

# Desarrollo de un cuadro de competición de voleibol

Aleix Yagües Marín

**Resumen**—En este documento se presenta el desarrollo de una aplicación web con el propósito de servir como cuadro de competición de voleibol. El objetivo principal es permitir la creación y gestión de un torneo, con todos los atributos que este conlleva, por parte de la entidad que lo hospeda, ofreciendo una herramienta amigable, intuitiva y *responsive*. Los datos del torneo serán visibles para el público, lo que le permitirá a este poder hacer un seguimiento del torneo, consultando los clubes, los equipos, la clasificación, etc. Los árbitros de los partidos podrán hacer el seguimiento de estos, de esta manera los usuarios tendrán la posibilidad de seguir los encuentros en directo, pudiendo ver la puntuación en todo momento.

**Palabras clave**—Cuadro de competición, Desarrollo Web, Aplicación Web, Deporte

**Abstract**—In this document will be presented the development of a web application with the purpose to serve as a competition scoreboard for volleyball. The main goal is to let create and manage a tournament with all the attributes it has by the entity that hosts it, offering a friendly, intuitive and responsive tool. The data of the tournament will be available for the public, they will be allowed to trace the tournament and will be able to see clubs, teams, classification, etc. Referees will be able to control matches via the application, it will allow the public to know in real time the result of a game.

**Index Terms**—Competition scoreboard, Web development, Web Application, Sport

## 1 INTRODUCCIÓN

La aplicación desarrollada es una propuesta hecha por el Club Voleibol Sant Cugat. Esta entidad anualmente celebra un torneo en el que participan varios equipos provenientes de diferentes países. Inicialmente la gestión de la competición suponía una tarea complicada ya que, en un primer momento, no se hacía uso de una herramienta que la facilitara.

Una vez que se implantó una aplicación para ayudar a administrar mejor los resultados de los partidos, ésta proporcionó problemas que obligaron al club a volver a la metodología de gestión inicial.

Por esta razón el club decidió marcar como objetivo para la siguiente competición la implantación de una aplicación web propia, adaptada a las necesidades que tiene este, con una gestión sencilla e intuitiva con la que el torneo tenga un desarrollo más fluido y se evite la tediosa administración de los equipos y los resultados de los partidos. La herramienta también sería de utilidad para que los asistentes del torneo puedan hacer un seguimiento en directo de los partidos a los que no puedan asistir, al disputarse partidos en paralelo, y así poder conocer los resultados de los encuentros.

En este documento se recogerán los aspectos generales del proyecto, como son: los objetivos principales de la aplicación web, el estado del arte respecto a otras aplicaciones que poseen el mismo objetivo, la metodología utilizada, el

desarrollo del proyecto y las diferentes fases que ha tenido, los resultados obtenidos y las conclusiones.

## 2 OBJETIVOS

Los objetivos principales del proyecto son crear una aplicación web que sirva de herramienta para el Club Voleibol Sant Cugat con la finalidad de gestionar los torneos y proporcionar información para los usuarios interesados en este.

Después de conocer los objetivos generales estos se desglosaron en requerimientos, para proporcionar mayor exactitud respecto a las necesidades que debía cubrir la aplicación. Por lo tanto, los requerimientos fueron creados después de tener una primera reunión con la entidad y estos se vieron modificados a lo largo del proyecto, en las posteriores reuniones que se llevaron a cabo, a causa de cambios en el formato del torneo o especificaciones que no se habían tenido en cuenta en un primer momento.

TABLA 1: OBJETIVOS DEL PROYECTO

N.	Objetivo
1	Crear una base de datos consistente y eficiente valorando todas las dependencias que los objetos de la aplicación poseen.
2	Crear una aplicación web con un diseño intuitivo, amigable y <i>responsive</i> para posibilitar su utilización en dispositivos móviles.
3	Permitir crear diversos formatos de torneo para distintos números de equipos.

- E-mail de contacto: [aleix.yagues@e-campus.uab.cat](mailto:aleix.yagues@e-campus.uab.cat).
- Mención realizada: Ingeniería de Tecnologías de la Información
- Trabajo tutorizado por: Enric Vergara Carreras
- Curso 2019/20

4	Los administradores deben poder acceder a la gestión de la página web a través de un usuario.
5	Un administrador debe poder crear un torneo asignando todas las dependencias que este posee: equipos, fases y partidos
6	Un administrador debe poder crear un club asignándole una imagen.
7	Un administrador debe poder crear una categoría.
8	Un administrador debe poder crear, editar y eliminar un equipo asignándole las dependencias que este posee: club y categoría.
9	Un administrador debe poder crear, editar y eliminar un grupo definiendo el tipo de este: liga o eliminatoria.
10	Un administrador debe poder crear, editar y eliminar una fase asignándole los grupos que posee y definiendo las características de esta: número de sets, número de puntos por set, número de puntos por <i>tiebreak</i> , el último set del partido, con menos puntos, con el objetivo de desempatar, y la posibilidad de obligar a jugar el <i>tiebreak</i> .
11	Un administrador debe poder crear, editar y eliminar un partido asignándole las dependencias que posee: equipos, grupo y categoría; y las características de este: número, fecha y hora, lugar y, en caso de no estar definidos los equipos, partido previo del que estos provendrán.
12	Un administrador debe poder crear otro usuario administrador.
13	Un administrador puede editar sus propios datos de usuario.
14	Un usuario debe poder acceder a la anotación de un partido y anotarlo utilizando un código alfanumérico único para cada partido generado automáticamente al crear el partido.
15	Un usuario debe poder acceder a los diferentes torneos que se hayan disputado seleccionando la categoría concreta.
16	Un usuario debe poder consultar el listado de equipos de un torneo y categoría concretos.
17	Un usuario debe poder consultar los datos de un equipo que forme o haya formado parte de un torneo.
18	Un usuario debe poder consultar los partidos que se han llevado o se llevarán a cabo en un torneo y categoría en concreto.
19	Un usuario debe poder consultar la clasificación de un torneo o categoría en concreto.
20	Un usuario debe poder seguir en directo el resultado de un partido.

