



# FootCoach

**Alumno: Jesús Cela Boisa**

Máster Universitario en Desarrollo de Aplicaciones para Dispositivos Móviles

**Consultor: Francesc D'Assís Giralt Queralt**

**Director de la asignatura: Carles Garrigues Olivella**

Junio 2018



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-

SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	FootCoach
<b>Nombre del autor:</b>	Jesús Cela Boisa
<b>Nombre del consultor:</b>	Francesc D'Assís Giralt Queralt
<b>Fecha de entrega (mm/aaaa):</b>	06/2018
<b>Titulación:</b>	<i>Máster Universitario en Desarrollo de Aplicaciones para Dispositivos Móviles</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b>	
<p>En este trabajo de final de máster se propone la creación de una aplicación para dispositivos móviles Android que permita a los entrenadores de equipos de fútbol realizar una planificación y gestión más ágil de las sesiones de entrenamiento de su equipo.</p> <p>Esta idea nace de la falta de informatización en este sector, que obliga a los entrenadores a realizar de forma manual una gran cantidad de trabajo previo y posterior a los entrenamientos y partidos de fútbol, lo que conlleva un considerable esfuerzo para llevar a cabo una correcta gestión del equipo.</p> <p>Mediante este trabajo se pretende obtener una aplicación móvil que cumpla con los objetivos propuestos y poner en práctica los conocimientos adquiridos a lo largo de todo el máster.</p>	
<b>Abstract (in English, 250 words or less):</b>	
This master's degree final project will cover the development process of a mobile application for Android devices, whose purpose will be to allow football	

team coaches to plan, prepare and manage training sessions in a more agile manner.

This idea originated from observing the lack of proper computerized systems for this purpose, which forces coaches to do large amounts of work manually before and after training sessions and football matches. This in turn leads to greater difficulty and effort required in order to properly manage their teams.

The project's objective is to create a mobile application that fulfills the goals stated and to apply the knowledge and skills acquired throughout the course.

**Palabras clave (entre 4 y 8):**

Fútbol, Android, aplicación, móvil, entrenamiento.

# Índice

<b>1. INTRODUCCIÓN</b> .....	<b>1</b>
1.1. CONTEXTO Y JUSTIFICACIÓN DEL TRABAJO .....	1
1.1.1. <i>Estudio de mercado</i> .....	3
1.2. OBJETIVOS DEL TRABAJO .....	4
1.3. ENFOQUE Y MÉTODO SEGUIDO.....	4
1.3.1. <i>Estrategia a seguir</i> .....	5
1.3.2. <i>Metodología de desarrollo</i> .....	5
1.3.3. <i>Plataforma y tecnologías</i> .....	6
1.4. PLANIFICACIÓN DEL TRABAJO .....	7
1.5. BREVE SUMARIO DE PRODUCTOS OBTENIDOS .....	9
1.6. BREVE DESCRIPCIÓN DE LOS OTROS CAPÍTULOS DE LA MEMORIA .....	10
<b>2. ANÁLISIS</b> .....	<b>11</b>
2.1. DEFINICIÓN DEL USUARIO Y CONTEXTO DE USO. ....	11
2.1.1. <i>Identificación de grupo de usuarios</i> .....	11
2.1.2. <i>Fichas de personas y escenarios</i> .....	12
2.2. CASOS DE USO.....	15
2.2.1. <i>Descripción de los actores y diagrama de casos de uso</i> .....	16
2.2.2. <i>Especificación de casos de uso</i> .....	16
2.3. REQUISITOS DE LA APLICACIÓN. ....	28
2.3.1. <i>Requisitos funcionales</i> .....	30
2.3.2. <i>Requisitos de usabilidad</i> .....	38
2.3.3. <i>Requisitos de interfaz</i> .....	39
2.3.4. <i>Requisitos de software</i> .....	39
2.3.5. <i>Requisitos de diseño e implementación</i> .....	40
2.4. MATRIZ DE TRAZABILIDAD.....	40
<b>3. DISEÑO</b> .....	<b>42</b>
3.1. DISEÑO DEL PROTOTIPO DE LA APLICACIÓN.....	42
3.1.1. <i>Pantalla de Login</i> .....	42

3.1.2.	<i>Pantalla de inicio.</i>	43
3.1.3.	<i>Gestión de jugadores.</i>	44
3.1.4.	<i>Gestión de ejercicios</i>	45
3.1.5.	<i>Gestión de sesiones de entrenamiento.</i>	46
3.1.6.	<i>Gestión de partidos</i>	47
3.2.	DISEÑO DE LA BASE DE DATOS.	49
3.3.	DISEÑO DE LA ARQUITECTURA.	50
3.3.1.	<i>Patrón MVC vs patrón MVP.</i>	50
3.3.2.	<i>Persistencia de los datos.</i>	51
<b>4.</b>	<b>IMPLEMENTACIÓN.</b>	<b>52</b>
4.1.	ENTORNO DE DESARROLLO.	52
4.2.	DECISIONES Y CAMBIOS EN EL DISEÑO.	52
4.2.1.	<i>Portada.</i>	52
4.2.2.	<i>Detalles de los ejercicios.</i>	53
4.2.3.	<i>Asociar jugadores y ejercicios a los entrenamientos.</i>	53
4.2.4.	<i>Gestión de los partidos.</i>	54
4.2.5.	<i>Estadísticas del jugador.</i>	54
4.3.	DECISIONES SOBRE LA FUNCIONALIDAD.	55
4.3.1.	<i>Registro y autenticación del usuario.</i>	55
4.3.2.	<i>Validación de los formularios.</i>	55
4.3.3.	<i>Tratamiento de los campos fecha y hora.</i>	56
4.3.4.	<i>Reutilización de código mediante Fragments.</i>	56
4.3.5.	<i>Almacenamiento de los datos.</i>	57
4.4.	ESTRUCTURACIÓN DEL CÓDIGO.	59
4.5.	PROBLEMAS ENCONTRADOS.	61
<b>5.</b>	<b>VALIDACIÓN Y PRUEBAS.</b>	<b>62</b>
5.1.	PRUEBAS.	63
5.2.	VALIDACIÓN DE REQUISITOS.	70
<b>6.</b>	<b>CONCLUSIONES.</b>	<b>79</b>
6.1.	GRADO DE CUMPLIMIENTO DE LOS OBJETIVOS PROPUESTOS.	79
6.2.	SEGUIMIENTO DE LA PLANIFICACIÓN.	80

6.3.	LÍNEAS DE TRABAJO FUTURO .....	80
7.	GLOSARIO .....	82
8.	BIBLIOGRAFÍA .....	83
9.	ANEXOS.....	86
9.1.	ANEXO A: INSTRUCCIONES DE COMPILACIÓN.....	86
9.2.	ANEXO B: MANUAL DE USUARIO.....	86

## Lista de ilustraciones

Ilustración 1: Planificación temporal general	7
Ilustración 2: Carga trabajo general recurso	8
Ilustración 3: Gantt tareas PEC1	8
Ilustración 4: Gantt tareas PEC2	8
Ilustración 5: Gantt tareas PEC3	8
Ilustración 6: Gantt tareas Entrega Final	8
Ilustración 7: Carga temporal de recurso por tarea específica	9
Ilustración 8: Diagrama de casos de uso.	16
Ilustración 9: Pantalla de login.	42
Ilustración 10: Pantalla de inicio.	43
Ilustración 11: Gestión de jugadores.	44
Ilustración 12: Gestión de ejercicios.	45
Ilustración 13: Gestión de sesiones de entrenamiento.	46
Ilustración 14: Gestión de partidos.	47
Ilustración 15: Diagrama Entidad-Relación.	49
Ilustración 16: Patrones MVC y MVP.	50
Ilustración 17: Cambios portada.	53
Ilustración 18: Cambios detalles entrenamiento.	54
Ilustración 19: Cambios detalles del jugador.	55
Ilustración 20: Estructura paquetes MVP.	59
Ilustración 21: Paquete firebase.	60
Ilustración 22: Pantalla registro.	86
Ilustración 23: Pantalla de Login.	87
Ilustración 24: Pantalla de Home.	87
Ilustración 25: Pantalla ejercicios 1.	88
Ilustración 26: Crear nuevo ejercicio.	89
Ilustración 27: Añadir nuevo jugador.	89
Ilustración 28: Crear nuevo entrenamiento.	90



Ilustración 29: Añadir ejercicios y jugadores al entrenamiento.	91
Ilustración 30: Próximos entrenamientos.	91
Ilustración 31: Generar nuevo entrenamiento.	92
Ilustración 32: Entrenamiento generado automáticamente por el sistema.	92
Ilustración 33: Nuevo partido.	93
Ilustración 34: Detalles del partido.	93
Ilustración 35: Detalles del jugador.	94
Ilustración 36: Cerrar sesión.	94

## Lista de tablas

Tabla 1: Esquema ficha persona.....	12
Tabla 2: Ficha de Iván.....	13
Tabla 3: Escenario de uso 1.....	14
Tabla 4: Escenario de uso 2.....	14
Tabla 5: Esquema caso de uso.....	15
Tabla 6: CU-1: Actualizar ejercicios.....	18
Tabla 7: CU-2: Actualizar jugadores.....	19
Tabla 8: CU-3: Actualizar sesiones de entrenamiento.....	20
Tabla 9: CU-4: Actualizar partidos.....	21
Tabla 10: CU-5: Listar.....	22
Tabla 11: CU-6: Generar sesión de entrenamiento.....	23
Tabla 12: CU-7: Recopilar datos del partido.....	24
Tabla 13: CU-8: Generar convocatoria.....	25
Tabla 14: CU-9: Ver estadísticas jugador.....	25
Tabla 15: CU-10: Definir preferencias.....	26
Tabla 16: CU-11: Registrarse.....	27
Tabla 17: CU-12: Loguearse.....	27
Tabla 18: Nomenclatura identificación de requisitos.....	29
Tabla 19: Plantilla de requisitos.....	30
Tabla 20: RF-1: Crear ejercicio.....	30
Tabla 21: RF-2: Modificar ejercicio.....	30
Tabla 22: RF-3: Eliminar ejercicio.....	31
Tabla 23: RF-4: Crear sesión de entrenamiento.....	31
Tabla 24: RF-5: Modificar sesión de entrenamiento.....	31
Tabla 25: RF-6: Eliminar sesión de entrenamiento.....	32
Tabla 26: RF-7: Crear jugador.....	32
Tabla 27: RF-8: Modificar jugador.....	32
Tabla 28: RF-9: Eliminar jugador.....	33

Tabla 29: RF-10: Crear partido.....	33
Tabla 30: RF-11: Modificar partido.....	33
Tabla 31: RF-12: Eliminar partido.....	34
Tabla 32: RF-13: Listar ejercicios.....	34
Tabla 33: RF-14: Listar sesiones de entrenamiento.....	34
Tabla 34: RF-15: Listar jugadores.....	35
Tabla 35: RF-16: Listar partidos.....	35
Tabla 36: RF-17: Guardar preferencias del usuario.....	35
Tabla 37: RF-18: Generar sesión de entrenamiento.....	36
Tabla 38: RF-19: Registrar usuario.....	36
Tabla 39: RF-20: Login del usuario.....	36
Tabla 40: RF-21: Generar fichero de texto.....	37
Tabla 41: RF-22: Guardar datos asociados a un partido.....	37
Tabla 42: RF-23: Consultar y resumir datos de un jugador.....	37
Tabla 43: RUsab-1: Formularios sencillos.....	38
Tabla 44: RUsab-2: Asistencia al usuario.....	38
Tabla 45: RIfaz-1: Contraste de colores.....	39
Tabla 46: RSoft-1: Plataforma Android 4.1 o superior.....	39
Tabla 47: RDis-1: Utilización de servicios Firebase.....	40
Tabla 48: Matriz de trazabilidad.....	41
Tabla 49: Plantilla especificación pruebas.....	62
Tabla 50: Pr-1: Crear ejercicio.....	63
Tabla 51: Pr-2: Modificar ejercicio.....	63
Tabla 52: Pr-3: Eliminar ejercicio.....	64
Tabla 53: Pr-4: Crear entrenamiento.....	64
Tabla 54: Pr-5: Modificar entrenamiento.....	64
Tabla 55: Pr-6: Asociar ejercicios a entrenamiento.....	64
Tabla 56: Pr-7: Asociar jugadores a entrenamiento.....	65
Tabla 57: Pr-8: Eliminar entrenamiento.....	65
Tabla 58: Pr-9: Crear jugador.....	65
Tabla 59: Pr-10: Modificar jugador.....	66
Tabla 60: Pr-11: Eliminar jugador.....	66
Tabla 61: Pr-12: Crear partido.....	66

Tabla 62: Pr-13: Modificar partido. ....	67
Tabla 63: Pr-14: Asociar jugadores a partido. ....	67
Tabla 64: Pr-15: Eliminar partido. ....	67
Tabla 65: Pr-16: Listar ejercicios. ....	68
Tabla 66: Pr-17: Listar entrenamientos. ....	68
Tabla 67: Pr-18: Listar jugadores. ....	68
Tabla 68: Pr-19: Listar partidos. ....	69
Tabla 69: Pr-20: Generar entrenamiento automático. ....	69
Tabla 70: Pr-21: Registrar usuario. ....	69
Tabla 71: Pr-22: Loguear usuario. ....	70
Tabla 72: Pr-23: Ver detalles del jugador. ....	70
Tabla 73: Plantilla especificación de validación de requisitos. ....	71
Tabla 74: Validación RF-1 ....	71
Tabla 75: Validación RF-2 ....	71
Tabla 76: Validación RF-3 ....	72
Tabla 77: Validación RF-4 ....	72
Tabla 78: Validación RF-5 ....	72
Tabla 79: Validación RF-6 ....	72
Tabla 80: Validación RF-7 ....	73
Tabla 81: Validación RF-8 ....	73
Tabla 82: Validación RF-9 ....	73
Tabla 83: Validación RF-10 ....	73
Tabla 84: Validación RF-11 ....	74
Tabla 85: Validación RF-12 ....	74
Tabla 86: Validación RF-13 ....	74
Tabla 87: Validación RF-14 ....	74
Tabla 88: Validación RF-15 ....	75
Tabla 89: Validación RF-16 ....	75
Tabla 90: Validación RF-17 ....	75
Tabla 91: Validación RF-18 ....	75
Tabla 92: Validación RF-19 ....	76
Tabla 93: Validación RF-20 ....	76
Tabla 94: Validación RF-21 ....	76

Tabla 95: Validación RF-22 .....	76
Tabla 96: Validación RF-23 .....	77
Tabla 97: Validación RUsab-1 .....	77
Tabla 98: Validación RUsab-2 .....	77
Tabla 99: Validación Rfz-1 .....	77
Tabla 100: Validación RSoft-1 .....	78
Tabla 101: Validación RDis-1 .....	78

# 1. Introducción

## 1.1. Contexto y justificación del Trabajo

En los orígenes del fútbol base, todo constaba de un mero entretenimiento para los chicos que se apuntaban a un equipo y para los entrenadores, que normalmente eran jugadores de los equipos sénior y/o los propios padres de los jugadores, a los que les entusiasmaba la idea de entrenar por hobby.

Todo ha ido evolucionando hacia una profesionalización del fútbol base, tanto en recursos económicos (cuotas a los padres de los niños por jugar, búsqueda de patrocinadores, etc.), necesarios para poder costear la compra de material deportivo y equipaciones, como en la formación del cuerpo técnico. A pesar de ello, los entrenadores han sido la última estación en la profesionalización del fútbol base.

Con la aparición de los cursos de entrenador de fútbol de nivel 1, 2 y 3, y posteriormente de los cursos de monitor de fútbol base, preparador físico y entrenador de porteros, se ha comenzado a dotar al cuerpo técnico de los equipos de los conocimientos necesarios para ofrecer una formación de calidad. En ese proceso formativo, económicamente hablando, todos los recursos se destinan normalmente a la compra de material deportivo (picas, setas, petos, balones, vallas...), que son fundamentalmente aprovechados por los jugadores para mejorar su rendimiento y capacidades, pero prácticamente ninguna vez se destina parte de ese presupuesto a la compra o inversión en material y/o tecnologías que facilite el trabajo previo al entrenamiento y a los partidos, a los entrenadores y el cuerpo técnico.

Este hecho implica un gran trabajo por parte del cuerpo técnico en las diferentes tareas asociadas a este. Planificar y gestionar los entrenamientos, recopilación de datos con el fin de obtener estadísticas de su equipo, preparación de la táctica de los partidos, etc.

De esta necesidad, surgió la idea de algunos formadores de introducir la informática en el fútbol base (hablando siempre de clubes que no tienen los presupuestos millonarios necesarios para invertir en hardware y software específico para la gestión deportiva) utilizando los medios disponibles tales como, hojas de cálculo, procesadores de texto, etc.

Puesto que estas tecnologías no están especializadas, precisan de un lento proceso de aprendizaje, y requieren de mucha intervención por parte del cuerpo técnico, la idea es ir un paso más allá y automatizar la gran mayoría de los procesos de gestión deportiva de un club de fútbol base, aportando soluciones rápidas y simples que facilite el trabajo a los entrenadores, aporte estadísticas al cuerpo técnico y que en general ahorre tiempo en todos los procesos.

Pudiendo dividir el proyecto en tres áreas claramente diferenciadas (Jugadores, Entrenamientos y Estadísticas), concentraremos el trabajo previo que deba realizar el entrenador en la parte de creación y configuración de los entrenamientos utilizando una interfaz lo más limpia y clara posible. Tras esa intervención inicial, deberá ser la aplicación la que gestione los entrenamientos, ofreciendo sesiones creadas a partir de los ejercicios existentes, y permitiendo la modificación por parte del cuerpo técnico si lo cree oportuno.

La segunda parte será la creación de los jugadores, realizada por el cuerpo técnico, que da lugar a la relación entre las sesiones de entrenamiento y los jugadores entrando en escena las estadísticas en la aplicación. Estas deben generarse sin intervención del entrenador, pero deberán ofrecer información útil para poder tomar decisiones a presente y futuro sobre la planificación deportiva (asistencia a entrenamientos, rendimiento en los ejercicios, etc.).

### 1.1.1. Estudio de mercado

Tras realizar una encuesta a algunas personas de la profesión a la cuál va dirigida la aplicación a desarrollar, el 100% de los resultados fueron positivos.

Todos los encuestados creen que de ser una aplicación gratuita o de tener un precio ajustado, sería una solución que aportaría un ahorro de tiempo importante al mismo tiempo que ayudaría con la planificación deportiva.

Una mayoría de los encuestados afirmó que presentarían a su club la aplicación para que la incluyesen en todas las categorías.

Si acudimos a la tienda de aplicaciones de Android podemos encontrar un gran número de aplicaciones relacionadas con el tema, pero que en ningún caso resuelven el problema en cuestión, ya que se limitan a ofrecer un gran catálogo de posibles ejercicios que incluir en el entrenamiento pero ninguna de ellas permite automatizar este proceso.

Algunos ejemplos de estas aplicaciones son:

1. **Entrenamiento de fútbol gratis (Android):** aplicación que incluye gran variedad de vídeos de ejercicios de fútbol. [1]
2. **Ejercicios de fútbol (Android):** ofrece un gran número de ejercicios de fútbol y formas de juego. [2]
3. **Ejercicios Fútbol Base (Android e iOS):** esta aplicación ofrece un amplio abanico de ejercicios de fútbol clasificados por edad, categorías, etc. [3,4]
4. **Entrenamiento de fútbol (iOS):** aplicación que ofrece diferentes metodologías de entrenamiento a base de diferentes ejercicios. [5]
5. **El entrenador de fútbol (iOS):** aplicación que permite crear entrenamientos en base a unos ejercicios predefinidos, se da la opción de obtener paquetes de entrenamientos ya diseñados pagando. [6]



Echando un vistazo al mercado de aplicaciones móviles Android e iOS y teniendo en cuenta los resultados de la encuesta, se observa que estos no ofrecen ninguna aplicación que dé solución al problema planteado. Por lo tanto, aunque el nicho de usuarios potenciales de la aplicación es más reducido de lo habitual al tratarse de una aplicación tan enfocada hacia un sector, sobran razones para creer que la aplicación podría ser un éxito.

## **1.2. Objetivos del Trabajo**

El principal objetivo de este trabajo es la creación de una aplicación móvil Android que facilite las tareas y ahorre tiempo al cuerpo técnico de los equipos de fútbol.

Por otro lado el cuerpo técnico dispondrá de estadísticas recopiladas por la aplicación que les ayude a tomar decisiones sobre su equipo.

