones de los resultados antes de la noche del ultimo día de campeonato.

La estructura actual de la base de datos, está compuesta por once tablas, nueve de ellas forman la base de datos en sí mismo y se han añadido una tabla de control de usuarios y una tabla temporal para poder hacer la publicación de los resultados acumulados del circuito, ya que se necesitan cálculos diversos que la usan como base. Como resultado final, el esquema de la base de datos es el siguiente: (página siguiente)

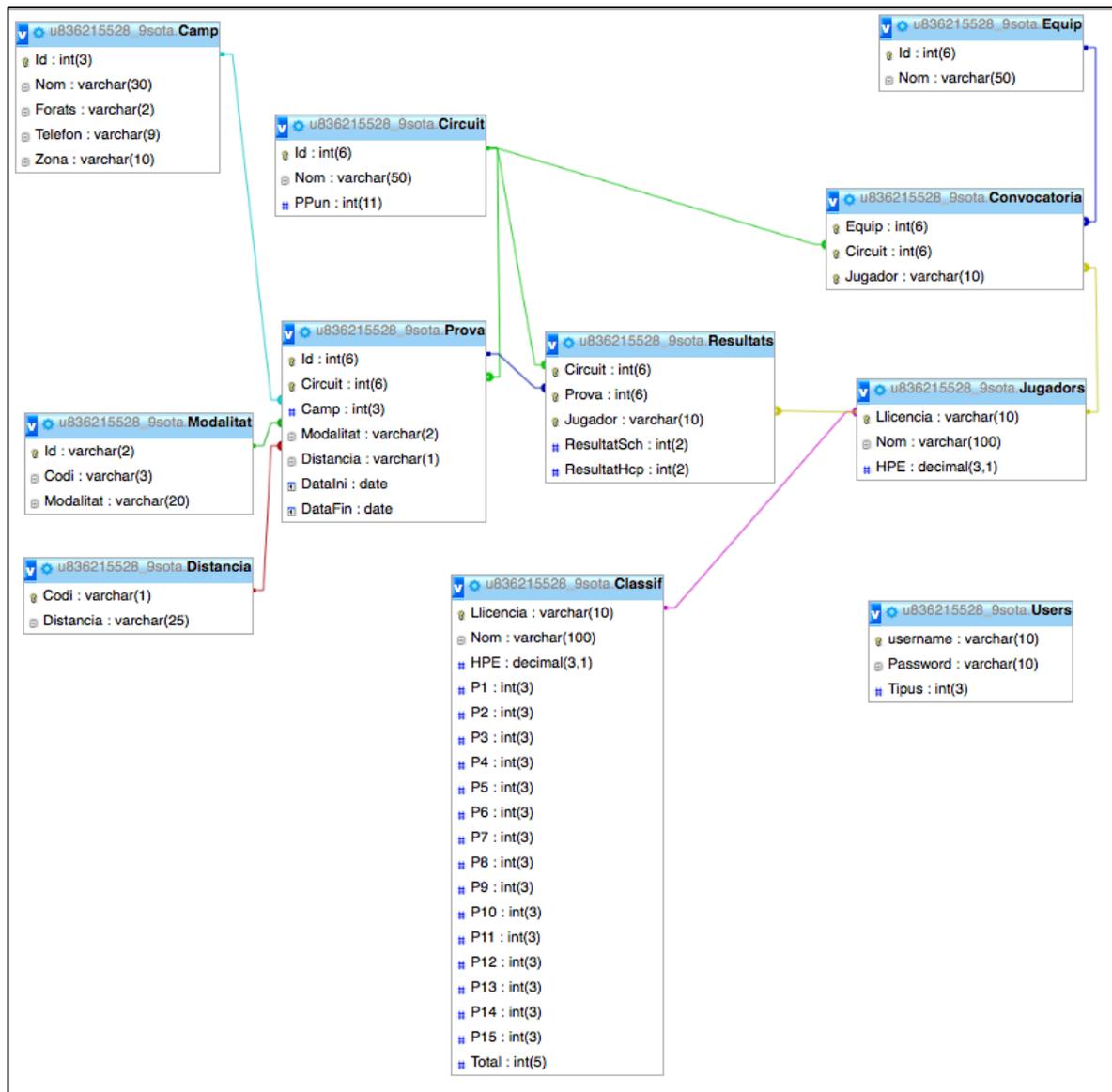


Fig. 3.1. Mapa conceptual de la base de datos

3.1. Resumen de las tablas

La información que almacenan las tablas son:

- **Camp**: Los datos de cada campo.
- **Circuit**: Nombre y numero de pruebas que puntúan del circuito.
- **Classif**: Temporal para realizar la publicación de las clasificaciones.
- **Convocatoria**: Lookup que informa que jugador juega en cada equipo para cada circuito.
- **Distancia**: Maestra la modalidad de distancia de juego.
- **Equip**: Información de los equipos.
- **Jugadors**: Datos de los jugadores.
- **Modalitat**: Maestra de las modalidades de juego.
- **Prova**: Contiene la información de campos y fechas en que se realizaran las pruebas de un circuito
- **Resultats**: Tabla principal donde se almacenan todos los resultados de las diferentes pruebas
- **Users**: Listado de usuarios administradores.

4. Diseño de la aplicación

En este apartado se explicará cómo se ha desarrollado la aplicación:

4.1. La base de datos

El tipo de base de datos que se utiliza en este proyecto web es MySQL; los principales motivos para usar este tipo de base de datos es que es open source (código abierto), cumple con las necesidades requeridas y su lenguaje es bastante sencillo.

Se han creado las siguientes tablas:

4.1.1. Camp

```
CREATE TABLE IF NOT EXISTS `Camp` (  
  
  `Id` int(3) NOT NULL,  
  
  `Nom` varchar(30) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT NULL,  
  
  `Forats` varchar(2) COLLATE utf8_unicode_ci NOT NULL,  
  
  `Telefon` varchar(9) COLLATE utf8_unicode_ci NOT NULL,  
  
  `Zona` varchar(10) COLLATE utf8_unicode_ci NOT NULL,  
  
  PRIMARY KEY (`Id`))
```

Id: Clave primaria que define la id de la tabla de forma automática e incremental.

Nom: Nombre del campo

Forats: Número de hoyos que tiene el campo (normalmente 9 o 18).

Telefon: Telefono del campo donde hacer las reservas de salidas

Zona: Provincia a la que corresponde el campo

4.1.2. Circuit

```
CREATE TABLE IF NOT EXISTS `Circuit` (  
  `Id` int(6) NOT NULL,  
  `Nom` varchar(50) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT NULL,  
  `PPun` int(11) NOT NULL,  
  PRIMARY KEY (`Id`))
```

Id: Clave primaria que define la ID de la tabla de forma automática e incremental.

Nom: Nombre del circuito

PPun: Número de pruebas que puntúan para el resultado total, ya que hay circuitos donde no es obligatorio jugar todas las pruebas.

4.1.3. Classif

```
CREATE TABLE IF NOT EXISTS `Classif` (  
  `Licencia` varchar(10) COLLATE utf8_unicode_ci NOT NULL,  
  `Nom` varchar(100) COLLATE utf8_unicode_ci NOT NULL,  
  `HPE` decimal(3,1) NOT NULL,  
  `P1` int(3) NOT NULL,  
  `P2` int(3) NOT NULL,  
  `P3` int(3) NOT NULL,  
  `P4` int(3) NOT NULL,  
  `P5` int(3) NOT NULL,  
  `P6` int(3) NOT NULL,
```

```
`P7` int(3) NOT NULL,  
  
`P8` int(3) NOT NULL,  
  
`P9` int(3) NOT NULL,  
  
`P10` int(3) NOT NULL,  
  
`P11` int(3) NOT NULL,  
  
`P12` int(3) NOT NULL,  
  
`P13` int(3) NOT NULL,  
  
`P14` int(3) NOT NULL,  
  
`P15` int(3) NOT NULL,  
  
`Total` int(5) NOT NULL,  
  
PRIMARY KEY (`Licencia`))
```

Licencia: Identificador del jugador, es único para todos los jugadores federados.

Nom: Nombre del Jugador.

HPE: Hándicap Exacto en el momento de sacar la clasificación (este hándicap es la última modificación que tenemos en nuestros campeonatos)

P1~P15: Puntuaciones para 15 pruebas.

Total: Resultado total de los mejores “PPun” (ver circuito) partidos jugados.

4.1.4. Convocatoria

```
CREATE TABLE IF NOT EXISTS `Convocatoria` (  
  `Equip` int(6) NOT NULL,  
  `Circuit` int(6) NOT NULL,  
  `Jugador` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  PRIMARY KEY (`Equip`,`Circuit`,`Jugador`))
```

Equip: Identificador del equipo.

Circuit: Identificador del circuito.

Jugador: Identificador del jugador, Licencia.

4.1.5. Distancia

```
CREATE TABLE IF NOT EXISTS `Distancia` (  
  `Codi` varchar(1) COLLATE utf8_unicode_ci NOT NULL,  
  `Distancia` varchar(25) COLLATE utf8_unicode_ci NOT NULL,  
  PRIMARY KEY (`Codi`))
```

Codi: Clave primaria que define el ID de la tabla de forma automática e incremental.

Distancia: Identificación literal de la distancia (EPPA o Catalanes)

4.1.6. Equip

```
CREATE TABLE IF NOT EXISTS `Equip` (  
  `Id` int(6) NOT NULL,  
  `Nom` varchar(50) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT NULL,  
  PRIMARY KEY (`Id`))
```

Id: Clave primaria que define la ID de la tabla de forma automática e incremental.

Nom: Nombre del equipo

4.1.7. Jugadors

```
CREATE TABLE IF NOT EXISTS `Jugadors` (  
  `Llicencia` varchar(10) COLLATE utf8_spanish_ci NOT NULL,  
  `Nom` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `HPE` decimal(3,1) NOT NULL,  
  PRIMARY KEY (`Llicencia`))
```

Llicencia: Clave primaria que define a cada jugador de forma única, este “código” lo entrega la federación de cada comunidad (Cataluña: ACPxxxxxx).