Los cambios llevados a cabo en los objetivos se encuentran en las características que poseen las fases y los grupos. En un primer momento el número de sets, número de puntos por set y número de puntos por *tiebreak* eran un atributo del torneo, un concepto general que sería igual para todos los partidos. Por otro lado, el tipo de los partidos, liga o eliminatoria, estaban definidos en las fases. Posteriormente el cliente pidió la posibilidad de

poder diferenciar como funcionarían los sets para cada fase, haciendo que en la primera fase el *tiebreak* pueda ser obligatorio. A parte, también se especificó la posibilidad de que el torneo pudiera ser de un número impar de equipos, dando resultado a fases que poseerían grupos con diferentes tipos de partido, ya que un grupo con un número impar de partidos no se puede jugar como una eliminatoria.

También fue añadida una característica a los partidos, la categoría, ya que no se había considerado el hecho de que se crearían partidos en fases adelantadas donde los equipos no estarían definidos, provocando el desconocimiento de la categoría a la que pertenecen los encuentros.

### 3 ESTADO DEL ARTE

Actualmente existe una gran variedad de aplicaciones con las que gestionar torneos de variedad de deportes. Antes de empezar el proyecto se hizo una investigación para valorar los productos que hay en el mercado y ofrecer una herramienta que no diste mucho del estándar con el que se trabaja para evitar una mayor complejidad para el usuario.

El software más utilizado en este sector son aplicaciones móviles:

- **Leverade**, una aplicación móvil con un gran número de descargas en la *Play Store* de Google ofrece la posibilidad de escoger una gran variedad de deportes, incluyendo el voleibol. También dispone de distintos formatos de torneo. La versión inicial de la aplicación es gratuita aunque al realizar un número concreto de partidos obliga al usuario a pagar mensualmente para proseguir con el torneo.
- **Winner**, una aplicación móvil gratuita que incluye publicidad, ofrece un servicio para gestionar competiciones de distintos deportes y la posibilidad de variar el formato de estas.
- **MatchApp**, una aplicación móvil utilizada por el club con el propósito de anotar los partidos y permitir al público hacer un seguimiento de los resultados de estos en directo. La herramienta generó problemas en el cambio de fase, al hacer la redistribución de equipos, esto conllevó más tiempo del esperado obligando al club a gestionar ellos mismos los resultados o a alargar la duración del torneo.

Conociendo diversas opciones que se encuentran en el mercado y valorando las necesidades que el cliente tenía, se llegó a la conclusión que este requería de un software propio hecho a medida. La herramienta debía permitir crear un torneo de voleibol con la posibilidad de generar diversos formatos de torneo, formatos propios con especificaciones particulares como albergar varias categorías o realizar una fase eliminatoria donde los equipos derrotados puedan seguir jugando entre ellos en grupos distintos. Este debía albergar distintos números de equipos, no tan solo un valor par, que sea exponente de 2. Específicamente el club pidió que se pudieran generar torneos con 12, 15 y 16 equipos, con distintos funcionamientos de torneo.

Por otro lado, también se requería la posibilidad para los asistentes del torneo de hacer un seguimiento del resultado de los partidos en directo, por lo tanto implicaba la funcionalidad de anotación de los partidos. El público también debía poder acceder a la información relativa al campeonato.

## 4 METODOLOGÍA

La metodología utilizada en el proyecto ha sido el Desarrollo adaptativo del software (ASD).

El objetivo que alberga esta metodología y, por lo tanto, el beneficio que aporta, es permitir al equipo que desarrolla la aplicación poder cambiar requerimientos o necesidades a lo largo del proyecto. La aplicación, al estar totalmente dedicada a las necesidades que posee el cliente era interesante no utilizar una metodología rígida en la que los posibles cambios pudieran afectar negativamente, y en gran medida, a la planificación y por lo tanto al resultado final.

Por esta razón la metodología ASD, que presenta un desarrollo iterativo en el que se caracterizan 3 fases:

- **Especulación:** En esta fase se conocen los requisitos básicos del proyecto y se especula sobre la posible planificación y los objetivos principales que tendrá el proyecto.
- **Colaboración:** En la segunda fase, después de haber hecho la previa especulación, conjuntamente con la parte interesada se perfilan los requisitos y objetivos que tendrá el proyecto.
- **Aprendizaje:** En la última fase del ciclo se adquiere el conocimiento después de haber implementado y probado la parte planificada para la iteración. Posteriormente, la parte funcional obtenida se entrega al cliente para valorar posibles ajustes futuros. En la siguiente iteración y a partir del trabajo realizado se volverá a especular decidiendo así el camino que se debería tomar, para seguidamente volver a la fase de colaboración y recibir el *feedback* que nos puede aportar la parte interesada.

El Desarrollo adaptativo del software permite acotar el tiempo del proyecto evitando retrasos, al hacer una planificación pesimista en la que se suponen cambios en el transcurso del proyecto.

Por otra parte, el equipo del proyecto está formado por un único integrante, este hecho hace que ciertas metodologías ágiles pierdan valor al estar centradas en gran parte a hacer una organización de los miembros del equipo para conseguir una mayor productividad con las tareas disponibles y permitiendo el mayor paralelismo posible valorando las dependencias entre ellas.

El control de versiones se ha realizado a través de GitHub para facilitar aspectos en el desarrollo como la consulta de versiones anteriores del proyecto.

## 5 PLANIFICACIÓN

En la planificación del proyecto se definieron unas fechas concretas, que al ser un trabajo de final de grado, coincidían con las diferentes entregas parciales de este.

