

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



PROYECTO FIN DE CARRERA

Waterpolo Improvement: aplicación Android para la mejora en dicho deporte y realización de estudios estadísticos

Iván del Castillo Gallardo

Febrero 2015

Waterpolo Improvement: aplicación Android para la mejora en dicho deporte y realización de estudios estadísticos

AUTOR: Iván del Castillo Gallardo

TUTOR: Juan José Sánchez Peña

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Febrero de 2015

Resumen

En el presente proyecto se ha desarrollado una aplicación Android para dispositivos electrónicos que utilicen dicho sistema operativo. Los objetivos principales de esta aplicación son ofrecer un soporte electrónico donde almacenar datos estadísticos y realizar consultas de jugadas o entrenamientos en el deporte de waterpolo. Además, se han añadido más funcionalidades enfocadas a la gestión estadística y a la de cubrir posibles necesidades de los practicantes o entrenadores. Por tanto, el perfil de usuario al que está destinada la aplicación sería el de alguien que tenga interés por el deporte del waterpolo, tanto a nivel de principiante como a un nivel que pudiera tener un entrenador.

Es decir, se han aprovechado las ventajas y opciones que ofrece hoy en día un teléfono inteligente (envío y recepción de datos por bluetooth, capacidad de procesamiento, almacenamiento de datos, posibilidad de crear una interfaz de usuario atractiva e intuitiva, etc) para permitir al usuario que en cualquier lugar tenga acceso a un registro estadístico y a diversas alternativas para mejorar en el waterpolo.

El proyecto fue motivado por el propio interés del waterpolo y el de desarrollar una aplicación móvil. Todo ello buscando cubrir necesidades reales o con el fin de poder ofrecer una aplicación realmente útil. Para ello, en primer lugar se realizaron una serie de entrevistas con entrenadores y se formó una lista de posibles funcionalidades que fueran verdaderamente interesantes.

Acto seguido, se realizó un profundo análisis del estado del arte, del cual se extrajeron conclusiones como la de qué sistema operativo sería el más conveniente. El siguiente paso fue el de crear un diseño y plasmarlo en un programa. Por último, se anotaron posibles mejoras futuras y se hizo un manual de usuario para la aplicación.

Palabras clave

Aplicación, Waterpolo, Android, Estadísticas, Jugadas, Bluetooth.

Abstract

The aim of this project is to develop an Android application which could be run on electronics devices. The application is oriented towards statistical analysis and improvement in waterpolo. This application has a lot of options to manage the statistics and other functions that are very useful for coaches and players. Therefore, the target user is anyone interested on waterpolo.

In other words, a smartphone has many advantages that can be very interesting in order to offer a way to improve in some sport or just to be used like a notebook.

This project started with the self interest on the sport and to learn how to create an application. Furthermore, a lot of needs have been met. Firstly, we have had some interviews with different coaches to make a list of functionalities that could offer our application.

Afterwards, we carry out researches on different OS and market applications. Then we designed the application and created the program. At last, we wrote down future improvements and made an user manual.

Key words

Application, Waterpolo, Android, Statistics, Plays, Bluetooth.

Agradecimientos

En primer lugar quiero dar las gracias a mi familia por el apoyo incondicional recibido durante todos estos años. Para los buenos y malos momentos siempre he tenido su apoyo.

Siempre estaré agradecido por haber conocido a Daniel, Alberto, Javier y Sergio, que son los mejores compañeros y amigos que se pudiera tener fuera y dentro de la carrera.

También debo agradecer a Eduardo Boemo su ayuda para esbozar las primeras ideas del proyecto y a Juan José por toda la ayuda prestada.

Por último dar las gracias a todos los amigos y personas que de una forma más o menos directa me han ayudado durante estos años.

ÍNDICE DE CONTENIDO

1	Introducción.....	14
1.1	Motivación	14
1.2	Objetivos.	15
1.3	Metodología de trabajo.	16
1.4	Organización de la memoria	17
2	Estado del arte	18
2.1	Introducción.	18
2.2	El mundo de las aplicaciones móviles.	18
2.2.1	Definición.	18
2.2.2	Objetivos.	19
2.2.3	Tipos de Aplicaciones.....	20
2.2.4	Facilidades.....	22
2.3	Sistema Operativo Android.....	23
2.3.1	Comienzos.....	23
2.3.2	Características generales.	23
2.3.3	Estructura.....	24
2.4	Aplicaciones en el mercado.	25
2.4.1	Waterpolo Analytics.....	25
2.4.2	H2Opolostats.....	26
2.4.3	Waterpolo.	28
2.4.4	Waterpolo sheets.	29
3	Diseño.....	31
3.1	Intención.	31
3.2	Diagrama de estados.	31
3.3	Funciones de la aplicación.	32
3.3.1	Entrenamientos.	33
3.3.2	Jugadas.....	33
3.3.3	Estadística.....	33
3.3.4	Cronómetro.....	34
3.3.5	Bluetooth.....	34
3.4	Requisitos.....	35
3.4.1	Elección del Sistema Operativo.	35
3.4.2	Versión Android.....	45
3.4.3	Dimensión de pantalla.	46
3.4.4	Idioma.	49
4	Desarrollo	50
4.1	Inicios.	50
4.2	Herramientas utilizadas.	50
4.2.1	Eclipse.	50

4.2.2	Emulador de Android.....	51
4.2.3	Herramienta Draw 9-patch.....	51
4.3	Archivos y directorios del proyecto de programación.....	53
4.3.1	Activity.....	54
4.3.2	Fragment.....	56
4.4	Inicio de la aplicación.....	58
4.4.1	Icono.....	58
4.4.2	Pantalla de inicio.....	58
4.5	Entrenamientos.....	58
4.6	Jugadas.....	60
4.6.1	Representación de las jugadas.....	60
4.6.2	Estructura del menú de jugadas.....	61
4.7	Gestión de Estadísticas.....	62
4.7.1	Introducción a la Base de Datos.....	62
4.7.2	Organización de la Base de datos.....	63
4.7.3	Base de datos creada.....	63
4.7.4	Modelo Entidad - Relación.....	66
4.7.5	Funciones.....	66
4.8	Cronómetro.....	68
4.8.1	Layout.....	69
4.8.2	Funcionamiento.....	69
4.9	Bluetooth.....	69
4.9.1	Transmitir.....	69
4.9.2	Recibir.....	70
4.9.3	Modificación.....	72
4.10	Idioma.....	72
4.11	Diagrama de clases.....	73
5	Conclusiones.....	76
6	Trabajo futuro.....	77
7	Manual de Usuario.....	78
	Referencias.....	91
	PRESUPUESTO.....	92
	PLIEGO DE CONDICIONES.....	93

ÍNDICE DE FIGURAS

FIGURA 1.- EXPANSIÓN DE LOS DISPOSITIVOS ELECTRÓNICOS	14
FIGURA 2.- USO DE MÓVIL POR EDADES	14
FIGURA 3.- ESQUEMA DEL PÚBLICO OBJETIVO VS FUNCIONALIDADES	16
FIGURA 4.- INTERÉS POR APP, BÚSQUEDAS EN GOOGLE	19
FIGURA 5.- ESTRUCTURA ANDROID	24
FIGURA 6.- ICONO APP WATERPOLO ANALYTICS	25
FIGURA 7.- INTERFACES APP WATERPOLO ANALYTICS.....	26
FIGURA 8.- ICONO APP H2OPOLOSTATS.....	27
FIGURA 9.- INTERFACES APP H2OPOLOSTATS.....	27
FIGURA 10.- ICONO APP WATERPOLO	28
FIGURA 11.- INTERFACES APP WATERPOLO.....	29
FIGURA 12.-ICONO APP WPSHEETS	30
FIGURA 13.- INTERFACES APP WPSHEETS	30
FIGURA 14.- DIAGRAMA DE ESTADOS	32
FIGURA 15.- GRÁFICA CON CUOTAS DE MERCADO SISTEMAS OPERATIVOS.	36
FIGURA 16.- DISTRIBUCIÓN DE PROGRAMADORES POR SO Y ZONA GEOGRÁFICA	37
FIGURA 17.- PORCENTAJES DE PROGRAMADORES POR PLATAFORMA PRINCIPAL.....	38
FIGURA 18.- PREFERENCIA DE SO POR DESARROLLADORES SEGÚN DISPOSITIVO	39
FIGURA 19.- DIFERENCIA DE PRECIOS ENTRE SO	40
FIGURA 20.- BENEFICIO DE APLICACIONES POR CATEGORÍAS	40
FIGURA 21.-PROYECCIÓN FINANCIERA	41
FIGURA 22.- USO DE SO POR IDIOMA.....	42
FIGURA 23.- PROYECCIÓN USUARIOS DE LOS SO	43

FIGURA 24.- REPRESENTACIÓN MUNDIAL DE SOS.....	43
FIGURA 25.- GRÁFICA CON PORCENTAJES DE USO DE CADA SO A NIVEL GLOBAL.....	44
FIGURA 26.- PREDOMINIO EN DIFERENTES DISPOSITIVOS	45
FIGURA 27.- DISTRIBUCIÓN DE DISPOSITIVOS POR VERSIÓN ANDROID.....	46
FIGURA 28.- TAMAÑOS DE PANTALLA.....	47
FIGURA 29.- GRÁFICO CON LA DISTRIBUCIÓN DE LOS TAMAÑOS DE PANTALLA.....	48
FIGURA 30.- GRÁFICO CON LA DISTRIBUCIÓN DE LAS DENSIDADES DE PANTALLA.....	49
FIGURA 31.- CAPTURA DEL PROGRAMA DRAW 9-PATCH.....	52
FIGURA 32.- DIRECTORIO	53
FIGURA 33.- ESTADOS DE UN ACTIVITY	55
FIGURA 34.- ESTADOS DE UN FRAGMENT	57
FIGURA 35.- ICONO APP WATERPOLO IMPROVEMENT.....	58
FIGURA 36.- PANTALLA INICIO APLICACIÓN.....	58
FIGURA 37.- MODIFICACIÓN DE CODIFICACIÓN 1	59
FIGURA 38.- MODIFICACIÓN DE CODIFICACIÓN 2	60
FIGURA 39.- MUESTRA DE JUGADA.....	61
FIGURA 40.- ESTRUCTURA DE JUGADAS	62
FIGURA 41.- DIAGRAMA DE CLASES, 1	74
FIGURA 42.- DIAGRAMA DE CLASES, 2	75
FIGURA 43.- IMAGEN DE CARGA DE APLICACIÓN	78
FIGURA 44.-MENÚ INICIAL	78
FIGURA 45.-MENÚ ENTRENAMIENTOS	79
FIGURA 46.- EJEMPLO DE ENTRENAMIENTO.....	80
FIGURA 47.- MENÚ JUGADAS.....	81

FIGURA 48.- JUGADA EJEMPLO	82
FIGURA 49.- AGREGAR UN EQUIPO.....	83
FIGURA 1 50.- AGREGAR UN JUGADOR.....	84
FIGURA 51.- ANOTAR DATOS ESTADÍSTICOS	85
FIGURA 52.- CONSULTA DE ESTADÍSTICAS.....	86
FIGURA 53.- MODIFICACIÓN DE JUGADOR.....	87
FIGURA 54.- ANOTACIÓN DE ESTADÍSTICAS.....	87
FIGURA 1 55.- CONSULTA DE ESTADÍSTICAS AMPLIADAS.....	88
FIGURA 56.- CRONÓMETRO	89
FIGURA 57.- FUNCIONES DE BLUETOOTH	90
FIGURA 58.- SELECCIÓN DEL DISPOSITIVO RECEPTOR (BLUETOOTH).....	90

ÍNDICE DE TABLAS

TABLA 1.- TABLA COMPARATIVA DE LOS TIPOS DE APLICACIONES.....	22
TABLA 2.- CUOTA DE MERCADO SISTEMAS OPERATIVOS	35
TABLA 3.- DISTRIBUCIÓN DE MÓVILES POR TAMAÑO Y DENSIDAD DE PANTALLA	48
TABLA 4.- EJEMPLO DE BASE DE DATOS.....	63
TABLA 5.- TABLA DEL JUGADOR.....	64
TABLA 6.- TABLA EQUIPO	64
TABLA 7.- TABLA ESTADÍSTICAS.....	65
TABLA 8.- FUNCIONES PARA LA GESTIÓN DE LA BD	67
TABLA 9.- FUNCIÓN BASE DE TRANSMISIÓN BLUETOOTH	70
TABLA 10.- FUNCIONES PARA IMPORTAR LA INFORMACIÓN RECIBIDA.....	72

1 Introducción

1.1 Motivación

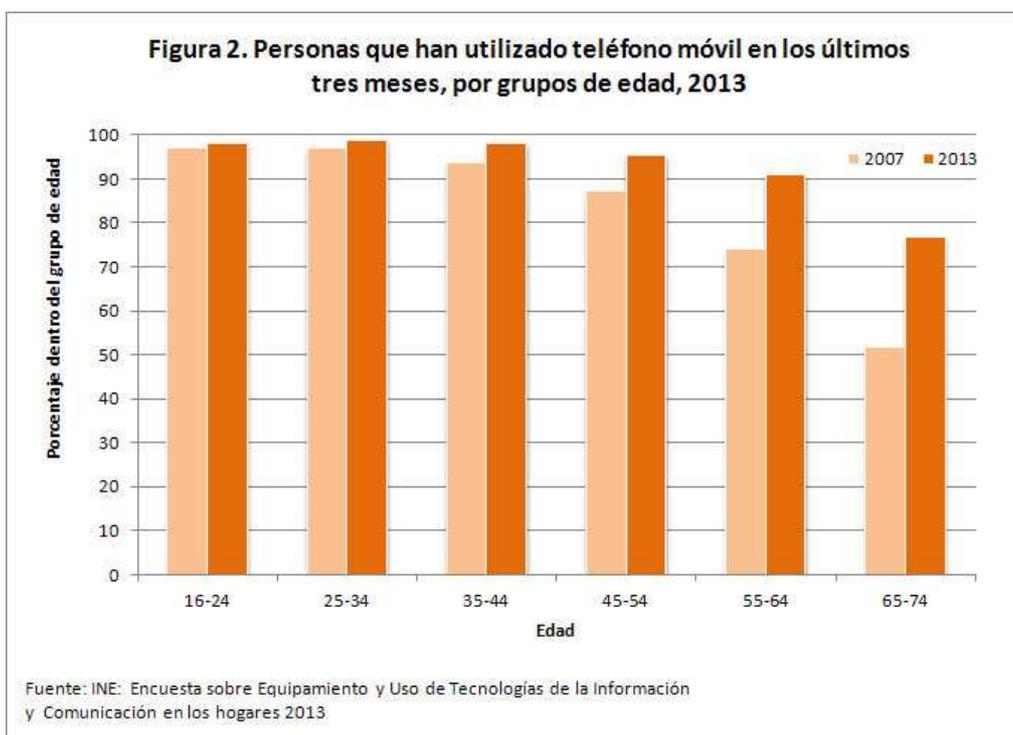
En los últimos años y gracias al gran éxito que están teniendo deportistas profesionales de distintas especialidades, se puede observar un mayor conocimiento y atracción de las personas por “nuevos deportes”. Además, si se junta este creciente interés a la accesibilidad de la mayor parte de la población a un teléfono inteligente, el desarrollo de aplicaciones informáticas orientadas a estos deportes puede ser una buena idea de negocio.

Figura 1.- Expansión de los dispositivos electrónicos



Fuente: lvaole.blogspot.com.es/2015/01/moviles-escuela-y-familias-mi-hij-me.html

Figura 2.- Uso de móvil por edades



Se puede observar la gran inmersión de los dispositivos electrónicos y teléfonos inteligentes en todos los rangos de edad de la sociedad.

Llevando este tema hacia lo personal, habiéndolo practicado y vivido de primera mano, se ha elegido un deporte en concreto para desarrollar una aplicación Android. Este deporte es el waterpolo. Por tanto, uniendo el análisis “social” previo con la visión formada desde dentro de un deporte se ha realizado dicha aplicación con las siguientes herramientas:

- ❖ Acceso a una base de datos con entrenamientos y posibilidad de utilizar un cronómetro para ejecutarlos de forma más rigurosa.
- ❖ Anotaciones estadísticas que serán archivadas en el dispositivo móvil, teniendo la posibilidad de asociar un jugador a diferentes equipos y posiciones.
- ❖ Opción de transmitir dichas estadísticas a otros teléfonos inteligentes.
- ❖ Posibilidad de consultar un libro de jugadas tanto de ataque como de defensa.