Nombre: Nombre completo del jugador.

HPE: Ultimo hándicap conocido por la organización, se obtiene de la modificación del ultimo campeonato incorporado a la base de datos.

4.1.8. Modalitat

```
CREATE TABLE IF NOT EXISTS `Modalitat` (  
  `Id` varchar(2) COLLATE utf8_unicode_ci NOT NULL,  
  `Codi` varchar(3) COLLATE utf8_unicode_ci NOT NULL,  
  `Modalitat` varchar(20) COLLATE utf8_unicode_ci NOT NULL,  
  
  PRIMARY KEY (`Id`))
```

Id: Clave primaria que define la ID de la tabla de forma automática e incremental.

Codi: Código internacional que se le da a las diferentes modalidades de juego.

Modalitat: Descripción literal de la modalidad.

4.1.9. Prova

```
CREATE TABLE IF NOT EXISTS `Prova` (  
  `Id` int(6) NOT NULL,  
  `Circuit` int(6) NOT NULL,  
  `Camp` int(3) NOT NULL,  
  `Modalitat` varchar(2) COLLATE utf8_unicode_ci NOT NULL,  
  `Distancia` varchar(1) COLLATE utf8_unicode_ci NOT NULL,  
  `DataIni` date NOT NULL,  
  `DataFin` date NOT NULL,  
  
  PRIMARY KEY (`Id`, `Circuit`))
```

4.1.10. Resultats

```
CREATE TABLE IF NOT EXISTS `Resultats` (  
  `Circuit` int(6) NOT NULL,  
  `Prova` int(6) NOT NULL,  
  `Jugador` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  `ResultatSch` int(2) NOT NULL,  
  `ResultatHcp` int(2) NOT NULL,  
  PRIMARY KEY (`Circuit`, `Prova`, `Jugador`))
```

Circuit: Identificador de Circuito.

Prova: Identificador de número de prueba

Jugador: Identificador de la licencia del jugador.

ResultatSch: Resultado scratch obtenido en la prueba.

ResultatHcp: Resultado hándicap obtenido en la prueba.

4.1.11. Users

```
CREATE TABLE IF NOT EXISTS `Users` (  
  `username` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  `Password` varchar(10) CHARACTER SET utf8 COLLATE utf8_spanish_ci NOT  
  NULL,  
  `Tipus` int(3) NOT NULL DEFAULT '1',  
  PRIMARY KEY (`username`))
```

Username: Nombre del usuario o nickname

Password: Contraseña para acceder a la parte privada del aplicativo.

Tipus: Tipo de usuario, de momento sólo existe el tipo 1 que serán los administradores.

4.2. Diseño de la aplicación

La aplicación esta dividida en dos partes; zona pública y zona de administración.

Para estructurar todo el contenido de la web, se ha hecho uso del estilo de maquetación (framework) 960 grid; en la siguiente web, <http://960.gs>, se pueden encontrar varias demos de otras webs y/o aplicaciones web que han usado este framework como base, así como los códigos CSS que son necesarios para hacer la maquetación del proyecto. Se ha elegido utilizar la variante de 12 columnas ya que es la que mejor se adaptaba a la idea de la presentación de la información.

4.2.1. Zona publica



Fig. 4.1. Portada aplicación

La zona pública, empieza con la portada de la aplicación, en ella se encuentra el menú desplegable que permitirá moverse por la información suministrada a los usuarios/jugadores.

Este menú está formado por código Html animado mediante código de hoja de estilo (CSS) y para poder tener la opción multilingüe se le ha añadido código PHP que permite que se cargue el contenido de las variables de idioma dependiendo que idioma se ha elegido.

HTML del menú:

```
<div id="header">
  <div class="container_12">
    <div id="logo" class="grid_4">
      <a href="/">
        
      </a>
    </div>
    <div class="grid_8">
      <ul id="nav">
        <li><a href="index.php" class="">Home</a></li>
        <li><a><?php echo $stext['M_CLAS']; ?></a>
          <ul>
            <li><a href="FrontPage/clasificacio_s.php"><?php echo $stext['CLA_S']; ?></a></li>
            <li><a href="FrontPage/clasificacio_p.php"><?php echo $stext['CLA_P']; ?></a></li>
            <li><a href="FrontPage/clasificacio_0.php"><?php echo $stext['CLA_0']; ?></a></li>
            <li><a href="FrontPage/clasificacio_V.php"><?php echo $stext['CLA_V']; ?></a></li>
            <li><a href="FrontPage/clasificacio_X.php"><?php echo $stext['CLA_X']; ?></a></li>
            <li><a href="FrontPage/premis.php"><?php echo $stext['PREMI']; ?></a></li>
          </ul>
        </li>
        <li><a><?php echo $stext['M_RESU']; ?></a>
          <ul>
            <li><a href="FrontPage/resultat_s.php"><?php echo $stext['RES_S']; ?></a></li>
            <li><a href="FrontPage/resultat_h.php"><?php echo $stext['RES_H']; ?></a></li>
          </ul>
        </li>
        <li><a href="FrontPage/calendari.php"><?php echo $stext['M_CALE']; ?></a></li>
        <li><a href="FrontPage/camps.php"><?php echo $stext['M_CAMP']; ?></a></li>
        <li><a href="FrontPage/glosari.php"><?php echo $stext['M_GLOS']; ?></a></li>
        <li><a><?php echo $stext['M_IDIOMA']; ?></a>
          <ul>
            <li><a href="?local=cat"><?php echo $stext['I_CAT']; ?></a></li>
            <li><a href="?local=es"><?php echo $stext['I_ESP']; ?></a></li>
          </ul>
        </li>
        <li><a href="Private/Login/Logon.php">Login</a></li>
      </ul>
    </div>
    <div class="clear"></div>
  </div>
</div>
```

Fig. 4.2. HTML menú desplegable

CSS correspondiente al menú:

```
body {
  padding: 0px;
  margin: 0px;
  background-color: #999;
  font: 13px/1.5 'Helvetica Neue',Arial,'Liberation Sans',FreeSans,sans-serif;
  font-family: franklin-gothic-urw,helvetica,arial,sans-serif;
  font-style: normal;
  font-weight: 400;
  line-height: 18px;
  color: #333;
```

```
display: block;
min-width:960px;
}

#header{
padding: 0px;
margin: 0px;
color: #fff;
border-bottom: solid #DAA520 1px;
background: #555;
}

#logo a{
display: block;
padding: 5px 0;
text-align: center;
}

#nav {
list-style-type: none;
padding-top: 34px;
margin: 0 -15px 0 0;
float: right;
font-size: 14px;
line-height: 18px;
text-shadow: 1px 1px 2px #000;
text-transform: uppercase;
white-space: nowrap;
}

#nav li {
position: relative;
float: left;
margin: 0;
padding: 0;
}

#nav li:hover ul { left:0; }
#nav li:hover ul li a:hover {
color: #fff;
background: #333;
}

#nav li a:hover { background: #717171; }

#nav li ul {
box-shadow: 0px 15px 5px -5px #333;
```

```
margin: 0px;
padding: 0px;
text-transform: capitalize;
}

#nav ul {
list-style-type: none;
position: absolute;
left: -9999px;
z-index: 5000;
min-width: 110px;
background: #717171;
}

#nav ul li {
margin-left: 0;
padding-top: 1px;
float: none;
}

#nav ul a { white-space: nowrap; padding: 2px 5px;}

#nav a {
display: block;
padding: 5px;
color: #ddd;
text-decoration: none;
font-size: 12px;
}

#selldioma {
font-size: 14px;
line-height: 18px;
text-transform: capitalize;
}

#selldioma a:link, a:visited {
color: #fff;
text-decoration: none;
}

#loggin { text-align: right;}

#loggin a:link, a:visited { color: #ff0; text-decoration: none; }
```

Fig. 4.3 CSS menú desplegable

4.2.1.1. Clasificación

Al pasar el ratón por encima de “Clasificación”, aparecerá un menú desplegable en el que se encontrarán los diferentes tipos de clasificación a la que se puede acceder:



Fig. 4.4. Clasificación - Menú Clasificación

Una vez en su interior, se puede elegir el torneo del que se quiere obtener las clasificaciones. Actualmente los torneos no se eliminan del menú de selección, lo que permite seguir viendo la información de éstos, incluso después de finalizar el campeonato.