TABLA 2: PLANIFICACIÓN DEL PROYECTO

Actividades y tareas	Fecha de entrega
<b>Estudio del proyecto</b>	
Entrevista inicial con el cliente	16/02/2020
Documento de especificaciones y requisitos del proyecto	16/02/2020
<b>Planteamiento del proyecto</b>	
Investigación sobre las herramientas disponibles	16/02/2020
Valoración de las distintas herramientas disponibles	
Estructuración de la base de datos	16/02/2020
Primera aproximación de las vistas de la aplicación web	08/03/2020
<b>Desarrollo del software</b>	
Desarrollo del back-end	19/04/2020
Implementación y comunicación con la base de datos	19/04/2020
Gestión de los equipos	19/04/2020
Gestión de los torneos	19/04/2020
Gestión de los usuarios	19/04/2020
Desarrollo del front-end	24/05/2020
<b>Fase final</b>	
Documentación	14/06/2020
Test	
Test de todos los módulos en servidor	28/06/2020
Test general	28/06/2020

Como podemos ver en la tabla 2 las fechas están muy marcadas, ya que, al tratarse de un trabajo de final de grado que posee unas fechas de entrega, estas supusieron los distintos *milestones* donde se aplicaba la parte de la entrega, la especulación y la colaboración de la metodología, recibiendo *feedback* y planeando posibles ajustes futuros.

Podemos ver que el proyecto consta de diferentes iteraciones:

- **Iteración 1:** En esta primera fase, el objetivo era coger constancia del proyecto, recoger todos los requisitos posibles y generar un documento en el cual se mostrarían ordenadamente. También se investigó sobre las mejores opciones disponibles para abordar el problema en cuestión.
- **Iteración 2:** En la segunda iteración se empezó a crear una primera aproximación a las vistas que poseería la aplicación, creando un mock up y valorándolo posteriormente con el cliente.
- **Iteración 3:** En la tercera iteración empezó el desarrollo del software, y el aprendizaje de las herramientas escogidas. Se focalizó el trabajo en el back-end, generando un front-end básico para la introducción de datos.
- **Iteración 4:** En la cuarta iteración se finalizó la parte del back-end que no se había concluido en la iteración anterior. Posteriormente el trabajo se dedicó a crear el front-end de la aplicación web.
- **Iteración 5:** En la quinta iteración se acabó de finalizar y perfilar el código.

En el listado de iteraciones anterior podemos ver que apareció un pequeño retraso en el desarrollo del software, este empezó en el desarrollo del back-end y se propagó al desarrollo del front-end. Este retraso, causado por los diferentes cambios en el código y la estructura, mencionados anteriormente, fue resuelto gracias a la planificación que poseía cierta holgura con la que se podían tratar los imprevistos sin un retraso en el proyecto entero.

## 6 DESARROLLO

El proyecto desarrollado ha sido un proyecto de tipo Maven, un tipo de proyectos con una configuración definida en ficheros XML, de esta manera las dependencias se pueden especificar fácilmente.

Las herramientas utilizadas en el proyecto han sido variadas y han requerido de un tiempo de aprendizaje. El proyecto sigue el formato MVC, modelo, vista y controlador. A continuación se comentarán las herramientas utilizadas en cada sección del proyecto.

### 6.1 Modelo

El modelo se encarga de la gestión de los datos, está directamente relacionado con la base de datos, en concreto una base de datos MySQL. La herramienta utilizada para el mapeo de los objetos y la gestión de la introducción y la recuperación de datos ha sido Hibernate, esta requirió un aprendizaje previo pero facilitó el trabajo evitando crear algunas sentencias SQL.

La imagen siguiente representa el modelo relacional y por lo tanto la estructura de la base de datos del proyecto.

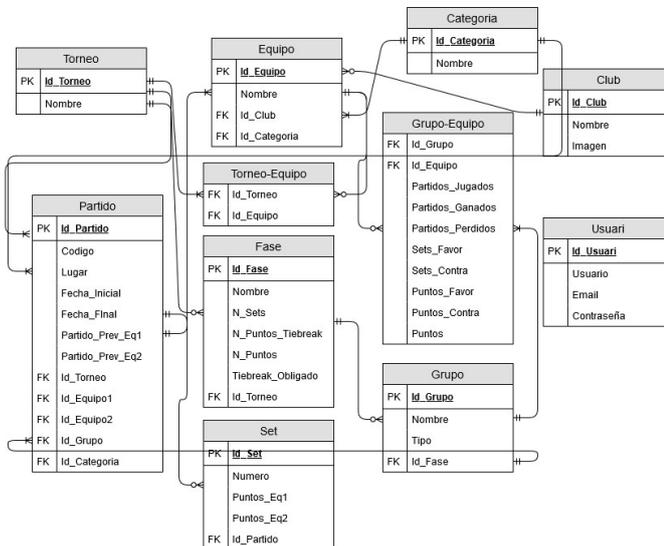


Fig. 1. Modelo relacional de la base de datos.

### 6.2 Vista

La vista es todo aquel código dedicado a la interfaz del usuario. El objetivo principal ha sido crear un software intuitivo, amigable y responsive ya que los dispositivos móviles serán aquellos los usados mayoritariamente para acceder a la aplicación por los asistentes del torneo.

Las herramientas utilizadas para el desarrollo de esta

parte han sido Spring Framework, utilizando las librerías que ofrece Spring y a partir del modelo creado en Java permite crear código HTML de forma dinámica. El proceso de conversión del código propio de Spring a HTML es llevado a cabo por el servidor. También se ha utilizado el lenguaje HTML en ciertas partes.

En relación al estilo se ha utilizado CSS y para las partes dinámicas de la aplicación se ha hecho uso de JavaScript, JQuery y AJAX.

También se han incluido Bootstrap para facilitar la edición del estilo y crear un software responsive.

Las vistas finales han sido creadas a partir de varias versiones hechas previamente para alcanzar un consenso con el cliente y evitar la modificación masiva de código dedicado a la interfaz una vez programado.