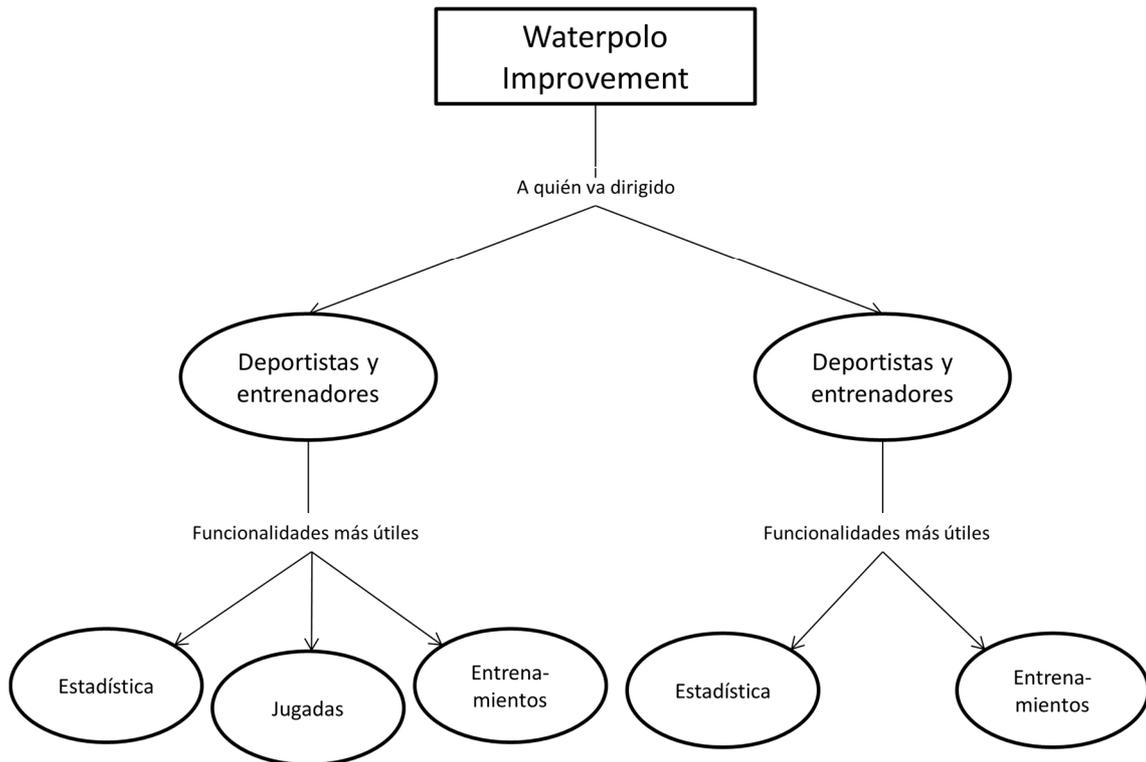
1.2 Objetivos.

La aplicación se ha desarrollado con dos objetivos principales. El primero es el de permitir adentrarse a personas que no hayan practicado nunca el waterpolo en este deporte cada vez más conocido. El segundo consiste en que la aplicación sea utilizada como herramienta para entrenadores o practicantes de dicho deporte sean del nivel que sean. Es decir, el público destinado serán tanto los técnicos y personas ya experimentadas como aquellos que quieran iniciarse o tengan ya unos conocimientos del tema.

Se ha logrado que dicha aplicación sea muy intuitiva y fácil de utilizar. El único requerimiento es disponer de un dispositivo con un Sistema Operativo Android que tenga instalada una versión a partir de la tercera, “Ice Cream Sandwich”.

En línea con lo explicado anteriormente, este PFC permitirá dotar de una herramienta, para calcular y gestionar estadísticas, a los deportistas y entrenadores. De esta forma podrán hacer un análisis más completo de los partidos y entrenamientos. (Los datos a recopilar y a analizar han sido seleccionados tras haber consultado a distintos entrenadores y con el objetivo de cubrir la información de un “acta de partido”). También, servirá como inventario de jugadas, de ataque y de defensa, y de entrenamientos. Todos ellos con una explicación detallada. Por tanto, un esquema del objetivo seguido en este proyecto sería el siguiente:

Figura 3.- Esquema del público objetivo VS funcionalidades



Para poner al alcance de los usuarios la aplicación desarrollada, ésta será subida a “Google Play” donde podrá ser descargada sin ninguna restricción. Asimismo, una vez instalada la aplicación está podrá ejecutarse en dos idiomas: castellano e inglés.

Todo esto ha sido realizado bajo el objetivo primordial de aprender un lenguaje de programación y una serie de herramientas necesarias para crear una aplicación Android, siendo estas cualidades bastante demandadas por el mercado, aparte de las cualidades no medibles que se obtienen o se mejoran tras desarrollar un proyecto de este tipo.

1.3 Metodología de trabajo.

El plan de trabajo llevado a cabo ha ido siguiendo diferentes puntos, los cuales se pueden representar de la siguiente manera.

1. Estudio del estado del arte. En este punto se observaron las aplicaciones ya desarrolladas y se analizó qué ofrecían, así como una lista con aspectos valorables y otros todavía por mejorar o con opción de introducir cambios que representasen mejoras.
2. Aprendizaje y recopilación de herramientas necesarias para crear aplicación Android.
3. Parte troncal de desarrollo de la aplicación: programación.
4. Realización de pruebas y soluciones para distintos errores con el fin de verificar el correcto funcionamiento de la aplicación.

5. Documentación del proyecto.

1.4 Organización de la memoria

❖ Capítulo 1: Introducción.

En este apartado se explican las motivaciones que propiciaron este proyecto y los objetivos buscados.

❖ Capítulo 2: Estado del arte.

Una vez realizada una introducción sobre en qué consiste el proyecto se hace un análisis del Estado del arte

❖ Capítulo 3: Diseño del proyecto.

En este tercer bloque se expone el diseño que se ha llevado a cabo en la realización de la aplicación. Además, se comentan las funcionalidades que ofrece la herramienta desarrollada y se presentan argumentos defendiendo la utilización de la plataforma Android.

❖ Capítulo 4. Desarrollo.

Aquí se hace una explicación más técnica de cómo funciona la aplicación, de los archivos y de los elementos que se han sido necesarios para desarrollarla.

❖ Capítulo 5. Conclusiones y trabajo futuro.

Breve evaluación sobre la herramienta creada, línea de futuro y posibles modificaciones de mejora.

❖ Capítulo 6. Manual de usuario.

Por último se muestra una guía de usuario de cómo se maneja la aplicación y se hace uso de las distintas funcionalidades. En este manual se incorporan imágenes con los diferentes menús para hacer más fácil las explicaciones sobre el funcionamiento.

2 Estado del arte

En esta sección se realizará una breve introducción y análisis de los elementos que componen el proyecto. Además se intentará mostrar una visión y un estudio del estado del arte en cuanto a las aplicaciones que pertenecen a la misma temática que la desarrollada. Todo ello, con el objetivo de extraer características positivas y negativas de las herramientas ya ofertadas. De esta manera, entender las líneas generales que normalmente se buscan y buscar ofrecer algo nuevo o mejorado.

2.1 Introducción.

Primeramente se muestra el dispositivo electrónico que será el portador de la aplicación desarrollada y en el que se podrá ejecutar ésta. Esto es, el teléfono inteligente, o más conocido por su nombre en inglés, el Smartphone. La principal diferencia o avance de este tipo de teléfono móvil respecto a las versiones anteriores es que, aparte de la realización y recepción de llamadas y mensajes telefónicos, dispone de funcionalidades que bien lo asemejan a un ordenador. Es decir, este nuevo teléfono dispone de hardware y software, con un Sistema Operativo propio que le permite ofrecer funcionalidades propias de un ordenador.

La característica principal que comúnmente se comenzó a utilizar para diferenciar los teléfonos inteligentes de los anteriores era si éstos disponían de soporte completo para el correo electrónico. Pero ahora, hay muchos más rasgos que definen a un Smartphone como pueden ser: la posibilidad de multitarea, el acceso a Internet, funciones multimedia (como son cámara y reproductor de música y vídeos comprimidos), GPS y sobre todo, la que más nos importa, la posibilidad de instalar diferentes aplicaciones con diversos fines u objetivos.

2.2 El mundo de las aplicaciones móviles.

2.2.1 Definición.

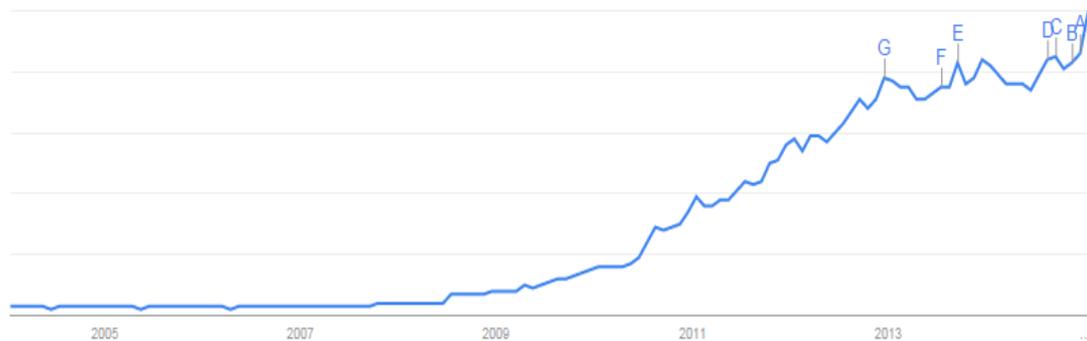
Una app es una aplicación de software que se instala en teléfonos móviles inteligentes o tabletas para ayudar al usuario en una labor concreta, ya sea de carácter profesional o de ocio y entretenimiento.

Por lo general, las aplicaciones se encuentran disponibles a través de plataformas de distribución, siendo éstas operadas por las compañías propietarias de los sistemas operativos móviles como Android, iOS, BlackBerry OS y Windows Phone. Existen aplicaciones móviles gratuitas y otras de pago. En el caso de las segundas, un promedio del 20-30% del costo de la aplicación, al ser vendida en la plataforma de distribución, se destina al distribuidor y el resto es para el desarrollador.http://es.wikipedia.org/wiki/Aplicaci%C3%B3n_m%C3%B3vil_-_cite_note-VB--1

Como curiosidad y prueba de la gran y rápida extensión de las aplicaciones móviles en por mundo, se encuentra las siguientes:

- ❖ El término app fue nombrado como “Word of the Year” (Palabra del Año) por la “American Dialect Society” en el año 2010.
- ❖ Si se utiliza la herramienta de Google de “Explorar el año en tendencias” se obtiene una muestra estadística de cómo ha ido creciendo el interés por las app a lo largo del tiempo.

Figura 4.- Interés por app, búsquedas en Google



Fuente: Herramienta estadística de búsqueda de Google.

Los tres hitos, todos ocurridos en el año 2008, que provocaron que comenzase el despegue del interés por las aplicaciones móviles fueron: el lanzamiento del “App Store” en julio, la publicación del primer SDK para desarrolladores de Android en agosto y la apertura del “Android Market” en octubre.

2.2.2 Objetivos.

El objetivo de una aplicación es facilitarnos la consecución de una tarea determinada o asistirnos en operaciones y gestiones del día a día. Además de servir como una fuente de información o de consulta y como entretenimiento.

Claro ejemplo de las diferentes finalidades que pueden tener las aplicaciones son los numerosos tipos que existen de éstas y la gran variedad disponible: aplicaciones de noticias (de diversos periódicos), herramientas de comunicación (“Whatsapp”), herramientas de localización (GPS), juegos y aplicaciones de pasatiempo, redes sociales (“Facebook”), etc.

Resumiendo, una aplicación puede tener un objetivo meramente comercial y lucrativo mediante el aprovechamiento de la tendencia actual del uso de los teléfonos inteligentes y aplicaciones que encandilan, atrapan a los usuarios y que además aprovechen ese medio para obtener beneficios por publicidad. Sin embargo, una aplicación puede también nacer con el fin de cubrir una necesidad, a modo de herramienta y además reportar beneficios a sus creadores. Es decir, no sólo el propósito de enriquecerse es el buscado.

2.2.3 Tipos de Aplicaciones.

Las aplicaciones móviles se pueden dividir en tres tipos según su naturaleza:

❖ **Aplicaciones Nativas.**

Las que pertenecen a este tipo se desarrollan específicamente para cada Sistema Operativo, iOS, Android, Windows Phone... De tal manera, se adaptan a cada uno de los lenguajes de programación: lenguaje "Objective-C" para iOS, "Java" para Android (con sus variantes y modificaciones), y ".Net" para Windows Phone.

+Ventajas.

1. Acceso completo al dispositivo. Se puede acceder a todos los componentes hardware del móvil: cámara, GPS, agenda, dispositivos de almacenamiento, etc.
2. No precisa de conexión a Internet para funcionar.
3. La descarga e instalación de este tipo de aplicaciones es sencilla a través de las "app store" de los fabricantes. Además, esto facilita el marketing porque promociona y permite que se conozca una aplicación.
4. Siempre se dispone de actualizaciones de la aplicación.
5. Todos los puntos anteriores provocan que la experiencia de usuario sea muy buena.

-Inconvenientes

1. El código que se ha desarrollado no es reutilizable, o no se puede ejecutar en diferentes plataformas.
2. La creación de este tipo suele ser más cara.
3. No hay una biblioteca de consulta con información y herramientas que se pueda aplicar por igual a todas las plataformas.

❖ **Aplicaciones Web.**

En este apartado estarían las aplicaciones que se crean a partir de la utilización de lenguaje Javascript, CSS o HTML. Éstas, se ejecutan dentro del propio navegador web del dispositivo a través de una dirección, "URL". Esto último también implica que una aplicación no llega a estar instalada como tal en el dispositivo.

+Ventajas.

1. Se puede programar con un objetivo global. Es decir, la programación no depende del Sistema Operativo en el que se utilizará la aplicación y un código base podría ser reutilizable en múltiples plataformas.
2. El desarrollo es más sencillo y económico.
3. No necesita la aprobación externa para ser publicada (las nativas deben tener una autorización para poder ser visibles en el "app store").

4. El usuario no necesitará realizar una actualización porque siempre accederá a la última versión.

-Inconvenientes

1. Es necesaria una conexión a Internet.
2. El acceso a los elementos hardware del dispositivo es muy limitado.
3. Los dos puntos anteriores más el tiempo de respuesta de la aplicación pueden empeorar la experiencia del usuario.
4. Es más difícil dar a conocer, publicitar la aplicación.

❖ Aplicaciones Híbridas.

Se denominan aplicaciones híbridas porque combinan aspectos de las aplicaciones nativas y de las aplicaciones web según más convenga. Por ello, contiene las mejores características de cada una. Se desarrollan con lenguajes propios de las “webabpp”, es decir, HTML, Javascript y CSS por lo que permite su uso en diferentes plataformas, pero también ofrecen la posibilidad de acceder a varios elementos y funcionalidades hardware del dispositivo.

+Ventajas.

1. Se puede distribuir en tiendas de aplicaciones de diferentes plataformas (iOS, Android...).
2. La instalación tiene todas las ventajas que tiene una aplicación de tipo nativa.
3. Se habilita el acceso a componentes hardware del dispositivo.

-Inconvenientes

1. El diseño visual no será siempre tan familiar y tan agradable al usuario como en una aplicación nativa, principalmente porque no estará fijado a un determinado Sistema Operativo.
2. La experiencia del usuario no es igual de satisfactoria como en el caso de las nativas.

Se muestra una tabla resumen y comparativa de los tipos de aplicaciones.

Tabla 1.- Tabla comparativa de los Tipos de Aplicaciones

TIPO DE APLICACIÓN	RESUMEN	VENTAJAS	INCONVENIENTES
Nativa	Se desarrollan específicamente para cada SO	Acceso completo al dispositivo	Código no reutilizable
		No precisa de Internet para funcionar	Desarrollo más caro
		Utiliza "app store" de los fabricantes	No hay biblioteca de consulta general
		Actualizaciones siempre disponibles	
Web	Se ejecutan dentro del propio navegador web a través de una dirección	Se pueden programar con un objetivo global	Necesaria conexión a Internet
		Desarrollo más sencillo y económico	Limitado acceso al dispositivo
		No necesita de aprobación externa para ser publicada	Lento tiempo de respuesta
		El usuario no necesita hacer la actualización	Más difícil de publicitar
Híbrida	Combina aspectos de la Nativa y la Web	Se puede distribuir en "app stores"	Diseño visual poco familiar
		Ventajas en la instalación como las de la nativa	Experiencia de usuario no tan satisfactoria como en las nativas
		Acceso habilitado a los componentes del dispositivo	

2.2.4 Facilidades.

Como se ha mencionado en el apartado de Objetivos, una aplicación puede nacer con el fin de cubrir una necesidad detectada por el diseñador de esta. Dicha necesidad puede tener un origen empresarial, es decir, que una empresa precise de una aplicación para ofertar sus servicios y llegar mejor al cliente o un origen en la sociedad. Las de este segundo tipo, al igual que las primeras pueden tener el propósito de enriquecer a los creadores pero también buscan ser una buena herramienta.

Por ejemplo, una aplicación que descargase los folletos y las ofertas de una famosa empresa de decoración sueca pertenecería al primer grupo, mientras que una que informe de los horarios de los autobuses se correspondería con la segunda clase mencionada.

Dicho de una forma más breve, una aplicación móvil nos puede ofrecer facilidades para el día a día.