Fig. 4.5. Clasificación - Selección campeonato

Este sería el resultado de haber elegido, clasificación scratch y el torneo "Circuit complet":

9 sota par La Revista de Pitch & Putt														
HOME CLASIFICACION RESULTADOS CALENDARIO CAMPOS GLOSARIO IDIOMA LOGIN														
Circuit Complet Clasificacion Scratch														
Licencia	Nombre	HPE	Resultados										Total	
1	ACPP000179	MONTERO SANCHEZ, PILAR	-3.6	28	36	35	0	0	0	0	0	0	0	71
2	ACPP002173	PLATERO SANCHEZ, ALBERT	2.3	28	33	35	0	0	0	0	0	0	0	68
3	ACPP022297	GOMEZ MARTA, MARIO	-0.9	34	0	31	0	0	0	0	0	0	0	65
4	ACPP007324	SANTAMARIA SOBERON, JULIO	1.8	35	29	22	0	0	0	0	0	0	0	64
5	ACPP001975	CORTES GUERRA, NICOLAS	4.3	35	26	27	0	0	0	0	0	0	0	62
6	ACPP000180	CORTADA RECOLONS, JOSEP	5.3	33	28	26	0	0	0	0	0	0	0	61
7	ACPP022296	CABRERA GIBAJA, ANTONIO	5.2	30	0	31	0	0	0	0	0	0	0	61
8	ACPP010431	LOSADA FERNANDEZ, JOSE LUIS	3.5	0	29	30	0	0	0	0	0	0	0	59
9	ACPP010584	LOSADA FERNANDEZ, MIGUEL ANGEL	2.4	27	24	30	0	0	0	0	0	0	0	57
10	ACPP011844	GASULL MARTINEZ, JOSEP MARIA	1.6	0	27	29	0	0	0	0	0	0	0	56
11	ACPP001974	CORTES PONS, FERRAN	6.7	26	22	30	0	0	0	0	0	0	0	56
12	ACPP022904	RUBIO MARTIN, JUAN	7.6	0	27	27	0	0	0	0	0	0	0	54
13	ACPP009872	CANO YA	3.0	34	19	0	0	0	0	0	0	0	0	53
14	ACPP025727	GIMENO GARCIA, MARC	14.6	23	0	30	0	0	0	0	0	0	0	53
15	ACPP004882	BACARDIT GIRIBETS, PEPITA	8.1	0	25	27	0	0	0	0	0	0	0	52
16	ACPP008571	BARJA GIL, FRANCISCO	4.4	22	28	0	0	0	0	0	0	0	0	50
17	ACPP001445	HERNANDEZ LORENTE, ORIOL	12.0	24	22	26	0	0	0	0	0	0	0	50
18	ACPP003842	PEREZ HERNANDEZ, DIEGO	5.5	0	19	30	0	0	0	0	0	0	0	49
19	ACPP009789	BICARDI QUERO, ABEL	6.2	0	17	31	0	0	0	0	0	0	0	48
20	ACPP010227	CANTOS VERGES, GUSTAVO	5.5	21	19	24	0	0	0	0	0	0	0	45
21	ACPP005549	ALARCON MESTRES, VICEN	4.3	24	0	19	0	0	0	0	0	0	0	43
22	ACPP001423	CLOTET RIFE, JORDI	-1.6	43	0	0	0	0	0	0	0	0	0	43
23	ACPP001389	PONSDOMENECH CALVET XAVIER	-5.6	42	0	0	0	0	0	0	0	0	0	42

Prueba	F. Inicio	F. Fin	Campo de la prueba	Modalidad	Distancia
1	25/01/14	25/01/14	Papalus P&P	Individual	Catalanes
2	26/04/14	30/04/14	P&P Centre de Golf El Vendrell	Individual	Catalanes
3	17/05/14	18/05/14	P&P Teia	Individual	Catalanes
4	15/08/14	17/08/14	Mas Pages P&P	Individual	Catalanes

Circuit Complet - OK

Fig. 4.6. Clasificación - Scratch

Y este sería el resultado de haber elegido la clasificación de hándicap positivo:

9 sota par La Revista de Pitch & Putt														
HOME CLASIFICACION RESULTADOS CALENDARIO CAMPOS GLOSARIO IDIOMA LOGIN														
Circuit Complet Clasificacion Handicap Positivo														
Licencia	Nombre	HPE	Resultados										Total	
1	ACPP000179	MONTERO SANCHEZ, PILAR	-3.6	24	36	34	0	0	0	0	0	0	0	70
2	ACPP022297	GOMEZ MARTA, MARIO	-0.9	33	0	31	0	0	0	0	0	0	0	64
3	ACPP001423	CLOTET RIFE, JORDI	-1.6	43	0	0	0	0	0	0	0	0	0	43
4	ACPP020354	HERNANDEZ GARCIA, JUAN JOSE	-0.9	40	0	0	0	0	0	0	0	0	0	40
5	ACPP002871	SERRA CATASUS, CARLOTA	-0.1	0	0	37	0	0	0	0	0	0	0	37
6	ACPP008122	SANZ RAMON, ENRIC	-4.4	37	0	0	0	0	0	0	0	0	0	37
7	ACPP005554	ROVIROSA REDO, ENRIC	-0.2	36	0	0	0	0	0	0	0	0	0	36
8	ACPP011428	SALCEDO RECIO, ANTONIO	-1.6	36	0	0	0	0	0	0	0	0	0	36
9	ACPP001389	PONSDOMENECH CALVET, XAVIER	-5.6	36	0	0	0	0	0	0	0	0	0	36
10	ACPP003442	SERRA GOLOBART, JOSE ISIDRO	-1.9	35	0	0	0	0	0	0	0	0	0	35
11	ACPP005213	LLOVERAS BARBOSA, RAMON	-0.5	35	0	0	0	0	0	0	0	0	0	35
12	ACPP015444	PETIT GARCIA, EDGAR	0.3	35	0	0	0	0	0	0	0	0	0	35
13	ACPP001894	ROGER PUJOL, MIQUEL	-5.2	35	0	0	0	0	0	0	0	0	0	35
14	ACPP008797	HERMOSO SANCHEZ, SERGIO	-2.2	35	0	0	0	0	0	0	0	0	0	35
15	ACPP016895	WATERSON, TERENCE	0.3	34	0	0	0	0	0	0	0	0	0	34
16	ACPP008666	SOLER BENET, FIDEL	-5.0	34	0	0	0	0	0	0	0	0	0	34
17	ACPP004837	BAUTISTA GALLARDO, GABRIEL	0.4	34	0	0	0	0	0	0	0	0	0	34
18	ACPP006305	FUENTES FUENTES, CELEDONIO	-0.3	0	34	0	0	0	0	0	0	0	0	34
19	ACPP000507	MARCO MARTINEZ, MERCEDES	-0.9	0	33	0	0	0	0	0	0	0	0	33
20	ACPP017287	ALCALA BADIA, IVAN	-0.4	0	33	0	0	0	0	0	0	0	0	33
21	ACPP010533	FERNANDEZ CORREDOR, RAUL	-1.4	33	0	0	0	0	0	0	0	0	0	33
22	ACPP010588	NAVARRO CA	-2.3	33	0	0	0	0	0	0	0	0	0	33
23	ACPP004927	HERNANDEZ HERNANDEZ, ANTONIO	-0.2	0	32	0	0	0	0	0	0	0	0	32

Prueba	F. Inicio	F. Fin	Campo de la prueba	Modalidad	Distancia
1	25/01/14	25/01/14	Papalus P&P	Individual	Catalanes
2	26/04/14	30/04/14	P&P Centre de Golf El Vendrell	Individual	Catalanes
3	17/05/14	18/05/14	P&P Teia	Individual	Catalanes
4	15/08/14	17/08/14	Mas Pages P&P	Individual	Catalanes

Circuit Complet - OK

Fig. 4.7. Clasificación - Hcp Positivo

Para poder realizar esta parte de la aplicación aparte de los lenguajes comentados para realizar la portada, se ha tenido que añadir el AJAX, ya que permite poder moverse entre los diferentes torneos sin necesidad de tener que estar cargando la página de forma constante, y también se han añadido las funciones SQL para poder obtener y tratar los datos almacenados en la base de datos. Estas funciones están almacenadas en “sentencias.php”.

Código AJAX:

```
<!-- Libreria jQuery -->
<script type='text/javascript' src="//ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>
<!-- Acción sobre el botón con id=boton y actualizamos el div con id=capa -->
<script type="text/javascript">
    $(document).ready(function() {
        $("#boton").click(function(event) {
            var v_Circuit=document.getElementById("des_circuit").value;
            $("#clasif").load('clasificacio_10.php',{circuit: v_Circuit, tipus:'SCH'});
        });
    });
</script>
```

Fig. 4.8. Clasificación - Código Ajax

Código HTML página principal:

```
</br>
<div id="content" class="container_12">
    <div id="clasif" class="grid_12"></div>
    <div class="clear"></div>

    <div id="selclasif" class="grid_12">
        <center>
            <form method="post" id="Selecc">
                <?php
                    $result_cons = Desp_Circuit();
                    echo "<select name='campionat' id='des_circuit'>";
                    while($fila=mysql_fetch_array($result_cons)){
                        echo "<option value='". $fila['Id']. "-" . $fila['Nom']. "'>". $fila['Nom']. "</option>";
                    }
                    echo "</select>";
                ?>
                <input name="boton" id="boton" type="button" value="OK" />
            </form>
        </center>
    </div>
    <div class="clear"></div>
</div>
```

Fig. 4.9. Clasificación - HTML principal Clasificación

Código PHP página secundaria:

```

<?php
$cirucuit = $_POST['circuit'];
$tipus = $_POST['tipus'];

$pos = strpos($cirucuit, '-');
$ex_campionat = substr($cirucuit,0,$pos);
$ex_nom = substr($cirucuit,$pos+1);

$query2 = Get_Circuit($ex_campionat);
$data2 = mysql_fetch_array($query2);

if ($tipus == 'SCH') {$query3 = CarClassSch($ex_campionat,
$data2['ppun']);}
else {$query3 = CarClassHcp($ex_campionat, $data2['ppun'], $tipus);}

$cont = 0;

//var_dump($_POST); // Muestra el contenido de una variable

echo
"<table class='taula5'>
  <caption>".
    $data2['nom']."<br/>"; // Nombre Circuito
    if ($tipus == 'SCH') {echo $text['CL_TS']."<br>";}
    if ($tipus == 'POS') {echo $text['CL_TP']."<br>";}
    if ($tipus == 'CL0') {echo $text['CL_T0']."<br>";}
    if ($tipus == 'CLV') {echo $text['CL_TV']."<br>";}
    if ($tipus == 'CLX') {echo $text['CL_TX']."<br>";}
    echo "&nbsp;";
  </caption>

  <thead>
    <tr>
      <th class='th1' scope='col'>&nbsp;</th>
      <th class='th2' scope='col'>".$text['CL_LIC']."</th>
      <th class='th3' scope='col'>".$text['CL_NOM']."</th>
      <th class='th4' scope='col'>HPE</th>
      <th class='th5' scope='col'
colspan='10'>".$text['CL_RES']."</th>
      <th class='th6' scope='col'>".$text['CL_TOT']."</th>
      <th class='th7' scope='col'>&nbsp;</th>
    </tr>
  </thead>

  <tfoot>
    <td colspan=16'>&nbsp;</td>