Para cumplir con los objetivos anteriores la aplicación debe cumplir con estos objetivos en específico:

1. Permitir identificarse a los usuarios a través de un sistema de login.
2. Poder añadir ejercicios de entrenamiento de forma simple.
3. Ofrecer al usuario la posibilidad de planificar sesiones de entrenamiento de la forma más automática posible.
4. Permitir al usuario gestionar los datos de los jugadores del equipo.
5. Gestión de la convocatoria y recopilación de datos durante el partido.

## **1.3. Enfoque y método seguido**

En esta sección se describirá la estrategia a seguir y la metodología utilizada para el desarrollo de la aplicación. También se definirá la plataforma para la cual estará destinada y que tecnologías se emplearán para su desarrollo.

### 1.3.1. Estrategia a seguir.

Tras analizar el mercado de aplicaciones se decide crear una nueva aplicación desde el principio ya que ninguna de las aplicaciones del mercado actual ofrece una solución alternativa o busca un fin similar para el problema que se pretende resolver. Dado que las características de la aplicación que se desea obtener son totalmente innovadoras en este mercado obliga a partir de un proyecto totalmente nuevo.

### 1.3.2. Metodología de desarrollo.

En el panorama actual de gestión de proyectos informáticos existen diferentes metodologías de desarrollo que se adaptan a diferentes situaciones y condiciones según el proyecto.

Algunas de estas metodologías son:

1. **Modelo en cascada:** es el método clásico de desarrollo que consiste en realizar las fases del desarrollo de forma lineal, análisis, diseño, implementación, pruebas y mantenimiento. Requiere de que los requisitos estén claros desde la primera fase del proyecto y los cambios presentan un gran costo a nivel de trabajo [7,8]
2. **Modelo en espiral:** este modelo añade al modelo en cascada una gestión de los riesgos de manera que el desarrollo del proyecto se gestione secuencialmente en función de los riesgos analizados. Generalmente se emplea para proyectos de gran tamaño y son muy costosos. [9,10]
3. **Metodologías ágiles:** estas metodologías se basan en el desarrollo rápido de diferentes iteraciones con parte de la funcionalidad de la aplicación final para entregar al cliente, con el que posteriormente se decidirá que nuevas funcionalidades añadir. Dentro de estas metodologías se puede optar por el desarrollo de prototipos, incrementos, la programación extrema, etc. [11,12]

Dado que en el caso de esta aplicación se dispone de un pequeño grupo de usuarios que pueden aportar ideas durante la fase de análisis y diseño y

posteriormente colaborar en la fase de desarrollo y pruebas, me he decantado por aplicar una metodología ágil basada en el desarrollo iterativo por incrementos. [13]

Mediante el desarrollo por incrementos podré establecer unos requisitos funcionales iniciales que se podrán ir refinando a lo largo de las diferentes entregas o etapas del desarrollo. Es una buena opción al tratarse de un equipo de trabajo compuesto de una única persona y al disponer de un pequeño grupo de usuarios de prueba que podrían jugar el papel de cliente.

### **1.3.3. Plataforma y tecnologías.**

Tras realizar el estudio de mercado se opta por desarrollar la aplicación móvil para dispositivos Android ya que ocupan una mayor cuota de mercado en Europa y son muy asequibles, puede que por esto acaparen gran parte del mercado. [14,15]

Para llevar a cabo el desarrollo se dispondrá de los siguientes recursos de hardware:

- Ordenador portátil con Windows 10.
- Teléfono móvil Xiaomi Redmi 4 PRO.
- Tablet Nexus 9.

Serán necesarios los siguientes recursos software para el desarrollo:

- Android Studio 3.0 como IDE para programar
- Microsoft Windows Word para la redacción de la memoria
- ProjectLibre para realizar la planificación temporal y de recursos humanos.
- StarUML para realizar los diagramas de diseño de la aplicación.

Las tecnologías utilizadas serán:

- Kotlin como lenguaje de programación de Android nativo.
- SQLite como gestor de base de datos local de la aplicación

- Firebase Realtime Database como base de datos remota.
- UML para la realización de los diagramas de diseño.

## 1.4. Planificación del Trabajo

A continuación se detalla a través de un diagrama de Gantt la planificación temporal del proyecto. Para ello se debe realizar un pequeño estudio de la carga de trabajo que se estima que puede tener el trabajo de final de máster.

Partiendo de que el proyecto de fin de máster consta de 12 créditos, y valorando la carga de trabajo de las asignaturas realizadas en otros semestres se le asignará a cada crédito una carga de trabajo temporal de 21 horas.

Para una planificación inicial tomando estos valores como “buenos” este proyecto requerirá de un trabajo total de 252 (12 x 21) horas repartidas a lo largo de 76 días laborales. A estos 76 días les descontaremos los días festivos quedando un total de 71 días de trabajo. Con todo lo anterior, la carga de trabajo diario será de 3,55 (252 / 71) horas. Sabiendo la carga de trabajo temporal diaria se puede realizar una estimación temporal en horas de cada una de las PEC de las que consta el trabajo.

La planificación temporal general de cara a las PEC se muestran en la siguiente imagen:





		Nombre	Duracion	Inicio	Terminado
1		PEC1-Plan de trabajo	36,571 days	21/02/18 8:00	14/03/18 17:00
2		PEC2-Diseño	34,286 days?	15/03/18 8:00	6/04/18 17:00
3		PEC3-Implementación	68,571 days?	9/04/18 8:00	23/05/18 17:00
4		Entrega Final	22,857 days?	24/05/18 8:00	6/06/18 17:00

Ilustración 1: Planificación temporal general

La duración de las tareas está indicado en número de días naturales, esta duración se puede apreciar en la siguiente imagen, asignada al recurso ‘Jesús Cela Boisa’ con una carga de trabajo diario de 3,55 horas y descontando los días festivos y fines de semana:

	Nombre	Trabajo
1	Jesús Cela Boisa	252,05 horas
	PEC1-Plan de trabajo	56,8 horas
	PEC2-Diseño	53,25 horas
	PEC3-Implementación	106,5 horas
	Entrega Final	35,5 horas

Ilustración 2: Carga trabajo general recurso

Ahora se desglosarán las cuatro entregas principales en tareas más pequeñas asignándolas a nuestro único recurso, en las siguientes imágenes se mostrará el desglose de tareas para cada entrega.

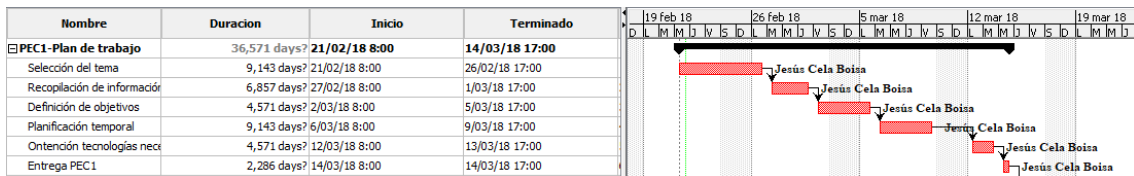


Ilustración 3: Gantt tareas PEC1



Ilustración 4: Gantt tareas PEC2

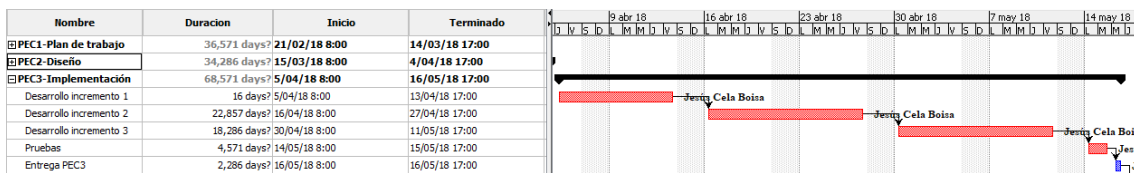


Ilustración 5: Gantt tareas PEC3



Ilustración 6: Gantt tareas Entrega Final

Finalmente se adjunta el desglose de horas por tarea en la siguiente imagen:

<b>Nombre</b>	<b>Trabajo</b>
Jesús Cela Boisa	252,05 horas
<i>Selección del tema</i>	<i>14,2 horas</i>
<i>Recopilación de información</i>	<i>10,65 horas</i>
<i>Definición de objetivos</i>	<i>7,1 horas</i>
<i>Planificación temporal</i>	<i>14,2 horas</i>
<i>Obtención tecnologías necesarias</i>	<i>7,1 horas</i>
<i>Entrega PEC1</i>	<i>3,55 horas</i>
<i>Identificación casos de uso</i>	<i>7,1 horas</i>
<i>Análisis de requisitos</i>	<i>10,65 horas</i>
<i>Diseño de alto nivel</i>	<i>7,1 horas</i>
<i>Diseño de arquitectura</i>	<i>10,65 horas</i>
<i>Diseño interfaz</i>	<i>7,1 horas</i>
<i>Entrega PEC2</i>	<i>3,55 horas</i>
<i>Desarrollo incremento 1</i>	<i>24,85 horas</i>
<i>Desarrollo incremento 2</i>	<i>35,5 horas</i>
<i>Desarrollo incremento 3</i>	<i>28,4 horas</i>
<i>Pruebas</i>	<i>7,1 horas</i>
<i>Entrega PEC3</i>	<i>3,55 horas</i>
<i>Manual de usuario</i>	<i>10,65 horas</i>
<i>Revisión de memoria</i>	<i>14,2 horas</i>
<i>Vídeo presentación</i>	<i>21,3 horas</i>
<i>Final Proyecto</i>	<i>3,55 horas</i>

Ilustración 7: Carga temporal de recurso por tarea específica

## 1.5. Breve resumen de productos obtenidos

1. Código fuente de la aplicación.
2. APK de la aplicación.
3. Memoria del proyecto.
4. Vídeo de la presentación del proyecto.
5. Manual de usuario de la aplicación (probablemente incluido en Memoria como un Anexo).

## 1.6. Breve descripción de los otros capítulos de la memoria

Esta memoria cuenta además de este primer capítulo de introducción con estos otros capítulos:

1. **Análisis:** Engloba toda la parte relacionada con los casos de uso y diferentes documentos que se emplearan en la fase posterior de "Diseño". Se identificarán los distintos casos de uso que darán lugar a los requisitos funcionales que debe tener la aplicación.
2. **Diseño:** Se realizará, en primer lugar, un diseño de alto nivel de la aplicación para luego pasar a un diseño centrado en el usuario. También se llevará a cabo un diseño de la arquitectura del sistema.
3. **Implementación:** Como en la mayoría de proyectos esta será la fase más larga. Es aquí donde se llevará a cabo la realización de cada uno de los incrementos y la aplicación se irá desarrollando y tomando forma hasta la entrega del producto final.
4. **Validación y pruebas:** Esta fase estará centrada en la realización de diferentes pruebas para detectar posibles errores a la hora de integrar los elementos creados en cada incremento.
5. **Conclusiones:** Se revisará la memoria, que será realizada a lo largo de cada una de las diferentes fases, si se detecta algún error en ella se procederá a su corrección. También se revisarán los manuales de usuario en busca de cualquier fallo y se redactarán las conclusiones del proyecto.

## **2.Análisis**

A continuación se documentará el proceso de análisis de la aplicación. En primer lugar se definirá el grupo de usuarios a los que va dirigida la aplicación. Posteriormente se especificarán los casos de uso, donde a partir de ellos se llevará a cabo un proceso de extracción de requisitos de la aplicación. Finalmente obtendremos un documento de especificación de requisitos que será empleado durante las siguientes etapas del proceso de desarrollo.

### **2.1. Definición del usuario y contexto de uso.**

En esta sección se identificará el grupo o grupos de usuarios a los que va destinada la aplicación. También se definirán fichas concretas de personas que serán potenciales usuarios de la aplicación y se un posible escenario en los que se utilizaría la aplicación para resolver una necesidad.

#### **2.1.1. Identificación de grupo de usuarios.**

Debido a la temática tan específica de la aplicación y a lo comentado en la introducción sobre el actual trabajo que conlleva la preparación de los entrenamientos y gestión de un equipo de fútbol por parte del cuerpo técnico resulta sencillo localizar un grupo de usuarios potenciales a los que va dirigida la aplicación. Este grupo de usuarios es el conjunto de entrenadores de equipos de fútbol de humilde presupuesto del marco español.

Se analiza este grupo de usuarios. Actualmente para ser entrenador se debe pasar por un periodo de formación. Pasado este periodo de formación los aspirantes deben someterse a una prueba teórica y una prueba práctica en la cual demostrarán que los conocimientos y habilidades adquiridas son suficientes para obtener el título de entrenador. [16]

Debido al proceso de formación y a las pruebas evaluadoras a los que están sometidos los aspirantes a entrenadores se puede decir que estos poseen los conocimientos necesarios para utilizar la aplicación mientras que a otro tipo de usuarios les podría resultar confuso.



Por tanto se puede concluir que para el uso correcto de la aplicación se deben poseer:

- Conocimientos medio-altos del sistema de gestión de un equipo de fútbol.
- Conocimientos de la terminología utilizada.

### 2.1.2. Fichas de personas y escenarios.

Ahora se va a definir a una persona que será potencial usuario de la aplicación ya que es un entrenador cualificado y en activo. También se definirán un par de escenarios en los que sea útil el uso de la aplicación para facilitar al usuario el trabajo y ahorrarle tiempo.

La ficha de las personas sigue el siguiente esquema:

<b>Foto</b>	Foto de la persona
<b>Nombre</b>	Nombre de la persona
<b>Edad</b>	Edad de la persona
<b>Profesión</b>	Situación laboral y ocupación de la persona
<b>Lugar de residencia</b>	Lugar donde vive actualmente la persona
<b>Tipo de usuario</b>	Tipo de usuario de la aplicación
<b>Descripción general</b>	Descripción de la persona en cuanto a su relación con la aplicación.

Tabla 1: Esquema ficha persona.

*\*Nota: los datos de esta persona (reales) están aportados bajo su consentimiento.*

<b>Foto</b>	
<b>Nombre</b>	Iván Pardal Coca
<b>Edad</b>	29 años
<b>Profesión</b>	Informático y entrenador de fútbol base
<b>Lugar de residencia</b>	Santiago de Compostela
<b>Tipo de usuario</b>	<b>Focal</b>
<b>Descripción general</b>	Iván se inició en el deporte con 6 años, cuando por primera vez sus padres lo apuntaron al equipo de fútbol sala de su

	<p>localidad. Con el tiempo, comenzó a interesarse también por el fútbol campo y con ayuda de su hermano mayor encontró un equipo cerca de donde vivía para poder entrenar.</p> <p>Durante toda su infancia y adolescencia, Iván jugó al fútbol campo federado en el mismo equipo. En ese equipo conoció a muchos de sus amigos y aprendió unos valores de compañerismo, trabajo y esfuerzo que le fueron marcando durante todo su crecimiento.</p> <p>Iván con 18 años, comenzó a trabajar de monitor de campamentos con niños y descubrió que le encantaba enseñarles y despertar su curiosidad y motivación por la vida. Como también le encantaba todo lo relacionado con el fútbol, llegado el momento le surgió la idea de convertirse en entrenador de un equipo de base y poder mezclar así dos de sus pasiones, enseñar y el fútbol.</p> <p>Tras comenzar con un equipo de chicos de 8 a 10 años, y obtener una experiencia más que positiva, enseguida sintió la necesidad de mejorar para ofrecer a los niños que entrenaba y a él mismo una calidad en el trabajo que desempeñaba. Fue entonces cuando se sacó el título de entrenador y descubrió infinidad de entrenamientos, tareas, ejercicios.</p>
--	---

**Tabla 2: Ficha de Iván.**

Se presentan a continuación dos posibles escenarios en los cuáles resulta útil la aplicación.

<b>Escenario 1</b>	<p>Iván necesita para preparar el entrenamiento de la semana una media de entre 15 y 30 minutos, según el tipo de entrenamiento que tenga pensado realizar. Se sienta, consulta algunos ejercicios en internet para hacerse una idea y los adapta a las necesidades de su equipo (edad, nivel, cantidad de chavales, material necesario...). Anota cada ejercicio y las modificaciones en un folio en sucio. Tras</p>
--------------------	---

	<p>terminar pasa todo a un papel limpio para llevárselo al entrenamiento y poder consultar los ejercicios que tiene preparado.</p> <p>En el escenario anterior, Iván invierte sistemáticamente una media de 30 minutos en realizar cada una de las sesiones de entrenamiento, sin importar cuantas veces se repitan los ejercicios a lo largo de las semanas. Esto provoca que en numerosas ocasiones, por culpa de falta de tiempo, no pueda pararse suficiente tiempo en pensar que ejercicios son más adecuados y esto repercute negativamente en la calidad de las sesiones. Con la aplicación, e invirtiendo un tiempo previo en la creación de los ejercicios, Iván puede programar una sesión en un par de minutos, ya que el sistema le sugiere una serie de tareas adecuadas a la edad, nivel y número de jugadores disponibles.</p>
--	---

**Tabla 3: Escenario de uso 1.**

<p><b>Escenario 2</b></p>	<p>Iván lleva un control de la asistencia a los entrenamientos, lo cual le sirve en parte para realizar las convocatorias y para a la larga identificar que jugadores vinieron a qué sesiones. Un día que sale tarde de trabajar, no le da tiempo a imprimir la hoja de asistencia, por lo que pierde los datos de ese día, y a la hora de hacer la convocatoria deberá hacerla con solo parte de los datos necesarios.</p> <p>En el escenario 2, Iván ha perdido una serie de datos necesarios para el correcto desempeño de sus funciones de entrenador. La tarea de llevar un control de asistencia se ve interrumpida y los datos falseados. Con la aplicación, Iván podría prescindir del gasto y de la incomodidad del papel, y pasar a introducir los datos directamente desde su móvil de manera más cómoda y rápida. A final de semana este control de asistencia, le ofrecería una convocatoria provisional modificable, ahorrándole tiempo a la hora de revisar las hojas de asistencia y de pasar posteriormente los datos a la hoja de convocatoria.</p>
---------------------------	---

**Tabla 4: Escenario de uso 2.**

## 2.2. Casos de uso.

En este apartado se presenta el diagrama de casos de uso de la aplicación, posteriormente se especificará cada uno de los casos de uso siguiendo la plantilla de especificación de casos de uso. Los casos de uso son una descripción de los pasos o actividades que deben realizarse para llevar a cabo un proceso. En el diagrama de casos de uso se podrá ver las relaciones entre los distintos actores y los casos de uso del sistema. La principal ventaja de los casos de uso es la facilidad para su interpretación, lo que hace que sean especialmente útiles en la comunicación con el cliente. [17,18]

<b>ID</b>	<b>Clave que identifica el caso de uso. Se representa mediante el código CU-x donde <u>x</u> representa el número de caso de uso.</b>
<b>Nombre</b>	Nombre asociado al caso de uso
<b>Descripción</b>	Descripción del caso de uso.
<b>Actor</b>	Nombre del actor o actores que se relacionan con el caso de uso.
<b>Precondición</b>	Condición que se debe cumplir para que se pueda realizar el caso de uso.
<b>Escenario principal</b>	Comportamiento del sistema en una situación válida o normal del caso de uso.
<b>Escenario alternativo</b>	Comportamiento del sistema en una situación especial o no válida (error) del caso de uso.
<b>Post condición</b>	Define el estado en el que debe quedar el sistema una vez realizado el caso de uso.
<b>Prioridad</b>	Define el grado de importancia del caso de uso dentro del sistema. La prioridad puede ser: Alta, Media o Baja.
<b>Observaciones</b>	Información adicional sobre el caso de uso.

Tabla 5: Esquema caso de uso.

### 2.2.1. Descripción de los actores y diagrama de casos de uso

En el sistema se identifica un único actor que será el usuario que maneja la aplicación. Este actor será el encargado de introducir los datos de los ejercicios, jugadores, entrenamientos, etc. pudiendo consultarlos posteriormente y obteniendo vistas simplificadas generadas por la aplicación.