Primeramente se creó un *mock-up* inicial hecho a mano, con las principales vistas acordadas con el cliente. Posteriormente se trabajó en un *mock-up* con más detalles y con un estilo más formal con el que el cliente se pudo hacer más a la idea de cómo se había planteado la aplicación y que cosas debían modificarse.

### 6.3 Controlador

El controlador es la parte del código que se encarga de hacer la gestión entre el modelo y las vistas. El uso de Spring Framework ha simplificado la gestión de las peticiones para ofrecer las vistas.

Por otro lado, este ofrece librerías relacionadas con la seguridad y la autenticación con las que la implementación de diferentes secciones con distintos niveles de privilegio ha supuesto de una dificultad menor.

El lenguaje principal utilizado en el back-end ha sido Java, con este se ha realizado toda la lógica que envuelve al transcurso del torneo y que gestiona los diferentes formatos de torneo.

En un primer momento el número de equipos por categoría que poseería el campeonato no estaba definido, el objetivo principal era conseguir un torneo funcional de 16 equipos con las especificaciones concretas del cliente. Posteriormente, causa del desconocimiento, se consideró la posibilidad que el torneo albergara diferentes números de equipos, 12 o 15, obligando a una gestión particular del torneo.

El formato de 16 equipos se conforma de cuatro fases:

- Una primera fase formada por 4 grupos de 4 equipos. Estos grupos son de tipo liga. Al acabar la fase cada equipo habrá adquirido unos puntos y en función de estos se reasignan los equipos a los partidos de la siguiente fase.
- La segunda fase consta de 2 grupos de 8 equipos. Los grupos son de tipo eliminatoria, cada equipo disputa tan solo un partido. Al acabar cada encuentro se configuran los equipos para los partidos de la siguiente fase.
- La tercera fase consta de 4 grupos de 4 equipos. Los grupos son de tipo eliminatoria.
- Por último, la cuarta fase, consta de 8 grupos de 2 equipos, estos partidos decidirán el resultado final del torneo.

El formato de 15 equipos se conforma de cuatro fases:

- Una primera fase formada por 5 grupos de 3 equi-

pos. Estos grupos son de tipo liga. Al acabar la fase cada equipo habrá adquirido unos puntos y en función de estos se reasignan los equipos a los partidos de la siguiente fase.

- La segunda fase consta de 2 grupos, el primero tiene 8 equipos, el segundo, en cambio, tiene 6. El equipo que ocuparía un lugar en el segundo grupo pasa directamente a la tercera fase, este sería el que ocupaba el noveno lugar en la primera fase. Los grupos son de tipo eliminatoria. Al finalizarse los partidos los equipos se reasignan a los partidos de la fase siguiente.
- La tercera fase consta de 4 grupos, 3 grupos tienen 4 equipos y el último grupo de la clasificación tiene 3 equipos. Los grupos de 4 equipos son de tipo eliminatoria, el grupo de 3 equipos es de tipo liga y para los equipos que forman parte de este se convierte en la última fase.
- La cuarta fase consta de 6 grupos de 2 equipos. Los grupos son de tipo eliminatoria y el resultado de los partidos decide la clasificación final.

El formato de competición de 12 equipos queda como una funcionalidad pendiente, esta se implementará en un futuro en el proyecto.

## 7 RESULTADOS

El resultado obtenido en este proyecto es una aplicación web responsive, permitiendo la versatilidad de esta pudiéndose utilizar en diferentes dispositivos, con el objetivo de gestionar un torneo de voleibol y ofrecer la información de este de una forma sencilla e intuitiva, asegurando así el correcto y fluido desarrollo de la competición.

Primeramente se van a mostrar las vistas accesibles únicamente por los administradores:

- **Log in:** El log in es una vista que incluye un formulario, esta tiene el objetivo de permitir el registro a los usuarios administradores con un nombre de usuario y una contraseña.

El acceso a la vista será a través del botón "Log in" o en el caso de que un usuario intente acceder a una vista dedicada al administrador, y este no esté registrado, será redirigido.

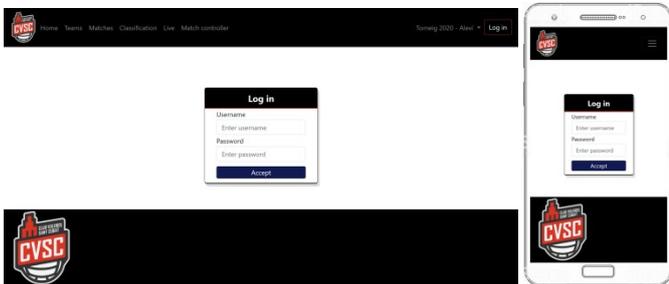


Fig. 2. Vista del Log in.

- **Crear usuario:** La vista crear usuario muestra un formulario con los campos necesarios para registrar un nuevo usuario que tendrá privilegios de administrador.

El acceso a la vista se realiza a través del menú de configuración, un menú al que se puede acceder una vez el usuario está registrado.

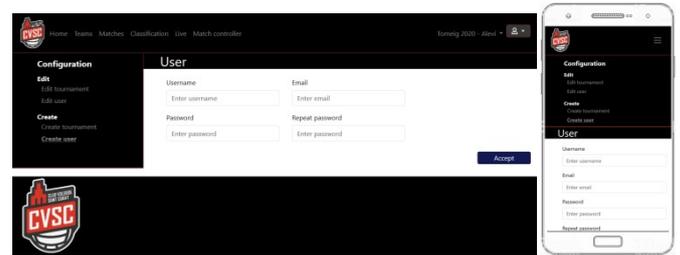


Fig. 3. Vista Crear usuario.

- **Crear torneo:** Crear torneo es una funcionalidad que es accedida a través del menú de configuración, en esta se muestran las diferentes vistas que conforman la creación de un torneo y todas las dependencias que este engloba.
  - **Torneo:** La vista torneo es el primer paso que forma la creación del torneo, en ella se encuentra un campo donde el administrador especificará el nombre del torneo a crear.