```

```

</tfoot>

<tbody>
  <tr><td colspan='16'>
    <div class='innerb'>
      <table class='taula6'>;
      while($cont < mysql_num_rows($query3)){
        $row = mysql_fetch_array($query3);
        $pos = $cont +1;
        if ($cont%2) { echo "<tr>";} else {echo "<tr
class='dk'>";}

        echo "
          <th class='td1' scope='row'>".$pos."</th>
          <td class='td2'>".$row['Licencia']."</td>
          <td class='td3'>".$row['Nom']."</td>
          <td class='td4'>".$row['HPE']."</td>
          <td class='td5'>".$row['P1']."</td>
          <td class='td5'>".$row['P2']."</td>
          <td class='td5'>".$row['P3']."</td>
          <td class='td5'>".$row['P4']."</td>
          <td class='td5'>".$row['P5']."</td>
          <td class='td5'>".$row['P6']."</td>
          <td class='td5'>".$row['P7']."</td>
          <td class='td5'>".$row['P8']."</td>
          <td class='td5'>".$row['P9']."</td>
          <td class='td5'>".$row['P10']."</td>
          <td class='td7'>".$row['Total']."</td>

          </tr>";
          $cont++;
        }
        echo "</table>
      </div>
    </td></tr>
  </tbody>
</table><br><br>";

echo "<table id='calendari'>
  <thead>
    <tr>
      <th class='th0'>".$text['C_PR']."</th>
      <th class='th1'>".$text['C_D1']."</th>
      <th class='th1'>".$text['C_D2']."</th>
      <th class='th2'>".$text['C_CA']."</th>
      <th class='th3'>".$text['C_MO']."</th>
      <th class='th4'>".$text['C_DI']."</th>
    </tr>
  </thead>

```

```
<tbody>";
$query= Calendari($ex_campionat);
$cont= 0;
while($cont<mysql_num_rows($query)){
    $row=mysql_fetch_array($query);
    echo "<tr>
        <td class='td0'>".$row['id']. "</td>
        <td class='td1'>".$row['dataini']. "</td>
        <td class='td1'>".$row['datafin']. "</td>
        <td class='td2'>".$row['nom']. "</td>
        <td class='td3'>".$row['modalitat']. "</td>
        <td class='td4'>".$row['distancia']. "</td>
    </tr>";
    $cont++;}
echo "</tbody></table><br><br>";
```

?>

Fig. 4.10. Clasificación - CSS principal Clasificación

Como se puede ver en el código de la figura 4.10 esta pagina secundaria, creará el cuadro de clasificación para todos los tipos de torneos y el calendario de pruebas del torneo elegido. Esta pagina es común para todas las paginas de clasificación y se llama desde el código Ajax, junto con los parámetros necesarios para su ejecución, que en este caso serán el tipo de torneo que se está consultando y el identificador del torneo.

4.2.1.2. Resultados

La funcionalidad del apartado de resultados es muy parecida al de clasificaciones, funcionando de la misma forma: una página principal o externa, con código AJAX, ejecuta una pagina secundaria o interior que crea la tabla de resultados para el campeonato y prueba elegidos.

En este caso en la función de llamada de la página interior se le pasa el tipo de resultado deseado y los identificadores de campeonato y prueba.



Posicion	Licencia	Nombre	HPE	Puntos
1	ACPP002871	SERRA CATASUS, CARLOTA	-0.1	37
2	ACPP002173	PLATERO SANCHEZ, ALBERT	2.3	35
3	ACPP000124	GARCIA SAMPEDRO, JORDI	2.8	35
4	ACPP000179	MONTERO SANCHEZ, PILAR	-3.6	35
5	ACPP009611	POZO ALCANTARA, ANTONIO	0.7	33
6	ACPP021793	ARMISEN MIGUEL, FREDERIC	5.1	32
7	ACPP013058	OLIVERAS COLL, JORDI	2.8	32
8	ACPP001081	PAU SALINAS, RAUL	4.9	32
9	ACPP001049	PAU AVENTIN, SALVADOR	2.4	32
10	ACPP016507	PI	5.6	31
11	ACPP009789	BICARDI QUERO, ABEL	6.2	31
12	ACPP022296	CABRERA GIBAJA, ANTONIO	5.2	31
13	ACPP022297	GOMEZ MARTA, MARIO	-0.9	31
14	ACPP012782	MARTINEZ LOPEZ, FRANCISCO	4.7	30

Prueba nº: 3

Circuit Complet - Prueba 3 OK

Fig. 4.11. Resultados - Scratch

Código AJAX:

```

<!-- Libreria jQuery -->
<script type='text/javascript' src="//ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>
<!-- Acción sobre el botón con id=boton y actualizamos el div con id=capa -->
<script type="text/javascript">
    $(document).ready(function() {
        $("#boton").click(function(event) {
            var v_Circuit=document.getElementById("des_camp").value;
            $("#result").load('resultat.php',{circuit: v_Circuit, tipus:'SCH'});
        });
    });
</script>

```

Fig. 4.12. Resultados - Código Ajax

Código PHP página interior:

```

<?php
    $circuit = $_POST['circuit'];
    $pos = strpos($circuit,'-');
    $sex_campionat = substr($circuit,0,$pos);
    $sex_prova = substr($circuit,$pos+1);
    $tipus = $_POST['tipus'];

    $query1 = DadesProvaCircuit($sex_campionat, $sex_prova);
    $query2 = Get_Circuit($sex_campionat);
    $query3 = GetResultatsProva($sex_campionat, $sex_prova, $tipus);
    $data1 = mysql_fetch_array($query1);
    $data2 = mysql_fetch_array($query2);
    $cont = 0;

    //var_dump($_POST); // Muestra el contenido de una variable

    echo
    "<table class='taula1'>
        <caption>".
            $data2['nom']." - ".$data1['Moda']."<br/>". // Nombre Circuito +
Modalidad
            $data1['Camp']." - Mides ".$data1['Dis']."<br/>"; // Nombre Campo +
Distancias
            if ($tipus == 'SCH') {echo $text['RE_SCH']."<br/>";} else {echo
$text['RE_HCP']."<br/>";}
            echo "&nbsp;";
        </caption>

        <thead>
            <tr>
                <th class='th1' scope='col'>".$text['RE_POS']."</th>
                <th class='th2' scope='col'>".$text['CL_LIC']."</th>
                <th class='th3' scope='col'>".$text['CL_NOM']."</th>
                <th class='th4' scope='col'>HPE</th>
                <th class='th5' scope='col'>".$text['RE_PUN']."</th>
                <th class='th6' scope='col'>&nbsp;</th>
            </tr>
        </thead>

        <tfoot>
            <td colspan='6'>".$text['C_PR']." n°: ".$sex_prova."</td>
        </tfoot>

        <tbody>
            <tr><td colspan='6'>
                <div class='innerb'>
                    <table class='taula2'>";
                    while($cont<mysql_num_rows($query3)){

```

```
class='dk'>";}

scope='row'>".$row['id']."</th>

$row=mysql_fetch_array($query3);
if ($cont%2) { echo "<tr>";} else {echo "<tr

echo "<th class='td1'

<td class='td2'>".$row['Jugador']."</td>
<td class='td3'>".$row['Nom']."</td>
<td class='td4'>".$row['Hpe']."</td>
<td class='td5'>".$row['Resultat']."</td>";
$cont++;
}
echo "</table>

</div>
</td></tr>
</tbody>
</table><br><br>";

?>
```

4.2.1.3. Calendario

La funcionalidad del calendario está realizada en una página PHP que llama a una función descrita en la página “sentencias.php”. Actualmente se ha fijado el torneo 1 (SotaPar 15 aniversario) como los datos a mostrar. En esta página se puede ver donde se realizará cada prueba de un circuito, en que fechas y que modalidad y distancias será cada prueba.



Prueba	F. Inicio	F. Fin	Campo de la prueba	Modalidad	Distancia
1	03/01/14	05/01/14	P&P Badalona	Individual	Catalanes
2	24/01/14	26/01/14	P&P Tennis Mora	Individual	Catalanes
3	31/01/14	02/02/14	Can Cuyas	Individual	Catalanes
4	14/02/14	16/02/14	P&P Sant Cebria	Individual	Catalanes
5	07/03/14	09/03/14	P&P Centre de Golf El Vendrell	Individual	Catalanes
6	21/03/14	23/03/14	Franciac P&P	Individual	Catalanes
7	04/04/14	06/04/14	Papalus P&P	Individual	Catalanes
8	18/04/14	20/04/14	La Garriga Golf P&P	Individual	Catalanes
9	25/04/14	27/04/14	HCP1 P&P	Individual	Catalanes
10	02/05/14	04/05/14	P&P Vallromanes	Individual	Catalanes
11	16/05/14	18/05/14	P&P Teia	Individual	Catalanes

Fig. 4.13. Calendario

Código HTML:

```

</br>
<div id="content">
  <div class="container_12">
    <div class="grid_9 prefix_1">
      <table id="calendari">
        <thead>
          <tr>
            <th class="th0"><?php echo $text['C_PR']; ?></th>
            <th class="th1"><?php echo $text['C_D1']; ?></th>
            <th class="th1"><?php echo $text['C_D2']; ?></th>
            <th class="th2"><?php echo $text['C_CA']; ?></th>
            <th class="th3"><?php echo $text['C_MO']; ?></th>
            <th class="th4"><?php echo $text['C_DI']; ?></th>
          </tr>
        </thead>
        <tbody>
          <?php // Consulta
          $query= Calendari('1');
          $cont= 0;

          // Resultados y Tratamiento
          while($cont<mysql_num_rows($query)){ $row=mysql_fetch_array($query); ?>
            <tr>
              <td class="td0"><?php echo $row['id']; ?></td>
              <td class="td1"><?php echo $row['dataini']; ?></td>
              <td class="td1"><?php echo $row['datafin']; ?></td>
              <td class="td2"><?php echo $row['nom']; ?></td>
              <td class="td3"><?php echo $row['modalitat']; ?></td>
              <td class="td4"><?php echo $row['distancia']; ?></td>
            </tr>
            <?php $cont++; }?>
          </tbody>
        </table>
      </div>
    <div class="grid_2"></div>
    <div class="clear"></div>
  </div>
</div>

```

Fig. 4.14. Calendario - Código HTML

4.2.1.4. Campos



Fig. 4.15. Campos

Esta parte de la aplicación es una de las mejoras que se están desarrollando actualmente y la idea es poder crear una página con información de cada campo, al seleccionar la imagen del campo deseado.