A continuación se muestra el diagrama de casos de uso:

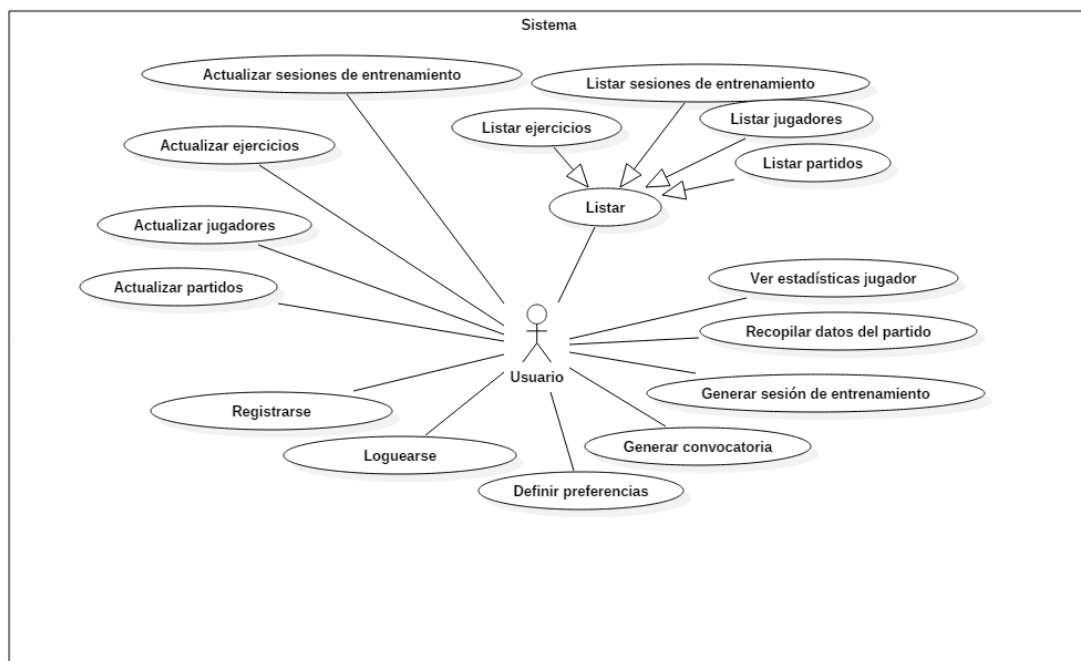


Ilustración 8: Diagrama de casos de uso.

En el diagrama anterior se reflejan varios casos de uso de mantenimientos como los de jugadores, ejercicios, partidos y sesiones de entrenamiento. También están aquellos casos de uso referentes al apartado de resultados y estadísticas así como los casos de uso relacionados con el registro y login del usuario y las preferencias de este a nivel de aplicación.

### 2.2.2. Especificación de casos de uso

A continuación se pasará a especificar en detalle cada uno de los casos de uso del diagrama del apartado anterior. Se comenzará por especificar los casos de

uso de los mantenimientos (CRUD) ya que son aquellos casos de usos básicos e indispensables para el funcionamiento del sistema y posteriormente pasaremos a especificar aquellos casos de uso más complejos.

<b>ID</b>	<b>CU-1</b>
<b>Nombre</b>	Actualizar ejercicios
<b>Descripción</b>	El usuario podrá actualizar la información de los ejercicios.
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. <b>Añadir ejercicio:</b> el caso de uso comienza cuando el usuario elige añadir nuevo ejercicio.  El sistema presenta un formulario con la información a proporcionar por el usuario.  El usuario provee la información requerida.  El sistema valida la información introducida.  El caso de uso termina.</li> <li>2. <b>Modificar ejercicio:</b> el caso de uso comienza cuando el usuario elige modificar un ejercicio.  El sistema presenta un formulario con la información actual del ejercicio.  El usuario modifica la información que desea.  El sistema valida que la nueva información es correcta.  El caso de uso termina.</li> <li>3. <b>Eliminar ejercicio:</b> el caso de uso comienza cuando el usuario elige eliminar un ejercicio.  El sistema pide confirmación al usuario de que desea eliminar el ejercicio.  El usuario confirma la acción.  El sistema valida que el ejercicio puede ser eliminado.  El sistema elimina el ejercicio del sistema.  El caso de uso termina.</li> </ol>
<b>Escenario alternativo</b>	<p>El usuario cancela la operación a realizar.</p> <p>Los datos introducidos por el usuario no son válidos.</p>

	El sistema no permite la eliminación del ejercicio.
<b>Post condición</b>	La información de los ejercicios queda actualizada en el sistema.
<b>Prioridad</b>	Alta
<b>Observaciones</b>	

Tabla 6: CU-1: Actualizar ejercicios.

<b>ID</b>	<b>CU-2</b>
<b>Nombre</b>	Actualizar jugadores
<b>Descripción</b>	El usuario podrá actualizar la información de los jugadores.
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. Añadir jugador: el caso de uso comienza cuando el usuario elige añadir nuevo jugador.  El sistema presenta un formulario con la información a proporcionar por el usuario.  El usuario provee la información requerida.  El sistema valida la información introducida.  El caso de uso termina.</li> <li>2. Modificar jugador: el caso de uso comienza cuando el usuario elige modificar un jugador.  El sistema presenta un formulario con la información actual del jugador.  El usuario modifica la información que desea.  El sistema valida que la nueva información es correcta.  El caso de uso termina.</li> <li>3. Eliminar jugador: el caso de uso comienza cuando el usuario elige eliminar un jugador.  El sistema pide confirmación al usuario de que desea eliminar el jugador.  El usuario confirma la acción.  El sistema valida que el jugador puede ser eliminado.  El sistema elimina el jugador del sistema.</li> </ol>

	El caso de uso termina.
<b>Escenario alternativo</b>	El usuario cancela la operación a realizar. Los datos introducidos por el usuario no son válidos. El sistema no permite la eliminación del jugador.
<b>Post condición</b>	La información de los jugadores queda actualizada en el sistema.
<b>Prioridad</b>	Alta.
<b>Observaciones</b>	

Tabla 7: CU-2: Actualizar jugadores.

<b>ID</b>	<b>CU-3</b>
<b>Nombre</b>	Actualizar sesiones de entrenamiento
<b>Descripción</b>	El usuario podrá actualizar la información de las sesiones de entrenamiento.
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. Añadir sesión de entrenamiento: el caso de uso comienza cuando el usuario elige añadir nueva sesión de entrenamiento. El sistema presenta un formulario con la información a proporcionar por el usuario. El usuario provee la información requerida. El sistema valida la información introducida. El caso de uso termina.</li> <li>2. Modificar sesión de entrenamiento: el caso de uso comienza cuando el usuario elige modificar una sesión de entrenamiento. El sistema presenta un formulario con la información actual de la sesión. El usuario modifica la información que desea. El sistema valida que la nueva información es correcta. El caso de uso termina.</li> <li>3. Eliminar sesión de entrenamiento: el caso de uso</li> </ol>



	<p>comienza cuando el usuario elige eliminar una sesión de entrenamiento.</p> <p>El sistema pide confirmación al usuario de que desea eliminar la sesión de entrenamiento.</p> <p>El usuario confirma la acción.</p> <p>El sistema valida que la sesión puede ser eliminada.</p> <p>El sistema elimina la sesión del sistema.</p> <p>El caso de uso termina.</p>
<b>Escenario alternativo</b>	<p>El usuario cancela la operación a realizar.</p> <p>Los datos introducidos por el usuario no son válidos.</p> <p>El sistema no permite la eliminación del ejercicio.</p>
<b>Post condición</b>	<p>La información de las sesiones de entrenamiento queda actualizada en el sistema.</p>
<b>Prioridad</b>	Alta.
<b>Observaciones</b>	

Tabla 8: CU-3: Actualizar sesiones de entrenamiento.

<b>ID</b>	<b>CU-4</b>
<b>Nombre</b>	Actualizar partidos
<b>Descripción</b>	El usuario podrá actualizar la información de los partidos
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. Añadir partido: el caso de uso comienza cuando el usuario elige añadir nuevo partido. El sistema presenta un formulario con la información a proporcionar por el usuario. El usuario provee la información requerida. El sistema valida la información introducida. El caso de uso termina.</li> <li>2. Modificar partido: el caso de uso comienza cuando el usuario elige modificar un partido. El sistema presenta un formulario con la información actual del partido.</li> </ol>

	<p>El usuario modifica la información que desea.  El sistema valida que la nueva información es correcta.  El caso de uso termina.</p> <p>3. Eliminar partido: el caso de uso comienza cuando el usuario elige eliminar un partido.  El sistema pide confirmación al usuario de que desea eliminar el partido.  El usuario confirma la acción.  El sistema valida que el partido puede ser eliminado.  El sistema elimina el partido del sistema.  El caso de uso termina.</p>
<b>Escenario alternativo</b>	<p>El usuario cancela la operación a realizar.  Los datos introducidos por el usuario no son válidos.  El sistema no permite la eliminación del ejercicio.</p>
<b>Post condición</b>	<p>La información de los partidos queda actualizada en el sistema.</p>
<b>Prioridad</b>	Alta
<b>Observaciones</b>	

Tabla 9: CU-4: Actualizar partidos.

<b>ID</b>	CU-5
<b>Nombre</b>	Listar
<b>Descripción</b>	El usuario podrá visualizar listados de diferentes datos del sistema
<b>Actor</b>	Usuario.
<b>Precondición</b>	Deben existir datos que listar.
<b>Escenario principal</b>	<ol style="list-style-type: none"> <li>1. Listar ejercicios: el caso de uso comienza cuando el usuario accede a la lista de ejercicios. El sistema muestra la lista de ejercicios almacenados.</li> <li>2. Listar sesiones de entrenamiento: el caso de uso comienza cuando el usuario accede a la lista de sesiones de entrenamiento. El sistema muestra la lista de sesiones de entrenamiento almacenadas.</li> <li>3. Listar jugadores: el caso de uso comienza cuando el usuario accede a la lista de jugadores. El sistema muestra la lista de jugadores almacenados.</li> <li>4. Listar partidos: el caso de uso comienza cuando el usuario accede a la lista de partidos. El sistema muestra la lista de partidos almacenados.</li> </ol>
<b>Escenario alternativo</b>	El sistema no puede realizar el listado.
<b>Post condición</b>	El usuario visualiza el listado solicitado
<b>Prioridad</b>	Alta
<b>Observaciones</b>	

Tabla 10: CU-5: Listar.

<b>ID</b>	<b>CU-6</b>
<b>Nombre</b>	Generar sesión de entrenamiento
<b>Descripción</b>	El usuario podrá generar una sesión de entrenamiento de manera automática a partir de los ejercicios añadidos anteriormente y las indicaciones dadas.
<b>Actor</b>	Usuario
<b>Precondición</b>	Debe existir algún ejercicio en el sistema.
<b>Escenario principal</b>	<p>El caso de uso comienza cuando el usuario elige generar una sesión de entrenamiento.</p> <p>El sistema muestra un formulario de preferencias de la sesión:</p> <p>El usuario cubre los datos requeridos por el sistema.</p> <p>El sistema valida los datos.</p> <p>El sistema genera una sesión de entrenamiento en base a las preferencias del usuario.</p> <p>El caso de uso termina.</p>
<b>Escenario alternativo</b>	<p>El usuario cancela la operación a realizar.</p> <p>Los datos introducidos por el usuario no son válidos.</p> <p>El sistema no es capaz de generar la sesión.</p>
<b>Post condición</b>	La aplicación muestra una sesión de entrenamiento generada de forma automática de acuerdo a las preferencias del usuario.
<b>Prioridad</b>	Alta.
<b>Observaciones</b>	

Tabla 11: CU-6: Generar sesión de entrenamiento.

<b>ID</b>	<b>CU-7</b>
<b>Nombre</b>	Recopilar datos del partido
<b>Descripción</b>	El usuario podrá recopilar datos durante el partido
<b>Actor</b>	Usuario
<b>Precondición</b>	Debe haber un partido creado.
<b>Escenario principal</b>	<p>El caso de uso comienza cuando el usuario elige recopilar datos del partido.</p> <p>El sistema proporciona una lista de datos a rellenar por el usuario.</p> <p>El usuario recopila información del partido y la anota en el sistema.</p> <p>El sistema valida que los datos introducidos sean correctos.</p> <p>El sistema guarda los datos recopilados por el usuario en la base de datos.</p>
<b>Escenario alternativo</b>	<p>El usuario cancela la operación a realizar.</p> <p>Los datos introducidos por el usuario no son válidos.</p>
<b>Post condición</b>	Los datos recopilados por el usuario quedan almacenados en el sistema
<b>Prioridad</b>	Media.
<b>Observaciones</b>	

Tabla 12: CU-7: Recopilar datos del partido.

<b>ID</b>	<b>CU-8</b>
<b>Nombre</b>	Generar convocatoria
<b>Descripción</b>	El usuario podrá generar la convocatoria al partido.
<b>Actor</b>	Usuario
<b>Precondición</b>	Debe haber un partido creado.
<b>Escenario principal</b>	<p>El caso de uso comienza cuando el usuario elige generar la convocatoria a un partido.</p> <p>El sistema genera la convocatoria al partido en base a los datos del mismo.</p>
<b>Escenario</b>	El sistema devuelve un error al generar la convocatoria.

<b>alternativo</b>	
<b>Post condición</b>	La convocatoria al partido es generada por el sistema.
<b>Prioridad</b>	Media.
<b>Observaciones</b>	

Tabla 13: CU-8: Generar convocatoria.

<b>ID</b>	<b>CU-9</b>
<b>Nombre</b>	Ver estadísticas jugador
<b>Descripción</b>	El usuario podrá consultar datos del jugador recopilados por el sistema a lo largo del tiempo.
<b>Actor</b>	Usuario
<b>Precondición</b>	Debe existir un jugador creado.
<b>Escenario principal</b>	El caso de uso comienza cuando el usuario selecciona ver estadísticas del jugador. El sistema recopila información del jugador y la muestra al usuario de forma ordenada. El usuario consulta la información del jugador.
<b>Escenario alternativo</b>	---
<b>Post condición</b>	El usuario puede ver un resumen estadístico de los datos del jugador.
<b>Prioridad</b>	Media.
<b>Observaciones</b>	

Tabla 14: CU-9: Ver estadísticas jugador.

<b>ID</b>	<b>CU-10</b>
<b>Nombre</b>	Definir preferencias
<b>Descripción</b>	El usuario podrá establecer ciertas preferencias a nivel de aplicación.
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	El caso de uso comienza cuando el usuario elige definir las preferencias. El sistema proporciona una lista de preferencias ajustables por el usuario. El usuario informa al sistema de sus preferencias. El sistema guarda los datos proporcionados por el usuario.
<b>Escenario alternativo</b>	---
<b>Post condición</b>	Las preferencias quedarán guardadas correctamente. Y el sistema se comportará en base a ellas.
<b>Prioridad</b>	Baja
<b>Observaciones</b>	

Tabla 15: CU-10: Definir preferencias.

<b>ID</b>	<b>CU-11</b>
<b>Nombre</b>	Registrarse
<b>Descripción</b>	El usuario podrá registrarse en una base de datos de usuarios.
<b>Actor</b>	Usuario
<b>Precondición</b>	---
<b>Escenario principal</b>	El caso de uso comienza cuando el usuario elige registrarse. El sistema solicita la información requerida al usuario. El usuario provee la información al sistema. El sistema registra la información del usuario en la base de datos.
<b>Escenario</b>	El usuario cancela la operación a realizar.

<b>alternativo</b>	Los datos introducidos por el usuario no son válidos. El sistema no tiene conexión a internet.
<b>Post condición</b>	El usuario queda registrado en la base de datos
<b>Prioridad</b>	Alta
<b>Observaciones</b>	

**Tabla 16: CU-11: Registrarse.**

<b>ID</b>	<b>CU-12</b>
<b>Nombre</b>	Loguearse
<b>Descripción</b>	El usuario podrá iniciar sesión en el sistema
<b>Actor</b>	Usuario
<b>Precondición</b>	El usuario debe estar registrado en el sistema.
<b>Escenario principal</b>	El sistema solicita las credenciales al usuario. El usuario informa de sus credenciales. El sistema valida las credenciales del usuario. El sistema da acceso al usuario.
<b>Escenario alternativo</b>	El usuario cancela la operación a realizar. Los datos introducidos por el usuario no son válidos. El sistema no tiene conexión a internet
<b>Post condición</b>	El usuario queda logueado correctamente en el sistema.
<b>Prioridad</b>	Alta
<b>Observaciones</b>	

**Tabla 17: CU-12: Loguearse.**



## 2.3. Requisitos de la aplicación.

En este apartado se llevará a cabo una descripción completa de las funcionalidades y requisitos que debe implementar la aplicación. En primer lugar se introducirá el concepto de requisito y se explicarán los distintos tipos de requisitos que pueden aparecer en nuestra aplicación para posteriormente clasificar nuestros requisitos y especificarlos.

Un requisito es una condición necesaria para que algo se cumpla o se realice, y los vamos a clasificar en dos tipos: requisitos funcionales y requisitos no funcionales.

- Los **requisitos funcionales** son aquellos que definen una función del sistema de software, es decir, describen el conjunto de entradas, comportamientos y salidas del sistema de software. [19]
- Los **requisitos no funcionales** son aquellos que no describen un comportamiento o una función que realizar. En la aplicación nos encontraremos con distintos tipos de requisitos no funcionales: [20]
  - **Requisitos de usabilidad:** describen los niveles apropiados de usabilidad, dados los usuarios finales del producto, para ello deben revisarse las especificaciones de los perfiles de usuarios y las clasificaciones de sus niveles de experiencia.
  - **Requisitos de interfaz:** este tipo de requisito describe la apariencia del producto. Es importante destacar que no se trata del diseño de la interfaz en detalle sino que especifican cómo se pretende que sea la interfaz externa del producto. También pueden ser necesidades de cumplir con normas estándares, o con los estándares de la empresa para la cual se esté desarrollando el software.
  - **Requisitos de integridad:** la información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes, de la misma forma será considerada igual

a la fuente o autoridad de los datos. Pueden incluir también mecanismos de chequeo de integridad y realización de auditorías.

- **Requisitos de software:** debe mencionarse el software del que se debe disponer, después de implementado el sistema.
- **Requisitos de diseño e implementación:** especifica o restringe la codificación o construcción de un sistema, son restricciones que han sido ordenadas y deben ser cumplidas estrictamente.

Una vez visto los distintos tipos de requisitos se establecerá una nomenclatura para identificar cada requisito de forma inequívoca (véase tabla 18).

Tipo de requisito	Codificación (siendo <u>x</u> el número de requisito)
Requisito funcional	RF- <i>x</i>
Requisito de usabilidad	RUsab- <i>x</i>
Requisito de interfaz	RIfaz- <i>x</i>
Requisito de integridad	RIdad- <i>x</i>
Requisito de software	RSoft- <i>x</i>
Requisito de diseño	RDis- <i>x</i>

Tabla 18: Nomenclatura identificación de requisitos.

Todos los requisitos especificados seguirán la plantilla propuesta (véase Tabla 19, página siguiente) donde se describirán las condiciones de aceptación de dicho requisito, el caso de uso en el que se basa y la importancia de su cumplimiento.

<b>ID</b>	<b>Según codificación (véase tabla 18)</b>
<b>Nombre</b>	Nombre de requisito
<b>Condiciones de aceptación</b>	Condiciones que se deben dar para que el requisito se dé por satisfecho
<b>Caso de uso relacionado</b>	Caso de uso con el que se relaciona el requisito.
<b>Importancia</b>	Importancia de cumplimiento del requisito dentro de la aplicación.

**Tabla 19: Plantilla de requisitos.**

### 2.3.1. Requisitos funcionales

En este apartado se especificarán los requisitos funcionales que debe cumplir la aplicación.