2.3 Sistema Operativo Android.

2.3.1 Comienzos.

Android fue fundado por un grupo de jóvenes en el año 2003. Cuando en 2005 Google lo compró era muy poco conocido. Finalmente, en noviembre del 2007 en la celebración de la “Open Handset Alliance*”, que agrupaba a numerosos fabricantes de teléfonos móviles, chipsets y a Google, se mostró la primera versión de Android, junto con el SDK para que los programadores pudieran comenzar a crear sus aplicaciones para este sistema.

2.3.2 Características generales.

Android es un Sistema Operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y Blackberry OS. La diferencia con ellos es que está basado en Linux, es decir, un núcleo de sistema operativo libre, gratuito y multiplataforma.

Este sistema permite programar aplicaciones en una variación de Java llamada Dalvik. Esta plataforma dispone de todas las interfaces necesarias para desarrollar aplicaciones que puedan acceder a las diferentes funciones del teléfono como pueden ser el GPS, el registro de llamadas, la agenda de contactos, Bluetooth, etc.

Además, existen herramientas de programación gratuitas y eso conlleva que una inmensa cantidad de aplicaciones puedan ser ofrecidas y estén disponibles, de tal manera que el usuario tiene a su alcance y disposición infinidad de opciones y tipos de aplicaciones.

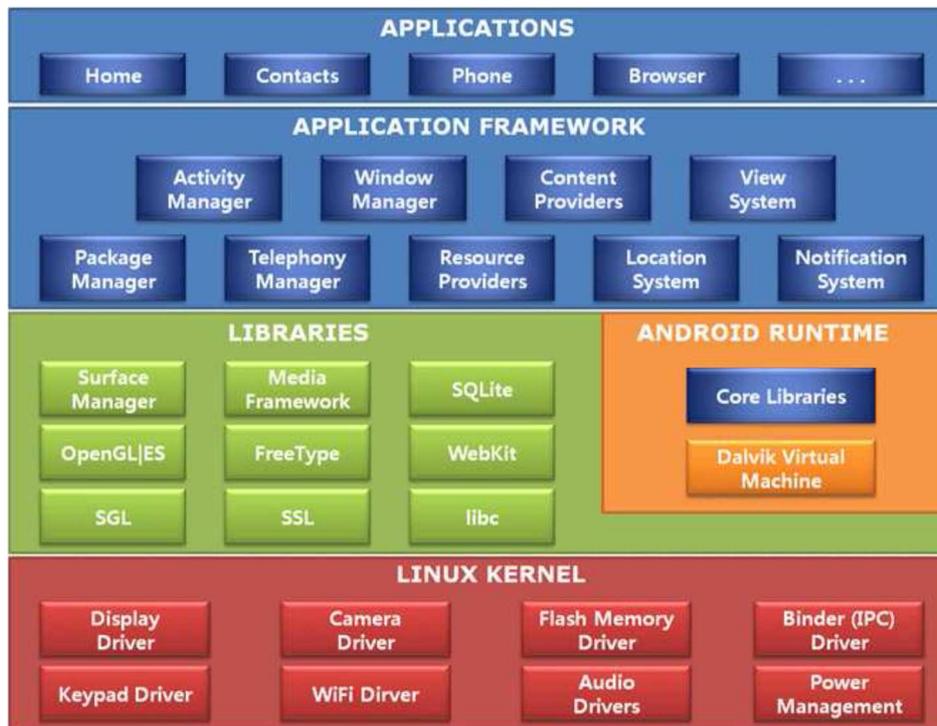
En el tercer apartado (Diseño), se muestran más características de este Sistema Operativo y se busca argumentar porqué es una buena elección como plataforma para desarrollar una aplicación móvil.

*Open Handset Alliance (OHA) es una alianza comercial de 84 compañías que se dedica a desarrollar estándares abiertos para dispositivos móviles. Algunos de sus miembros son Google, HTC, Dell, Intel, Motorola, Qualcomm, Texas Instruments, Samsung, LG, T-Mobile, Nvidia y Wind River Systems.

2.3.3 Estructura.

El Sistema Operativo Android se ejecuta sobre “DVM” (Dalvik Virtual Machine) es decir, está basado en el kernel, núcleo de Linux. La estructura sería la siguiente:

Figura 5.- Estructura Android



Fuente: www.cubrid.org

❖ Aplicaciones.

Este apartado engloba las aplicaciones desarrolladas con Java. Generalmente son correo electrónico, programa de SMS, calendario, navegador y contactos.

❖ Marco de trabajo de las aplicaciones.

Los desarrolladores tienen acceso completo a todos los tipos de APIs (Application Programming Interface, Interfaz de Programación de Aplicaciones) del marco de trabajo que utiliza la aplicación base. La arquitectura está diseñada para que se puedan reutilizar los componentes, es decir, cualquier aplicación puede ofrecer unas determinadas funcionalidades y cualquier otra puede hacer uso de dichas funcionalidades, aunque siempre obedeciendo a unas reglas del “framework”. También es posible modificar el ciclo de vida de las aplicaciones, esto es, que sean reemplazadas por el usuario.

❖ Bibliotecas.

Se incluyen un conjunto de bibliotecas de “C/C++” que pueden ser utilizadas sobre Android. Éstas son mostradas a los programadores a través del marco de trabajo de las aplicaciones (Application Framework) y puestas a su disposición.

❖ Runtime de Android.

A pesar de que Android se encuentra desarrollado en lenguaje Java, se utiliza DVM (Dalvik Virtual Machine) en vez de JVM (Java Virtual Machine). De tal manera que un código, archivo tipo Java (tipo Class) compilado por un compilador Java es transformado en un “Dalvik Executable” a través de las herramientas “DX” (herramienta incluida en el “SDK”, Kit de Desarrollo de Software de Android). El motivo de esto es que “Dalvik” está optimizado para ejecutarse en dispositivos con la memoria limitada.

❖ Núcleo Linux.

Android depende de Linux para servicios básicos del sistema como son la seguridad, gestión de memoria y de procesos, pila de red y el modelo de controladores.

2.4 Aplicaciones en el mercado.

Si se realiza un filtro por “waterpolo” en “Google Play” se puede comprobar que hay varias aplicaciones relacionadas con dicha temática. Éstas se pueden subdividir según la función que desempeñan: herramientas, entretenimiento o informativas.

El proyecto que se ha desarrollado tiene el objetivo de servir como herramienta y por tanto, se va a proceder a hacer un análisis de las aplicaciones que pudieran ser similares. Además se intentan mostrar las características positivas o negativas de cada una de ellas.

2.4.1 Waterpolo Analytics.

- Descripción.

Se muestra la descripción propia de la aplicación que aparece en el “Google Play”:

Figura 6.- Icono app Waterpolo Analytics



“Waterpolo Analytics” es una herramienta para entrenadores y jugadores. Es hora de poner en práctica la ventaja de las aplicaciones móviles y adaptarlas al análisis deportivo. Descubra las fortalezas y debilidades de su equipo para diseñar la estrategia que mejor se adapte a su estilo de juego.”

La característica más identificativa y original de esta aplicación es que permite marcar, en una pizarra que simula el campo de juego, dónde se ha producido un lanzamiento a portería. Una vez que dicho lanzamiento se dibuja en la pantalla se puede anotar qué jugador, identificado por un número, ha realizado el disparo y el resultado de éste. Además, permite relacionar la ocasión con un contexto: si era en una superioridad numérica, en igualdad, en un contra-ataque o tras un penalti. Una característica más es que cada registro realizado se puede ver desde ambos lados, desde el punto de ataque o de defensa.

Por último, existen varias opciones para consultar los datos registrados: primero, se puede ver en qué zona de la piscina han ocurrido las diferentes acciones, filtrando si fue en un momento de igualdad o superioridad numérica, y segundo, existe la posibilidad de observar los datos obtenidos de forma estadística y agrupados.

Figura 7.- Interfaces app Waterpolo Analytics



- **Características positivas.**

1. Posibilidad de marcar en una pizarra dónde ha habido ocasiones de gol.
2. Forma visual de mostrar los datos anotados, posibilidad de realizar un mejor análisis de los acontecimientos del partido.
3. Aplicación con un uso bastante intuitivo y sencillo.

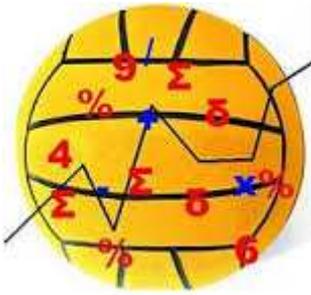
- **Características negativas.**

1. Aplicación únicamente enfocada al registro estadístico de ocasiones de gol.
2. No permite la opción de enviar la información recopilada.
3. No ofrece más funcionalidades a parte que de las relacionadas con la anotación de datos estadísticos referentes a los lanzamientos.

2.4.2 H2Opolostats.

- **Descripción.**

Figura 8.- Icono app H2Opolostats



Se muestra la descripción propia de la aplicación que aparece en el “Google Play”:

“Deja la carpeta, el papel y el bolígrafo. Desempeña tus funciones de delegado de waterpolo desde el móvil. Registra los aciertos y errores de los jugadores de tu equipo y publica la ficha del partido”.

La principal funcionalidad de esta aplicación es llevar un registro estadístico de un encuentro de waterpolo. En la primera pantalla se muestra la lista de jugadores de los dos equipos que se enfrentan. Se introduce los nombres de los equipos y después se puede seleccionar cada jugador, estando éste identificado por el número del gorro, y anotar las diferentes acciones que haya realizado. Si se pulsa un botón representado como un ojo, se pueden ver las estadísticas de cada jugador que hayan sido registradas.

Por otro lado, se ofrece la posibilidad de enviar los datos, a través de un correo electrónico, que hayan sido registrados, en diferentes idiomas: castellano, catalán, francés, inglés, gallego y euskera.

Figura 9.- Interfaces app H2Opolostats



- **Características positivas.**

1. Posibilidad de enviar las estadísticas recopiladas por correo electrónico.
2. Los datos estadísticos recopilados se pueden enviar en varios idiomas.

- **Características negativas.**

1. La navegación por la aplicación es bastante difícil y nada intuitiva.
2. Es algo complicado de entender el funcionamiento completo de la aplicación.
3. La única gran funcionalidad que presenta es el registro estadístico.

2.4.3 Waterpolo.

- **Descripción.**

Figura 10.- Icono app Waterpolo Se muestra la descripción que aparece en el “Google Play”:



“With this app you can keep track of your team statistics! create a team and manage users / goals and personal errors. With a handy user interface all your team statistics at hand!”

Esta aplicación se basa en el registro de los aciertos y fallos en los lanzamientos a portería durante un encuentro y de los resultados finales de los partidos.

En primer lugar, se debe crear un equipo nuevo o unirse a uno ya creado. Entonces, se podrán anotar los aciertos y errores de cada jugador. En el caso de formar un equipo nuevo será necesario introducir los nombres de los jugadores y los correos electrónicos de los participantes para que sean informados. De esta manera, si los demás jugadores, usuarios tienen instalada dicha aplicación, podrán consultar la información recopilada para un mismo equipo. Esto, de forma muy probable, se consigue utilizando un servidor externo para almacenar la base de datos.

Una vez que los componentes de un equipo están introducidos en el sistema se puede proceder a apuntar los goles y el resultado de un partido.

Figura 11.- Interfaces app Waterpolo



- **Características positivas.**

1. Informa a los participantes de un equipo por correo electrónico.
2. Utilización de servidor externo para almacenamiento de información.
3. Diferentes usuarios pueden acceder a consultar los datos de un mismo equipo a través de otros dispositivos móviles con solo tener la aplicación instalada.
4. Posibilidad de introducir la fecha de un partido seleccionando el día en un calendario desplegable.

- **Características negativas.**

1. La navegación por la aplicación es bastante difícil y nada intuitiva.
2. Algunas veces la aplicación reporta errores.
3. Publicidad en el menú principal.
4. Existe una versión de pago que no tiene publicidad y la forma de mostrar las estadísticas está mejorada.

2.4.4 Waterpolo sheets.

- **Descripción.**

Figura 12.-Icono app WPSheets



Se muestra la descripción que aparece en el “Google Play”:

“This is an app that register all the events of a water polo match and at the makes you can get the game sheet.”

Esta herramienta se enfoca en la función de apuntar los goles y faltas cometidas durante un partido, dividiendo éste entre los diferentes cuartos de tiempo. Como información complementaria se puede añadir la fecha y el lugar donde se disputa el encuentro. Una vez que se han recopilado todos los datos, se exporta un documento con la información estadística.

Figura 13.- Interfaces app WPSheets

Championship	Referee1	TimeOut White Team
st	st	0
Date	Referee2	TimeOut Blue Team
12	fo	0
Location	st	
Game Number	1	

Quarter	W	B
1 ^o quarter	1	0
2 ^o quarter	0	0
3 ^o quarter	0	0
4 ^o quarter	0	0
final	1	0

WHITE TEAM	White Team:	Time	Cap	Number	Action	Score
PlayerNumber	PlayerName	GoalsNumber	FaultsNumber	5	W 2	G 1-0
0	Player 1	0	0	6	W 8	E 1-0
1	Player 2	1	0		0	
2	Player 3	0	0		0	

- **Características positivas.**

1. Posibilidad de añadir información complementaria en cuanto a un partido: nombre del campeonato, lugar, fecha...
2. El registro de los datos contiene el detalle de en qué cuarto de tiempo sucedió.
3. Se exporta un documento con la información registrada.

- **Características negativas.**

1. No se muestran las estadísticas directamente en la aplicación sino que hay que consultar el documento que ésta exporta.
2. Aplicación solamente enfocada a la gestión estadística.
3. El diseño no está depurado.

3 Diseño

En esta sección se lleva a cabo un análisis sobre las principales características, funcionamiento y puntos fuertes y débiles de la aplicación. Además se explican las decisiones tomadas durante el desarrollo y diseño del proyecto.

3.1 Intención.

Como se ha comentado en apartados anteriores, el objetivo es el de cubrir una “necesidad” de información, en iniciados y potenciales practicantes del waterpolo, que ha surgido a raíz de la fuerte entrada que ha tenido el waterpolo en los medios de comunicación.

Para ello, se ha aprovechado la gran integración en la sociedad de los dispositivos electrónicos como son los teléfonos inteligentes. De tal forma, se ha buscado desarrollar una aplicación de fácil uso y cimentada en las necesidades y los consejos propuestos por practicantes y técnicos del deporte tratado, distinguiéndose de esta manera de otras ya ofertadas en el mercado.

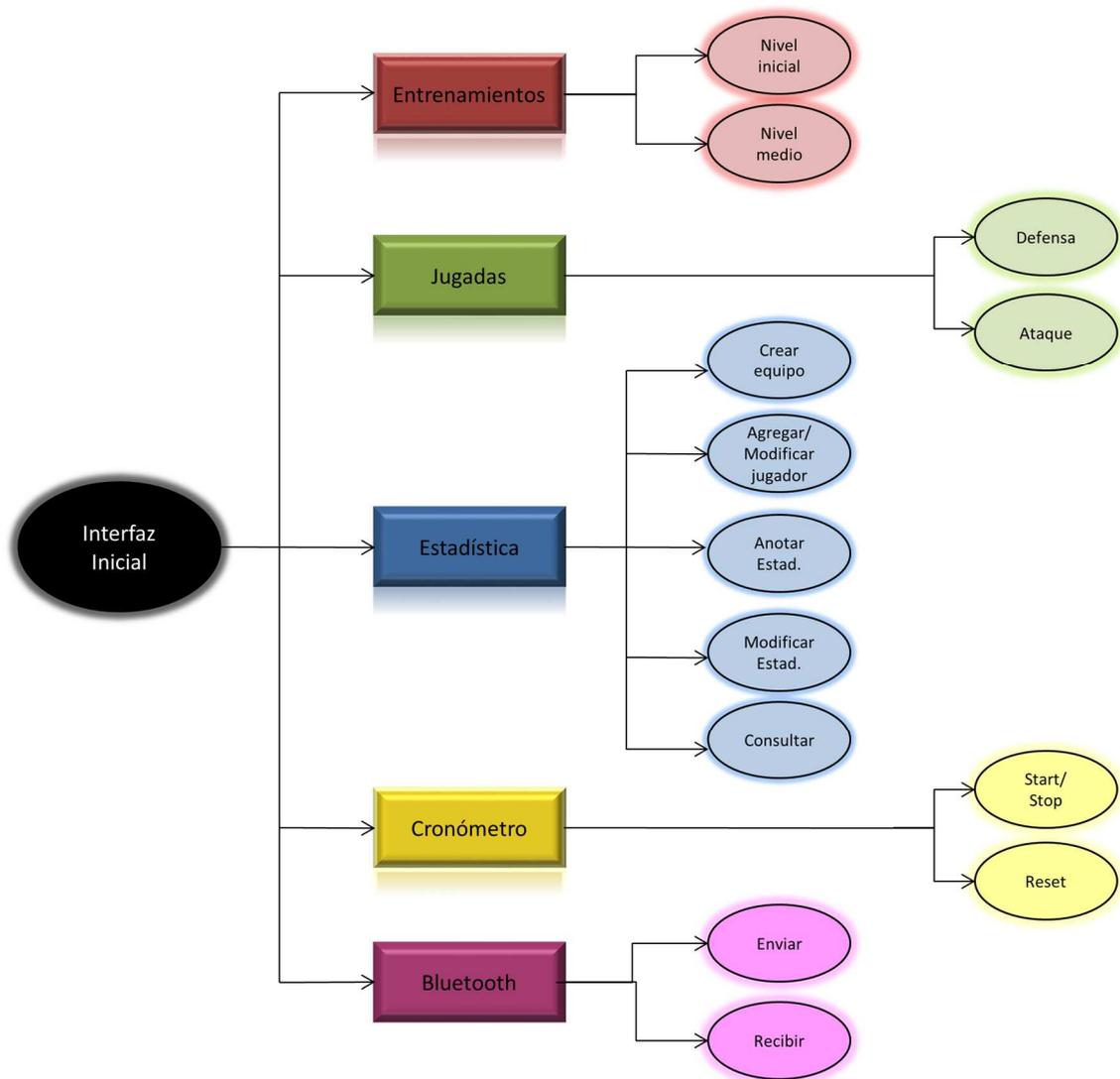
En los siguientes apartados se mostrarán las diferentes opciones y recursos de que dispone la aplicación creada.