4.2.1.5. Idioma



Fig. 4.16. Idioma

Para poder cambiar de idioma es tan simple como hacer click en el idioma que se quiere seleccionar, todos los literales de la aplicación se cambiarán directamente de idioma; esto es posible ya que todas las paginas deberían de tener incluida la página “idiomes.php” que lee las diferentes variables de idioma que hay en las paginas y las traduce por los literales que hay declarados en la pagina de idioma correspondiente. Como idioma por defecto será el catalán.

Código PHP página *idiomes.php*:

```

<?php session_start();
/*Lo que vamos a comprobar primero es si se esta solicitando un nuevo idioma
Si se esta solicitando, verificamos que el idioma existe tambien*/
if(!empty($_REQUEST['local']) && file_exists('Lang/'.$_REQUEST['local'].'.php')){
    /*Entonces lo que vamos a hacer ahora, es decir que a partir de ahora,
    nuestro idioma por defecto es este, al menos que se solicite cambiarlo de vuelta*/
    $_SESSION['local'] = $_REQUEST['local'];
    //y esto lo vamos a usar despues
    $language = $_REQUEST['local'];
}
//Sino se solicito ningun idioma, verificamos si quedo guardado en nuestra session y lo mismo que antes, esto para despues
elseif(isset($_SESSION['local']))
    $language = $_SESSION['local'];
//Y por ultimo, si nada de lo anterior cumple los requisitos, cargamos el idioma, que seria el idioma por defecto
else
    $language = 'cat';
require_once('Lang/'.$language.'.php');
?>

```

Fig. 4.17. Idioma – idiomes.php

4.2.1.6. Login

Ésta es la puerta para acceder a la zona de administración de la aplicación; por lo que al elegir esta opción aparecerá una pantalla donde poderse identificar.

Fig. 4.18. Login – Identificación

En caso de no poner correctamente el usuario o el password devolverá de nuevo a la pantalla de identificación mostrando un error de identificación.

Fig. 4.19. Login – Error login

Para evitar los ataques de SQL Injection se han tomado dos líneas de defensa que recomiendan en varios foros sobre protección de estos ataques, el primero de ello es limitar los dos campos a que únicamente se puedan introducir caracteres alfabéticos y el segundo es hacer la comprobación del password en la propia aplicación.

```

1 <?php
2 //include ("sentencias.php");
3
4 $username=$_POST['usuari'];
5 $password=$_POST['password'];
6
7 if (isset($username) && !empty($username) && ctype_alpha($username) && ctype_alpha($password)) {
8     if ($password==NULL) {header("Location: Logon.php?errorusuari=NPass");}
9     else{
10         $enlace = connect(); // Me conecto al MySQL
11         if (!$enlace) { die("No pudo conectarse: ".mysql_error()); }
12
13         $db=mysql_connect($enlace) or die("No puedo elegir la Base de Datos"); // Selecciono la BD
14         $query = GetPass($username, $enlace) or die(mysql_error());
15         $row = mysql_fetch_array($query);
16
17         if($row['password'] != $password) { header("Location: Logon.php?errorusuari=PassErr");}
18     } else{
19         session_start();
20         $_SESSION["s_username"] = $username;
21         $_SESSION["s_valida"] = "SI";
22         header("Location: ../SelecPanel.php");
23     }
24     mysql_close($enlace); // Cerramos Conexión
25 }
26 } else { header("Location: Logon.php?errorusuari=NUsuari");}
27 ?>

```

Fig. 4.20. Login – SQL Injection

4.2.2. Zona administración

9 sota par La Revista de Pitch & Putt Log Off

Has accedido como: **Admin**

Circuito	Prueba	Campo
<ul style="list-style-type: none"> Crear Circuito Buscar Circuito 	<ul style="list-style-type: none"> Crear Prueba (-) Modificar Prueba (X) Eliminar Prueba (X) 	<ul style="list-style-type: none"> Añadir Campo (-) Buscar Campo (-) Eliminar Campo (X) 
Equipo	Jugador	Modalidad
<ul style="list-style-type: none"> Crear Equipo (-) Modificar Equipo (-) Fichar Jugador (-) Borrar Equipo (X) 	<ul style="list-style-type: none"> Nuevo Jugador (-) Modificar Jugador (-) Buscar por Hep (X) 	<ul style="list-style-type: none"> Añadir Modalidad (-) Modificar Modalidad (X) Eliminar Modalidad (X) 
Usuario	Resultado	
<ul style="list-style-type: none"> Añadir Usuario (X) Modificar Usuario (X) Eliminar Usuario (X) 	<ul style="list-style-type: none"> Introduir Resultado Cargar Resultado Consultar Resultado (X) Borrar Resultado (X) 	

Fig. 4.21. Menú Administración

La zona de administración es la parte privada de la aplicación a la que se accede mediante el login del apartado anterior; en este menú se puede administrar todas las tablas de la base de datos que podrían tener modificaciones o nuevos registros.

El menú está formado por nueve apartados, ocho siempre visibles y un noveno que se usará para los menús emergentes (actualmente sólo para cargar resultados). Las opciones básicas de cada apartado son las de crear un registro, modificarlo y, si corresponde, borrarlo. Para esta primera fase se ha implementado el 100% de las opciones para “Circuito” y se han creado las pantallas de muestra de conceptos para el resto de opciones, que se irán implementando en futuras fases.

El menú de administración está codificado en HTML, al que se le ha añadido código en PHP para poder tener la opción de idioma en los literales de la aplicación y se ha mejorado

su aspecto visual y el estilo de la página con CSS, para hacer los menús emergentes se ha creado una función JavaScript.

Todas las páginas a las que se puede acceder a partir del menú, incluido el menú de administración, se les ha incluido el control de login “Seguretat.php”, cuya función es controlar que el usuario se ha identificado antes de poder entrar en la URL solicitada. Si la sesión es válida, se podrá acceder a ella de forma transparente, en caso de que no sea una sesión válida, se abrirá la pantalla de login con un error de falta de registro.

```
<?php
//Inicio la sesión
session_start();

//COMPRUEBA QUE LA SESION SEA VALIDA
if ($_SESSION["s_valida"] != "SI") {
    //si no existe, envio a la página de autentificacion
    header("Location: Logon.php?errorusuari=NValid");
    //ademas salgo de este script
    exit();
}
?>
```

Fig. 4.22. Menú Administración – Verificación de la sesión

La codificación de cada apartado está compuesta por una tabla con dos columnas, una para los literales y la otra para la imagen. Los literales llevan una referencia de hipertexto con la url que se ha de abrir cuando se seleccionan.

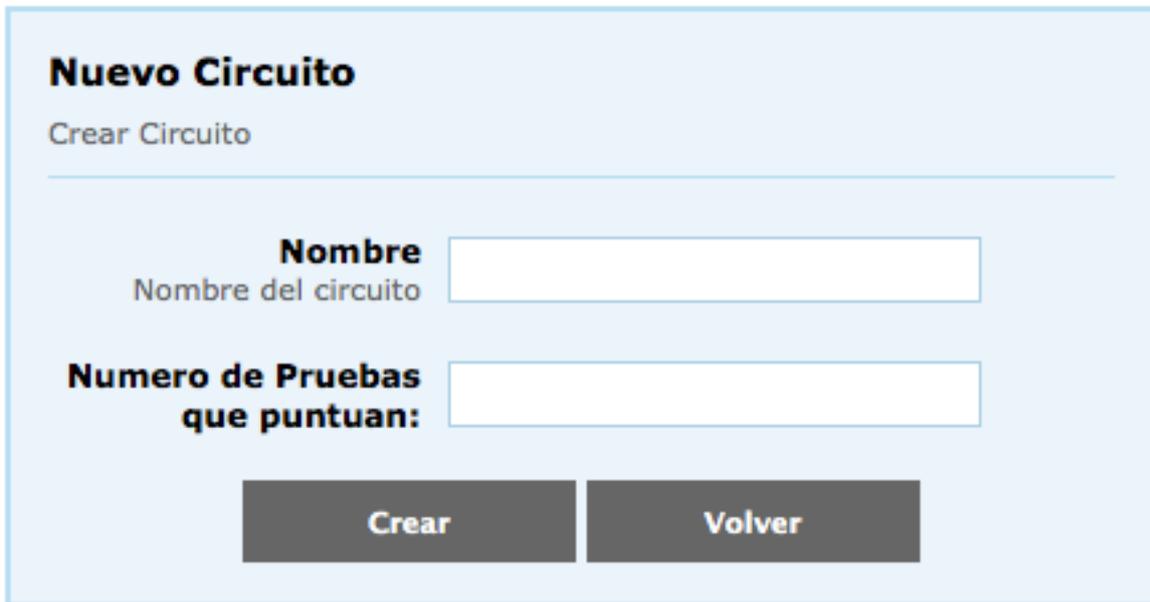
```
<div id="Circuit" class="grid_4">
  <b><?php echo $text['MP_CIR']; ?><hr/></b>
  <table width="300">
    <td width="165" valign="top">
      <a href="New/New_Circuit.php"><?php echo $text['MP_CREAR']; echo $text['MP_CIR']; ?></a><hr/>
      <a href="Buscar/Bus_Circuit.php"><?php echo $text['MP_BUSCA']; echo $text['MP_CIR']; ?></a><hr/>
    </td>
    <td width="135" valign="middle">
      
    </td>
  </table>
</div>
```

Fig. 4.23. Menú Administración – Codificación apartado

4.2.2.1. Circuito

4.2.2.1.1. Crear Circuito

Esta primera opción permite crear un nuevo circuito, los datos que solicita son el nombre del circuito y el número de pruebas que puntuaran, a partir de alcanzar ese número de pruebas se empezaran a substituir los peores resultados por el nuevo resultado si corresponde.