<b>ID</b>	<b>RF-1</b>
<b>Nombre</b>	Crear ejercicio
<b>Condiciones de aceptación.</b>	La aplicación podrá crear instancias de ejercicios y almacenarlas en base de datos.
<b>Caso de uso relacionado.</b>	CU-1
<b>Importancia.</b>	Alta

**Tabla 20: RF-1: Crear ejercicio.**

<b>ID</b>	<b>RF-2</b>
<b>Nombre</b>	Modificar ejercicio
<b>Condiciones de aceptación.</b>	La aplicación permitirá modificar los datos asociados a un ejercicio de la base de datos.
<b>Caso de uso relacionado.</b>	CU-1
<b>Importancia.</b>	Alta

**Tabla 21: RF-2: Modificar ejercicio.**

<b>ID</b>	<b>RF-3</b>
<b>Nombre</b>	Eliminar ejercicio
<b>Condiciones de aceptación.</b>	La aplicación podrá eliminar ejercicios de la base de datos.
<b>Caso de uso relacionado.</b>	CU-1
<b>Importancia.</b>	Media

**Tabla 22: RF-3: Eliminar ejercicio.**

<b>ID</b>	<b>RF-4</b>
<b>Nombre</b>	Crear sesión de entrenamiento
<b>Condiciones de aceptación.</b>	La aplicación podrá crear instancias de sesiones de entrenamientos y almacenar los datos de los mismos en base de datos.
<b>Caso de uso relacionado.</b>	CU-3
<b>Importancia.</b>	Alta

**Tabla 23: RF-4: Crear sesión de entrenamiento.**

<b>ID</b>	<b>RF-5</b>
<b>Nombre</b>	Modificar sesión de entrenamiento
<b>Condiciones de aceptación.</b>	La aplicación permitirá modificar los datos de las sesiones de entrenamiento de la base de datos.
<b>Caso de uso relacionado.</b>	CU-3
<b>Importancia.</b>	Alta

**Tabla 24: RF-5: Modificar sesión de entrenamiento.**

<b>ID</b>	<b>RF-6</b>
<b>Nombre</b>	Eliminar sesión de entrenamiento
<b>Condiciones de aceptación.</b>	La aplicación será capaz de eliminar sesiones de entrenamiento de la base de datos.
<b>Caso de uso relacionado.</b>	CU-3
<b>Importancia.</b>	Baja

**Tabla 25: RF-6: Eliminar sesión de entrenamiento.**

<b>ID</b>	<b>RF-7</b>
<b>Nombre</b>	Crear jugador
<b>Condiciones de aceptación.</b>	La aplicación podrá insertar en la base de datos la información de los jugadores
<b>Caso de uso relacionado.</b>	CU-2
<b>Importancia.</b>	Alta

**Tabla 26: RF-7: Crear jugador.**

<b>ID</b>	<b>RF-8</b>
<b>Nombre</b>	Modificar jugador
<b>Condiciones de aceptación.</b>	La aplicación podrá modificar los datos de los jugadores de la base de datos.
<b>Caso de uso relacionado.</b>	CU-2
<b>Importancia.</b>	Alta

**Tabla 27: RF-8: Modificar jugador.**

<b>ID</b>	<b>RF-9</b>
<b>Nombre</b>	Eliminar jugador
<b>Condiciones de aceptación.</b>	La aplicación podrá eliminar el registro de un jugador de la base de datos.
<b>Caso de uso relacionado.</b>	CU-2
<b>Importancia.</b>	Alta

Tabla 28: RF-9: Eliminar jugador.

<b>ID</b>	<b>RF-10</b>
<b>Nombre</b>	Crear partido
<b>Condiciones de aceptación.</b>	La aplicación podrá crear partidos y guardarlos en base de datos.
<b>Caso de uso relacionado.</b>	CU-4
<b>Importancia.</b>	Media

Tabla 29: RF-10: Crear partido.

<b>ID</b>	<b>RF-11</b>
<b>Nombre</b>	Modificar partido
<b>Condiciones de aceptación.</b>	La aplicación podrá modificar los datos asociados a un partido.
<b>Caso de uso relacionado.</b>	CU-4
<b>Importancia.</b>	Media.

Tabla 30: RF-11: Modificar partido.

<b>ID</b>	<b>RF-12</b>
<b>Nombre</b>	Eliminar partido
<b>Condiciones de aceptación.</b>	La aplicación podrá eliminar los datos de los partidos.
<b>Caso de uso relacionado.</b>	CU-4
<b>Importancia.</b>	Media.

**Tabla 31: RF-12: Eliminar partido.**

<b>ID</b>	<b>RF-13</b>
<b>Nombre</b>	Listar ejercicios
<b>Condiciones de aceptación.</b>	La aplicación podrá consultar los ejercicios existentes y presentarlos en forma de lista.
<b>Caso de uso relacionado.</b>	CU-5
<b>Importancia.</b>	Alta

**Tabla 32: RF-13: Listar ejercicios.**

<b>ID</b>	<b>RF-14</b>
<b>Nombre</b>	Listar sesiones de entrenamiento
<b>Condiciones de aceptación.</b>	La aplicación podrá consultar las sesiones de entrenamiento existentes y presentarlas en forma de lista.
<b>Caso de uso relacionado.</b>	CU-5
<b>Importancia.</b>	Alta

**Tabla 33: RF-14: Listar sesiones de entrenamiento.**

<b>ID</b>	<b>RF-15</b>
<b>Nombre</b>	Listar jugadores
<b>Condiciones de aceptación.</b>	La aplicación podrá consultar los jugadores existentes y presentarlos en forma de lista.
<b>Caso de uso relacionado.</b>	CU-5
<b>Importancia.</b>	Alta

**Tabla 34: RF-15: Listar jugadores.**

<b>ID</b>	<b>RF-16</b>
<b>Nombre</b>	Listar partidos
<b>Condiciones de aceptación.</b>	La aplicación podrá consultar los partidos existentes y presentarlos en forma de lista.
<b>Caso de uso relacionado.</b>	CU-5
<b>Importancia.</b>	Alta

**Tabla 35: RF-16: Listar partidos.**

<b>ID</b>	<b>RF-17</b>
<b>Nombre</b>	Guardar preferencias del usuario
<b>Condiciones de aceptación.</b>	La aplicación podrá guardar las preferencias del usuario.
<b>Caso de uso relacionado.</b>	CU-10
<b>Importancia.</b>	Media

**Tabla 36: RF-17: Guardar preferencias del usuario.**



<b>ID</b>	<b>RF-18</b>
<b>Nombre</b>	Generar sesión de entrenamiento
<b>Condiciones de aceptación.</b>	La aplicación podrá generar automáticamente una sesión de entrenamiento a partir de la información de ejercicios y jugadores almacenados en base de datos.
<b>Caso de uso relacionado.</b>	CU-6
<b>Importancia.</b>	Alta.

Tabla 37: RF-18: Generar sesión de entrenamiento.

<b>ID</b>	<b>RF-19</b>
<b>Nombre</b>	Registrar usuario.
<b>Condiciones de aceptación.</b>	La aplicación podrá registrar los datos del usuario en el servidor.
<b>Caso de uso relacionado.</b>	CU-11
<b>Importancia.</b>	Alta

Tabla 38: RF-19: Registrar usuario.

<b>ID</b>	<b>RF-20</b>
<b>Nombre</b>	Login del usuario.
<b>Condiciones de aceptación.</b>	La aplicación será capaz de loguear al usuario en la aplicación.
<b>Caso de uso relacionado.</b>	CU-12
<b>Importancia.</b>	Alta

Tabla 39: RF-20: Login del usuario.

<b>ID</b>	<b>RF-21</b>
<b>Nombre</b>	Generar fichero de texto
<b>Condiciones de aceptación.</b>	La aplicación será capaz de generar un fichero de texto con los datos de la convocatoria de un partido.
<b>Caso de uso relacionado.</b>	CU-8
<b>Importancia.</b>	Baja

**Tabla 40: RF-21: Generar fichero de texto.**

<b>ID</b>	<b>RF-22</b>
<b>Nombre</b>	Guardar datos asociados a un partido
<b>Condiciones de aceptación.</b>	La aplicación podrá guardar los datos de un partido recopilados por el usuario durante el transcurso del mismo.
<b>Caso de uso relacionado.</b>	CU-7
<b>Importancia.</b>	Baja

**Tabla 41: RF-22: Guardar datos asociados a un partido.**

<b>ID</b>	<b>RF-23</b>
<b>Nombre</b>	Consultar y resumir datos de un jugador
<b>Condiciones de aceptación.</b>	La aplicación podrá consultar los datos de la actividad de los jugadores y mostrarla por pantalla de manera resumida y organizada.
<b>Caso de uso relacionado.</b>	CU-9
<b>Importancia.</b>	Alta

**Tabla 42: RF-23: Consultar y resumir datos de un jugador.**

### 2.3.2. Requisitos de usabilidad.

Ahora se procederá a especificar los distintos requisitos de usabilidad.

ID	RUsab-1
Nombre	Formularios sencillos.
Condiciones de aceptación.	Los formularios de la aplicación se deben presentar de manera simplificada proporcionando listas y filtros que permitan al usuario apuntar los datos con rapidez.
Caso de uso relacionado.	---
Importancia.	Alta.

Tabla 43: RUsab-1: Formularios sencillos.

ID	RUsab-2
Nombre	Asistencia al usuario.
Condiciones de aceptación.	El usuario debe escribir lo menos posible en la aplicación por lo que los campos se rellenarán de forma automática en la medida de lo posible.
Caso de uso relacionado.	---
Importancia.	Alta.

Tabla 44: RUsab-2: Asistencia al usuario.

### 2.3.3. Requisitos de interfaz.

La aplicación tiene el siguiente requisito de interfaz.

ID	Rlfaz-1
Nombre	Contraste de colores.
Condiciones de aceptación.	La aplicación debe mostrar una interfaz con un contraste de colores que sea perceptible por la mayor cantidad de personas posibles.  La elección de los colores debe permitir el uso de la aplicación cuando las condiciones ambientales son adversas, por ejemplo, días muy soleados.
Caso de uso relacionado.	---
Importancia.	Alta.

Tabla 45: Rlfaz-1: Contraste de colores.

### 2.3.4. Requisitos de software.

En el caso de esta aplicación existe un requisito de software.

ID	RSoft-1
Nombre	Plataforma Android 5.0 o superior.
Condiciones de aceptación.	El dispositivo móvil debe tener el sistema operativo móvil Android en su versión 5.0 o superiores.
Caso de uso relacionado.	---
Importancia.	Alta.

Tabla 46: RSoft-1: Plataforma Android 4.1 o superior.

### 2.3.5. Requisitos de diseño e implementación.

Existe un requisito de diseño que se especifica detalladamente en la siguiente tabla.

<b>ID</b>	<b>RDis-1</b>
<b>Nombre</b>	Utilización de servicios Firebase.
<b>Condiciones de aceptación.</b>	El dispositivo móvil intercambiará datos con el servidor de firebase mediante su API oficial, también se utilizará el formato JSON para la transmisión de datos a través de los servicios.
<b>Caso de uso relacionado.</b>	---
<b>Importancia.</b>	Alta.

Tabla 47: RDis-1: Utilización de servicios Firebase.

## 2.4. Matriz de trazabilidad.

A continuación se creará una matriz de trazabilidad donde se podrá apreciar con más facilidad la relación entre los casos de uso y los requisitos funcionales descritos anteriormente. Se marcará con una 'X' la casilla en la que un requisito funcional corresponda con un caso de uso.

CU/ RF	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7	CU-8	CU-9	CU-10	CU-11	CU-12
RF-1	X											
RF-2	X											
RF-3	X											
RF-4			X									
RF-5			X									
RF-6			X									
RF-7		X										
RF-8		X										

RF-9		X										
RF-10				X								
RF-11				X								
RF-12				X								
RF-13					X							
RF-14					X							
RF-15					X							
RF-16					X							
RF-17										X		
RF-18						X						
RF-19											X	
RF-20												X
RF-21								X				
RF-22							X					
RF-23									X			

Tabla 48: Matriz de trazabilidad.

## 3. Diseño

En esta sección se documentará la fase de diseño. En primer lugar se expondrá el diseño de un prototipo de la aplicación para posteriormente hablar del diseño de la arquitectura de la misma.

### 3.1. Diseño del prototipo de la aplicación.

Se expone a continuación un diseño a nivel de usuario de las diferentes pantallas con las que contará la aplicación para hacernos una idea general de los elementos de los cuales constará cada una de ellas.

#### 3.1.1. Pantalla de Login.

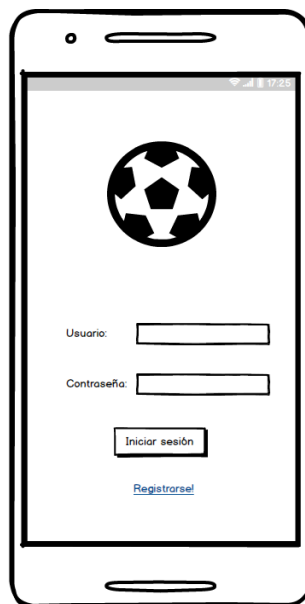


Ilustración 9: Pantalla de login.

En la imagen de la anterior se puede ver cómo será la pantalla de login en la aplicación. Esta será la primera pantalla que aparecerá una vez abierta la aplicación, en ella se muestra un formulario mediante el cual el usuario podrá iniciar sesión en la aplicación para pasar a la pantalla principal. Si el usuario ya ha iniciado sesión en la aplicación esta pantalla se omitirá y este accederá directamente a la pantalla de inicio.

### 3.1.2. Pantalla de inicio.

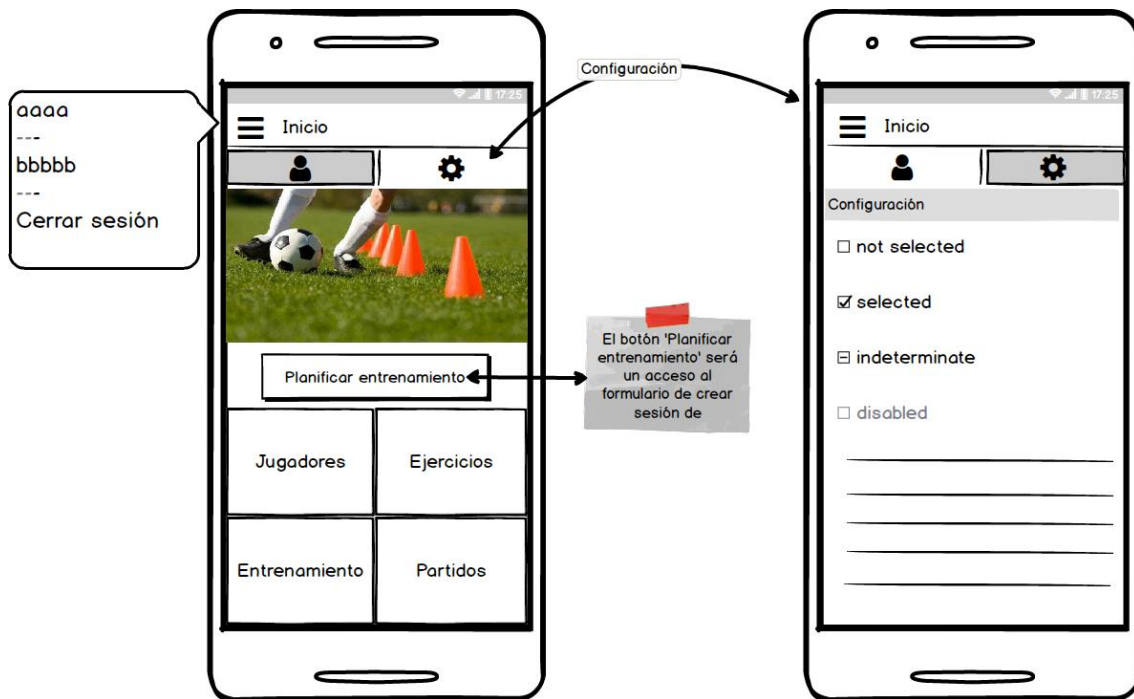


Ilustración 10: Pantalla de inicio.

En la pantalla de inicio se mostrarán las opciones más comunes a utilizar por el usuario que serán 'jugadores', 'ejercicios', 'entrenamiento' y partidos, también se mostrará un acceso rápido a la planificación de un nuevo entrenamiento ya que es considerado por el usuario focal de nuestra aplicación como funcionalidad que más será utilizada.

Desde esta pantalla también se podrá acceder a un menú lateral de opciones así como saltar a la pantalla de preferencias de la aplicación mediante el botón habilitado para ello. En esta pantalla de preferencias se mostrará al usuario diferentes configuraciones que permitirán al usuario adaptar un poco la aplicación as su gusto.



### 3.1.3. Gestión de jugadores.

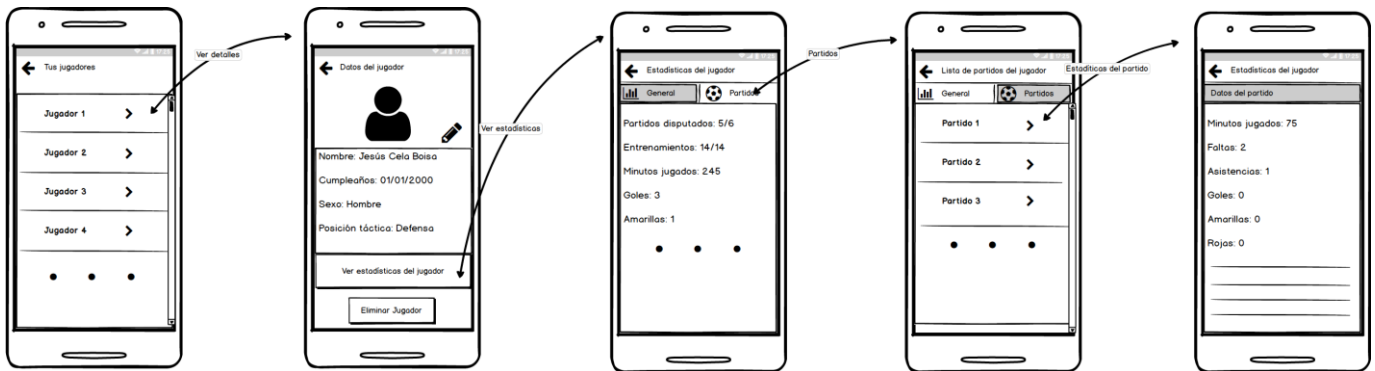


Ilustración 11: Gestión de jugadores.

Al acceder a la sección de jugadores desde la pantalla inicial se saltará a una lista de los jugadores registrados por el usuario en la aplicación. Desde esta lista el usuario tendrá la opción de gestionar la información de sus jugadores así como añadir nuevos o eliminar los existentes.

Si el usuario selecciona un jugador podrá ver sus datos y desde este mismo lugar podrá editar su información o ver estadísticas de la actividad de este jugador en los entrenamientos y partidos.

### 3.1.4. Gestión de ejercicios

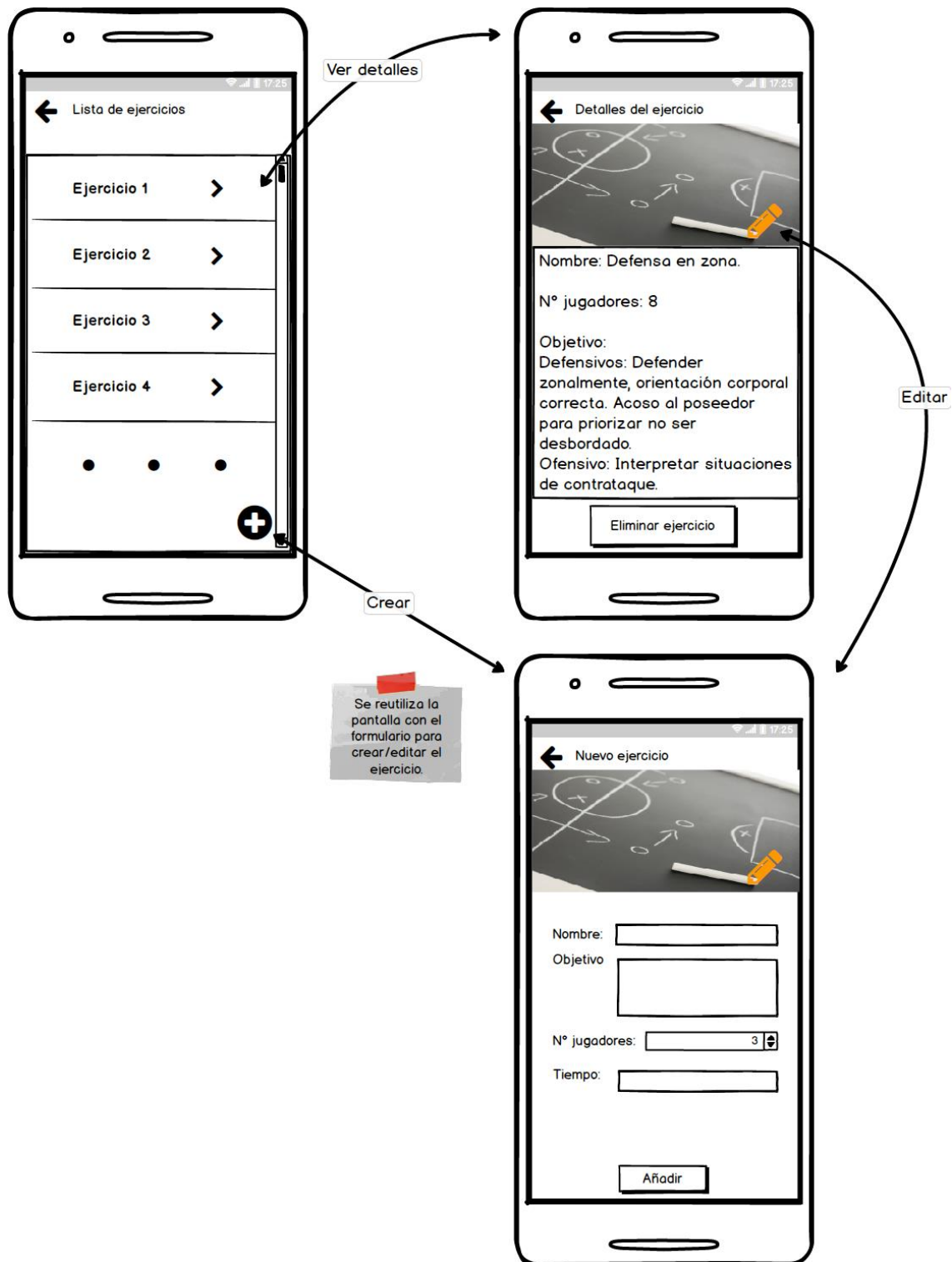


Ilustración 12: Gestión de ejercicios.