3.2 Diagrama de estados.

Si observamos el mercado, la característica principal de una aplicación ya consolidada es la interfaz de usuario utilizada, siendo ésta de fácil manejo e intuitiva.

En el siguiente esquema se muestra la posible navegación que se puede realizar a través de la aplicación en base a sus funcionalidades y apartados:

Figura 14.- Diagrama de estados



La aplicación no dispone de un primer apartado definido, es decir, no presenta un menú inicial como tal, sino que se puede navegar por los diferentes apartados de forma directa. Todo esto se consigue a través de cambiar de pantalla, se arrastra el dedo de forma horizontal por la pantalla y de esa manera se accede a las diferentes funcionalidades de la aplicación. También se puede acceder a cada sub-menú seleccionando unos botones identificativos de cada uno. Posteriormente, como se observa en el esquema, cada apartado tiene sus diferentes ramas y opciones.

3.3 Funciones de la aplicación.

La aplicación cuenta con varias funcionalidades, las cuales se vienen comentando anteriormente y ahora se va a proceder a describirlas.

3.3.1 Entrenamientos.

En este apartado se muestran una serie de entrenamientos divididos en dos niveles: Nivel Inicial o Principiante y Nivel Intermedio. Estos entrenamientos están compuestos por diferentes ejercicios para varias sesiones, en los cuales se explica en qué consisten y el descanso a realizar en cada uno. La forma de introducir más entrenamientos es bastante simple, por tanto, es una ventaja a tener en cuenta porque siempre se podría actualizar. Por ejemplo, como línea futura se podría llevar un registro de entrenamientos por meses, es decir, mostrar planificaciones de preparaciones físicas y tácticas mensuales.

3.3.2 Jugadas.

Existe un repertorio de jugadas, tanto para ataque como para defensa, a disponibilidad del usuario. Todas las jugadas están identificadas por su propio nombre y organizadas en jugadas de ataque y defensa. Para hacer la explicación completa de las jugadas, éstas han sido divididas en fases y se han añadido textos explicativos para cada una de ellas. La estructura de las jugadas está muy bien representada en el apartado de Desarrollo. Pero un breve esquema sería el siguiente:

❖ Ataque.

1. Entrada cruzada.
2. Segundo Boya.
3. Romper defensa en zona.
4. Bloqueo 5-4.
5. Jugada de 1+.

❖ Defensa.

1. Defensa en press.
2. Defensa en M.
3. Defensa en zona.
4. La India.
5. Jugada de 1-.

3.3.3 Estadística.

La creación y modificación de registros estadísticos es la parte principal de la aplicación. Con el fin de ofrecer un robusto sistema de almacenamiento de datos se ha creado una base de datos. El procedimiento es bastante automático e intuitivo, a grandes rasgos sería: crear un equipo, agregar un jugador a dicho equipo e introducir los valores asociados al jugador que se está analizando.

- ❖ **Crear Equipo.** Permite introducir un equipo nuevo en la base de datos. Éste será el primer elemento en la jerarquía de la base de datos del cual colgarán los distintos registros dependientes de los jugadores que se añadan.
- ❖ **Agregar / Modificar Jugador.** Mediante pulsar un botón, claramente identificativo, se puede agregar un jugador introduciendo sus datos identificativos como son su nombre, la posición en la que juega y el equipo al que pertenece. Además, se puede relacionar un jugador a diferentes equipos y posiciones permitiendo así un número de registros, datos de un mismo jugador, estando éste identificado por su nombre.
En el caso de que se quisiera insertar un jugador, con un nombre que ya estuviera registrado en la base de datos, la aplicación no lo permitiría. (En un principio esto sí era posible puesto que se asignaban id diferentes aunque fueran el mismo nombre).
- ❖ **Anotar Estadística.** Se introducen, de forma rápida y cómoda, los valores de los diferentes parámetros que se deseen recoger durante un partido. Siempre para un determinado jugador, para una posición en la que juega y para el equipo al que pertenece.
- ❖ **Modificar Estadística.** Existe la posibilidad de modificar una estadística al cambiarse de posición el jugador, de equipo o simplemente creando un nuevo registro para dicho jugador con los mismos parámetros identificativos anteriores.
- ❖ **Consultar.** Para cada jugador se va almenando un historial estadístico que siempre puede ser consultado si el jugador no se encuentra desactivado.

3.3.4 Cronómetro.

La aplicación dispone de un cronómetro digital, una función interesante que presenta las siguientes ventajas:

- ❖ Es un complemento perfecto para medir los descansos de las series propuestas en los entrenamientos o la duración de los ejercicios. Así como diferentes tiempos de posesión o cuartos durante un partido.
- ❖ Puede estar en funcionamiento, es decir, contando segundos mientras se consultan los entrenamientos o se apaga la pantalla del móvil por el ahorro energético del propio dispositivo.
- ❖ Tiene la opción de detener la cuenta y proseguir con ella cuando se presione el botón correspondiente.
- ❖ Siguiendo la línea de una interfaz clara para el usuario, el cronómetro está centrado en una pantalla utilizada solamente para esta función.

3.3.5 Bluetooth.

Una funcionalidad más de la herramienta creada es la de permitir enviar los datos estadísticos registrados a otro dispositivo que disponga de bluetooth. Si ese dispositivo tiene instalada la

aplicación podrá ver los registros estadísticos de un jugador de la misma manera que en el origen.

3.4 Requisitos.

En los siguientes apartados se exponen y se argumentan las elecciones tomadas en cuanto a temas técnicos, a la vez que se informa de los requisitos necesarios para que la aplicación se ejecute correctamente en un dispositivo electrónico.

3.4.1 Elección del Sistema Operativo.

Se han estudiado diversos factores y cualidades de diferentes Sistemas Operativos y finalmente se eligió Android. Los factores estudiados han sido: cuota de mercado, facilidades de programación, diferencia de precios (facilidad de adquisición), categorías con más posibilidades de ser buena inversión, proyección financiera de las aplicaciones, idioma del público objetivo, línea de futuro, cifras a nivel mundial y el predominio en dispositivos electrónicos.

✓ Cuota de mercado.

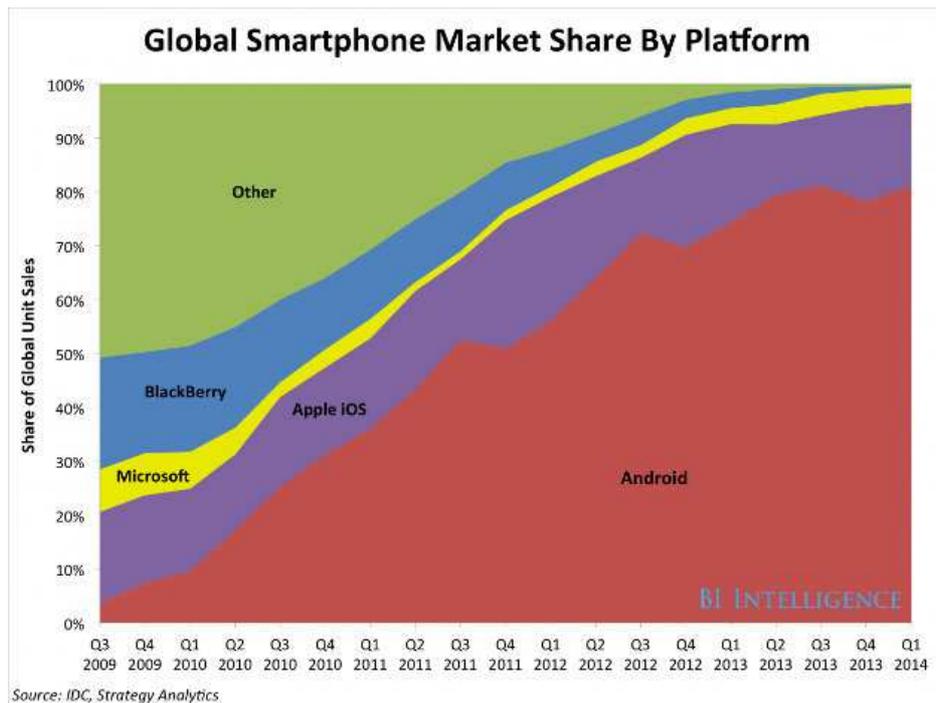
Cuando se desarrolla una aplicación el principal objetivo es buscar una salida en el mercado para la herramienta desarrollada y que llegue al mayor número de personas. Por tanto, se muestra la cuota de mercado que tienen diferentes Sistemas Operativos.

Tabla 2.- Cuota de mercado Sistemas Operativos

Worldwide Smartphone Sales to End Users by Operating System in 3Q14 (Thousands of Units)				
Operating System	3Q14 Units	3Q14 Market Share (%)	3Q13 Units	3Q13 Market Share (%)
Android	250,060.2	83.1	205,243	82.0
iOS	38,186.6	12.7	30,330	12.1
Windows	9,033.4	3.0	8,916	3.6
Blackberry	2,419.5	0.8	4,401	1.8
Other OS	1,310.2	0.4	1,407	0.6
Total	301,009.9	100.0	250,297	100.0

Fuente: Gartner (December 2014)

Figura 15.- Gráfica con cuotas de mercado Sistemas Operativos.



Fuente: "IDC, Strategy Analytics"

Se puede apreciar que la cuota de mercado es, en un alto nivel, dominada por Android. Por tanto es un Sistema Operativo fuerte y probado, con una gran participación en el mercado.

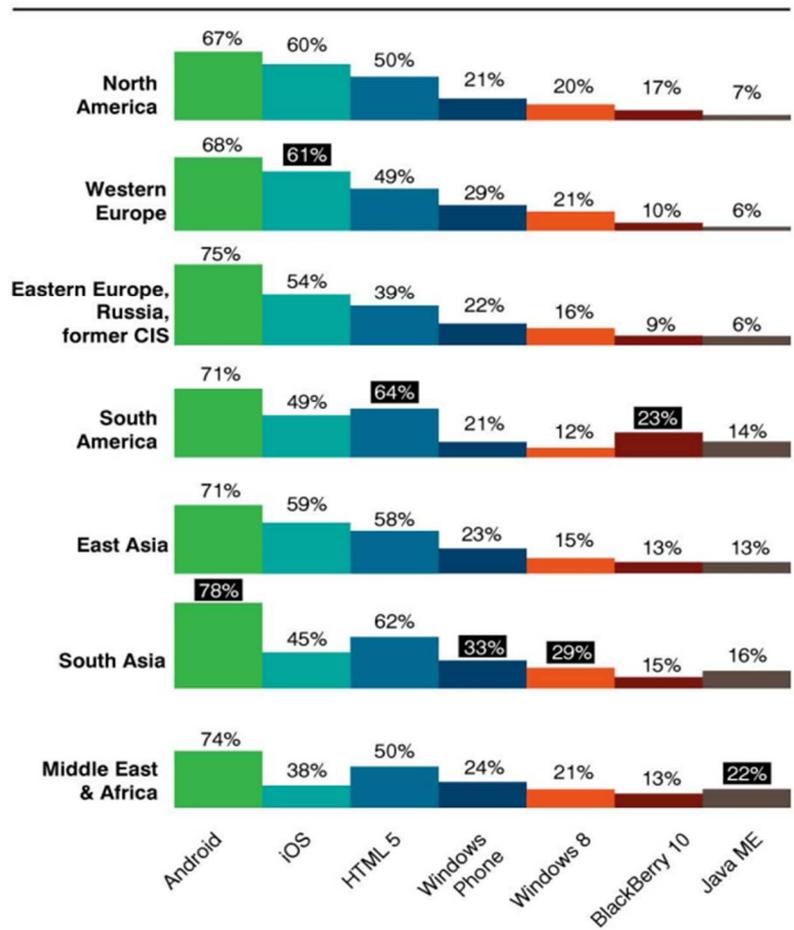
✓ **Facilidades de programación – Apertura a la creación de programas.**

El sistema Android tiene un gran número de facilidades a la hora de programar. Prueba de ello es el gran número de personas que se decantan por programar para este Sistema Operativo.

Figura 16.- Distribución de programadores por SO y Zona Geográfica

MOBILE DEVELOPER MINDSHARE BY REGION, Q1 2014

% of developers within each region using each platform (n=6,311)



% Highest regional Mindshare for the platform



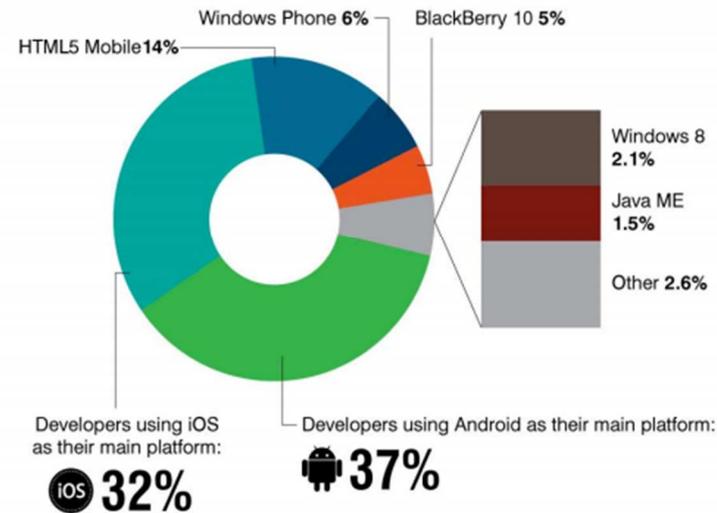
Licensed under CC BY ND | Copyright VisionMobile

Source: Developer Economics Q1 2014 | www.DeveloperEconomics.com/go

Figura 17.- Porcentajes de programadores por plataforma principal

DEVELOPER LOYALTY SPLIT BETWEEN IOS& ANDROID

% of developers using each platform as their main platform (n=6,311)



Licensed under CC BY ND | Copyright VisionMobile
Source: Developer Economics Q1 2014 | www.DeveloperEconomics.com/go

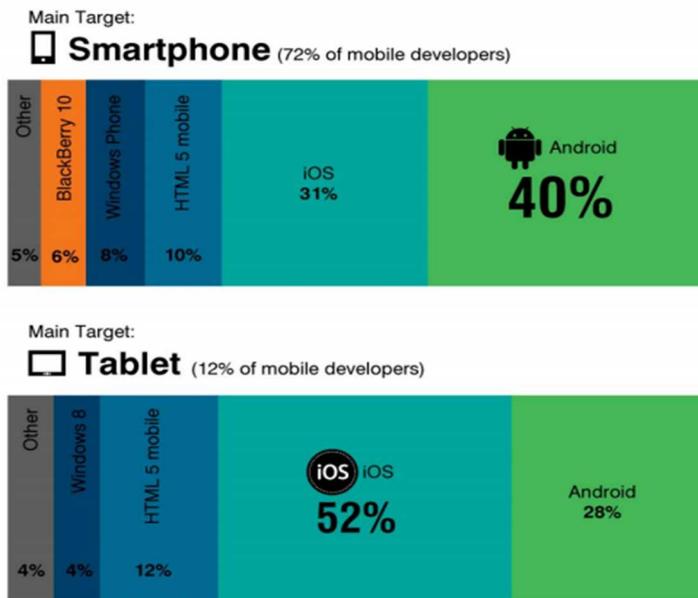
A pesar de que las aplicaciones desarrolladas para iOS (Sistema Operativo de Apple) suelen tener un porcentaje mayor de ganancias económicas, hay un mayor número de programadores que prefieren utilizar Android para crear sus aplicaciones.

Si bien, en el caso de las tabletas el sistema Operativo predominante continua siendo el iOS.