Fig. 4. Vista Torneo.

- **Crear equipo:** La vista crear equipo es el segundo paso de la creación de un torneo, en ella el administrador dispone de diversas funcionalidades.
  - Un listado con todos los equipos registrados con anterioridad en la aplicación, estos pueden ser editados o añadidos al torneo en proceso de creación. También dispone de un filtro de equipos por texto.

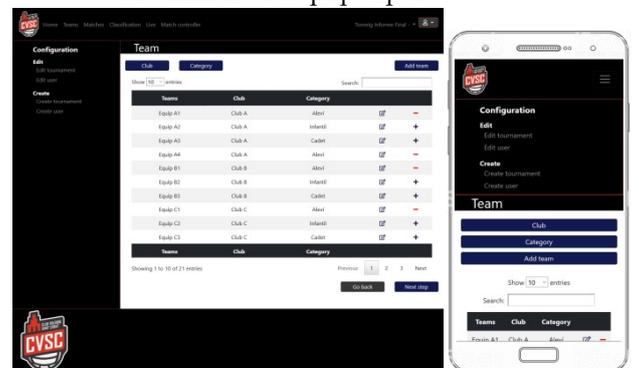


Fig. 5. Vista Crear equipo - Listado.

- En la vista del listado podemos ver diferentes botones, estos cubren varias funcionalidades, la primera de ellas, apretando el botón “Club”, es añadir un club especificando un nombre e importando una imagen que aparecerá en todos los equipos que formen parte del club.

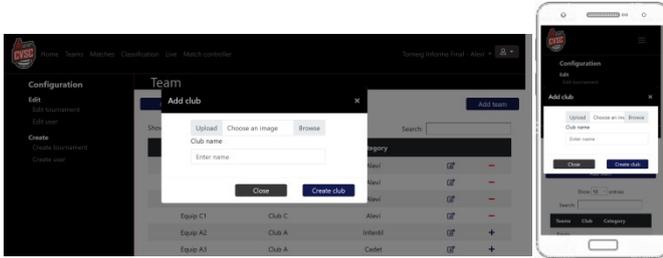


Fig. 6. Vista Crear equipo – Crear club

- La vista del listado de equipos también permite la posibilidad de crear una nueva categoría de la que podrán formar parte los equipos. A esta funcionalidad se accede pulsando el botón “Categoría”.

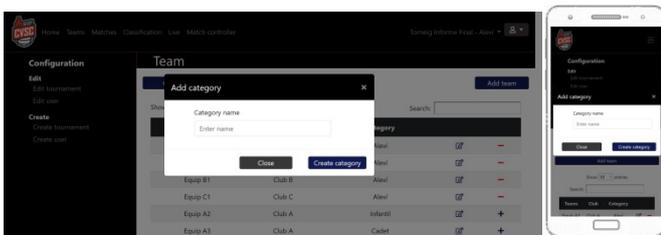


Fig. 7. Vista Crear equipo – Crear categoría.

- Por último, en la vista existe un botón con el cual crear equipos especificando el nombre de este, el club y la categoría al que forman parte.

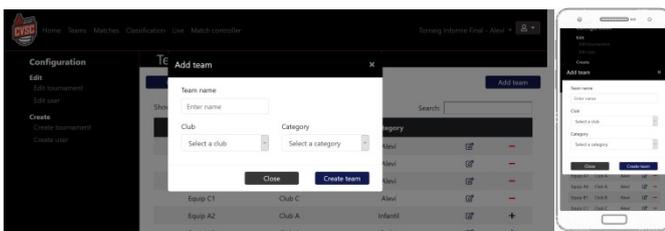


Fig. 8. Vista Crear equipo.

- Crear fase:** La vista crear fase es el tercer paso del proceso de crear un torneo. La vista posee diferentes funcionalidades:
  - La primera funcionalidad de la vista es el listado de fases creadas. Los elementos que forman el listado poseen diferentes posibilidades como ser editados, eliminados o acceder a la creación de grupos específicos para la fase concreta.

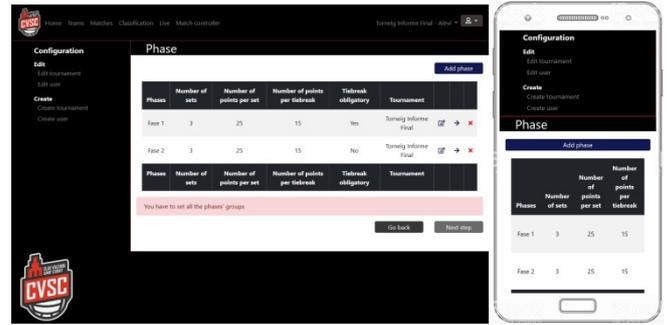


Fig. 9. Vista Crear fase - Listado.

- La segunda funcionalidad es accedida a través del botón “Crear fase”, en esta se especifican las diferentes características que posee la fase, el nombre, el número de sets por partido, el número de puntos por set, el número de puntos en el *tiebreak* y si el *tiebreak* es un set obligatorio.

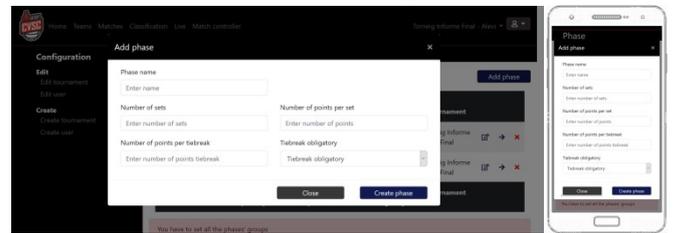


Fig. 10. Vista Crear fase.