En la imagen anterior se puede ver que a través de la lista de los ejercicios guardados en el sistema el usuario podrá ver los detalles de los mismos así como modificar sus datos y añadir o eliminar los ya existentes.

### 3.1.5. Gestión de sesiones de entrenamiento.



Ilustración 13: Gestión de sesiones de entrenamiento.

En la imagen anterior se muestra la funcionalidad más importante de la aplicación que es la gestión de los entrenamientos del equipo.

Partiendo de la lista de entrenamientos ya registrados el usuario podrá planificar una nueva sesión de entrenamiento, para ello deberá indicar algunos datos básicos como la fecha de la sesión y la hora de inicio y posteriormente podrá ir añadiendo ejercicios a esta sesión y asignando jugadores a los ejercicios anteriormente seleccionados. De esta forma el usuario podrá planificar sesiones de entrenamiento en las cuáles registrará los ejercicios

realizados y los jugadores que han intervenido en los mismos, pudiendo utilizar posteriormente estos datos para tomar decisiones estratégicas.

La opción más interesante será la de generar una sesión de entrenamiento de forma automática a partir de los datos básicos de la sesión (formulario de creación) y de las preferencias indicadas por el usuario, de esta forma en casos en los que el usuario no disponga del tiempo necesario para preparar de forma manual la sesión la aplicación aportará una sesión de entrenamiento generada que podrá modificar a su gusto, ahorrando así tiempo en el proceso.

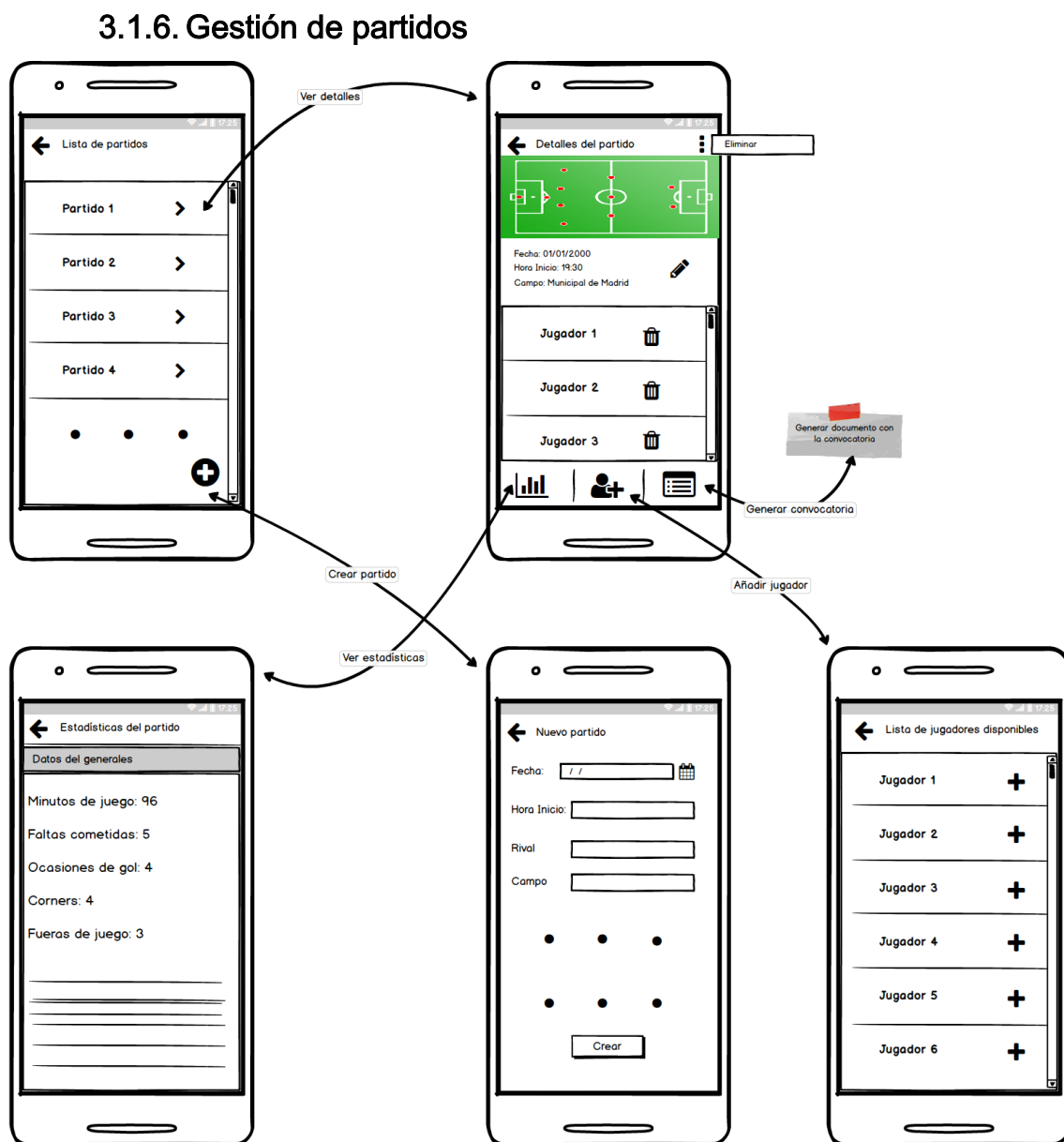


Ilustración 14: Gestión de partidos.

En la imagen de la página anterior se muestra la distribución de las pantallas que intervienen en la gestión de los partidos.

Mediante la lista de partidos ya disputados se podrán añadir nuevos partidos a partir de un formulario de datos básicos del mismo. Posteriormente el usuario podrá añadir los jugadores que participarán en el partido mediante una lista que se habilitará para ello. Una vez creado el partido y añadidos todos sus jugadores el usuario podrá generar la convocatoria a este teniendo la posibilidad de exportar esa convocatoria a un fichero de texto.

Una vez realizada la convocatoria el usuario podrá recopilar datos del partido durante el transcurso del mismo, estos datos podrán ser consultados posteriormente desde la sección de estadísticas de partidos accediendo a través del botón habilitado para ello.

### 3.2. Diseño de la base de datos.

A continuación se presenta el diagrama entidad-relación de la base de datos del sistema.

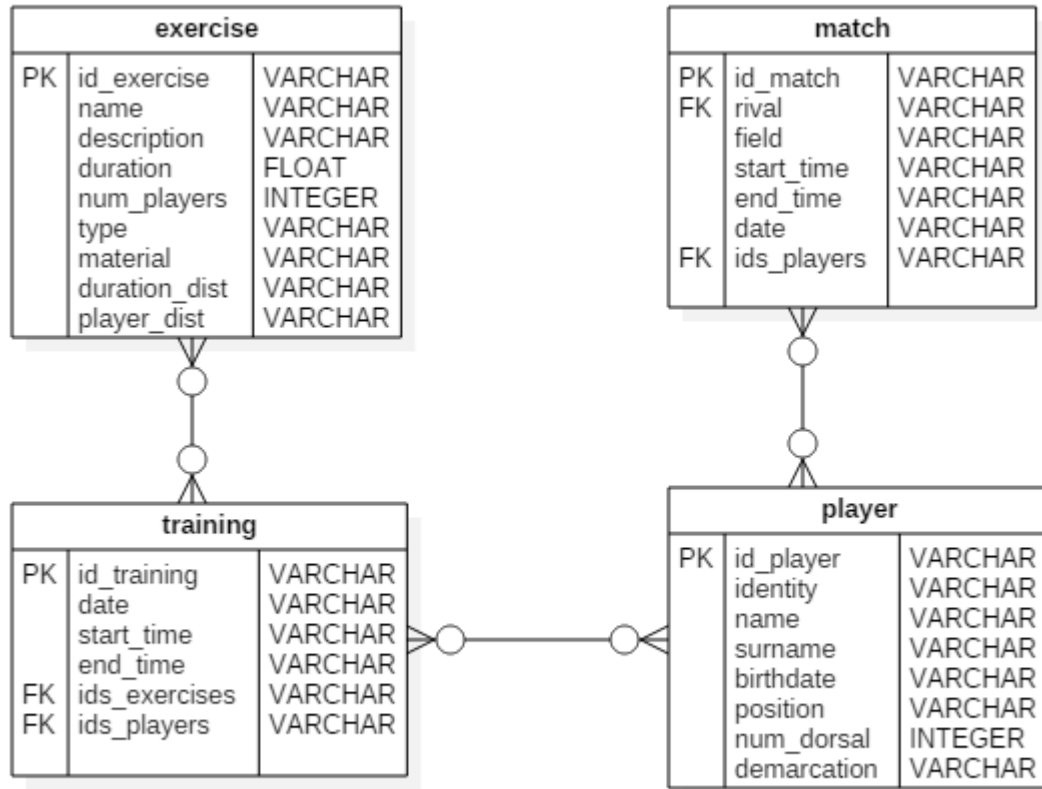


Ilustración 15: Diagrama Entidad-Relación.

Este consta de las entidades:

- **exercise:** almacena los datos de los ejercicios.
- **training:** almacena los datos de las sesiones de entrenamiento.
- **player:** almacena los datos de los jugadores
- **match:** almacena los datos del partido

### 3.3. Diseño de la arquitectura.

Se definirá un patrón para la organización del código de la aplicación y se explicará la capa de persistencia de datos.

#### 3.3.1. Patrón MVC vs patrón MVP.

A la hora de programar en Android surge la dificultad de decidir que patrón de arquitectura utilizar para organizar el código de nuestra aplicación. Entre las posibilidades estudiadas están el patrón MVC y el patrón MVP. Estos dos patrones se basan en la división del código en tres capas: el modelo, la vista y el controlador o presentador. [21,22]

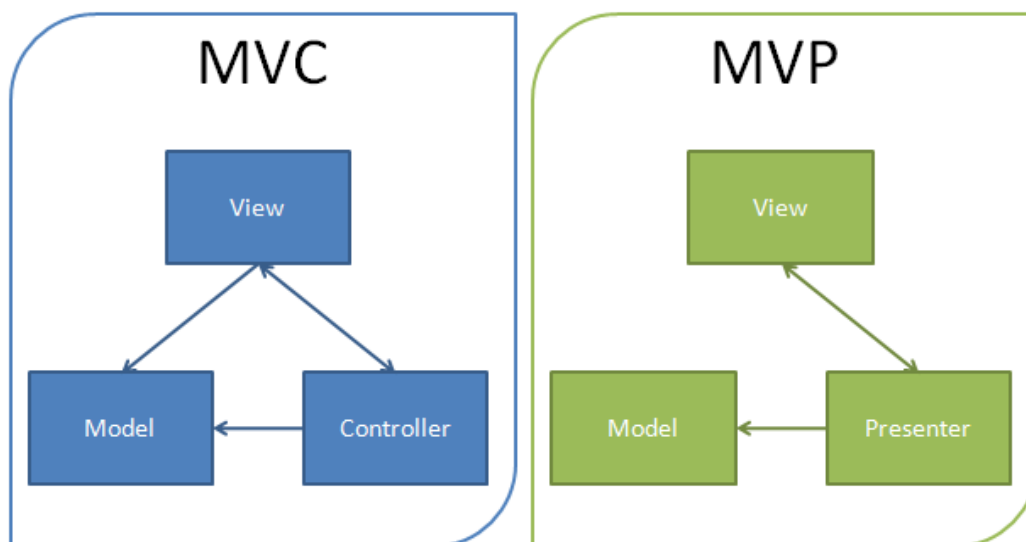


Ilustración 16: Patrones MVC y MVP.

- El **modelo** representa los datos con los que trabaja la aplicación.
- La **vista** es la parte que interactúa con el usuario y en la que se representan los datos del modelo.

En este punto se encuentra gran diferencia entre ambos patrones:

- Por un lado el MVC consta del **controlador** en donde se realiza la lógica de negocio de la aplicación y sirve de intermediador entre el modelo y la vista, aunque existen comunicación entre el modelo y la vista, si esto no

se gestiona bien es probable que en nuestras Activities o Fragments (vistas) acaben teniendo parte de lógica de negocio.

- Por otro lado en el MVP que consta de un **presentador** que actúa de intermediador entre la vista y el modelo y en donde estos últimos no tienen conexión, por lo tanto siempre que se produce un evento es el presentador el que lo recibe y decide qué hacer con los datos y cuándo actualizar la vista. De esta manera aseguras el desacople total de la vista y la lógica de negocio separando las responsabilidades de cada capa.

Se puede ver que MVP presenta algunas ventajas frente a MVC por lo que me decidiré a utilizar el patrón MVP para la implementación del código. Creo que mediante el patrón MVP podré dotar de mayor legibilidad y calidad al código de la aplicación y mediante un correcto desacople de los componentes será más fácil reaccionar ante cambios imprevistos o añadir nuevas funcionalidades.

### **3.3.2. Persistencia de los datos.**

Para el almacenamiento y persistencia de los datos de la aplicación se va a utilizar Firebase Realtime Database para la base de datos en servidor combinado con SQLite para la base de datos local del dispositivo.

Por una parte Firebase te aporta una base de datos NoSQL que almacena la información en formato JSON, lenguaje muy versátil para el tratamiento de datos desde el punto de vista de programación. Además consta de una API que puede resultar muy útil para realizar las comunicaciones con el servidor.

Gracias a Firebase, que resuelve de manera bastante directa el problema de la sincronización de datos, se permitirá al usuario trabajar con la aplicación aunque no disponga de acceso al servidor trabajando en local con los datos ya descargados y sincronizando los nuevos en el momento en que la conexión vuelva a estar disponible.



## 4. Implementación.

En este apartado se explicará cómo se ha llevado a cabo el proceso de implementación del producto.

### 4.1. Entorno de desarrollo.

Tal y como se ha comentado en el apartado 1.3.3 de esta memoria se ha utilizado Android Studio 3.0 como IDE para el desarrollo del código de la aplicación. Aunque existen entornos alternativos a este como el Eclipse o el Netbeans se ha optado por el entorno oficial proporcionado por Google para desarrolladores Android ya que ofrece una gran integración con el emulador y el SDK de Android [23].

### 4.2. Decisiones y cambios en el diseño.

Las decisiones y cambios de diseño que se han tomado durante el desarrollo de la aplicación son los siguientes:

#### 4.2.1. Portada.

Se ha decidido modificar el diseño de la portada con respecto al prototipo original, dotando a esta pantalla de una mayor información en pantalla mediante la lista de próximo entrenamientos planificados por el usuario. De esta forma los botones que ahí se ubicaban quedan desplazados a la parte inferior de la pantalla aprovechando el Widget de Android **BottomNavigationView**, permitiendo de esta forma el acceso rápido a las distintas secciones de la aplicación en todo momento desde la parte inferior (siguiente página).



Ilustración 17: Cambios portada.

#### 4.2.2. Detalles de los ejercicios.

Se ha decidido suprimir esta pantalla en consenso con el usuario de la aplicación ya que no aportaba ninguna información a mayores, de la que aporta el propio formulario de datos de los ejercicios. De esta forma se ahorra implementar una pantalla y desde el punto de vista del usuario puede acceder directamente a los detalles del ejercicio y modificarlos si lo ve necesario.

#### 4.2.3. Asociar jugadores y ejercicios a los entrenamientos.

En este punto se ha decidido cambiar el comportamiento anteriormente definido. En el prototipo presentado el usuario podía añadir ejercicios a los entrenamientos y después asociar jugadores del equipo a cada uno de los ejercicios. Comentando esto con nuestro usuario focal se ha decidido que realmente no necesita saber qué jugadores han participado en cada ejercicio sino que realmente lo que le interesa es guardar la información de qué jugadores han asistido o no al entrenamiento. Por todo esto en la pantalla de los detalles del entrenamiento se ha implementado un sistema pestañas mediante la cual se puede acceder a la lista de ejercicios y jugadores asociados al entrenamiento (siguiente página).

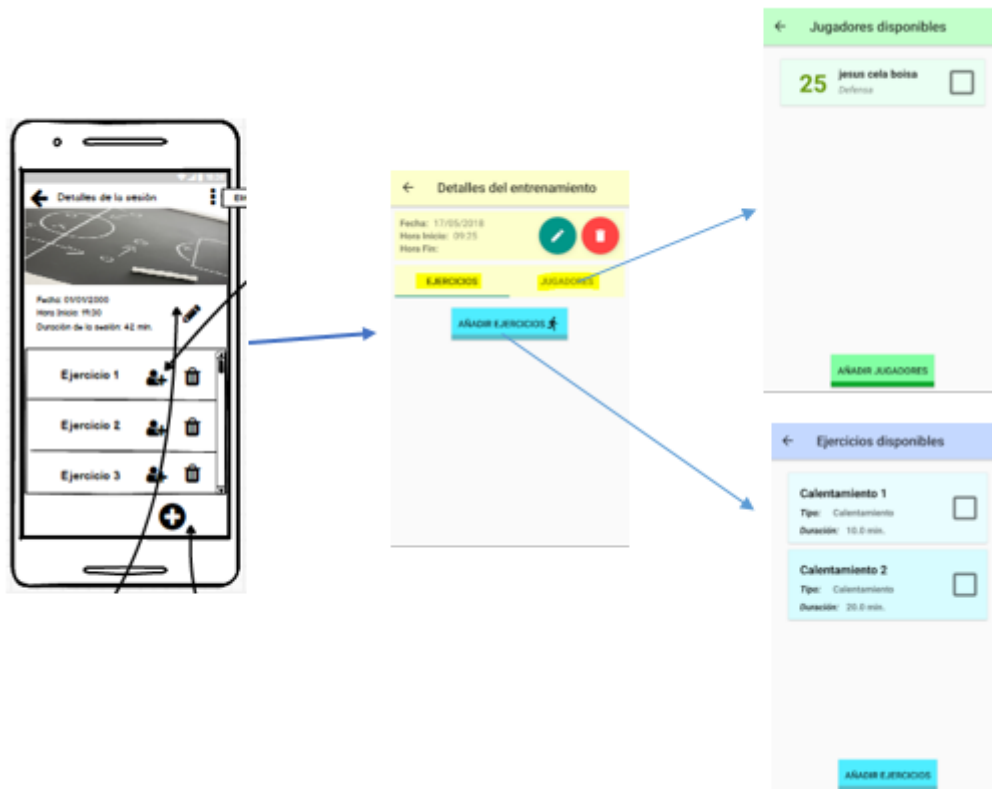


Ilustración 18: Cambios detalles entrenamiento.

#### 4.2.4. Gestión de los partidos.

Se han suprimido las pantallas de recopilación de datos del partido por falta de tiempo, se ha decidido así en consenso con el usuario focal ya que la parte importante de la aplicación es la gestión de los entrenamientos.

A consecuencia de suprimir esta parte de nacen los cambios de diseño explicados en el siguiente apartado

#### 4.2.5. Estadísticas del jugador.

Se suprime la pestaña de 'partidos' debido a lo ya comentado en el punto anterior y se exponen los datos en cuanto a número de entrenamientos y partidos convocados en la propia pantalla de detalles debajo de los datos personales del jugador (siguiente página).



Ilustración 19: Cambios detalles del jugador.

### 4.3. Decisiones sobre la funcionalidad.

En este apartado se describen las decisiones en cuanto a las tecnologías y librerías utilizadas para la simplificación del desarrollo de diferentes funcionalidades de la aplicación.

#### 4.3.1. Registro y autenticación del usuario.

Se ha utilizado Firebase Authentication para la gestión de los usuarios de la aplicación. Firebase Authentication nos proporciona los servicios de backend, SDK fáciles de usar y bibliotecas de IU ya elaboradas para autenticar a los usuarios de nuestra aplicación. De esta forma implementamos de manera sencilla una autenticación mediante email y contraseña [24].

Para incorporar Firebase Authentication a la aplicación se ha añadido la siguiente dependencia en el *build.gradle* de nuestra app.

```
implementation 'com.google.firebase:firebase-auth:12.0.1'
```

#### 4.3.2. Validación de los formularios.

Para la validación de los datos de los formularios se ha decidido incorporar la librería AwesomeValidation, esta librería nos permite seleccionar un estilo de validación, indicar las reglas necesarias que debe cumplir cada dato a validar y

elegir un punto en el cuál se ejecutará esa validación. De esta forma mediante esta librería se simplifica el código para validar los datos de los formularios y nos aporta una gran cantidad de herramientas para definir las reglas que debe cumplir cada dato [25].