Figura 18.- Preferencia de SO por desarrolladores según dispositivo

IOS REMAINS THE PREFERRED PLATFORM ON TABLETS

% of developers prioritising each platform, by priority screen (n=6,311)

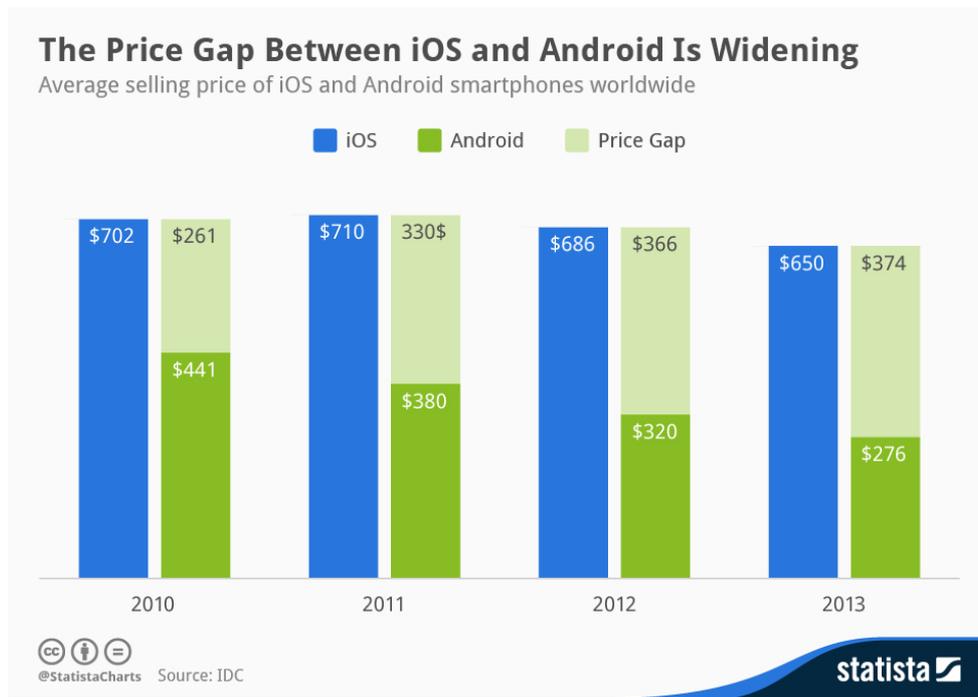


Licensed under CC BY ND | Copyright VisionMobile
Source: Developer Economics Q1 2014 | www.DeveloperEconomics.com/go

✓ **Diferencia de precios – Facilidad de adquisición.**

Un buen argumento a la hora de decantarse por un Sistema Operativo en el que desarrollar una aplicación puede ser la facilidad de adquisición de un teléfono inteligente, es decir que tenga un precio asequible para el usuario. Además en los tiempos de crisis actuales éste puede ser un factor muy importante.

Figura 19.- Diferencia de precios entre SO



Fuente: "IDC".

Como se puede observar, el precio promedio de un móvil con Sistema Operativo iOS de Apple es bastante mayor que uno que utilice Android.

✓ **Categorías con más posibilidades de ser buena inversión.**

Los juegos son la categoría de aplicaciones para dispositivos móviles que más ingresos tienen. Esto ocurre tanto para Android como para iOS.

Figura 20.- Beneficio de aplicaciones por categorías



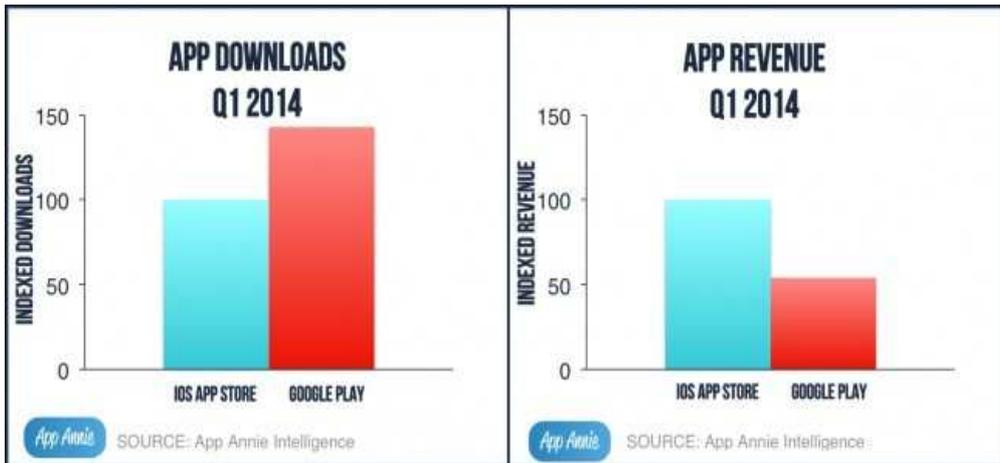
La aplicación que se ha desarrollado está más orientada como una herramienta (puesto cuarto en "Google Play") pero también recoge aspectos deportivos y de esa forma se relaciona un poco en temática con las aplicaciones que se encuentran catalogadas como juegos.

✓ **Proyección financiera de las aplicaciones.**

El número de aplicaciones descargadas a través de “Google Play”, sistema Android, es mayor que las obtenidas a través de la “iOS App Store”, sistema iOS de Apple.

En cambio el beneficio obtenido por dichas aplicaciones es mayor en el caso de las que se desarrollan para iOS.

Figura 21.-Proyección financiera



De estos dos datos se pueden extraer dos conclusiones:

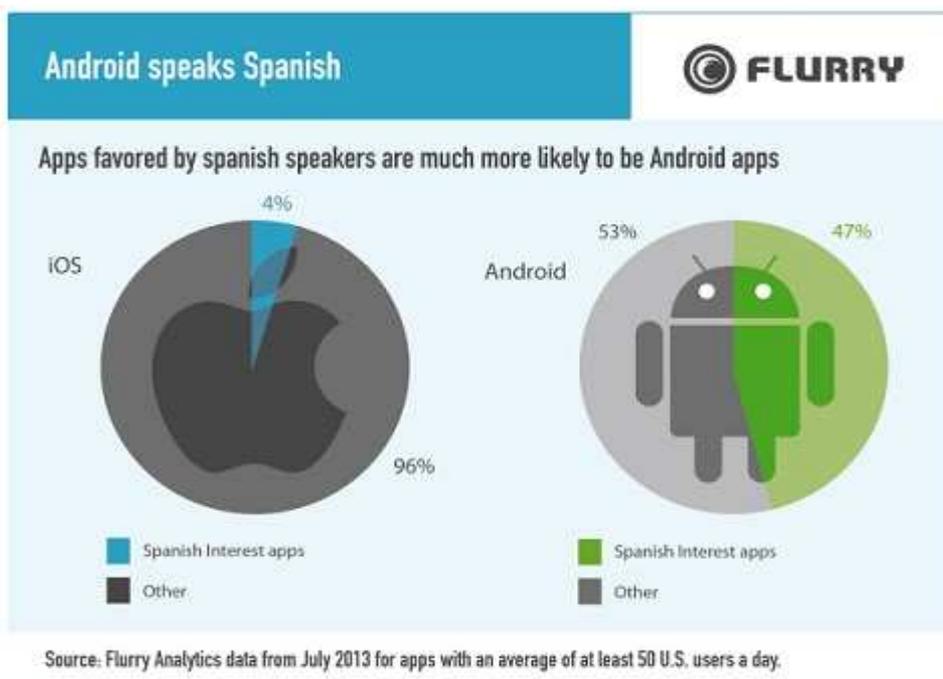
La primera es que si se acaba de comenzar a desarrollar aplicaciones es mejor empezar en sistema Android puesto que tiene más posibilidades de ser conocida y llegar a más gente.

La segunda, es que si se está más interesado en obtener beneficios sería bueno crear una aplicación en sistema iOS una vez ya se haya obtenido experiencia programando otras aplicaciones en Android.

✓ **Idioma del público objetivo.**

Las personas que hablan castellano, en un alto porcentaje, prefieren aplicaciones para Android.

Figura 22.- Uso de SO por idioma

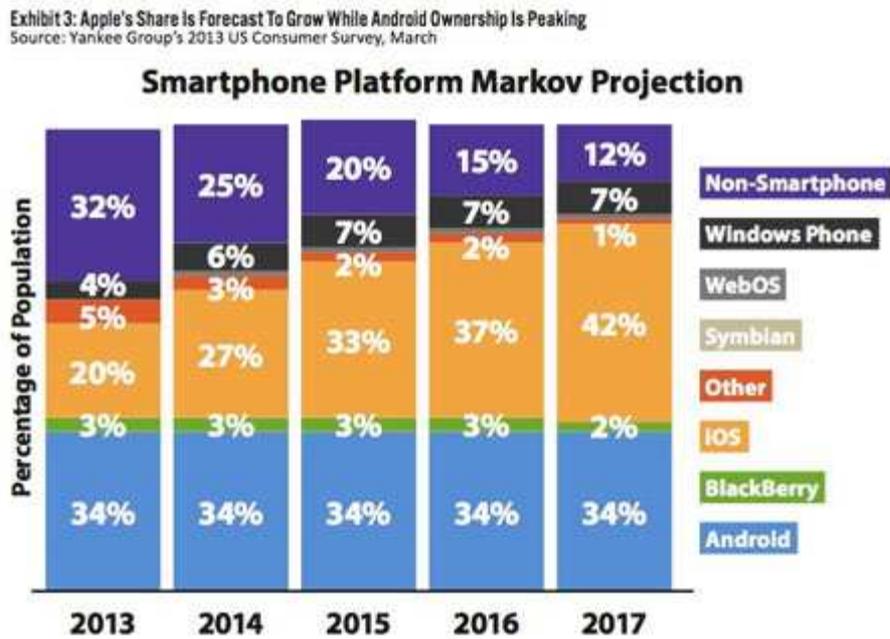


La aplicación ha sido desarrollada en España y por tanto tiene un origen en idioma castellano y por consecuencia podría tener más aceptación y mejor comprensión de la herramienta por parte de personas que hablen la misma lengua.

✓ **Línea de futuro.**

Aunque estos datos reflejan una previsión del porcentaje de población que utilizará un determinado Sistema Operativo en Estados Unidos, nos sirve bien como propuesta de buena perspectiva futura para los dispositivos móviles que utilicen Android, puesto que se mantiene estable para años futuros.

Figura 23.- Proyección usuarios de los SO



Fuente: "Yankee Group's 2013 US Consumer Survey, March

✓ **Cifras a nivel mundial.**

Representación a nivel geográfico mundial de la expansión de los Sistemas Operativos predominantes para teléfonos móviles inteligentes y tabletas.

Figura 24.- Representación mundial de SOs

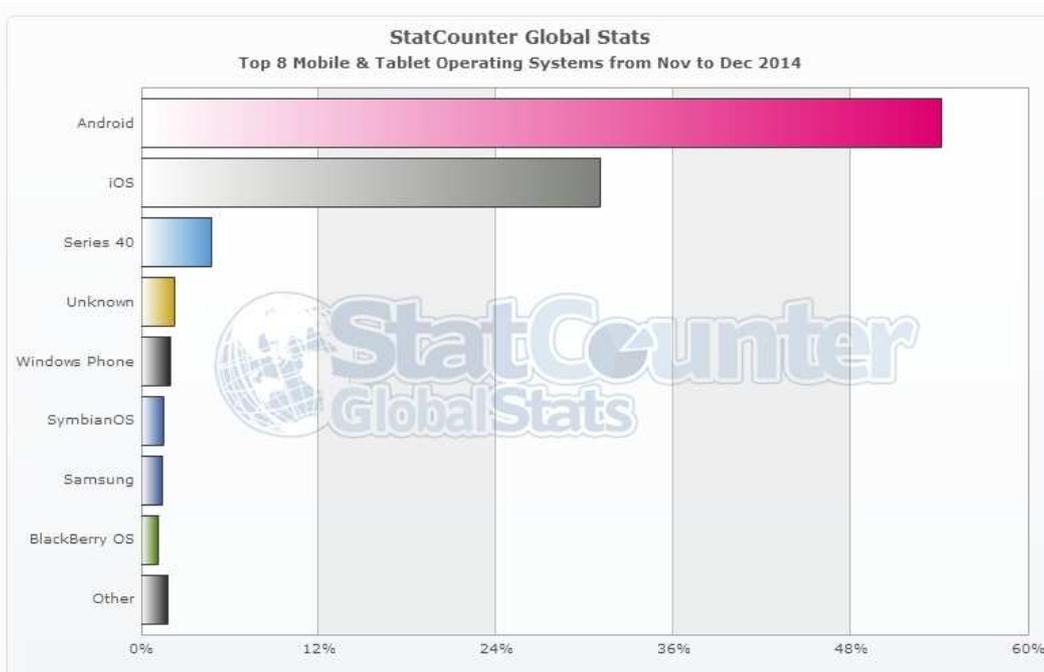


Fuente: "StatCounter", datos para la fechas comprendidas entre Noviembre y Diciembre de 2014. En el siguiente enlace se pueden consultar los porcentajes de uso para cada Sistema

Operativo en cada país: <http://gs.statcounter.com/#mobile+tablet-os-ww-monthly-201411-201412-map>.

Gráfica con los porcentajes de uso a nivel global.

Figura 25.- Gráfica con porcentajes de uso de cada SO a nivel global

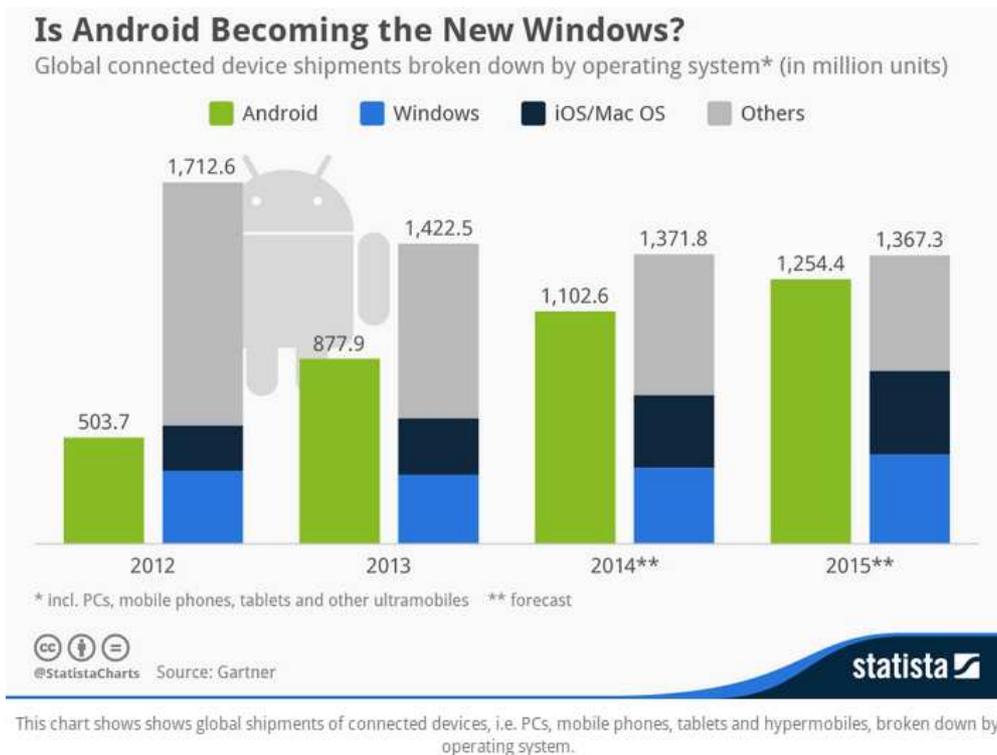


Fuente: "StatCounter", datos para la fechas comprendidas entre Noviembre y Diciembre de 2014.

✓ **Predominio en dispositivos electrónicos.**

En la siguiente gráfica se muestra el número de dispositivos con un determinado Sistema Operativo. En esta representación están incluidos los PCs, los teléfonos inteligentes y las tabletas.

Figura 26.- Predominio en diferentes dispositivos



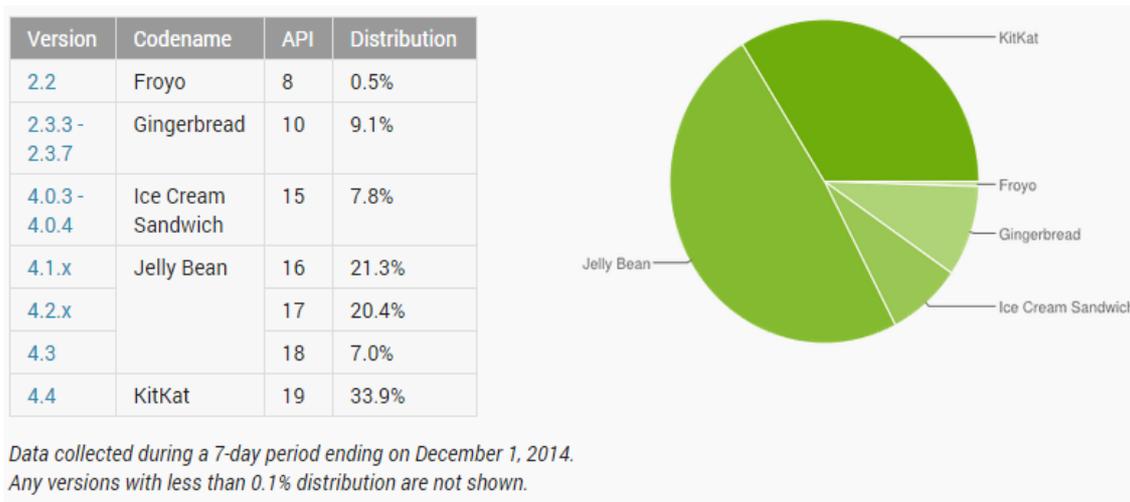
Como se puede apreciar, Android ha alcanzado un peso mayor que el del gigante Windows si unificamos en una gráfica todos los dispositivos que pudieran llevar instalado alguno de estos Sistemas Operativos.