- Crear grupo:** La vista crear grupo es accedida a través del listado de fases, encontrada en la vista crear fase, cada elemento del listado contiene una flecha con la que se accede a los grupos de la fase en concreto. La vista, como el resto de vistas mostrado hasta ahora posee diferentes funcionalidades.
  - La primera funcionalidad es mostrar un listado con los grupos creados, cada elemento del listado puede ser editado o eliminado.

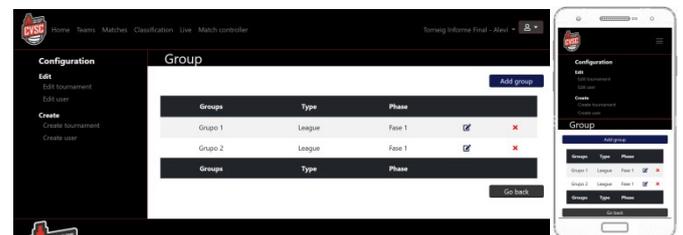


Fig. 11. Vista Crear grupo - Listado.

- La segunda funcionalidad es la de crear un grupo especificando el nombre y el tipo, liga o eliminatoria. El acceso es a través del botón “Crear equipo”.

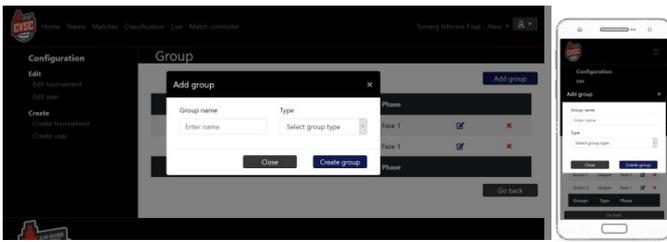


Fig. 12. Vista Crear grupo.

- **Crear partido:** La vista crear partido es el último paso de la creación de un torneo, esta vista contiene diferentes funcionalidades.
  - La primera funcionalidad que posee es la de mostrar un listado de partidos. El listado puede ser filtrado con texto, este es comparado con todos los campos que forman la tabla. Las filas, al contener mucha información pueden ser desplegadas para mostrar más detalles. Los elementos disponen de la posibilidad de ser editados o eliminados.

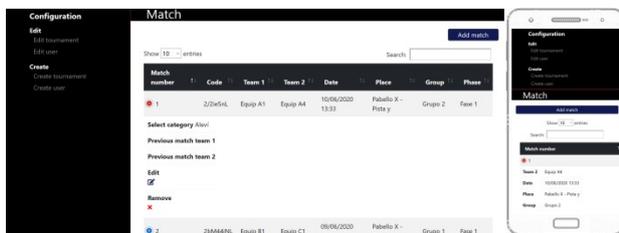


Fig. 13. Vista Crear partido – Listado.

- La segunda funcionalidad es la de crear un partido especificando todos los datos necesarios, número, equipo 1, equipo 2, fecha, categoría, lugar, grupo, partido previo 1 y partido previo 2

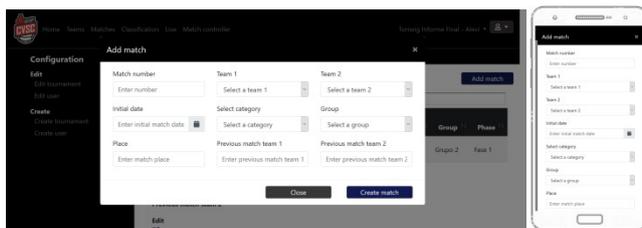


Fig. 14. Vista Crear partido.

- **Editar usuario:** El acceso a la vista se hace a través del menú de configuración. El formulario contiene los mismos campos que el que posee la vista crear usuario, la diferencia se encuentra en que este contiene los campos de nombre de usuario y email previamente rellenos con la información del usuario registrado.

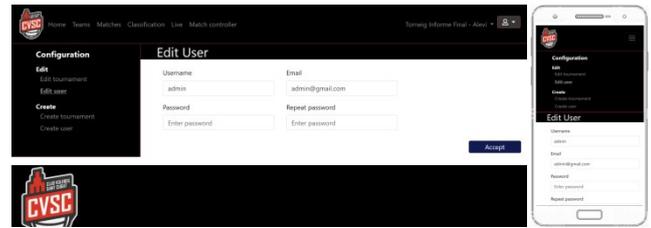


Fig. 15. Vista Editar usuario.

- **Editar torneo:** La vista editar torneo es accedida a través del menú de configuración. En ella se encuentra un listado con los torneos creados en la aplicación. Los elementos del listado disponen de un botón para ser editados con el cual se accedería al circuito de creación de torneo donde los campos estarían previamente rellenos, estos también pueden ser eliminados.

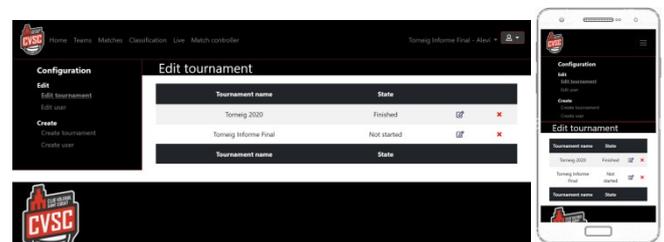


Fig. 16. Vista Editar torneo.

A continuación se van a mostrar las vistas dedicadas a los árbitros:

- **Controlar partido:** La vista para controlar un partido será la que utilizarán los árbitros o asistentes de los partidos para anotar el partido y hacer el seguimiento de este. El acceso a la vista se hace a través del menú principal de la aplicación, a través de la pestaña "Controlador de partido". La vista posee dos fases.
  - En la primera fase aparece un campo a rellenar en el cual se introduce el código del partido que se quiere anotar.