Para incorporar AwesomeValidation a la aplicación se ha añadido la siguiente dependencia en el fichero *build.gradle* de la app:

```
implementation 'com.basgeekball:awesome-validation:4.1'
```

#### 4.3.3. Tratamiento de los campos fecha y hora.

Para el manejo de los campos fechas y horas nos hemos decidido por el uso de la librería Joda-time que simplifica en gran medida el manejo con este tipo de datos que siempre son bastante problemáticos a la hora de hacer conversiones y cálculos con ellos [26].

Para incorporar Joda-time a la aplicación se ha añadido la siguiente dependencia en el fichero *build.gradle* de la app:

```
implementation 'net.danlew:android.joda:2.9.9.3'
```

#### 4.3.4. Reutilización de código mediante Fragments.

En cuanto al uso de Fragments en la aplicación se ha decidido utilizar estos elementos de Android para implementar la listas de datos (entrenamientos, jugadores, ejercicios y partidos). A partir del prototipo inicial se observa que se utilizan los mismos tipos de listas en diferentes partes de la aplicación, lo único que cambia son los datos a mostrar o bien los Adapters a utilizar. En base a lo anterior lo que se ha decidido ha sido implementar un Fragment para cada una de las listas a mostrar y en función de los parámetros recibidos por el constructor de este se decide que Adapter debe mostrar ese Fragment y que funcionalidades va a implementar.

Para la implementación de las listas en los Fragments se ha utilizado la RecyclerView combinado con la CardView para cada elemento de la lista:

```
implementation 'com.android.support:cardview-v7:21.0.+'  
implementation 'com.android.support:recyclerview-v7:21.0.+'
```

También ha sido útil el uso de Fragment en la Actividad principal de la aplicación (MainActivity), ya que en esta es en donde se despliega el menú de navegación inferior que permite la navegación sencilla entre las diferentes secciones de la aplicación, reemplazando así los Fragments de la sección a mostrar en cada caso y siendo el menú de navegación visible en todo momento.

#### 4.3.5. Almacenamiento de los datos.

Como ya se ha comentado anteriormente en el apartado de diseño referente a la persistencia de los datos se ha utilizado Firebase Realtime Database para guardar la información de los usuarios y todos sus datos.

Dado que Firebase Realtime Database es una base de datos no relacional, el diseño del modelo Entidad-Relación mostrado en el apartado 3.2 de esta memoria se ha adaptado de la siguiente forma durante el desarrollo de la aplicación:

```
{  
  "email" : "jesuscelaboisa@uoc.edu",  
  "exercises" : {  
    "-LCcL8XZsm5Hxh2UTpX_" : {  
      "description" : "Calentamiento prueba",  
      "duration" : 20,  
      "duration_dist" : "",  
      "id_exercise" : "-LCcL8XZsm5Hxh2UTpX_",  
      "ids_trainings" : {  
        "-LCcLXBh822J8gYOSzSX" : true  
      },  
      "material" : "",  
      "name" : "Ejercicio 1",  
      "num_players" : 22,  
      "players_dist" : "",  
      "type" : "PRETR"  
    },  
  },  
  "matches" : {  
    "-LCcLSQ_wjnbcmdYGPfa" : {
```

```

        "date" : "24/05/2018",
        "end_time" : "18:01",
        "field" : "Pumar",
        "id_match" : "-LcCtLSQ_wjnbcmdYGPfa",
        "ids_players" : {
            "-LcCtLHvrgnlNmzLpMHIS" : true
        },
        "rival" : "Barsa",
        "start_time" : "16:01"
    }
},
"name" : "jesuscb",
"pass" : "e10adc3949ba59abbe56e057f20f883e",
"players" : {
    "-LcCtLHvrgnlNmzLpMHIS" : {
        "birthdate" : "16/05/1993",
        "demarcation" : "DF",
        "id_player" : "-LcCtLHvrgnlNmzLpMHIS",
        "identity" : "12345678Z",
        "ids_matches" : {
            "-LcCtLSQ_wjnbcmdYGPfa" : true
        },
        "ids_trainings" : {
            "-LcCtLXBh822J8gYOSzSX" : true
        },
        "name" : "Jesús",
        "numDorsal" : 15,
        "position" : "CENL",
        "surname" : "Cela Boisa"
    }
},
"trainings" : {
    "-LcCtLXBh822J8gYOSzSX" : {
        "date" : "03/05/2018",
        "end_time" : "23:00",
        "id_training" : "-LcCtLXBh822J8gYOSzSX",
        "ids_exercises" : {
            "-LcCtL8XZsm5Hxh2UTpX_" : true
        },
        "ids_players" : {
            "-LcCtLHvrgnlNmzLpMHIS" : true
        },
        "start_time" : "21:00"
    }
},
"uid" : "mIzkkTGY5ORw7Au4m0YlSsxw3Dp1"
}

```

Los datos mostrados anteriormente son los generados por la aplicación para un usuario que ha añadido un ejercicio, un jugador, un entrenamiento y un partido asignando posteriormente al entrenamiento el ejercicio y el jugador creado y asignando al partido el jugador creado.

Para utilizar la librería de Realtime Database de Firebase se ha añadido la siguiente línea en el fichero build.gradle de la app:

```
implementation 'com.google.firebase:firebase-database:12.0.1'
```

También se ha habilitado la persistencia de los datos en modo sin conexión para que el usuario pueda seguir trabajando aun no teniendo conexión a internet, esto se hace mediante las siguientes líneas de código:

```
private var mDatabase: FirebaseDatabase  
mDatabase = FirebaseDatabase.getInstance()  
mDatabase.setPersistenceEnabled(true)
```

## 4.4. Estructuración del código.

El código se ha estructurado en diferentes paquetes dentro de la carpeta *java*:

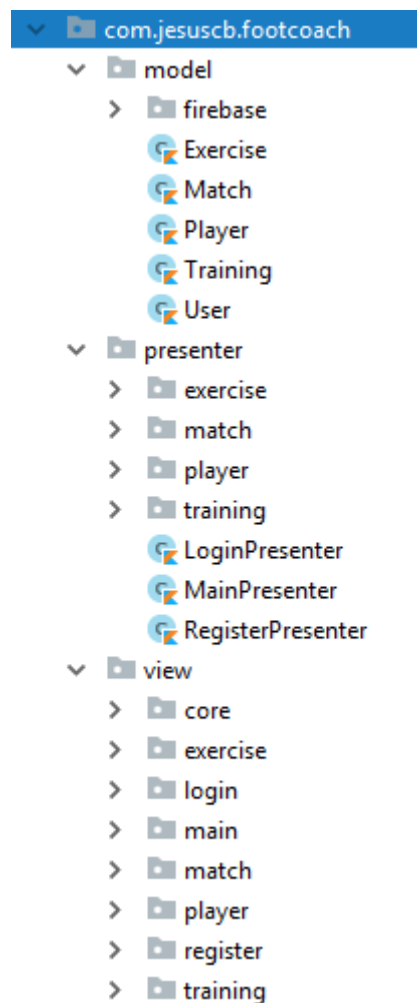


Ilustración 20: Estructura paquetes MVP.



Como se puede ver en la imagen anterior se han clasificado los paquetes por capas siguiendo la arquitectura MVP. En el caso de que existan varias vistas, presentadores o clases de modelo que pertenecen o trabajan con el mismo grupo de datos o funcionalidades se han agrupado en sub-paquetes según temática.

Dentro de esta estructura en la capa de modelo se ha creado un sub-paquete llamado *firebase*, destinado a albergar todas aquellas clases que se conectan a la base de datos de Firebase Realtime Database mediante su API, a estas clases e interfaces se les ha llamado *interactors* (acceden a la base de datos), funcionan a modo de DAO y son un punto de comunicación entre los presentadores y el modelo.

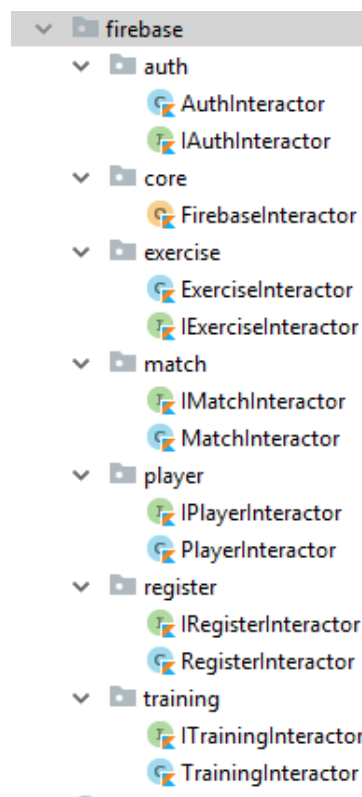


Ilustración 21: Paquete firebase.

Existe un interactor base (FirebaseInteractor), que será el encargado de inicializarse al arrancar la aplicación e inicializar los demás interactores en el momento preciso, así como devolver una instancia a los presentadores que los requieran.

## 4.5. Problemas encontrados.

El principal problema encontrado ha sido la falta de conocimiento y experiencia de uso de las tecnologías y librerías utilizadas, como Firebase Authentication, Firebase Realtime Database, la implementación de la propia arquitectura MVP o el uso del lenguaje de programación Kotlin.

Aunque habiendo realizado alguna práctica durante el máster en el que se ha utilizado Firebase Realtime Database, implementarlo a nivel de una aplicación más grande ha supuesto un gran esfuerzo y adaptación a la hora de trabajar con los datos en una base de datos no relacional.

La implementación de una estructura MVP en el proyecto ha hecho que la implementación de funcionalidades fuese más lenta durante las primeras semanas del desarrollo pero por su parte cuando la estructura ha estado definida completamente resultaba bastante ágil incorporar nuevas funcionalidades y pantallas a la aplicación.

El uso de un lenguaje de programación desconocido por el desarrollador como Kotlin también ha hecho que el desarrollo de las primeras líneas de código fuese mayor, aunque a medida que se cogía experiencia en el manejo de este nuevo lenguaje se observó un aumento en la productividad ya que ahorra bastantes líneas de código con respecto a Java.

## 5. Validación y pruebas.

En este apartado se llevará a cabo un proceso de pruebas sobre la aplicación que dará lugar a la validación, o no, de los requisitos definidos anteriormente.

Las pruebas son un instrumento que determina el estado de calidad de un producto de software y sirve para identificar posibles fallos en la implementación, calidad o usabilidad del software. El proceso de pruebas no se limita a la fase final de la implementación, deben ser definidas y ejecutadas a lo largo del proceso de implementación, en el caso de un ciclo de vida incremental a lo largo de cada incremento y al final de cada, de forma exhaustiva.

Las pruebas ejecutadas serán especificadas mediante la siguiente tabla:

<b>ID</b>	Identificador único de la prueba Pr-x siendo x el número de prueba.
<b>Descripción</b>	Explicación breve de la prueba.
<b>Requisitos</b>	Requisitos probados.
<b>Entradas</b>	Datos necesarios de entrada para ejecutar la prueba.
<b>Procedimiento</b>	Descripción de los pasos a realizar para llevar a cabo la prueba.
<b>Resultado</b>	Descripción del estado del sistema tras la ejecución de la prueba.

Tabla 49: Plantilla especificación pruebas.

## 5.1. Pruebas.

A continuación se exponen las pruebas funcionales llevadas a cabo en conjunto por el desarrollador y el usuario focal de la aplicación, no se han incluido pruebas unitarias en el proyecto por falta de tiempo.

<b>ID</b>	Pr-1
<b>Descripción</b>	Comprobar que se pueden crear ejercicios.
<b>Requisitos</b>	RF-1
<b>Entradas</b>	Datos del ejercicio.
<b>Procedimiento</b>	Se accede al formulario para añadir un ejercicio, se rellenan los datos y se envía el formulario.
<b>Resultado</b>	El ejercicio queda creado en la base de datos.

Tabla 50: Pr-1: Crear ejercicio.

<b>ID</b>	Pr-2
<b>Descripción</b>	Comprobar que se pueden modificar los datos de un ejercicio.
<b>Requisitos</b>	RF-2
<b>Entradas</b>	Datos modificados del ejercicio.
<b>Procedimiento</b>	Se accede al formulario de editar un ejercicio, se modifican los datos necesarios y se envía el formulario.
<b>Resultado</b>	El ejercicio queda modificado en la base de datos.

Tabla 51: Pr-2: Modificar ejercicio.

<b>ID</b>	Pr-3
<b>Descripción</b>	Comprobar que se puede eliminar un ejercicio.
<b>Requisitos</b>	RF-3
<b>Entradas</b>	El id del ejercicio.
<b>Procedimiento</b>	Se accede a la lista de ejercicios y se mantiene pulsado sobre el ejercicio que se desea eliminar, posteriormente se confirma la acción.

<b>Resultado</b>	El ejercicio queda eliminado de la base de datos.
------------------	---

**Tabla 52: Pr-3: Eliminar ejercicio.**

<b>ID</b>	Pr-4
<b>Descripción</b>	Comprobar que se puede crear una sesión de entrenamiento.
<b>Requisitos</b>	RF-4
<b>Entradas</b>	Datos básicos del entrenamiento.
<b>Procedimiento</b>	Se accede al formulario de crear entrenamiento, se rellenan los datos y se envía el formulario.
<b>Resultado</b>	El entrenamiento queda registrado en el sistema.

**Tabla 53: Pr-4: Crear entrenamiento.**

<b>ID</b>	Pr-5
<b>Descripción</b>	Comprobar que se pueden modificar los datos de una sesión de entrenamiento.
<b>Requisitos</b>	RF-5
<b>Entradas</b>	Datos a modificar en el entrenamiento.
<b>Procedimiento</b>	Se accede al formulario para editar los entrenamientos, se cambian los datos deseados y se envía el formulario.
<b>Resultado</b>	Los datos del entrenamiento se modifican en el sistema.

**Tabla 54: Pr-5: Modificar entrenamiento.**

<b>ID</b>	Pr-6
<b>Descripción</b>	Comprobar que se puede asociar ejercicios a los entrenamientos.
<b>Requisitos</b>	RF-5
<b>Entradas</b>	Ejercicios a asociar.
<b>Procedimiento</b>	Se accede a los detalles del entrenamiento y mediante el botón de “añadir ejercicios” se seleccionan los ejercicios deseados y se envía el formulario.
<b>Resultado</b>	Los ejercicios quedan asociados al entrenamiento.

**Tabla 55: Pr-6: Asociar ejercicios a entrenamiento.**

<b>ID</b>	Pr-7
<b>Descripción</b>	Comprobar que se puede asociar jugadores a un entrenamiento.
<b>Requisitos</b>	RF-5
<b>Entradas</b>	Jugadores a asociar.
<b>Procedimiento</b>	Se accede a los detalles del entrenamiento y mediante el botón de “añadir jugadores” se seleccionan los jugadores deseados y se envía el formulario.
<b>Resultado</b>	Los jugadores quedan asociados al entrenamiento.

**Tabla 56: Pr-7: Asociar jugadores a entrenamiento.**

<b>ID</b>	Pr-8
<b>Descripción</b>	Comprobar que se pueden eliminar los datos de un entrenamiento.
<b>Requisitos</b>	RF-6
<b>Entradas</b>	El id del entrenamiento.
<b>Procedimiento</b>	Se accede a la lista de entrenamientos y se mantiene pulsado sobre el entrenamiento que se desea eliminar, posteriormente se confirma la acción.
<b>Resultado</b>	El entrenamiento queda eliminado de la base de datos.

**Tabla 57: Pr-8: Eliminar entrenamiento.**

<b>ID</b>	Pr-9
<b>Descripción</b>	Comprobar que se puede crear jugadores.
<b>Requisitos</b>	RF-7
<b>Entradas</b>	Datos del jugador.
<b>Procedimiento</b>	Se accede al formulario de crear jugador, se rellenan los datos y se envía el formulario.
<b>Resultado</b>	El jugador queda guardado en la base de datos.

**Tabla 58: Pr-9: Crear jugador.**

<b>ID</b>	Pr-10
<b>Descripción</b>	Comprobar que se puede modificar los datos de los jugadores.
<b>Requisitos</b>	RF-8
<b>Entradas</b>	Datos a modificar del jugador.
<b>Procedimiento</b>	Se accede al formulario para editar los jugadores, se cambian los datos deseados y se envía el formulario.
<b>Resultado</b>	Los datos del jugador se modifican en la base de datos.

**Tabla 59: Pr-10: Modificar jugador.**

<b>ID</b>	Pr-11
<b>Descripción</b>	Comprobar que se pueden eliminar los jugadores.
<b>Requisitos</b>	RF-9
<b>Entradas</b>	El id del jugador.
<b>Procedimiento</b>	Se accede a la lista de jugadores y se mantiene pulsado sobre el jugador que se desea eliminar, posteriormente se confirma la acción.
<b>Resultado</b>	El jugador queda eliminado de la base de datos.

**Tabla 60: Pr-11: Eliminar jugador.**

<b>ID</b>	Pr-12
<b>Descripción</b>	Comprobar que se puede crear partidos.
<b>Requisitos</b>	RF-10
<b>Entradas</b>	Datos básicos del partido.
<b>Procedimiento</b>	Se accede al formulario de crear partido, se rellenan los datos y se envía el formulario.
<b>Resultado</b>	El partido queda guardado en el sistema.

**Tabla 61: Pr-12: Crear partido.**

<b>ID</b>	Pr-13
<b>Descripción</b>	Comprobar que se puede modificar los datos del partido.
<b>Requisitos</b>	RF-11
<b>Entradas</b>	Datos a modificar del partido.
<b>Procedimiento</b>	Se accede al formulario para editar los partidos, se cambian los datos deseados y se envía el formulario.
<b>Resultado</b>	Los datos del partido quedan modificados en la base de datos.

**Tabla 62: Pr-13: Modificar partido.**

<b>ID</b>	Pr-14
<b>Descripción</b>	Comprobar que se pueden asociar jugadores convocados a un partido.
<b>Requisitos</b>	RF-11
<b>Entradas</b>	Lista de jugadores a asociar.
<b>Procedimiento</b>	Se accede a los detalles del partido y mediante el botón de “añadir jugadores” se seleccionan los jugadores deseados y se envía el formulario.
<b>Resultado</b>	Los jugadores seleccionados quedan asociados al partido.

**Tabla 63: Pr-14: Asociar jugadores a partido.**

<b>ID</b>	Pr-15
<b>Descripción</b>	Comprobar que se puede eliminar los datos de los partidos.
<b>Requisitos</b>	RF-12
<b>Entradas</b>	El id del partido.
<b>Procedimiento</b>	Se accede a la lista de partidos y se mantiene pulsado sobre el partido que se desea eliminar, posteriormente se confirma la acción.
<b>Resultado</b>	El partido queda eliminado de la base de datos.

**Tabla 64: Pr-15: Eliminar partido.**



<b>ID</b>	Pr-16
<b>Descripción</b>	Comprobar que se puede visualizar la lista de ejercicios.
<b>Requisitos</b>	RF-13
<b>Entradas</b>	Lista de ejercicios del usuario.
<b>Procedimiento</b>	Acceder a la lista de ejercicios mediante el botón habilitado para ello.
<b>Resultado</b>	Se visualiza la lista de entrenamientos.

**Tabla 65: Pr-16: Listar ejercicios.**

<b>ID</b>	Pr-17
<b>Descripción</b>	Comprobar que se puede visualizar la lista de entrenamientos.
<b>Requisitos</b>	RF-14
<b>Entradas</b>	Lista de entrenamientos del usuario.
<b>Procedimiento</b>	Acceder a la lista de entrenamientos mediante el botón habilitado para ello.
<b>Resultado</b>	Se visualiza la lista de entrenamientos.

**Tabla 66: Pr-17: Listar entrenamientos.**

<b>ID</b>	Pr-18
<b>Descripción</b>	Comprobar que se puede visualizar la lista de los jugadores.
<b>Requisitos</b>	RF-15
<b>Entradas</b>	Lista de jugadores del usuario.
<b>Procedimiento</b>	Acceder a la lista de jugadores mediante el botón habilitado para ello.
<b>Resultado</b>	Se visualiza la lista de jugadores.

**Tabla 67: Pr-18: Listar jugadores.**

<b>ID</b>	Pr-19
<b>Descripción</b>	Comprobar que se puede visualizar la lista de partidos.
<b>Requisitos</b>	RF-16
<b>Entradas</b>	Lista de partidos del usuario.
<b>Procedimiento</b>	Acceder a la lista de partidos mediante el botón habilitado para ello.
<b>Resultado</b>	Se visualiza la lista de partidos añadidos.