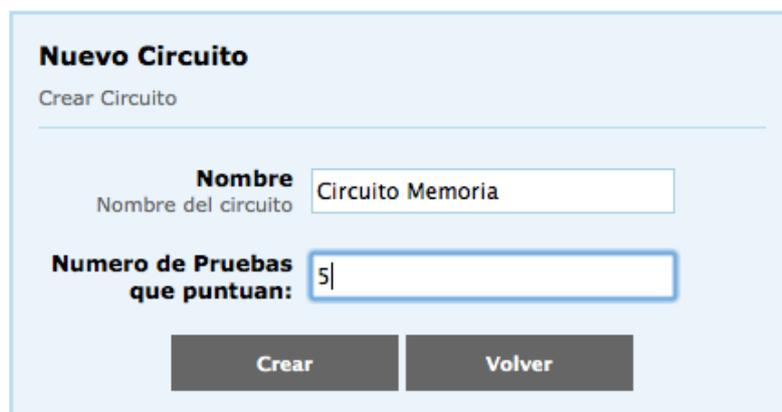
Nuevo Circuito
Crear Circuito

Nombre
Nombre del circuito

Numero de Pruebas que puntuan:

Crear **Volver**

Fig. 4.24. Circuito – Nuevo Circuito 01



Nuevo Circuito
Crear Circuito

Nombre
Nombre del circuito

Numero de Pruebas que puntuan:

Crear **Volver**

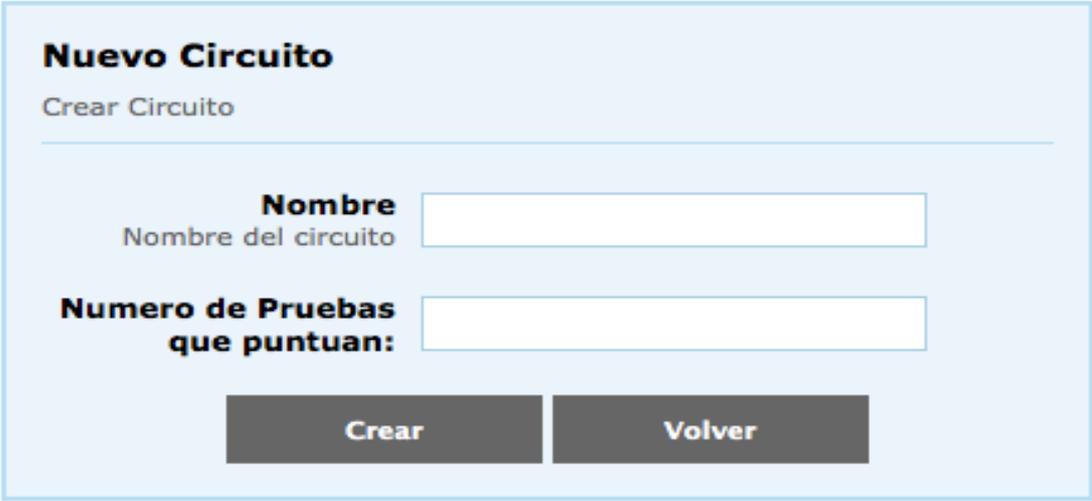
Fig. 4.25. Circuito – Nuevo Circuito 02

Una vez rellenos los dos campos se ha de apretar en el botón “Crear” para que se ejecute la rutina sql que creara el circuito en la base de datos añadiéndole el identificador único que hará más fácil la identificación futura.

```
function CrearCircuit($p_nomCircuit, $p_ppunt) {  
    $enlace = Connect();  
    if (!$enlace) { die($text['BD_ERR_CONECT'].mysql_error()); }  
    SelBBDD($enlace) or die('No puedo elegir la Base de Datos');  
  
    if (!is_numeric($p_ppunt)) { die($text['BD_ERR_NUMBER']);}  
  
    $v_aux= mysql_query("Select max(id)+1 id from Circuit",$enlace);  
    $fila  = mysql_fetch_array($v_aux);  
  
    $v_id  = $fila['id'];  
    $sql   = "REPLACE INTO Circuit values ($v_id, '$p_nomCircuit', $p_ppunt)";  
  
    $result = mysql_query($sql,$enlace);  
    mysql_close($enlace);  
  
    return $result; }  
}
```

Fig. 4.26. Circuito – Función CrearCircuit

Una vez acabada la inserción, el sistema retorna un mensaje informando que se ha creado correctamente el circuito



Nuevo Circuito
Crear Circuito

Nombre
Nombre del circuito

Numero de Pruebas que puntuan:

Crear **Volver**

Se ha creado: **Circuito Memoria**

Fig. 4.27. Circuito – Nuevo Circuito 03

4.2.2.1.1. Buscar Circuito

La segunda opción permite buscar un circuito, y una vez localizado, ver de que pruebas está formado o, si ha habido un error al crearlo, cambiarle el nombre y/o el número de pruebas que puntúan.



Buscar Circuit

Id
Codi identificador

Nom
Nom circuit

Buscar **Volver**

Fig. 4.28. Circuito – Buscar Circuito 01

Si no se introduce ningún parámetro en la búsqueda devolverá todos los circuitos que se hayan creado.



Buscar Circuit

Id
Codi identificador

Nom
Nom circuit

Buscar **Volver**

Id	Nombre del circuito	
1	SotaPar 15 aniversari	 
2	Interclubs 4div C2	 
3	Circuit Complet	 
4	Circuito Memoria	 

Fig. 4.29. Circuito – Buscar Circuito 02

Si por el contrario se busca por su identificador (id=4 en el ejemplo de la imagen 4.26) o por una parte del nombre (imagen 4.27) retornará aquellos circuitos que se adapten a la búsqueda que se está realizando.

The image shows a web form titled "Buscar Circuit" with two input fields: "Id" (labeled "Codi identificador") and "Nom" (labeled "Nom circuit"). Below the fields are two buttons: "Buscar" and "Volver". Below the form is a green table with the following content:

Id	Nombre del circuito
4	Circuito Memoria

There is a small icon of a pencil and a magnifying glass in the bottom right corner of the table.

Fig. 4.30. Circuito – Buscar Circuito 03

The image shows the same "Buscar Circuit" form as in Fig. 4.30, but with the text "Memor" entered in the "Nom" input field. The "Buscar" and "Volver" buttons are still present. Below the form is the same green table as in Fig. 4.30, showing the search results for "Circuito Memoria".

Fig. 4.31. Circuito – Buscar Circuito 04

Para entrar en la consulta del circuito, se ha de clicar sobre la lupa y se obtendrá la descripción del torneo junto con las pruebas que lo componen.

Interclubs 4div C2				
Numero de Pruebas que puntuan: 7				
Id Campo	Modalidad	Distancia	F. Inicio	F. Fin
1 P&P Vallromanes	Individual	Catalanes	19/01/14	19/01/14
2 P&P Sant Cebria	Individual	Catalanes	08/02/14	08/02/14
3 Mas Gurumbau	Individual	Catalanes	02/03/14	02/03/14
4 La Garriga Golf P&P	Individual	Catalanes	29/03/14	29/03/14
5 P&P Badalona	Individual	Catalanes	27/04/14	27/04/14
6 P&P Tennis Mora	Individual	Catalanes	24/05/14	24/05/14
7 Roc 3 P&P	Individual	Catalanes	15/06/14	15/06/14

Volver Menu Seleccion

Fig. 4.32. Circuito – Consultar circuito

Si lo que se quiere es editar los datos del circuito, se ha de clicar sobre el lápiz y se abrirá de nuevo un formulario con los datos del circuito, una vez editados se ha de pulsar en “Modificar” y el sistema informará de si se ha podido hacer la modificación correctamente.

Consultar Circuito

Modificar Circuito

Nombre
Nombre del circuito

Numero de Pruebas que puntuan:

Modificar Volver

Fig. 4.33. Circuito – Editar Circuito 01

Se ha modificado: **Circuito Memoria**

Volver Menu Seleccion

Fig. 4.34. Circuito – Editar Circuito 02

4.2.2.2. Prueba

En esta sección se podrán crear las pruebas asociándolas a los circuitos mediante el formulario de la imagen 4.35, se podrá modificar, si es que se quieren cambiar las fechas o el campo... y si se ha cancelado una prueba se podrá eliminar.



Nova Prova
Crear nova Prova

Circuit
A quin circuit correspon

Camp
A quin camp es juga

Modalitat
Tipus de prova

Mides
Mides Catalanes/EPPA

Data inici
Dia inici de competicio

Data fi
Dia fi de competicio

Fig. 4.35. Prueba – Crear Prueba

Nota: A partir de este apartado hasta el apartado de Resultados las imágenes estarán en catalán ya que es el idioma con el que se hizo la presentación de la aplicación a los clientes y hasta que no se inicie el desarrollo de estos en las nuevas fases de mejora, se decidió que no se implementara el multidioma, ya que se tenía más necesidad de estar finalizados el resto de puntos.

4.2.2.3. Campo

En esta sección se podrán crear/añadir nuevos campos a la base de datos mediante el formulario estándar para crear objetos nuevos en la base de datos. Este formulario estará compuesto por los campos que se pueden ver en la imagen 4.32; del mismo modo que en el apartado de los circuitos se implementara una opción de búsqueda de campos que permita consultar la información del campo o editarla.