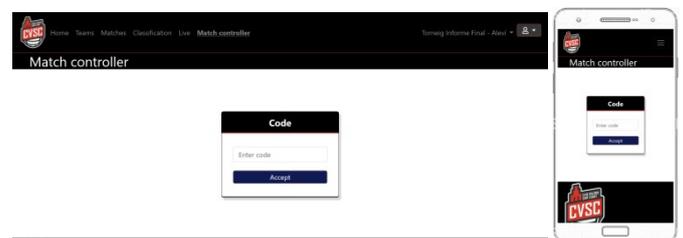


Fig. 17. Vista Controlar partido - Código.

- En la segunda fase, si el código es válido y el partido existe, aparece la interfaz con la que se hará el seguimiento del partido. En el centro de la vista aparecen botones para viajar por los distintos sets que conforman el partido, a los extremos de la vista aparecen los equipos que forman el partido con los respectivos botones de control para anotar el resultado del partido.

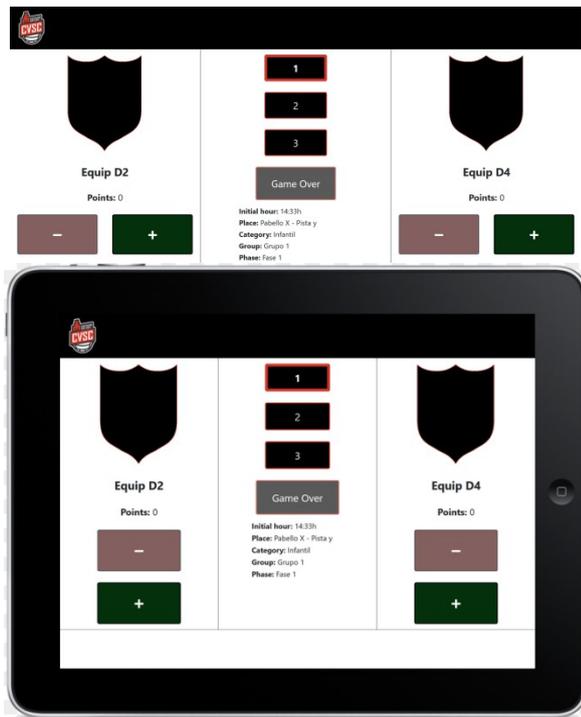


Fig. 18. Vista Controlar partido.

Seguidamente se mostraran las vistas dedicadas a informar a los asistentes de la competición.

- **Home:** La vista inicial con la que cualquier usuario se encuentra es el *Home*. En esta vista se encuentra un carrusel donde aparecen los partidos del torneo y categoría concretos, en el caso de que haya partidos en directo, aparece otro carrusel con estos encuentros. El acceso a la vista se puede realizar a través del menú principal, en la pestaña de "Home" o a través de la imagen del escudo del Club Voleibol Sant Cugat, situada en la parte superior a la izquierda de la pantalla.

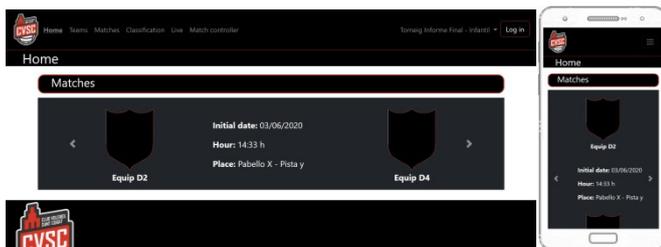


Fig. 19. Vista Home.

- **Equipos:** La vista equipos es un listado con los equipos de un torneo y una categoría concretos. El acceso a la vista se hace a través del menú principal. Los elementos de la lista, al ser pulsados, redirigen a una vista con mayor detalle del equipo seleccionado.

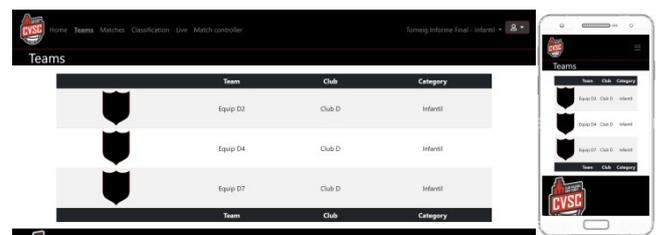


Fig. 20. Vista Equipos - Listado.

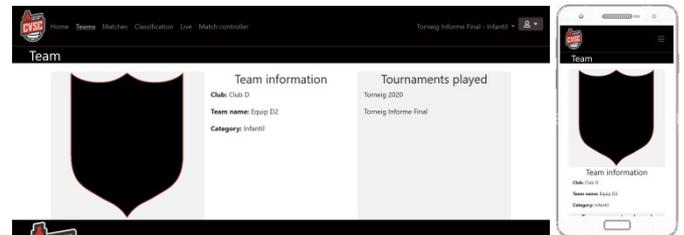


Fig. 21. Vista Equipo - Detalle.

- **Encuentros:** La vista encuentros es accedida a través del menú principal, en la pestaña "Encuentros". En la vista se encuentra un listado con los partidos del torneo y categoría concretos, definidos en la navegación. El listado está dividido por fases y grupos.

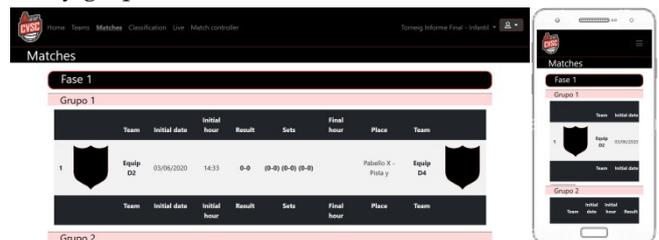


Fig. 22. Vista Encuentros.

- **Clasificatoria:** La vista clasificatoria muestra un listado de los equipos y los resultados de estos del torneo y categoría concretos seleccionados en la navegación. El listado está dividido en fases y grupos. El acceso a la vista se hace a través del menú principal.



Fig. 23. Vista Clasificatoria.