**Tabla 68: Pr-19: Listar partidos.**

<b>ID</b>	Pr-20
<b>Descripción</b>	Comprobar que la aplicación genera una sesión de entrenamiento de manera automática.
<b>Requisitos</b>	RF-18
<b>Entradas</b>	Datos básicos del entrenamiento y tipos de ejercicios que deberá incluir.
<b>Procedimiento</b>	El usuario accede al formulario y rellena los datos básicos del entrenamiento junto con el tipo de ejercicios que este debe incluir.
<b>Resultado</b>	La aplicación genera un entrenamiento con los datos solicitados por el usuario.

**Tabla 69: Pr-20: Generar entrenamiento automático.**

<b>ID</b>	Pr-21
<b>Descripción</b>	Comprobar que los usuarios se pueden registrar en el sistema.
<b>Requisitos</b>	RF-19
<b>Entradas</b>	Nombre y credenciales del usuario.
<b>Procedimiento</b>	El usuario accede al formulario de registro donde rellena todos los datos y envía el formulario.
<b>Resultado</b>	El usuario queda registrado correctamente en el sistema.

**Tabla 70: Pr-21: Registrar usuario.**

<b>ID</b>	Pr-22
<b>Descripción</b>	Comprobar que los usuarios pueden iniciar sesión en el sistema.
<b>Requisitos</b>	RF-20
<b>Entradas</b>	Credenciales del usuario.
<b>Procedimiento</b>	El usuario introduce sus credenciales y las envía para su validación.
<b>Resultado</b>	El usuario queda autenticado en el sistema.

**Tabla 71: Pr-22: Loguear usuario.**

<b>ID</b>	Pr-23
<b>Descripción</b>	Comprobar que la aplicación permite visualizar los datos del jugador.
<b>Requisitos</b>	RF-23
<b>Entradas</b>	El id del jugador.
<b>Procedimiento</b>	Se accede a la lista de jugadores y se pulsa sobre el jugador que se desea visualizar.
<b>Resultado</b>	El sistema muestra los detalles del mismo.

**Tabla 72: Pr-23: Ver detalles del jugador.**

## **5.2. Validación de requisitos.**

El proceso de validación de requisitos permite demostrar que los requerimientos definidos en el sistema son válidos, consistentes y completos (los que realmente quiere el cliente); además, revisa que no se haya omitido ninguno, que no sean ambiguos, inconsistentes o redundantes. Una vez finalizadas las pruebas se comprueba si el sistema cumple con los requisitos establecidos. A continuación, se mostrará un listado con todos los requisitos que han sido especificados para el sistema. Su estructura será la siguiente:

<b>Identificador del requisito</b>	
<b>Descripción</b>	Breve explicación del requisito
<b>Estado</b>	No implementado Parcial Completo
<b>Justificación</b>	Breve explicación acerca de los motivos del grado de cumplimiento en caso de no ser 'Completo'.
<b>Pruebas</b>	Pruebas empleadas para la validación.

**Tabla 73: Plantilla especificación de validación de requisitos.**

A continuación se validan los requisitos del proyecto, tanto los funcionales como los no funcionales.

<b>RF-1</b>	
<b>Descripción</b>	Crear ejercicio
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-1

**Tabla 74: Validación RF-1**

<b>RF-2</b>	
<b>Descripción</b>	Modificar ejercicio
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-2

**Tabla 75: Validación RF-2**

<b>RF-3</b>	
<b>Descripción</b>	Eliminar ejercicio
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-3

**Tabla 76: Validación RF-3**

<b>RF-4</b>	
<b>Descripción</b>	Crear sesión de entrenamiento
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-4

**Tabla 77: Validación RF-4**

<b>RF-5</b>	
<b>Descripción</b>	Modificar sesión de entrenamiento
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-5, Pr-6, Pr-7

**Tabla 78: Validación RF-5**

<b>RF-6</b>	
<b>Descripción</b>	Eliminar sesión de entrenamiento
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-8

**Tabla 79: Validación RF-6**

<b>RF-7</b>	
<b>Descripción</b>	Crear jugador
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-9

**Tabla 80: Validación RF-7**

<b>RF-8</b>	
<b>Descripción</b>	Modificar jugador
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-10

**Tabla 81: Validación RF-8**

<b>RF-9</b>	
<b>Descripción</b>	Eliminar jugador
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-11

**Tabla 82: Validación RF-9**

<b>RF-10</b>	
<b>Descripción</b>	Crear partido
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-12

**Tabla 83: Validación RF-10**

<b>RF-11</b>	
<b>Descripción</b>	Modificar partido
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-13, Pr-14

**Tabla 84: Validación RF-11**

<b>RF-12</b>	
<b>Descripción</b>	Eliminar partido
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-15

**Tabla 85: Validación RF-12**

<b>RF-13</b>	
<b>Descripción</b>	Listar ejercicios
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-16

**Tabla 86: Validación RF-13**

<b>RF-14</b>	
<b>Descripción</b>	Listar sesiones de entrenamiento
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-17

**Tabla 87: Validación RF-14**

<b>RF-15</b>	
<b>Descripción</b>	Listar jugadores
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-18

**Tabla 88: Validación RF-15**

<b>RF-16</b>	
<b>Descripción</b>	Listar partidos
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-19

**Tabla 89: Validación RF-16**

<b>RF-17</b>	
<b>Descripción</b>	Guardar preferencias del usuario
<b>Estado</b>	No implementado
<b>Justificación</b>	Descartado por el usuario, no se le veía utilidad.
<b>Pruebas</b>	---

**Tabla 90: Validación RF-17**

<b>RF-18</b>	
<b>Descripción</b>	Generar sesión de entrenamiento
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-20

**Tabla 91: Validación RF-18**



<b>RF-19</b>	
<b>Descripción</b>	Registro de usuario
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-21

**Tabla 92: Validación RF-19**

<b>RF-20</b>	
<b>Descripción</b>	Login de usuario
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	Pr-22

**Tabla 93: Validación RF-20**

<b>RF-21</b>	
<b>Descripción</b>	Generar fichero de texto
<b>Estado</b>	No implementado
<b>Justificación</b>	Falta de tiempo y prioridad baja
<b>Pruebas</b>	---

**Tabla 94: Validación RF-21**

<b>RF-22</b>	
<b>Descripción</b>	Guardar datos asociados a un partido
<b>Estado</b>	No implementado
<b>Justificación</b>	Falta de tiempo y prioridad baja
<b>Pruebas</b>	---

**Tabla 95: Validación RF-22**

RF-23	
Descripción	Consultar y resumir datos del jugador
Estado	Parcial
Justificación	Se podrían mostrar más datos
Pruebas	Pr-23

Tabla 96: Validación RF-23

RUsab-1	
Descripción	Formularios sencillos
Estado	Parcial
Justificación	Es muy subjetivo...
Pruebas	---

Tabla 97: Validación RUsab-1

RUsab-2	
Descripción	Asistencia al usuario
Estado	Parcial
Justificación	Se han utilizado los elementos de Android como timepickers, datepickers, dialogs con buttons, etc, puede quedar margen de mejora con los campos de texto.
Pruebas	---

Tabla 98: Validación RUsab-2

Rlfaz-1	
Descripción	Contraste de colores
Estado	Completo
Justificación	---
Pruebas	---

Tabla 99: Validación Rlfaz-1

<b>RSoft-1</b>	
<b>Descripción</b>	Plataforma 5.0 o superior
<b>Estado</b>	Parcial
<b>Justificación</b>	Funciona correctamente en todos los dispositivos en los que se ha podido probar.
<b>Pruebas</b>	---

**Tabla 100: Validación RSoft-1**

<b>RDis-1</b>	
<b>Descripción</b>	Firebase
<b>Estado</b>	Completo
<b>Justificación</b>	---
<b>Pruebas</b>	---

**Tabla 101: Validación RDis-1**

## 6. Conclusiones

### 6.1. Grado de cumplimiento de los objetivos propuestos.

Después de haber estudiado, analizado, diseñado, implementado y probado el producto resultante, se concluye esta memoria afirmando que se ha conseguido crear una aplicación que permite a los entrenadores de fútbol base ahorrar tiempo en sus tareas alcanzando ampliamente los objetivos inicialmente contemplados. La aplicación desarrollada para la plataforma móvil Android permite al usuario crear de manera sencilla y rápida la planificación de entrenamientos y registrar la actividad de sus jugadores tanto en entrenamientos como en partidos.

Se ha creado una interfaz de usuario que se adapta a las malas condiciones de visibilidad que se pueden dar en un campo de fútbol descubierto a diversas horas del día.

Se ha incluido un sistema de registro y login para asegurar la seguridad de los datos del usuario y se han utilizado los servicios de Firebase Realtime Database para asegurar la persistencia de estos datos en la nube.

No se ha podido implementar la gestión de la convocatoria a los partidos de la manera que se hubiera deseado ni la recopilación de datos durante el mismo por falta de tiempo. En este sentido la aplicación desarrollada sólo permite la creación de partidos de fútbol con la gestión de la lista de jugadores convocados al mismo pero no llega a cumplir con el objetivo contemplado inicialmente.

## **6.2. Seguimiento de la planificación.**

En este sentido se puede afirmar que en líneas generales se ha seguido y se ha cumplido la planificación estimada. Esta planificación constaba de varios hitos correspondientes a cada una de las entregas de las diferentes PEC.

Para la PEC1 se han cumplido los plazos en donde se desarrolló susodicha planificación para el resto del proyecto y se decidió el tema del mismo.

Para la PEC2, que comprendía la parte de análisis y diseño de la aplicación, también se cumplieron los plazos esperados ya que apenas hubo que hacer correcciones sobre la PEC anterior y las funcionalidades principales de la aplicación y su representación gráficas estaban bastante claras.

En la PEC3 referente a la implementación y pruebas del producto ha habido algún desajuste horas sobre todo en el primer incremento debido al desconocimiento de las tecnologías utilizadas, que supuso un tiempo de aprendizaje que hizo que la productividad se viese reducida en este primer incremento. Pese a esto se ha realizado la entrega de la PEC3 según los plazos establecidos, de modo que el producto final estuvo operativo según lo previsto.

## **6.3. Líneas de trabajo futuro**

A menudo, durante el desarrollo de un proyecto informático surgen nuevas ideas y mejoras complementarias a los objetivos principales. Se proponen a continuación posibles ampliaciones que podrían mejorar el producto en un futuro:

- Implementar el concepto de equipo: con esto se pretende proporcionar al usuario la capacidad de gestionar diferentes equipos con diferentes jugadores ya que se puede dar el caso que un mismo entrenador sea encargado de más de un equipo. De esta forma podría gestionar diferentes jugadores y planificar diferentes entrenamientos en función de

cada equipo, actualmente sería necesario que el usuario crease una nueva cuenta.

- **Mostrar más datos de los jugadores:** se podría recopilar más información de la actividad de los jugadores y mostrarla en sus detalles a modo de informe. Se podría mostrar un resumen de entrenamientos realizados agrupados por semanas o por meses, o también mostrar información recopilada durante el partido para el jugador.
- **Adaptar la aplicación para Tablets:** aunque es común que las personas lleven el móvil encima durante todo el día estaría bien que la aplicación se visualizase correctamente y aprovechara de manera eficiente el espacio en una pantalla más grande tratando de ofrecer mayor usabilidad al usuario a la hora de realizar las tareas.
- **Permitir gestionar el perfil del usuario:** el usuario elige su nombre y su contraseña durante el registro pero estos datos no los puede modificar posteriormente. En este aspecto se podría habilitar una sección en donde el usuario pudiera modificar sus datos y sobre todo, poder restablecer su contraseña.
- **Habilitar perfil de los jugadores:** permitir a los jugadores acceder a la aplicación con un perfil de 'jugador' para así poder ver los datos que el entrenador recopila de él.

## 7. Glosario

**UML:** (Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad

**CRUD:** es el acrónimo de "Crear, Leer, Actualizar y Borrar" (del original en inglés: Create, Read, Update and Delete), que se usa para referirse a las funciones básicas en bases de datos o la capa de persistencia en un software.

**API:** (Application Programming Interface) es un conjunto de rutinas que provee acceso a funciones de un determinado software.

**DAO:** (data access object) es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo.

**SDK:** (Software Development Kit) es un conjunto de herramientas que sirven para desarrollar software informático.

**Widget:** es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños que pretende dar fácil acceso a funciones frecuentemente usadas y proveer de información visual.

## 8. Bibliografía

[1] GooglePlay, *Entrenamiento de Fútbol Gratis*-[08/03/2018]:

<<https://play.google.com/store/apps/details?id=com.sportsapps.entrenamiento.de.futbol&hl=es>>

[2] GooglePlay, *Ejercicios de fútbol*-[08/03/2018]:

<<https://play.google.com/store/apps/details?id=net.easy2coach.exercisesDatabase&hl=es>>

[3] GooglePlay e iTunes, *Ejercicios Fútbol Base*-[08/03/2018]:

<[https://play.google.com/store/apps/details?id=com.grupocampus.ejercicios&hl=es\\_419](https://play.google.com/store/apps/details?id=com.grupocampus.ejercicios&hl=es_419)>

[4] iTunes, *Ejercicios Fútbol Base*-[04/04/2018]:

<<https://itunes.apple.com/es/app/ejercicios-f%C3%BAtbol-base/id579233423?mt=8>>

[5] iTunes, *Entrenamiento de fútbol*-[04/04/2018]:

<<https://itunes.apple.com/es/app/entrenamiento-de-f%C3%BAtbol/id486955502?mt=8>>

[6] iTunes, *El entrenador de fútbol*-[04/04/2018]:

<<https://itunes.apple.com/es/app/el-entrenador-de-f%C3%BAtbol/id415531025?mt=8>>

[7] Blogspot, *Metodología en cascada*-[04/04/2018]:

<<http://metodologiaencascada.blogspot.com.es/>>

[8] Wikipedia, *Desarrollo en cascada*-[04/04/2018]:

<[https://es.wikipedia.org/wiki/Desarrollo\\_%C3%A1gil\\_de\\_software](https://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software)>

[9] Blogspot, *Modelo en espiral*-[04/04/2018]:

<<http://modeloespiral.blogspot.com.es/>>



- [10] Wikipedia, Desarrollo en espiral-[04/04/2018]:  
<[https://es.wikipedia.org/wiki/Desarrollo\\_en\\_espiral](https://es.wikipedia.org/wiki/Desarrollo_en_espiral)>
- [11] Wikipedia, Desarrollo iterativo y creciente-[04/04/2018]:  
<[https://es.wikipedia.org/wiki/Desarrollo\\_iterativo\\_y\\_creciente](https://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente)>
- [12] Wikipedia, Desarrollo ágil de software-[04/04/2018]:  
<[https://es.wikipedia.org/wiki/Desarrollo\\_%C3%A1gil\\_de\\_software](https://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software)>
- [13] Proyectos agiles.org, *Desarrollo iterativo e incremental*-[08/03/2018]:  
<<https://proyectosagiles.org/desarrollo-iterativo-incremental/>>
- [14] okDiario.com, *Android es el sistema líder*-[04/04/2018]:  
<<https://canales.okdiario.com/tecnologia/2017/06/15/android-lider-ios-crecer-cuota-mercado-52890>>
- [15] andro4all.com, *Android pulveriza a iOS en la mayoría de países*-[04/04/2018]:  
<<https://andro4all.com/2017/09/android-vs-ios-lider-mercado>>
- [16] rfef.es, *Escuela Nacional de Entrenadores*-[04/04/2018]:  
< <http://www.rfef.es/formacion/escuela-de-entrenadores>>
- [17] IBM.com, *Definición de casos de uso*-[04/04/2018]:  
<[https://www.ibm.com/support/knowledgecenter/es/SS2U2U\\_10.0.0/com.ibm.pim.dev.doc/pim\\_tsk\\_arc\\_definingusecases.html](https://www.ibm.com/support/knowledgecenter/es/SS2U2U_10.0.0/com.ibm.pim.dev.doc/pim_tsk_arc_definingusecases.html)>
- [18] GoogleSites, *Estructuración y especificación de casos de uso*-[04/04/2018]:  
< <https://sites.google.com/site/alfonsoperezr/investigacion/estructuracin-y-especificacin-de-casos-de-uos>>
- [19] Wikipedia, *Requisito funcional*-[04/04/2018]:  
< [https://es.wikipedia.org/wiki/Requisito\\_funcional](https://es.wikipedia.org/wiki/Requisito_funcional)>
- [20] Wikipedia, *Requisito no funcional*-[04/04/2018]:  
< [https://es.wikipedia.org/wiki/Requisito\\_no\\_funcional](https://es.wikipedia.org/wiki/Requisito_no_funcional)>

[21] Katade.com, *MVP como patrón de arquitectura Android*-[04/04/2018]:

< <http://katade.com/2016/02/05/mvp-patron-arquitectura-android/>>

[22] develapps.com, *Modelo Vista Presentador en Android*-[04/04/2018]:

< <http://www.develapps.com/es/noticias/modelo-vista-presentador-mvp-en-android>>

[23] Developers Google, *Android Studio* -[15/05/2018]:

< <https://developer.android.com/studio/?hl=es-419> >

[24] Firebase Authentication -[16/05/2018]:

< <https://firebase.google.com/docs/auth/?authuser=0> >

[25] GitHub, *AwesomeValidation* -[16/05/2018]:

< <https://github.com/thyrlion/AwesomeValidation> >

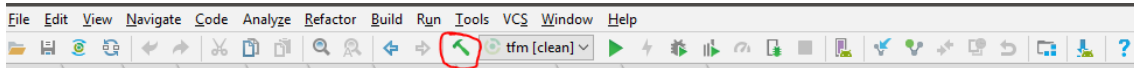
[26] GitHub, *Joda-time* -[16/05/2018]:

< <https://github.com/dlew/joda-time-android> >

## 9. Anexos

### 9.1. Anexo A: Instrucciones de compilación.

Abrir el proyecto con Android Studio y ejecutar el '*make project*'



### 9.2. Anexo B: Manual de usuario.

En esta sección se presentará un manual que guiará al usuario a través de todas las funcionalidades de la aplicación. Se trata de abordar todas las funcionalidades de una forma ordenada tal que sea lo más natural posible.

#### Registro y Login

Tras iniciar la aplicación por primera vez se cargará la pantalla de Login, antes de nada deberemos crear una nueva cuenta mediante el enlace de “Crear una cuenta!”, y accederemos al siguiente formulario.

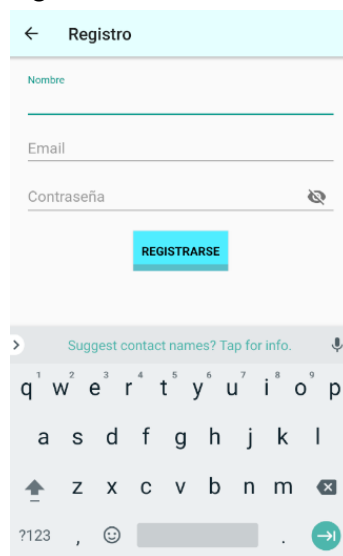
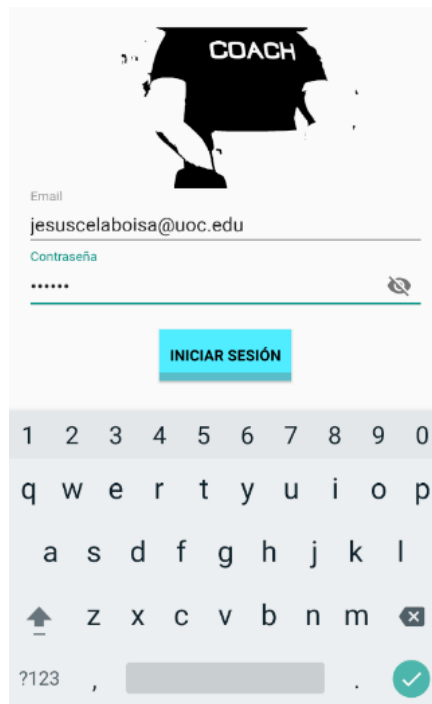


Ilustración 22: Pantalla registro.