Nou Camp
Donem d'Alta un nou camp on fer campionats

Nom
Nom de referencia

Forats
Camp de 9 o 18 forats

Telefon
On fer les inscripcions

Afegir **Tornar Enrera**

Fig. 4.36. Campo – Crear Campo



Buscar Camp

Id
Codi identificador

Nom
Nom camp

Buscar **Tornar Enrera**

Fig. 4.37. Campo – Buscar Campo

4.2.2.4. Equipo

En esta sección se podrá hacer la gestión de los equipos, teniendo las opciones para crear el equipo mediante un formulario estándar, modificar el equipo, fichar jugadores para el equipo-torneo y eliminar un equipo si decide no participar en el torneo.

Crear Equip
Nou Equip

Nom
Nom del equip

Crear **Tornar Enrera**

Fig. 4.38. Equipo – Crear Equipo

Buscar Equip

Id
Codi identificador

Nom
Nom equip

Buscar **Tornar Enrera**

Fig. 4.39. Equipo – Modificar Equipo

Afegir Jugador al Equip
Incorporar jugador al equip

Jugador
Llicencia jugador

Equip
Nom del equip

Circuit
Per a quin circuit

Crear **Tornar Enrera**

Fig. 4.40. Equipo – Fichar Jugador

4.2.2.5. Jugador

En esta sección se podrá hacer la gestión de los jugadores, teniendo las opciones para dar de alta a un nuevo jugador o modificación de los datos de un jugador (estas opciones las esta haciendo automáticamente la carga de resultados); también se quiere crear una opción para buscar jugadores por su hándicap.



Nou Jugador
Nou Jugador

Llicencia
ACPP*****

Nom
Nom del Jugador

Cognom1
Primer Cognom

Cognom2
Segon Cognom

HPE
Handicap Exacte

HPI
Handicap Inici

Telefon
Telefon de contacte

Mail
Correu electronic

Fig. 4.41. Jugador – Dar de alta nuevo Jugador



Buscar Jugador

Llicencia
NÂº de llicencia

Cognom1
Primer Cognom

Cognom2
Segon Cognom

Fig. 4.42. Jugador – Modificar Jugador

4.2.2.6. Modalidad

En esta sección se podrá hacer la gestión de las modalidades de juego; se crearan tres opciones para crear una nueva modalidad, modificar la modalidad o eliminarla. Esta sección junto con la de distancias son las que menos se van actualizar, llegando a decidirse que como los tipos de distancias no han variado desde la creación de la federación internacional de pitch and putt, ésta no tendría sección en el menú de administración. Desde la misma fecha no se han añadido nuevas modalidades de juego pero debido a que es posible que aparezcan de nuevas o que se inventen para algún campeonato esporádico se decidió no suprimir la sección de modalidades.



El formulario, titulado "Nova Modalitat", permite crear una nueva modalidad de juego. Incluye un subtítulo "Nova Modalitat de Joc" y dos campos de entrada: "Codi" (Codi de la modalitat) y "Modalitat" (Descripció). Al final del formulario hay dos botones: "Crear" y "Tornar Enrera".

Fig. 4.43. Modalidad – Crear nueva Modalidad de juego

4.2.2.7. Resultado

Esta sección es el motivo principal por el que se ha realizado esta aplicación web; sus dos primeras opciones de menú sirven para cargar un archivo Excel en formato (XLS) en el servidor y para leer este archivo, con el que se informará la tabla de resultados de la base de datos.

4.2.2.7.1. Introducir Resultado

Al seleccionar esta opción aparecerá la novena sección a la derecha de resultados; en ella se puede elegir el archivo Excel que se quiere subir a la aplicación, entre todos los archivos del ordenador desde el que se este conectado.



Fig. 4.44. Introducir Resultado 01

Una vez elegido el archivo aparecerá su nombre a la derecha del botón de selección (en el ejemplo Papalus.xls).



Fig. 4.45. Introducir Resultado 02

Acto seguido se cargará el archivo en el servidor de la aplicación pulsando el botón “Cargar”, si el archivo es correcto se informara de que el archivo ha sido almacenado y está listo para usarse para la importación de datos, en caso contrario se avisara de que los archivos válidos son los archivos XLS, el archivo del servidor se substituirá cada vez que se suba un archivo, por lo que si ha habido un error al seleccionar un archivo en lugar de otro es suficiente con volver a subir el archivo correcto.

Archivo cargado: Papalus.xls

Archivo guardado en: upload/result.xls

Recuerda que has de poner como hoja 1 los resultados y substituir las Ñ

Ok

Fig. 4.46. Introducir Resultado 03 – Archivo correcto

Tipo de archivo incorrecto. Ha de ser un Excel (xls)

Ok

Fig. 4.47. Introducir Resultado 04 – Archivo erróneo

Código PHP para hacer la carga del archivo al servidor:

```
<?php
include ("../idiomes.php");
include ("Login/Seguretat.php");

$allowedExts = array("xls");
$temp = explode(".", $_FILES["file"]["name"]);
$extension = end($temp);

if (($_FILES["file"]["type"] == "application/vnd.ms-excel") && in_array($extension
$allowedExts)){
    if ($_FILES["file"]["error"] > 0) { echo "Error:
".$_FILES["file"]["error"]."<br>"; }
    else
    {
        /*echo "Upload: ".$_FILES["file"]["name"]."<br>";
        echo "Type: ".$_FILES["file"]["type"]."<br>";
        echo "Size: " .($_FILES["file"]["size"]/1024). " kB<br>";
        echo "Stored in: ".$_FILES["file"]["tmp_name"].*/
```

```

        if (file_exists("upload/" . $_FILES["file"]["name"])){
            echo $_FILES["file"]["name"].$text['ERR_COPY'];
        }
        else {

            move_uploaded_file($_FILES["file"]["tmp_name"],"upload/".$_FILES["file"]["name"]);

            echo $text['MSG_LOAD'].$_FILES["file"]["name"]."<br>";
            rename("upload/".$_FILES["file"]["name"],
"upload/result.xls");
            echo
$text['MSG_SAVE']."upload/result.xls."<br>".$text['MSG_LOAD2']."<br><br>";
        }
    }
    else { echo $text['ERR_FILE']."<br><br>";}

    echo "<a href='SelePanel.php'><button type='button' style='width:70px;
height:20px'>Ok</button></a>";
?>

```

4.2.2.7.2. Cargar Resultado

Una vez se ha cargado el archivo en el servidor, se cargarán los datos en la tabla de resultados. Para poder hacer esto se ha de usar la opción de Resultados “Cargar Resultado” que activará una nueva sección a la derecha.

Resultado

[Introducir Resultado](#)

[Cargar Resultado](#)

[Consultar Resultado \(X\)](#)

[Borrar Resultado \(X\)](#)



Cargar Resultado

Circuito - Prueba:

Interclubs 4div C2 - Prueba 1 ↕

Registros: No ↕

Añadir resultados

Fig. 4.48. Cargar Resultado – Menú de selección

En esta nueva sección se podrá elegir a que circuito-prueba se va a hacer la importación de los resultados, así como elegir si queremos ver el registro de datos que se están incorporando o no (opción por defecto).

Start
Finish
Filas tratadas del Excel: 109

Fig. 4.49. Cargar Resultado – Carga sin registros

Strat
Parametro2-1

Campionat: 2
Prova: 1

Debug time: true
ACPP001423 CLOTET RIFE, JORDI -1.6 43 43
Jugador: 1
Resultado: 1

ACPP001389 PONSDOMENECH CALVET, XAVIER -5.6 42 36
Jugador: 1
Resultado: 1

ACPP008122 SANZ RAMON, ENRIC -4.4 41 37
Jugador: 1
Resultado: 1

ACPP020354 HERNANDEZ GARCIA, JUAN JOSE -0.9 40 40
Jugador: 1
Resultado: 1

Fig. 4.50. Cargar Resultado – Carga con registros 01

ACPP023967 MIRANDA QUINTERO, ANTONIO 12.6 -- --

Jugador: 1

Resultado:

ACPP024267 COROMINAS PONS, JOAN 9.7 -- --

Jugador: 1

Resultado:

ACPP010053 MARTINEZ PENDON, JOSE 1.7 -- --

Jugador: 1

Resultado:

Finish

Filas tratadas del Excel: 109

Ok

Fig. 4.51. Cargar Resultado – Carga con registros 02

Como se puede ver en las dos opciones carga los 109 jugadores que jugaron la prueba, en este caso de Papalus, en la opción “sin registros” informa únicamente de que lo ha cargado correctamente mientras que en la opción “con registros”, se puede ver para cada jugador el nuevo hándicap, el resultado scratch y el resultado hándicap de la prueba, también mostrará la información, como se muestra en la imagen 4.50, que se ha tratado la información del jugador y se ha añadido el resultado; mientras que en la imagen 4.51, que son los jugadores descalificados o retirados, se puede ver que solo ha informado los datos del jugador y el resultado no ha sido incorporado al sistema.

Para poder realizar esta carga de datos, se ha utilizado la librería “Spreadsheet” y, concretamente, la página reader.php en la cual están las funciones necesarias para poder tratar con la información de los archivos Excel.

Nombre	Fecha de modificación	Tamaño
▶ _notes	02/06/2014 20:47	--
oleread.inc	05/07/2007 23:09	8 KB
reader.php	06/04/2014 20:18	48 KB
▼ Spreadsheet	06/04/2014 10:01	--
▼ Excel	02/06/2014 20:47	--
▶ _notes	02/06/2014 20:47	--
▼ Reader	02/06/2014 20:47	--
▶ _notes	02/06/2014 20:47	--
OLERead.php	05/07/2007 23:09	8 KB
▼ Writer	02/06/2014 20:47	--
▶ _notes	02/06/2014 20:47	--
BIFFwriter.php	06/04/2014 20:14	8 KB
Format.php	06/04/2014 20:14	31 KB
Parser.php	06/04/2014 20:14	68 KB
Validator.php	06/04/2014 20:14	6 KB
Workbook.php	06/04/2014 20:14	58 KB
Worksheet.php	06/04/2014 20:14	116 KB
Writer.php	06/04/2014 20:14	3 KB

USB ALBERT ▶ PFC ▶ 9SotaPar ▶ Private ▶ Excel

Fig. 4.52. Cargar Resultado – Librería Spreadsheet

El código php que se ha usado para poder hacer la carga de los datos es el que se muestra en la siguiente figura y en el que se pueden ver los tratamientos previos que se hacen a los datos para que puedan ser insertados sin problemas en la base de datos. La única transformación que sigue sin tener aún solución es la de tratar con las “Ñ” de los nombres, por lo que se ha decidido que se substituirán por N para hacer la carga, como se puede ver en el mensaje de aviso de la imagen 4.42.