- **Live:** La vista *live* contiene los encuentros que están en directo y se muestra el resultado de estos. El resultado mostrado se actualiza de forma dinámica con AJAX, sin necesidad que los usuarios recar-

guen la página web. El acceso a la vista se encuentra en el menú principal.

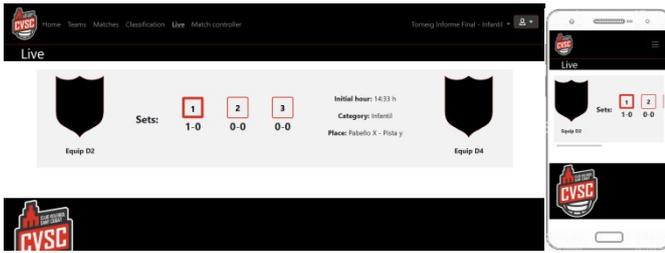


Fig. 24. Vista Live.

Por último, con el propósito de aclarar el funcionamiento de la aplicación web, a continuación se adjunta el link de un video en el cual se muestra un circuito completo: <https://youtu.be/f8qbP5gfp4>.

## 8 TEST

Una de las partes más importantes en la creación de software es el *testing*, donde se realizan las pruebas necesarias para garantizar un producto de calidad detectando y solventando errores y evitando así costes mayores en el futuro.

Actualmente, en el proyecto, se han utilizado los siguientes tipos de test:

- **Exploratory testing:** El test no toma como base unos casos concretos de prueba, a medida que se genera código se comprueba la correcta ejecución de este, verificando que la funcionalidad a la que está enfocado no contiene ningún tipo de *bug*. Las pruebas, como se ha mencionado, se han realizado de forma modular, para cada funcionalidad, a medida que se desarrollaban, a parte, también se han hecho pruebas comprobando la interacción entre los distintos módulos.
- **User testing:** El test se ha realizado con 14 personas, 8 de ellas tenían conocimientos de voleibol y 6 no habían estado relacionadas con el deporte anteriormente. En los test había 3 roles: el usuario, el facilitador y el observador. El objetivo de las pruebas propuestas al usuario era observar el comportamiento de este, como interactuaba con la herramienta intentando saber el porqué de los movimientos seguidos realizando preguntas o proponiendo que este piense en voz alta. De esta manera se pudo obtener *feedback* sobre la interfaz y el funcionamiento de la aplicación web permitiendo mejoras en todos los aspectos posibles.

## 9 CONCLUSIONES

Como conclusión del proyecto realizado, se ha podido desarrollar una aplicación web funcional, intuitiva y responsive con la que poder gestionar torneos de voleibol y permitir el seguimiento del público, ofreciendo una grata experiencia de usuario.

El proyecto ha sufrido ciertos retrasos a lo largo del proceso por diferentes dificultades que han surgido de

forma imprevista como los cambios en los requerimientos por parte del cliente, lo que ha provocado reestructuración del modelo, o errores en la planificación inicial del proyecto por falta de experiencia previa con trabajos de esta magnitud. De todas maneras se han podido solventar las dificultades gracias a la holgada planificación previa y la rápida adaptación a los cambios que se ha hecho en los momentos necesarios.

El proyecto ha supuesto un reto con el que se ha adquirido un mayor nivel de conocimiento, gracias al esfuerzo que se ha realizado en la creación de este.

El proyecto presenta líneas futuras, las cuales supondrán mejoras en la herramienta facilitando la gestión por parte de los administradores y ofreciendo mayor versatilidad a la hora de crear torneos. Existen funcionalidades con el objetivo de mejorar la aplicación que ya se están implementando, como son la posibilidad de insertar los partidos de un torneo a través de una hoja Excel o permitir la creación de un torneo con 12 equipos.

## BIBLIOGRAFÍA

- [1] Enunciat de la pràctica del MS-Project.pdf, [consulta: 14 junio 2020]
- [2] MDBootstrap.com, [consulta: 14 junio 2020], Disponible en: <https://mdbootstrap.com/docs/jquery/forms/multiselect/>
- [3] Oracle Corporation and/or its affiliates, [consulta: 14 junio 2020], <https://www.mysql.com>
- [4] The apache software foundation, [consulta: 14 junio 2020], <https://maven.apache.org/>
- [5] Getbootstrap.com, [consulta: 14 junio 2020], <https://getbootstrap.com/>
- [6] 2020 VMware, Inc. or its affiliates. [consulta: 14 junio 2020], <https://spring.io/>
- [7] Tutorials Point, [consulta: 14 junio 2020], [https://www.tutorialspoint.com/spring/spring\\_environment\\_setup.htm](https://www.tutorialspoint.com/spring/spring_environment_setup.htm)
- [8] TechTarget, [consulta: 14 junio 2020], <https://www.theserverside.com/tutorial/Spring-Without-XML-The-Basics-of-Spring-Annotations-vs-Spring-XML-Files>
- [9] TechTarget, [consulta: 14 junio 2020], <https://www.theserverside.com/tutorial/Spring-30-Tutorial-Setting-Up-Configuring-The-Environment>
- [10] DZone, Inc. dba Devada and its corporate affiliates, [consulta: 14 junio 2020], <https://dzone.com/articles/spring-security-5-form-login-with-database-provide>
- [11] Howtodoinjava.com, [consulta: 14 junio 2020], <https://howtodoinjava.com/spring5/webmvc/spring5-mvc-hibernate5-example/>
- [12] 2020 Red Hat, Inc. [consulta: 14 junio 2020], <https://hibernate.org/orm/documentation/5.4/>
- [13] Journaldev.com, [consulta: 14 junio 2020], <https://www.journaldev.com/2934/hibernate-many-to-many-mapping-join-tables>
- [14] MvnRepository, [consulta: 14 junio 2020], <https://mvnrepository.com/>
- [15] Fonticons, Inc. , [consulta: 14 junio 2020], <https://fontawesome.com/>