Deberemos rellenar nuestros datos de acceso con nuestro nombre para ser dados de alta en el sistema. Una vez hecho esto quedaremos registrados y autenticados en el sistema

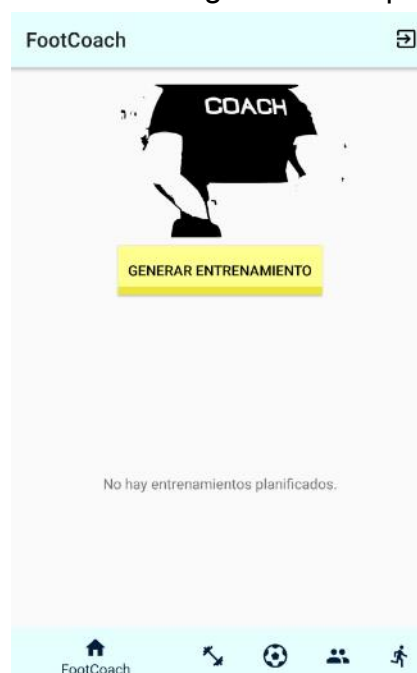
Las próximas veces que deseemos iniciar sesión solamente tendremos que rellenar nuestras credenciales en la pantalla de Login.



**Ilustración 23: Pantalla de Login.**

### **Pantalla de Home**

Al autenticarse en el sistema accederemos a la pantalla principal de la aplicación donde tendremos acceso rápido a las distintas secciones de la aplicación mediante una barra de navegación en la parte inferior.



**Ilustración 24: Pantalla de Home.**

En esta pantalla dispondremos de una lista de los próximos entrenamientos planificados y un botón que nos dará acceso al formulario de generación de un nuevo entrenamiento de forma automática, esto lo comentaremos más adelante, ahora comenzaremos a navegar por las distintas secciones de la aplicación.

### Gestión de ejercicios

Lo primero que haremos será crear los diferentes ejercicios de los que se compondrán nuestras sesiones de entrenamiento, para ello accederemos mediante el primer botón por la derecha a la sección de ejercicios.

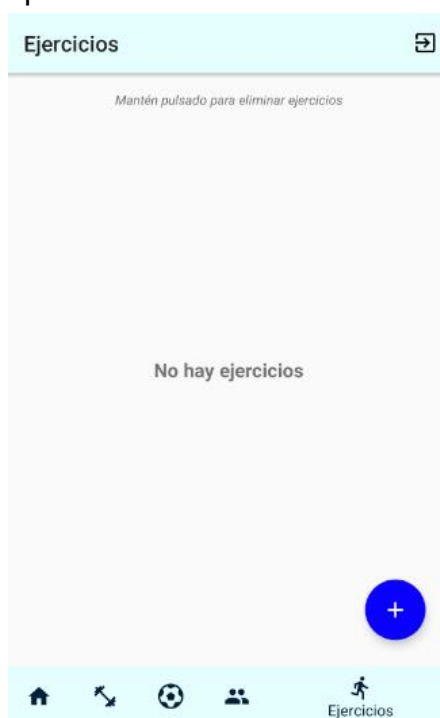


Ilustración 25: Pantalla ejercicios 1.

Mediante el botón azul podremos añadir un nuevo ejercicio a nuestra lista de ejercicios y posteriormente podremos añadir este ejercicio a nuestros entrenamientos. Una vez añadidos los ejercicios aparecerán en una lista mediante la cual podremos acceder a los detalles de cada uno de ellos y modificar sus datos o bien eliminarlos pulsando con el dedo sobre ellos más de un segundo.

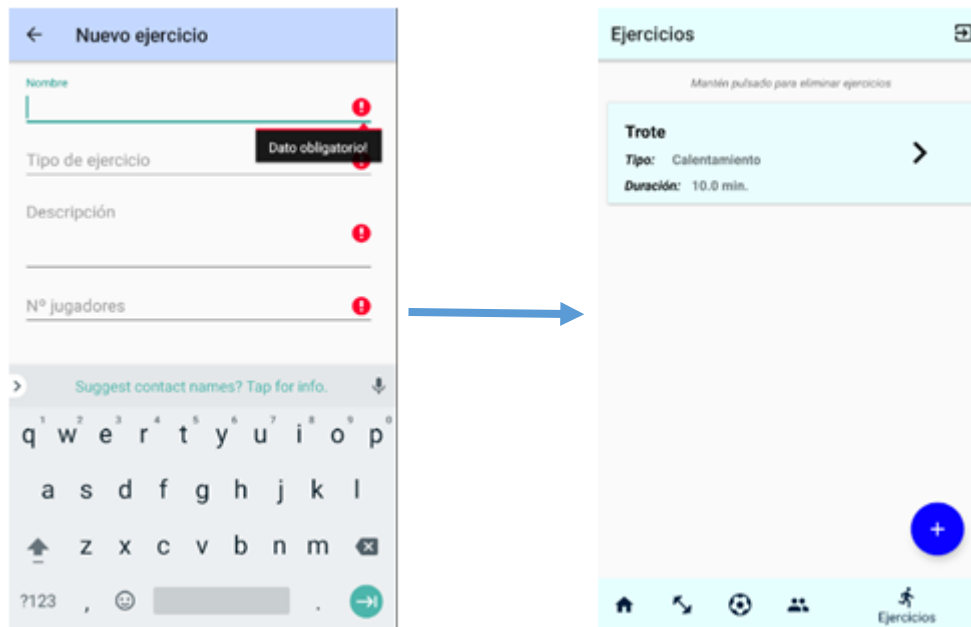


Ilustración 26: Crear nuevo ejercicio.

### Gestión de jugadores

Una vez que tenemos nuestros ejercicios añadidos podemos añadir nuestros jugadores, que participarán en nuestros entrenamientos y en los partidos. Para ello accedemos a la sección de los jugadores mediante el segundo icono por la derecha, al lado del icono de ejercicios. En esta sección nos encontraremos con una pantalla igual a la de ejercicios en la cual mediante el botón verde podremos ir añadiendo nuevos jugadores al sistema.

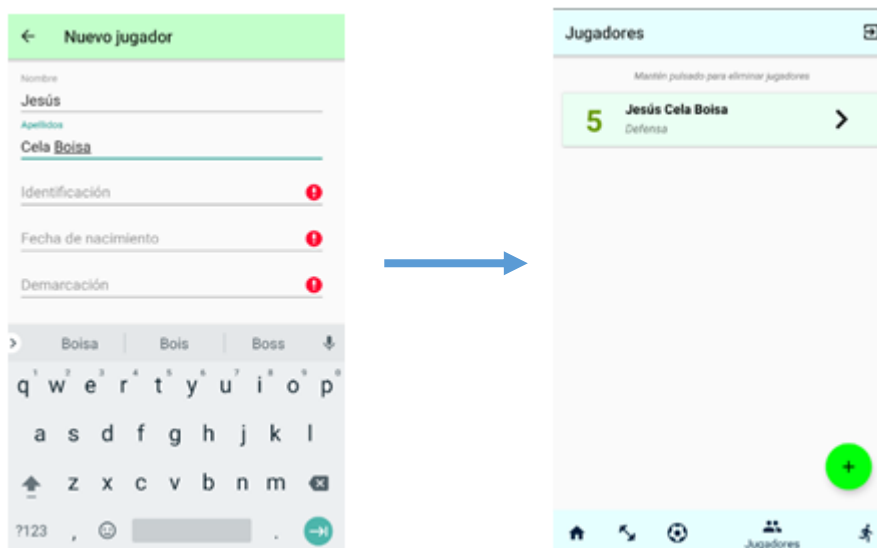


Ilustración 27: Añadir nuevo jugador.

Más adelante veremos cómo se puede acceder a los detalles de cada jugador para ver sus estadísticas de entrenamientos y partidos y cómo modificar sus datos personales, pero antes veamos cómo crear entrenamientos y partidos y cómo relacionar estos como los ejercicios y los jugadores.

### Gestión de entrenamientos

Para acceder a esta sección se debemos pulsar sobre el icono de una mancuerna situado en segunda posición por la izquierda, al lado del icono de la pantalla principal, representado por una casa.

Mediante el botón amarillo podremos crear entrenamientos en nuestra aplicación, para crear un entrenamiento solo será necesario indicar un día y una hora de inicio de la sesión, posteriormente con el entrenamiento ya creado se podrá modificar su hora de fin y añadir ejercicios y jugadores a dicho entrenamiento.

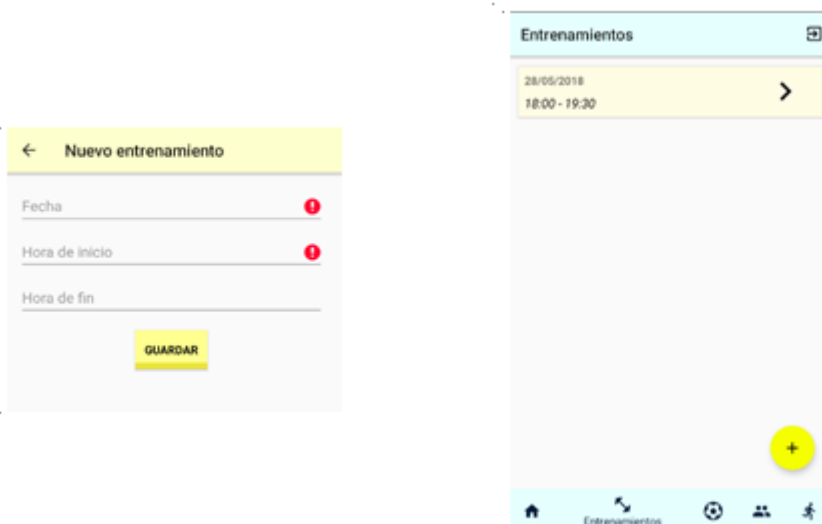


Ilustración 28: Crear nuevo entrenamiento.

Una vez creado podemos acceder a los detalles del mismo pulsado sobre el entrenamiento en la lista. Una vez dentro del entrenamiento tendremos la opción de editar sus datos, eliminarlo o bien añadir ejercicios y jugadores al mismo.

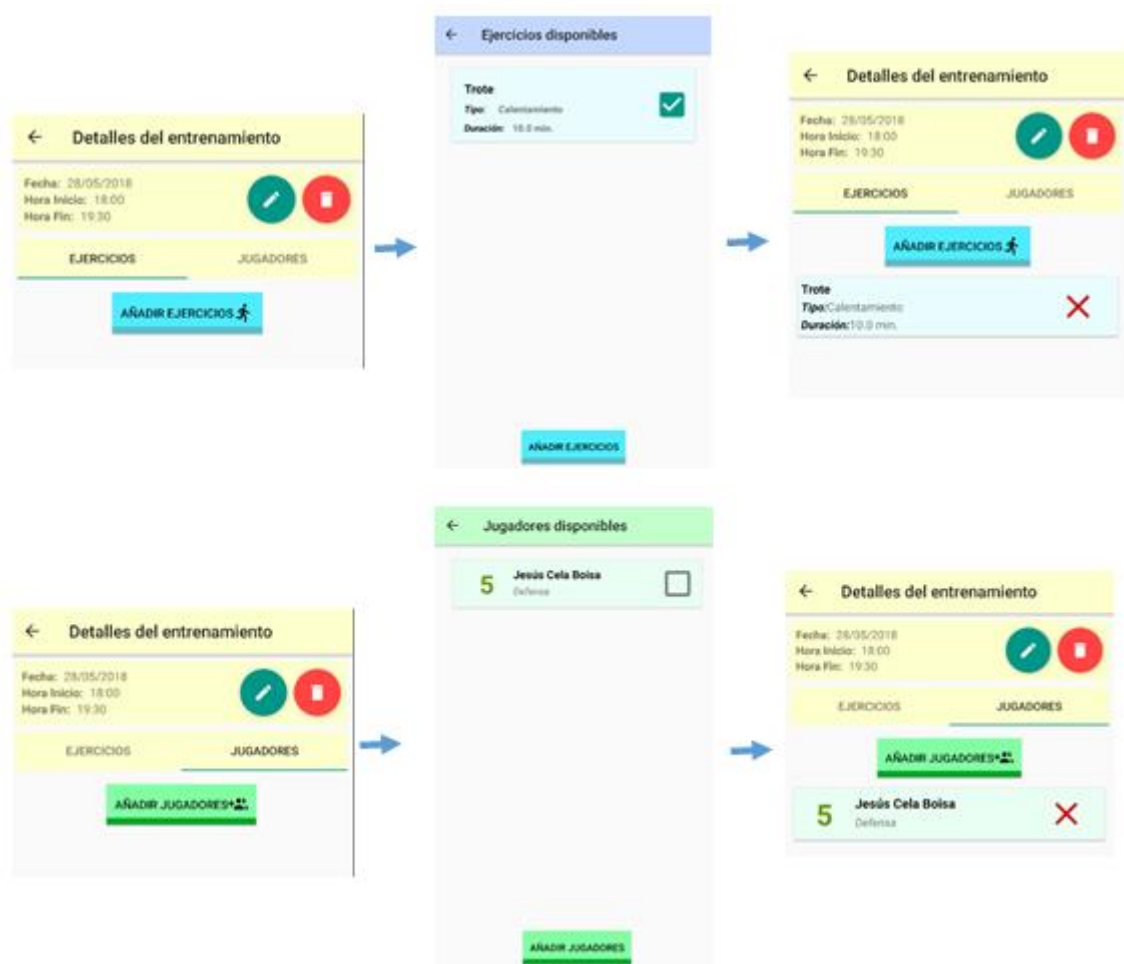


Ilustración 29: Añadir ejercicios y jugadores al entrenamiento.

Los ejercicios y jugadores añadidos se podrán eliminar mediante el botón **X** rojo habilitado para ello.

Se puede ver en la pantalla de inicio el entrenamiento creado ya que está planificado para una fecha posterior al día actual.

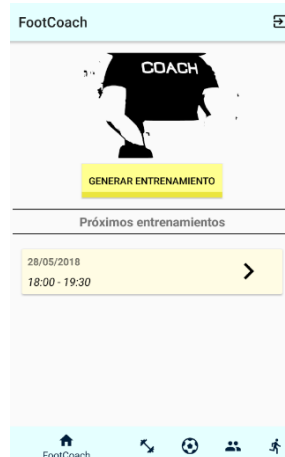
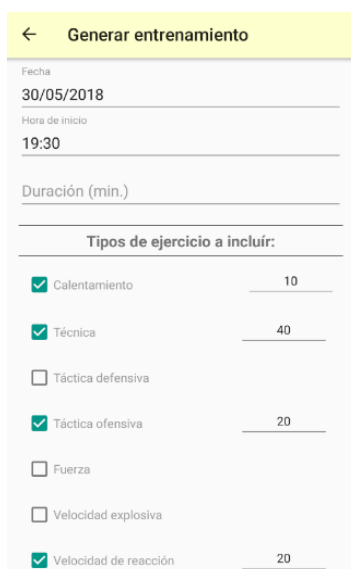


Ilustración 30: Próximos entrenamientos.



## Generación automática de entrenamientos

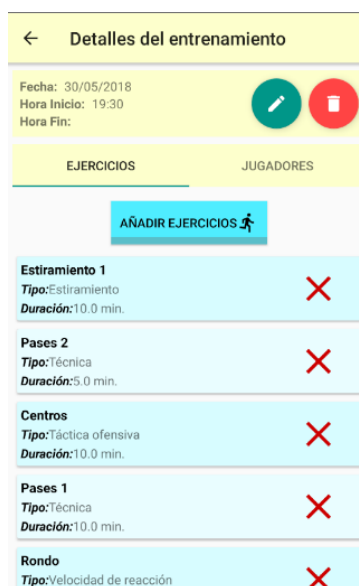
Mediante el botón de la pantalla de inicio “Generar entrenamiento” podemos acceder a un formulario donde introduciendo además de los datos básicos obligatorios para la creación de un entrenamiento, nuestras preferencias en cuanto al tipo de ejercicios que debe incluir y a la duración del trabajo de esos ejercicios el sistema nos generará un entrenamiento de forma automática.



Tipos de ejercicio a incluir:	
<input checked="" type="checkbox"/> Calentamiento	10
<input checked="" type="checkbox"/> Técnica	40
<input type="checkbox"/> Táctica defensiva	
<input checked="" type="checkbox"/> Táctica ofensiva	20
<input type="checkbox"/> Fuerza	
<input type="checkbox"/> Velocidad explosiva	
<input checked="" type="checkbox"/> Velocidad de reacción	20

Ilustración 31: Generar nuevo entrenamiento.

El formulario de la imagen anterior da lugar al entrenamiento generado de forma automática en la imagen siguiente, donde se puede ver que el sistema ha añadido ejercicios al entrenamiento siguiendo las instrucciones dadas en el formulario de generación.



EJERCICIOS	JUGADORES
<a href="#">AÑADIR EJERCICIOS</a>	
<b>Estiramiento 1</b> Tipo: Estiramiento Duración: 10.0 min.	✗
<b>Pases 2</b> Tipo: Técnica Duración: 5.0 min.	✗
<b>Centros</b> Tipo: Táctica ofensiva Duración: 10.0 min.	✗
<b>Pases 1</b> Tipo: Técnica Duración: 10.0 min.	✗
<b>Rondo</b> Tipo: Velocidad de reacción	✗

Ilustración 32: Entrenamiento generado automáticamente por el sistema.

Con el entrenamiento ya generado podremos asignar los jugadores que participarán en el mismo y de modificar libremente la lista de los ejercicios añadidos por el sistema.

### Gestión de partidos.

Para añadir nuevos partidos podemos acceder a la sección de partidos mediante el icono de una pelota situada en el centro del menú inferior, entre los entrenamientos y los jugadores. Mediante el botón rojo podemos acceder al formulario para añadir partidos como en el resto de secciones y estos aparecerán en forma de listado en esta sección.

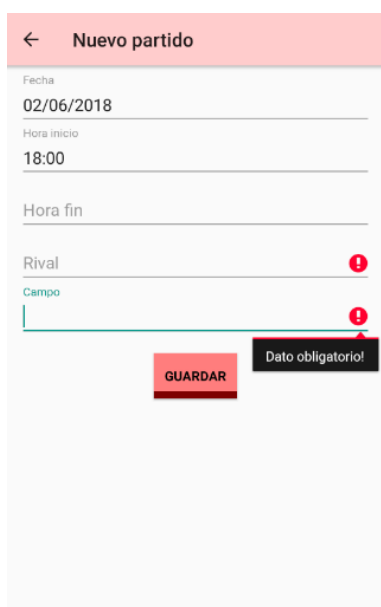


Ilustración 33: Nuevo partido.

Para acceder a los detalles del mismo y convocar jugadores se debe pulsar sobre el partido deseado y con el mismo mecanismo que en los entrenamientos se pueden añadir los jugadores disponibles al mismo.

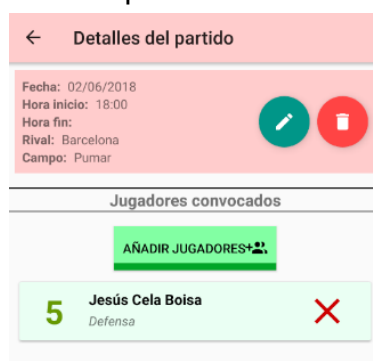


Ilustración 34: Detalles del partido.

## Detalles del jugador

Pulsando sobre el jugador deseado en la lista de estos se puede acceder a sus detalles, en esta pantalla también se visualizarán los datos en cuanto a entrenamientos y partidos en los que ha participado y un botón para modificar los datos del mismo.

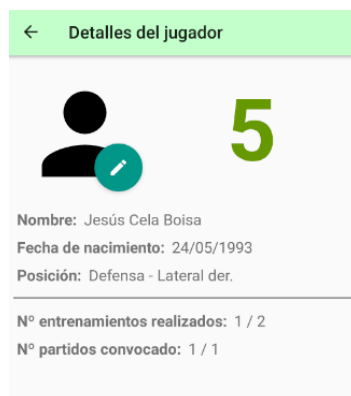


Ilustración 35: Detalles del jugador.

## Salir y cerrar sesión

Por último, podremos salir de la aplicación estando en la pantalla principal o cualquiera de los listados de la aplicación y pulsando dos veces el botón “atrás” del dispositivo en menos de dos segundos. Mediante este sistema simplemente saldremos de la aplicación, la próxima vez que la abramos seguiremos estando autenticados en el sistema por lo que podremos acceder a todos nuestros datos de la sesión anterior. Para cerrar nuestra sesión deberemos pulsar el botón indicado para ello en la parte derecha de la barra superior de la aplicación.

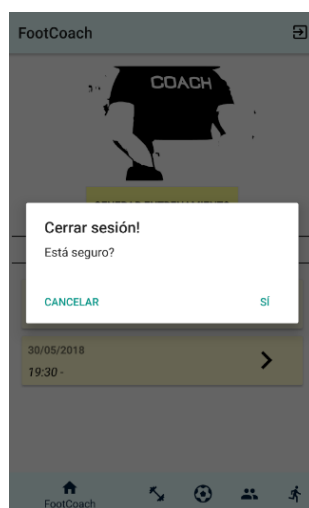


Ilustración 36: Cerrar sesión.