Las fases por las que pasa el tratamiento de los datos es:

1. Crear el objeto receptor de la información.
2. Indicar que codificación tendrá el archivo.
3. Leer el Excel que hay en la ubicación “upload/”.
4. Obtener el campeonato y la prueba en la que hay que informar los resultados.
5. Si han seleccionado el modo de registros, imprimir en pantalla que se ha iniciado el proceso junto con el campeonato y prueba que está informando. En caso de ser sin registros informar únicamente que se ha iniciado el proceso.
6. Abrir una nueva conexión con la base de datos.

7. Cambiar la codificación de los inserts a la base de datos a "utf8".
8. Iniciar la importación de los datos. Como las 6 primeras filas del Excel no interesan empieza a leer en la 7 que es donde se encuentran los resultados.
9. Recorrer para cada línea las 8 columnas obteniendo los diferentes campos necesarios para la importación de los resultados y la gestión de los jugadores.
10. Realizar el tratamiento de los datos.
11. Si se ha de informar de los registros, imprimir por pantalla los datos del jugador, los resultados y si se ha hecho el insert/update en la base de datos para el resultado y el jugador.
12. Imprimir que ha finalizado la importación de resultados.
13. Borrar el Excel para tener el servidor limpio.

```

<?php
include ("Login/sentencias.php");
include ("../idiomes.php");
require_once 'Excel/reader.php';

// ----- CARGAMOS EL EXCEL -----
$data = new Spreadsheet_Excel_Reader(); // lee el fichero
$data->setOutputEncoding('CP1251'); // indica el tipo de codificación
// $data->read($_FILES["file"]["name"]); // cargamos el Excel
$data->read('upload/result.xls');

$pos = strpos($_POST['campionat'],'-');
$sex_campionat = substr($_POST['campionat'],0,$pos);
$sex_prova = substr($_POST['campionat'],$pos+1);

if ($_POST['debug'] == 'true'){
    echo "Strat<br>Parametro".$_POST['campionat']."<br><br>Campionat:
".$_sex_campionat."<br>Prova: ".$_sex_prova."<br><br>";
    echo "Debug time: ".$_POST['debug']."<br>";
} else { echo "Start<br>"; }

$enlace = Connect();
if (!$enlace) { die('No pudo conectarse: ' . mysql_error()); }
$selBBDD($enlace) or die('No puedo elegir la Base de Datos');

mysql_query('SET NAMES utf8');

// ----- TRATAMOS EL EXCEL -----
for ($i = 7; $i <= count($data->sheets[0]["cells"]); $i++) {
// $variable = $data->sheets[0]["cells"][$row_number][$column_number];
    $sex_licencia = $data->sheets[0]["cells"][$i][2];
    $sex_nom = $data->sheets[0]["cells"][$i][3];
    $sex_handicap = $data->sheets[0]["cells"][$i][6];
    $sex_resultats = $data->sheets[0]["cells"][$i][7];
    $sex_resultath = $data->sheets[0]["cells"][$i][8];
}

```

```

        $p_caracters = array("&#42;","-");
        $p_enye =
array("&ntilde;","&Ntilde;","&#164","&#165");
        $sex_jugador = str_replace($p_caracters, "", $sex_nom);
        $sex_jugador = str_replace($p_enye, "", $sex_jugador);
        $p_decimal = array(",");
        $sex_hcp = str_replace($p_decimal, ".", $sex_handicap);
        $conv = array("(")" => "-");
        $sex_hcp = strtr($sex_hcp, $conv);

        if ($_POST['debug'] == 'true'){
            echo $sex_licencia." ".$sex_jugador." ".$sex_hcp." ".$sex_resultats."
".$sex_resultath."<br>";

            echo "Jugador: ".Act_Jugador($sex_licencia, $sex_jugador, $sex_hcp,
$enlace)."<br>";
            echo "Resultado: ".Add_Resultat($sex_campionat, $sex_prova,
$sex_licencia, $sex_resultats, $sex_resultath, $enlace)."<br><br>";
        } else{
            Act_Jugador($sex_licencia, $sex_jugador, $sex_hcp, $enlace);
            Add_Resultat($sex_campionat, $sex_prova, $sex_licencia, $sex_resultats
$sex_resultath, $enlace);
        }
    }
    // ----- FIN DE LA IMPORTACION -----
    $i = $i-7;
    echo "Finish<br>".$text['MSG_UPLOAD'].$i."<br>";
    mysql_close($enlace);

    // ----- BORRAMOS EL EXCEL -----
    unlink('upload/result.xls');

    echo "<a href='SelecPanel.php'><button type='button' style='width:70px;
height:20px'>Ok</button></a>";
?>

```

Fig. 4.53. Cargar Resultado – Código PHP

5. Valoración Económica

En este capítulo se detallan los costes de la creación de la aplicación web. Al final de este apartado, se muestra una tabla con todos los costes comentados en los diferentes subapartados :

5.1. Costes directos

Los costes que afectan directamente al desarrollo del proyecto se detallan en la siguiente tabla.

Tareas	Tiempo	Horas	Coste (30€/h)
Planificación y selección de herramientas	1 semana	35	1050 €
Análisis técnico	2 semanas	20	600 €
Diseño web	2 semanas	40	1200 €
Implementación casos de uso	5 semanas	400	12000 €
Revisión final	2 semanas	25	750 €
Documentación	2 semanas	40	1200 €
Viajes, reuniones y extras			150 €
TOTAL:			16950 €

Tabla 5.1. Costes directos

5.2. Amortización Software

Gran parte del software utilizado es gratuito, salvo el DreamWeaver; el alojamiento está creado en un hosting gratuito por lo que tampoco supone un coste adicional.

Teniendo en cuenta que el software se amortiza a los 24 meses y este proyecto se ha llevado a cabo en 4, amortizaremos solo un 17% del coste.

Productos	Unidades	Coste	Amortización
Hosting página web	1	0 €	0 €
Hosting base de datos	1	0 €	0 €
Herramientas control para le hosting	1	0 €	0 €
Registro en la Federación de P&P	1	0 €	0 €
Adobe Dreamweaver CS6	1	360 €	61.20 €
TOTAL:			61.20 €

Tabla 5.2. Amortización de Software

5.3. Amortización Hardware

Como hardware se considera la maquina con la que se ha programado el proyecto y como la amortización del hardware es de 48 meses, se amortizara un 12%.

Productos	Unidades	Coste	Amortización
Ordenador personal	1	1200 €	100 €
TOTAL:			100 €

Tabla 5.3. Amortización de Hardware

5.4. Resumen

Concepto	Coste
Costes directos	16 950.00 €
Amortización Software	61.20 €
Amortización Hardware	100.00 €
Subtotal:	17 111.20 €
20% costes indirectos	3 422.24 €
TOTAL:	20 533.44 €

Tabla 5.4. Resumen

6. Posibles mejoras

Las siguientes mejoras, como ya se ha comentado en diversos puntos a lo largo del proyecto, consisten en finalizar las opciones del menú de administración, ya que se tratan de tareas muy parecidas a las que actualmente están desarrolladas para la sección de Campeonatos.

Otro de los frentes abiertos es la página donde se encuentran los campos en el menú público, se está en negociaciones con otra web de la misma temática, pero que se dedica a recopilar información de los campos y se quiere enlazar con ellos, ya sea creando un link a su página o recuperando su información ya recopilada para poder acceder directamente desde la aplicación.

Se ha de preparar el sistema para poder interpretar de forma correcta los resultados de las modalidades Medal Play, y en lugar de ordenar por número de puntos descendente ordenar por número de golpes ascendente.

7. Conclusiones

El objetivo principal de la realización de este proyecto era la de aprender a programar en un entorno comercial en código PHP, así como a usar un framework de estilo para que quede más estructurada la página. También he podido realizar una pequeña incursión en el AJAX y el JQuery, aunque este segundo lenguaje ya lo había trabajado anteriormente y de forma mas amplia en las prácticas.

El proyecto fue muy ambicioso desde el primer momento y por suerte el cliente supo verlo también nada más empezar la modelización de lo que acabaría siendo la aplicación lo que nos ha ayudado en las fases de negociación para decidir que es lo que tendría mayor prioridad una vez montada la estructura de la aplicación; el problema de crear el proyecto desde cero dentro de una masa social de usuarios, es que todos querían ver avances muy rápido y se interesaban mucho por saber en que punto estaba casi cada semana la aplicación lo que ayuda en las fases más duras del proyecto tener siempre gente dándote apoyos e intentando que la cosa avance.

La experiencia nos ha satisfecho a todos, tanto al cliente como a los propios usuarios como a mi como programador/usuario, por lo que el proyecto no acabará aquí sino que seguiremos añadiendo contenido y esperemos que pueda llegar a ser un buen producto del que beneficiarnos todos.

8. Bibliografía y ayuda durante el proyecto

.
<http://www.php.net/>, Manual de PHP.

<http://www.w3schools.com/>, Manuales varios de HTML, PHP y MySQL.

<http://www.desarrolloweb.com/>, Foro y tutoriales de diversos lenguajes de programación.

Javier Hache, CEO de <http://bookemall.com>