3.4.2 Versión Android.

Si se consulta la página oficial de Android se pueden ver cuáles son las versiones más instaladas y su porcentaje. Con esta información se puede establecer una versión mínima con la que se cubriría la mayor parte de los dispositivos con sistema Android.

Además, el programa utilizado para el desarrollo de la aplicación permite fácilmente variar la versión Android objetivo, marcando una versión mínima y otra máxima que soportarán la aplicación. Un aspecto positivo a tener en cuenta es que las funciones orientadas a una versión antigua sí podrán ser ejecutadas en versiones nuevas, aunque esto no ocurre para el caso contrario. Es decir, se debe tener en cuenta que las funciones nuevas que se creen para futuras versiones de Android no podrán ser ejecutadas por versiones antiguas.

Figura 27.- Distribución de dispositivos por versión Android



Fuente: <http://developer.android.com/about/dashboards/index.html>

En el programa utilizado existe un fichero denominado como “AndroidManifest.xml” en el que están declaradas las versiones mínimas y objetivo o máximo del sistema Android.

```
android:minSdkVersion="13"
android:targetSdkVersion="19"
```

En este caso se ha seleccionado como versión mínima la decimotercera. Esto fue así porque el dispositivo móvil con el que se iniciaron las pruebas tenía dicha versión. Además, se ha decidido permanecer con este margen de versiones puesto que de esta manera, observando los datos mostrados por la página oficial de Android, se consigue abarcar un altísimo porcentaje de dispositivos y también utilizar funciones más nuevas.

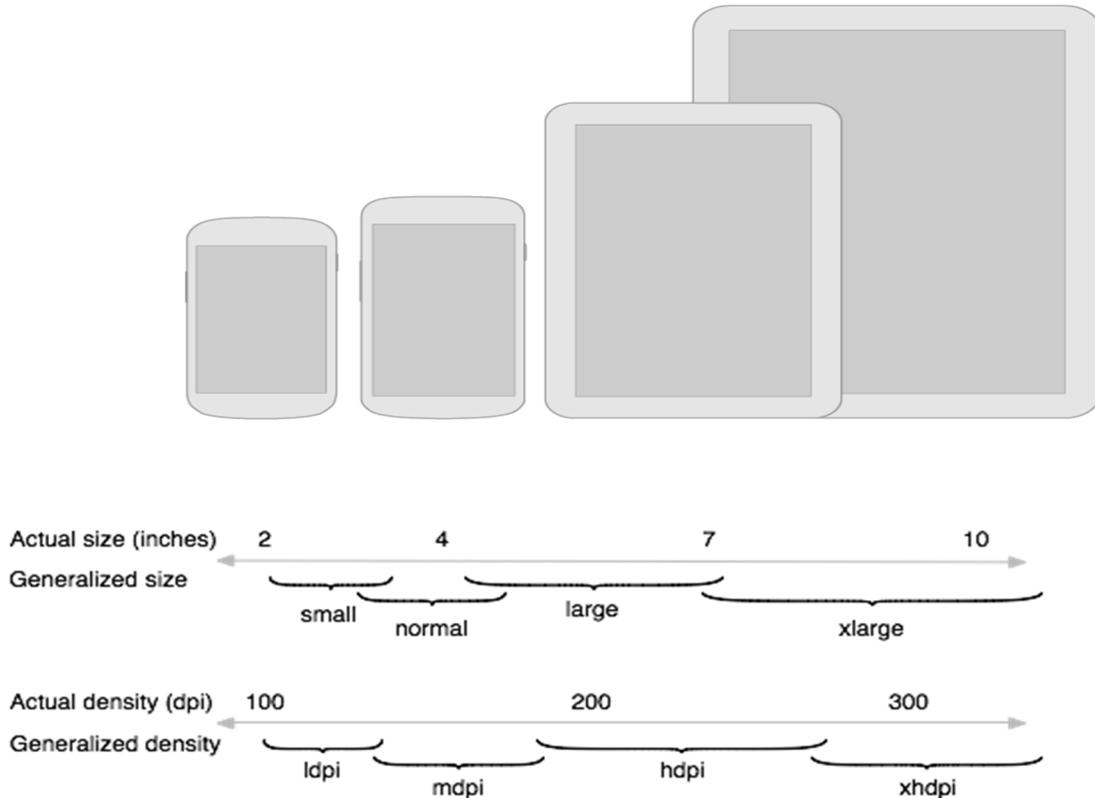
3.4.3 Dimensión de pantalla.

Una de las características más importantes de una aplicación es la forma en la que esta se muestra al usuario. Por tanto, no sólo es importante que la aplicación ofrezca funcionalidades variadas y realmente útiles sino que también, y más trascendente aún, que ofrezca una interfaz de usuario comprensible y atractiva. Por otro lado, la aplicación desarrollada está pensada como una herramienta de apoyo y de análisis de un deporte, con lo cual el teléfono inteligente sería el objetivo, puesto que es el dispositivo más adecuado por sus dimensiones y peso.

En línea con lo comentado anteriormente, se han analizado los diferentes tamaños de pantalla que se ofrecen en el mercado. Las medidas utilizadas son el tamaño y la densidad. El tamaño viene marcado por la longitud de la diagonal de la pantalla. La densidad hace referencia al número de píxeles por pulgada (en castellano “ppp”: puntos por pulgada, en inglés “dpi”: “dots per inch”). Esta segunda medida es la que realmente define la calidad de la imagen que podrá mostrar el dispositivo puesto que hace referencia directa a la resolución de la pantalla.

La plataforma utilizada ha sido Android y por tanto consultamos la clasificación que este Sistema Operativo realiza en cuanto a los tamaños y densidades de pantalla.

Figura 28.- Tamaños de pantalla



Fuente: Página web oficial de Android.

Como se puede observar hay cuatro tamaños distintos definidos de forma aproximada: pequeño (entre dos y tres pulgadas), normal (entre tres y cinco pulgadas), grande (entre cuatro y siete pulgadas) y extra-grande (entre siete y once pulgadas). En cuanto a la densidad se distingue entre baja, media, alta y extra-alta. Aunque en el dibujo anterior no se muestra, en la actualidad hay mayores grados de densidad siendo estos los siguientes:

- ❖ ldpi (low) ~ 120dpi
- ❖ mdpi (medium) ~ 160dpi
- ❖ hdpi (high) ~ 240dpi
- ❖ xhdpi (extra-high) ~ 320dpi
- ❖ xxhdpi (extra-extra-high) ~ 480dpi
- ❖ xxxhdpi (extra-extra-extra-high) ~ 640dpi

En la siguiente tabla se recogen la distribución de los dispositivos por la densidad y tamaño de sus pantallas.

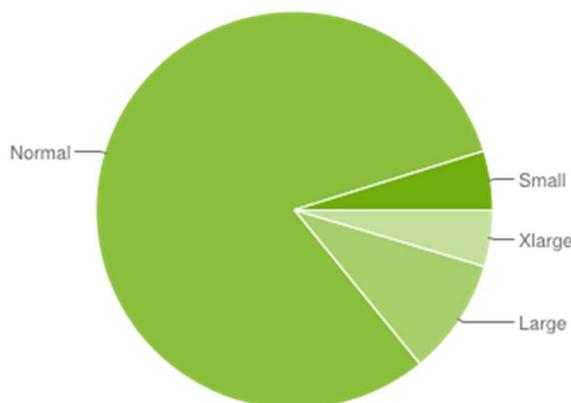
Tabla 3.- Distribución de móviles por tamaño y densidad de pantalla

	ldpi	mdpi	tvdpi	hdpi	xhdpi	xxhdpi	Total
Small	4.8%						4.8%
Normal		8.8%	0.1%	37.5%	18.4%	16.3%	81.1%
Large	0.6%	5.4%	2.3%	0.6%	0.6%		9.5%
Xlarge		3.7%		0.3%	0.6%		4.6%
Total	5.4%	17.9%	2.4%	38.4%	19.6%	16.3%	

Se puede apreciar que la mayor concentración de dispositivos se encuentra para las pantallas que tienen un tamaño normal (alrededor de tres y cinco pulgadas) y una densidad de pantalla alta (aproximadamente 240 “dpi”).

Si se representan en un gráfico la expansión de los diferentes tipos de tamaño se puede comparar, de forma más visual, que la dimensión predominante es la denominada como normal.

Figura 29.- Gráfico con la distribución de los tamaños de pantalla

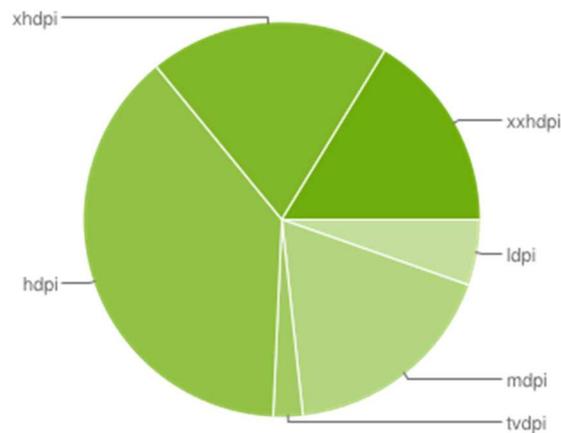


Data collected during a 7-day period ending on January 5, 2015. Any screen configurations with less than 0.1% distribution are not shown.

Fuente: Página web oficial de Android.

Por otro lado, la densidad, que es la otra característica analizada, tendrá una distribución de la siguiente forma.

Figura 30.- Gráfico con la distribución de las densidades de pantalla



Data collected during a 7-day period ending on January 5, 2015. Any screen configurations with less than 0.1% distribution are not shown.

Fuente: Página web oficial de Android.

Como se vio en la tabla mostrada anteriormente, la densidad de pantalla que más dispositivos utilizan es la denominada como alta, aunque las nuevas y mejoradas pantallas están aumentando considerablemente.

En el caso de nuestro software el “layout” escogido es el que engloba al mayor porcentaje de pantallas que se ha estudiado (tamaño normal y resolución alta). Además, el “layout” puede ser representado de manera óptima en diferentes tamaños de dispositivos. En cambio, las imágenes que se quieran mostrar a través de la aplicación sí que deberán estar adaptadas a las características de la pantalla.

3.4.4 Idioma.

Con el fin de abarcar un número mayor de usuarios de la herramienta desarrollada, ésta ha se puede representar en dos idiomas diferentes: castellano e inglés. Si el idioma definido en el dispositivo es el inglés la aplicación se mostrará en dicho idioma y el mismo efecto ocurre si el lenguaje del dispositivo es castellano.

4 Desarrollo

En este capítulo el objetivo que se seguirá será el de mostrar el funcionamiento de la aplicación desarrollada desde una perspectiva más técnica. Para ello, se explicará cómo funciona cada bloque del programa y la interconexión existente entre ellos.

4.1 Inicios.

Esta es la primera aplicación móvil desarrollada y por tanto fue necesario dar unos primeros pasos en cuanto a conocer la plataforma Android para entender cómo funciona y las opciones que ésta muestra. Por otro lado, se recabó información sobre el waterpolo para utilizar en la aplicación.

Los pasos dados fueron:

- ❖ Lectura de un libro de aprendizaje de Android.
- ❖ Lectura y estudio de información, relacionada con el tema de análisis, proporcionada en páginas de internet especializadas.
- ❖ Descarga y análisis de algunas aplicaciones a modo ejemplo.
- ❖ Realización de tutoriales.
- ❖ Consulta a entrenadores de waterpolo.

4.2 Herramientas utilizadas.

La base del proyecto ha sido el ejercicio de programación que se ha realizado y éste ha sido utilizando el programa Eclipse con la extensión de utilidades y herramientas que proporcionan el SDK Manager y el AVD Manager.

4.2.1 Eclipse.

Explicación de qué es Eclipse de la propia marca: “Eclipse is a platform that has been designed from the ground up for building integrated web and application development tooling. By design, the platform does not provide a great deal of end user functionality by itself. The value of the platform is what it encourages: rapid development of integrated features based on a plug-in model.

Eclipse provides a common user interface (UI) model for working with tools. It is designed to run on multiple operating systems while providing robust integration with each underlying OS. Plug-ins can program to the Eclipse portable APIs and run unchanged on any of the supported operating systems.

At the core of Eclipse is an architecture for dynamic discovery, loading, and running of plug-ins. The platform handles the logistics of finding and running the right code. The platform UI provides a standard user navigation model. Each plug-in can then focus on doing a small number of tasks well. What kinds of tasks? Defining, testing, animating, publishing, compiling, debugging, diagramming...the only limit is your imagination". (Fuente: <http://help.eclipse.org/luna/index.jsp>)

Resumiendo, Eclipse es un programa para crear entornos de desarrollo integrados (en inglés, IDE). Por tanto, tiene el gran atractivo de que se pueden agregar, instalar nuevas y diferentes herramientas a dicho programa. En el caso que nos atañe, se ha podido añadir un kit de desarrollo software denominado SDK que incluye herramientas de desarrollo Java (JDT, "Java Development Toolkit") y de compilador (ECJ).

La versión utilizada de este programa ha sido "Luna".

4.2.2 Emulador de Android.

Se ha utilizado el AVD (Android Virtual Device) con el objetivo de emular un dispositivo que tuviera el sistema operativo Android instalado. Con ello, se consigue probar y analizar directamente la aplicación desarrollada en un ordenador. Para configurar dicho emulador se introducen parámetros de software, como es la versión de Android que se simulará, y otros parámetros de hardware, como son características de la pantalla o de la tarjeta de memoria entre otros posibles.

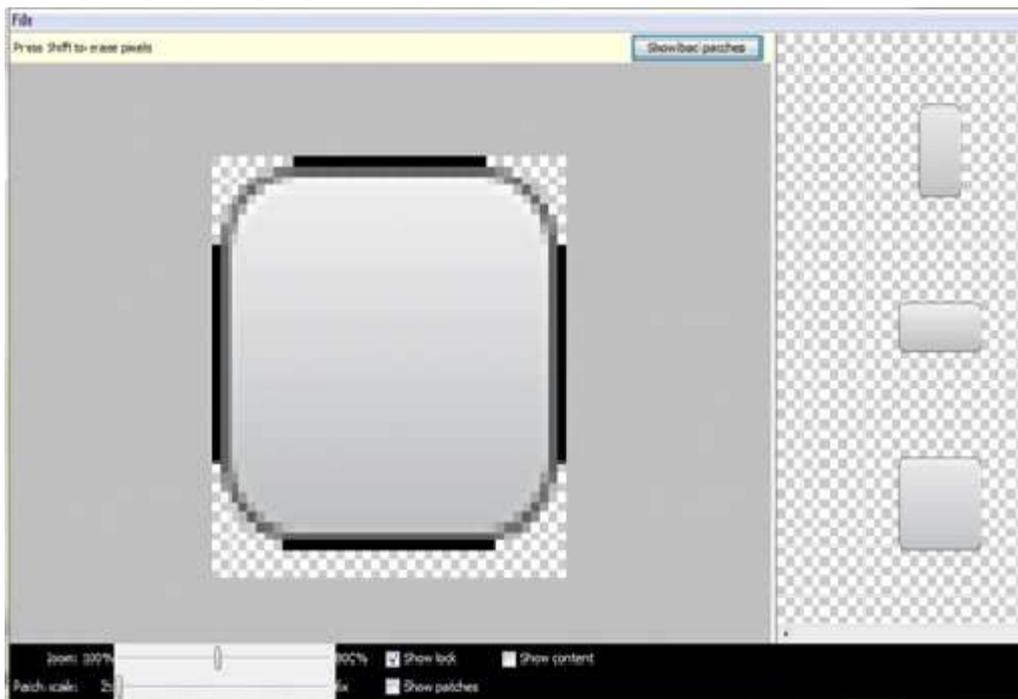
Además, se pueden crear varios AVDs como sean necesarios obteniendo así varias representaciones de cómo sería ejecutar la aplicación en distintos dispositivos con distintas características técnicas.

4.2.3 Herramienta Draw 9-patch.

Draw 9-patch es una herramienta proporcionada por el SDK de Android. La funcionalidad de ésta consiste en que permite adaptar las imágenes a los diferentes tamaños de pantalla sin que realmente se pierda calidad.

Esto se consigue a través de generar archivos de tipo "9-patch" a partir de imágenes tipo "png". Una representación tipo 9-patch es una imagen de mapa de bits extensible, o lo que es lo mismo, un tipo de imagen que el sistema Android es capaz de cambiar automáticamente de tamaño para de esta forma, adaptarla a la vista (view) a la que se encuentra asignada como "background".

Figura 31.- Captura del programa Draw 9-patch

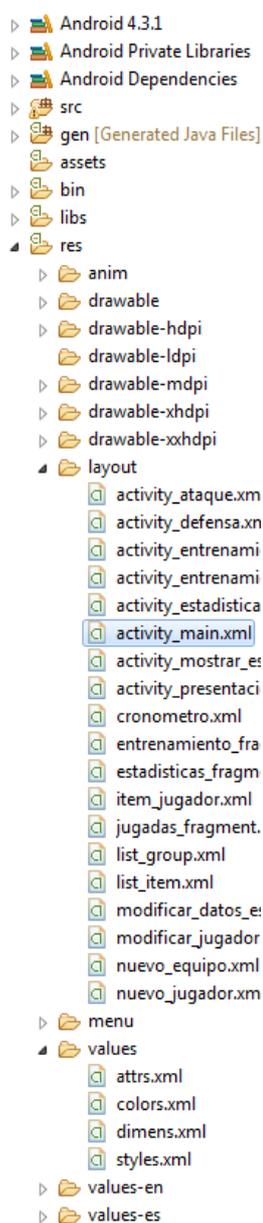


El modo de operar sería enmarcar en la imagen original las zonas que tendrán que ser redimensionadas en caso de ser necesario. Es decir, se mantendrán unas proporciones correctas en el caso de que se cambien las dimensiones de la imagen.

4.3 Archivos y directorios del proyecto de programación.

Una aplicación Android está compuesta por distintos archivos dependiendo del objetivo o de la función que cumplen dentro de la misma. El directorio es el siguiente:

Figura 32.- Directorio



❖ Librerías. Éstas son las librerías propias de cada versión Android y que se ponen a disposición del programador.

❖ Src. Carpeta que contiene el código fuente de la aplicación.

❖ Res. Aquí se encuentran los recursos utilizados por la aplicación: imágenes, archivos que definen la interfaz de usuario, archivos “xml” que contienen un conjunto de recursos como son las cadenas de texto o las definiciones de color, etc.

❖ AndroidManifest. Este archivo describe las características fundamentales de la aplicación y define los componentes. Además en él se declaran los permisos que requerirá la aplicación, se indica la versión mínima de Android para poder ejecutarla, etc.

4.3.1 Activity.

Una Activity (actividad) está formada por dos partes diferentes: una parte lógica y otra parte gráfica.

La parte lógica es la que hace referencia a un archivo de tipo java y es donde se encuentra el código que se encarga de controlar una determinada función de la aplicación. Es decir, es donde se encuentra el cuerpo de la programación que controla el funcionamiento de la aplicación.

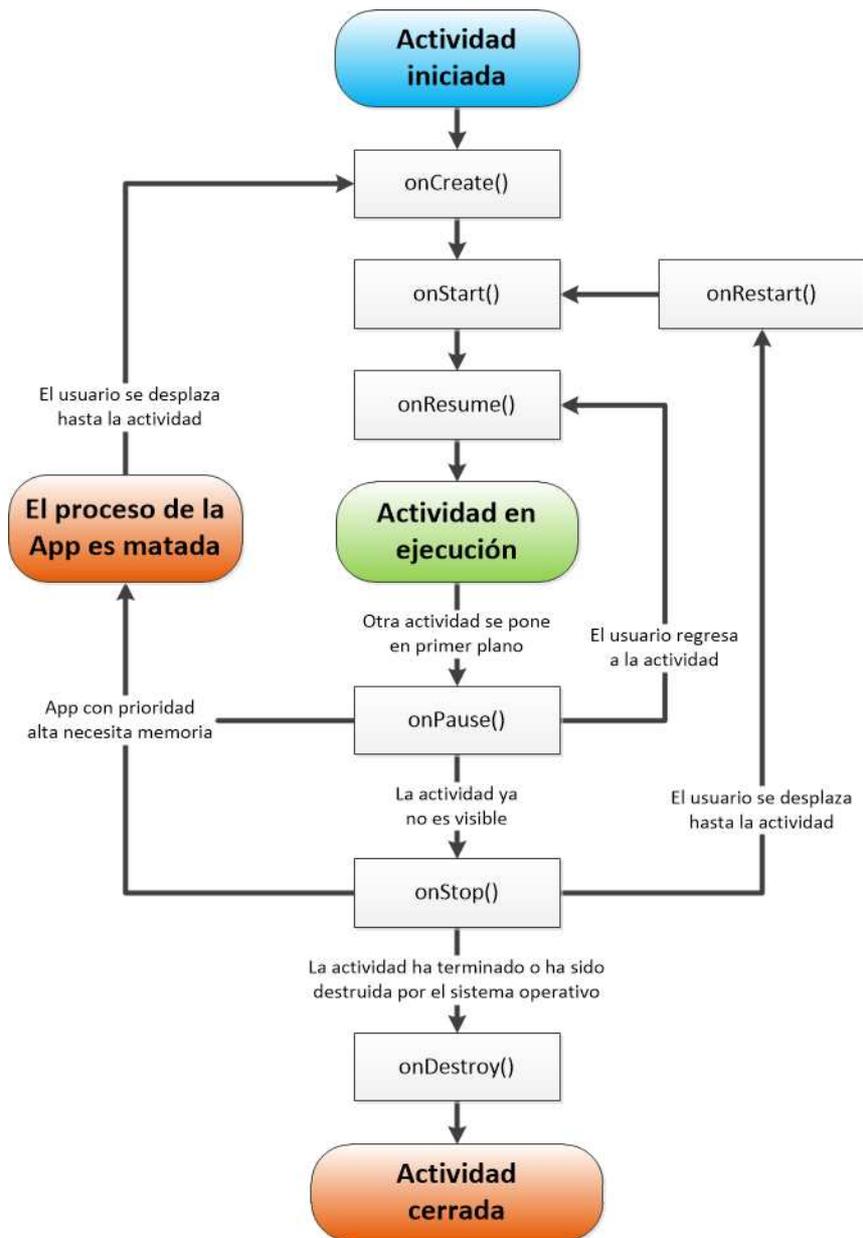
Por otro lado, la parte gráfica es un "xml" que contiene todos los elementos que se observan a través de la pantalla, es decir, la interfaz de usuario.

La principal característica es que las actividades tienen un ciclo de vida no lineal y se organizan en una pila (no necesariamente se crean y terminan por destruirse). Una actividad que es iniciada es situada en lo alto de la pila y se comienza a ejecutar. Sin embargo, la anterior actividad puede permanecer por debajo en la pila y volverse a ejecutar cuando se salga de la actual.

Una Activity tiene cuatro estados fundamentales:

1. Una actividad mostrada por pantalla, en primer plano se encuentra activada o ejecutándose. Es decir, está en la parte alta de la pila.
2. Si la actividad ya no se encuentra en primer plano pero aún es visible, significa que está en pausa. Es decir, está viva (mantiene todos los estados) pero puede ser matada por el sistema si se precisa de memoria.
3. Una actividad pasará al estado de stop si otra actividad más importante o más necesaria se quiere iniciar. En este momento ya no es visible por el usuario y la ventana es ocultada.
4. Si la actividad se encuentra en estado de stop o pausada, el sistema puede eliminarla matándola o finalizándola. (Más adelante, después de la figura se explica las posibles formas de terminar con una actividad).

Figura 33.- Estados de un Activity



Fuente: <http://developer.android.com/reference/android/app/Activity.html>

La actividad se puede terminar por dos motivos:

1. La actividad es matada. Se hace porque otra actividad se considera más importante o requiere atención. Por tanto, se retomará la actividad matada desde el principio cuando se desee. El ejemplo sería no cerrar una aplicación y cambiar de actividad o salir al menú principal de Android.
2. La actividad es cerrada. Se ha decidido claramente terminar con la actividad. En este caso el ejemplo más directo sería salir de la aplicación y apagar el smartphone.

4.3.2 Fragment.

El origen de los archivos de tipo fragment (fragmento) fue propiciado por la necesidad de adaptar la interfaz gráfica a dispositivos con diferentes tamaños de pantalla.

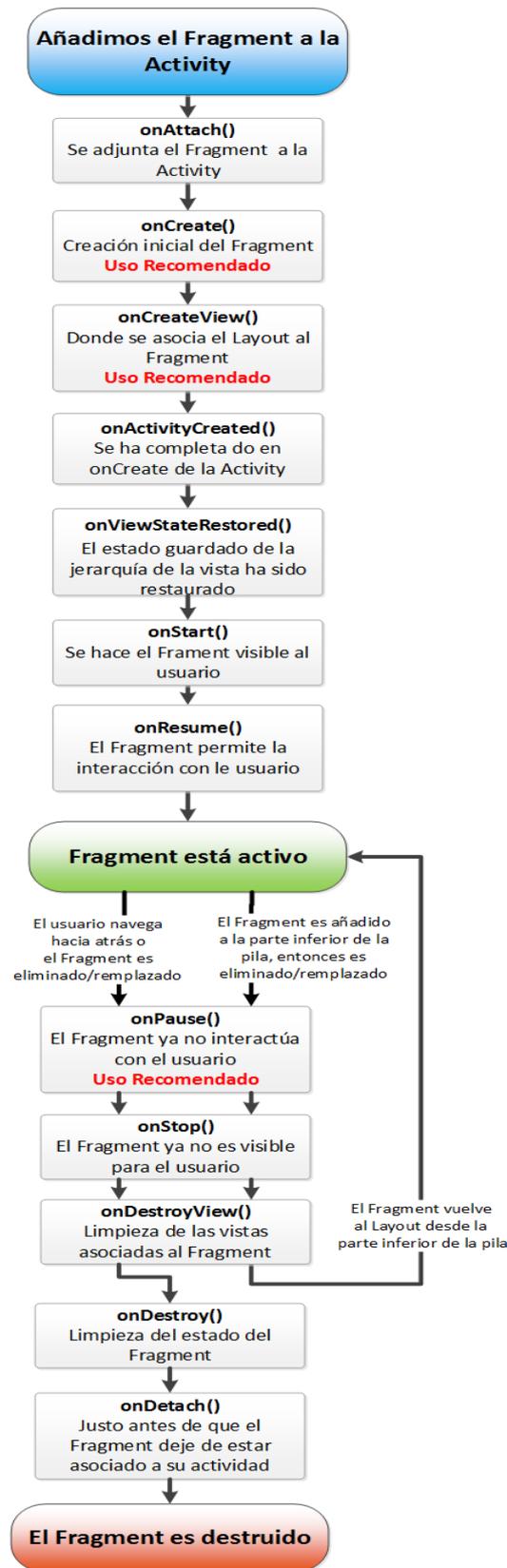
La definición de fragment es una porción de la interfaz de usuario que puede ser añadida o eliminada de una interfaz de forma independiente al resto de elementos o componentes de la actividad. Además, dicho fragment puede ser utilizado en distintas actividades. Por tanto, un fragment siempre tendrá su ciclo de vida asociado a una actividad y será utilizado para controlar fragmentos de la pantalla de la activity.

En el caso de la aplicación desarrollada, los fragments han sido utilizados para manejar las distintas pestañas en la ventana principal. Es decir, existe la actividad principal (MainActivity) que es mostrada con un layout y las distintas pestañas que se muestran se corresponden con los distintos fragments creados. Los cuatro fragmentos serían los siguientes:

1. Pestaña de Entrenamientos.
2. Pestaña de Jugadas.
3. Pestaña de Estadística.
4. Pestaña de Cronómetro.

El ciclo de vida de un fragment, sería el siguiente:

Figura 34.- Estados de un Fragment



Fuente: <http://developer.android.com/guide/components/fragments.html>

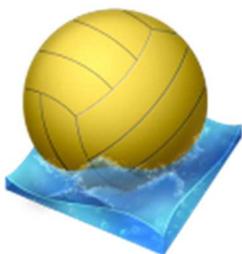
4.4 Inicio de la aplicación.

Una aplicación debe tener unas primeras imágenes muy identificativas para poder dar rápidamente una idea al usuario de la temática de dicha aplicación. Éstas dos primeras representaciones de la aplicación son el icono y una “Splash screen” o pantalla de presentación.

4.4.1 Icono.

Éste es el icono elegido para la aplicación, el cual es bastante original y representativo de la temática tratada.

Figura 35.- Icono app Waterpolo Improvement



El proceso realizado ha sido el siguiente: encontrar la imagen y editarla añadiendo el nombre de la aplicación que se puede hacer mediante unas instrucciones al principio del “AndroidManifest”.

4.4.2 Pantalla de inicio.

La pantalla de presentación muestra también una imagen clara. Con esto se consigue que el usuario se sumerja en la aplicación que acaba de ejecutar puesto que ocupa toda la pantalla y muestra el nombre de la aplicación durante unos instantes.

Figura 36.- Pantalla Inicio aplicación Como es un fondo de pantalla o “background”, la imagen que se ha seleccionado ha tenido que ser editada, adaptada con el programa “Draw 9-patch” mencionado anteriormente. De esta manera se han conseguido representar unas medidas proporcionales en la pantalla del móvil.



La llamada a la “Activity” (actividad) encargada de mostrar la pantalla de presentación o “Splash screen” se realiza en el “AndroidManifest” en primer lugar. En esta “Activity” se puede definir el tiempo que será mostrada la imagen de presentación aunque también se puede elegir que solamente se muestre durante el tiempo que tarde en cargarse la

aplicación. Al final de ella, se deberá dar paso a la actividad principal del programa.

4.5 Entrenamientos.

Esta es la parte menos compleja de la aplicación. Primero se ha realizado un layout para poder ver los entrenamientos en una misma pantalla deslizando el dedo en vertical. Después, se han incluido los textos en la carpeta de *res/values* y de ahí la actividad propia de los entrenamientos obtiene la información.

Por otro lado, se ha tenido que hacer una modificación en el tipo de codificación de los textos para poder representar correctamente todas las palabras que tuvieran acentos o signos.

Primeramente se ha establecido en el apartado de editor de textos la codificación "UTF-8*" y luego la misma para el workspace:

Figura 37.- Modificación de codificación 1

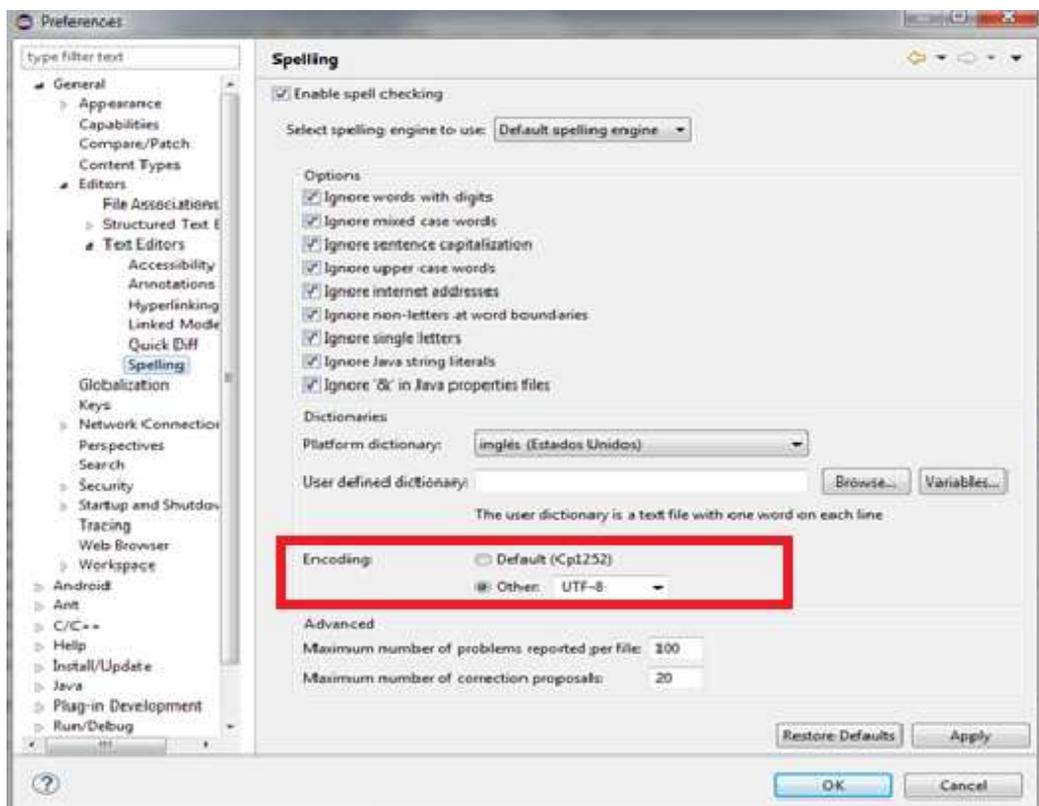
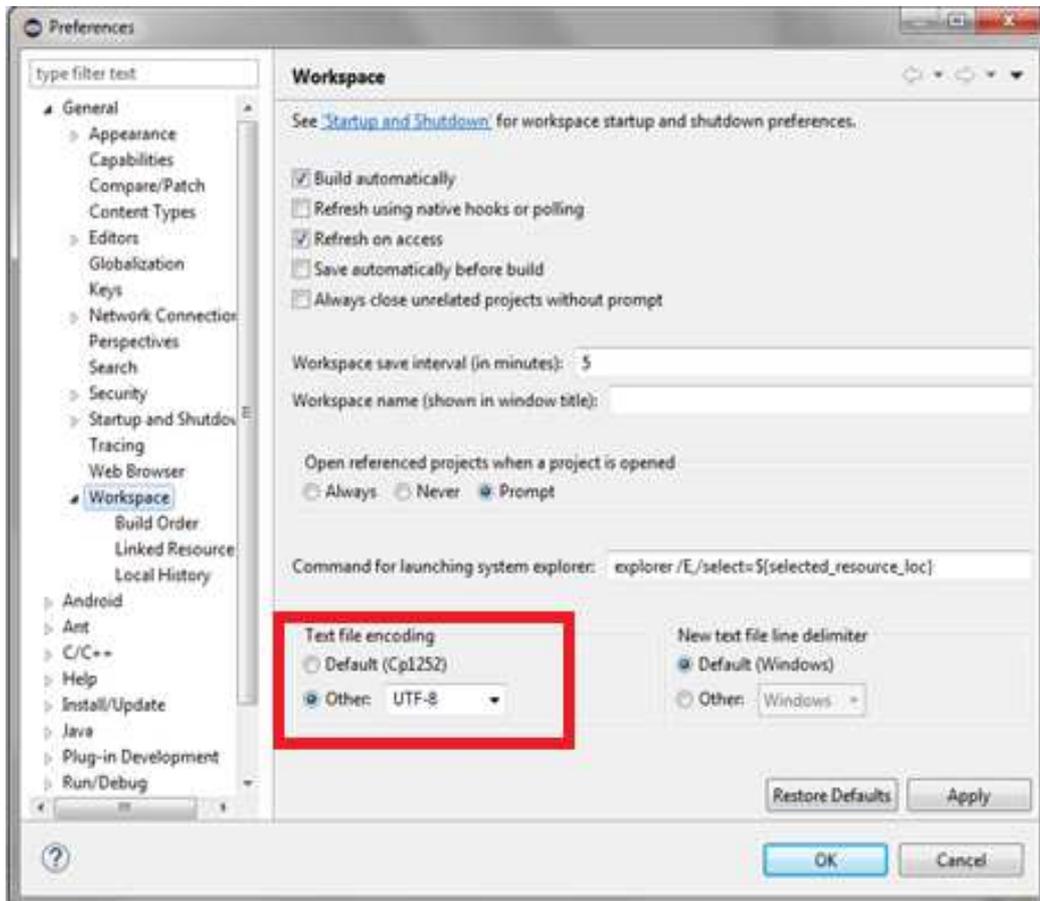


Figura 38.- Modificación de codificación 2



4.6 Jugadas.

Una de las mayores diferencias con las aplicaciones analizadas en el apartado “Estado del arte” es la creación de un libro de jugadas. En dicho libro, se muestran diez jugadas con sus respectivos textos explicativos. Las jugadas se encuentran referenciadas por su propio nombre y están organizadas en dos tipos: de ataque y de defensa.

4.6.1 Representación de las jugadas.

Las imágenes de las jugadas se almacenan en la carpeta de *res/drawable* con un determinado nombre.

*UTF-8 (8-bit Unicode Transformation Format) es un formato de codificación de caracteres Unicode e ISO 10646 utilizando símbolos de longitud variable.

El primer paso es definir el layout para poder mostrar en la pantalla la imagen con la representación de la mitad de la piscina y los jugadores, los textos explicativos y el botón que se utiliza para cambiar de fase de la jugada. Para ello se ha dividido la pantalla en cuatro partes:

- ❖ En la parte más alta se indica el grupo de jugadas en el que se encuentra (jugadas de ataque o de defensa).
- ❖ Seguidamente en la superficie más grande de todas las divisiones, se muestra la imagen de la jugada.
- ❖ Debajo de la imagen de la jugada aparece el texto explicativo.
- ❖ Por último, se diseña un botón, para cambiar de fase de la jugada, que va en la parte baja de la pantalla.

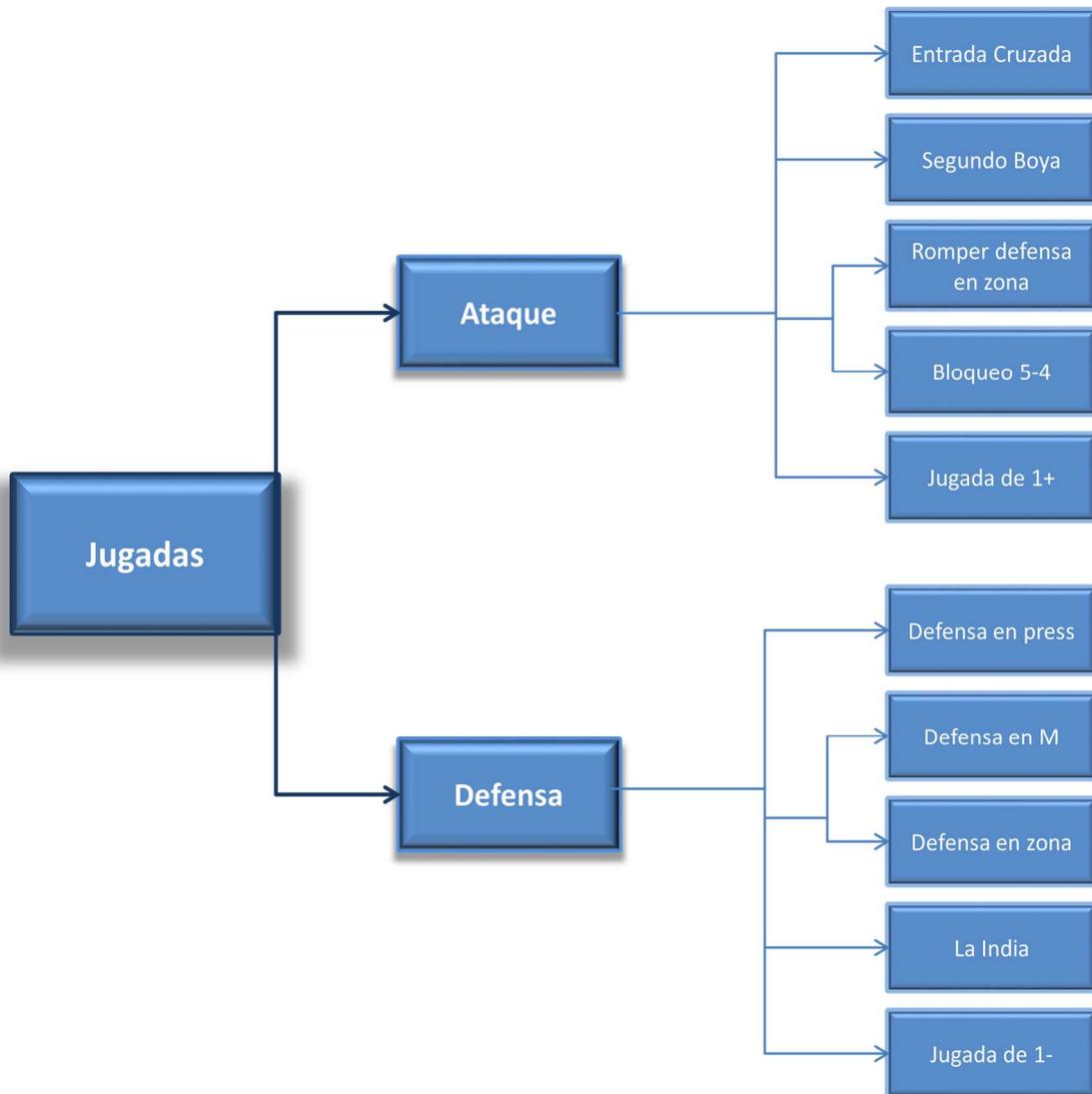


Posteriormente, la actividad de las jugadas recibe el id de la jugada que se quiere mostrar (lo recibe del fragment encargado de la pestaña de Jugadas). En base a ese id, en un “switch” se cargan las cuatro imágenes que se quieren mostrar y los textos asociados a cada una de ellas en variables distintas (las imágenes de las jugadas son almacenadas en la carpeta *res/drawable*). Por último, el botón definido en el layout se encarga de ir mostrando las imágenes y los textos correspondientes según un índice para seguir un orden y poder volver al principio de la explicación.

4.6.2 Estructura del menú de jugadas.

En la siguiente figura se muestra los niveles que tiene la pestaña de Jugadas, de qué manera se encuentra dividida y las jugadas que se ofrecen.

Figura 40.- Estructura de Jugadas



4.7 Gestión de Estadísticas.

Una de las funcionalidades más importantes de la aplicación desarrollada es la que permite realizar un registro de diferentes datos estadísticos, asociados a un jugador, que se pueden dar durante un partido. Para ello se ha diseñado una base de datos que guardará las diversas anotaciones y además permitirá realizar una consulta sobre la información guardada.

4.7.1 Introducción a la Base de Datos.

Una base de datos es una colección de información estructurada y bien organizada. Es decir, no solo tiene la función de almacenar gran cantidad de datos sino que además presenta una

estructura que permite consultar los registros que en ella hay de forma fácil. Además de consultar y añadir datos guardados en la base de datos, éstos pueden ser modificados y extraídos.

Como se comentó en la parte del Diseño, Android dispone de una biblioteca de SQLite. Esta es la principal razón para utilizar una base de datos de este tipo. Aunque SQLite también es un motor de base de datos muy potente y al ser de código abierto se puede encontrar mucha información acerca de cómo sacarle partido. Además ofrece varias ventajas como la de no precisar de un servidor y ocupar muy poco tamaño en el almacenamiento. (Las estadísticas que se exportan se guardan en un documento “txt”).

4.7.2 Organización de la Base de datos.

La base de datos SQLite, como se ha mencionado anteriormente, ocupa muy poco tamaño y además puede tener un número ilimitado de registros. Para organizar la base de datos es necesario utilizar un identificador, una clave única. Una vez establecido esto se podrán asociar información a dicho identificador que serán tratados como diferentes campos. Un ejemplo en el que una matrícula sería un pseudo-identificador sería el siguiente:

Tabla 4.- Ejemplo de Base de datos

ID	MATRÍCULA	VEHÍCULO	COLOR	COMBUSTIBLE
1	8523BJK	Monovolumen	Azul marino	Diésel
2	3251UWN	Turismo	Amarillo	Gasolina
3	6756XYQ	Berlina	Negro	Diésel

La matrícula podría ser la clave única, conceptualmente hablando, y a dicha clave se le puede asociar diferentes datos, campos de una tabla que pueden aplicar a más claves, ser generales. Para no tener problemas de duplicidades de identificador puro se hace una numeración para dar valores a éste.

4.7.3 Base de datos creada.

Se han definido dos tablas para recoger diferentes entidades. Una denominada Tabla Jugador y otra Tabla Equipo para cumplir el modelo de Entidad – Relación. Con ello conseguimos que cada tabla refleje una cosa del mundo real que estamos modelando:

- La Tabla Jugador guarda a los jugadores con sus atributos. Es decir, un único id que lo identifica, nombre, id del equipo, posición y campo activo o no activo (dato booleano).
- La Tabla Equipo recoge los equipos y solo guarda el nombre del equipo. Con esto lo que se consigue es que en la “Tabla Jugador” se almacena el id del equipo porque si no habría que repetir el nombre del equipo en cada jugador.

Las tablas creadas serían las siguientes:

1. La primera tabla recoge la información del jugador. Tabla Jugador.

Tabla 5.- Tabla del jugador

ID	NOMBRE	EQUIPO	POSICIÓN	ACTIVO
1	Iván	id equipo (1,Canoe)	1 o Extremo derecho	1
2	Luis	id equipo (2,Moscardó)	3 o cubre boya	1
3	Marta	id equipo (3,Colmenar)	2 o lateral derecho	0

2. Tabla Equipo.

Tabla 6.- Tabla Equipo

ID	EQUIPO
1	Canoe
2	Moscardó
3	Colmenar

3. La tercera tabla es la Tabla Estadísticas. En esta tabla se recogen los datos estadísticos correspondientes a cada jugador. Se almacenan las estadísticas actuales, es decir si el jugador ha cambiado de equipo o de posición se guardarán las correspondientes a la última situación.

Tabla 7.- Tabla Estadísticas

ID	JUGADOR	GOLES	TIROS	ASISTENCIAS	EXPULSIONES	EXPULS. PROVOCADAS	ROBOS	PÉRDIDAS	PARADAS	SPRINT GANADOS	SPRINT REALIZADOS
1	id jugador(1, Luis)	2	5	3	2	1	3	1	0	1	2
2	id jugador(2, Marta)	6	10	1	1	0	4	0	0	2	3
3	id jugador(3, Carmen)	1	2	4	0	1	1	2	0	0	1
4	id jugador(4, Iván)	0	0	0	0	0	0	2	1	0	0

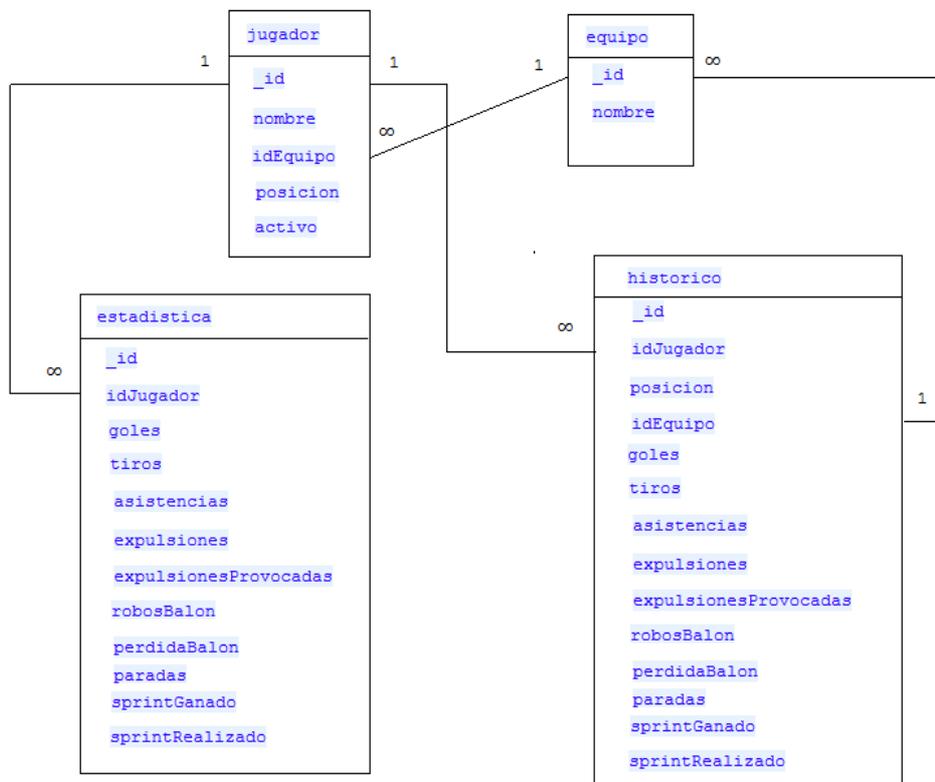
Además de estos campos, que se guardan en la base de datos, se pueden consultar el porcentaje de acierto y el porcentaje de los sprint ganados que son calculados directamente en la activity. El campo que recoge el número de paradas sólo está habilitado para un jugador que esté en la posición de portero.

4. La última tabla es la Tabla Histórico. Aquí se recopilan los mismos campos que en la Tabla Estadísticas más el id del Equipo y la posición. Con el objetivo de no perder los datos, cuando un jugador cambia de posición o de equipo, éstos son almacenados en dicha tabla. Es decir, se mantiene registrado un histórico del jugador. Por otro lado, cuando se consultan las estadísticas, aparte de ver todos los registros anotados para cada situación del jugador, también se puede ver la suma de los valores recopilados. Un ejemplo sería el siguiente:

Luis marca dos goles y comete una falta jugando en la posición de lateral izquierdo. Vuelve a jugar un partido, pero esta vez es en la posición de lateral derecho, anota tres goles y comete dos faltas. Por tanto, en la suma de los valores recopilados en la Tabla Histórico se observarán cinco goles y tres faltas cometidas. Esta suma la realiza el activity no se guarda el valor de la suma en la base de datos.

4.7.4 Modelo Entidad - Relación.

El modelo Entidad – Relación creado sería el siguiente:



4.7.5 Funciones.

La primera función utilizada es la de crear una base de datos y después poder insertar los valores deseados y consultarlos. Para ello el primer paso ha sido importar las librerías de

SQLite que ofrece Android. Estas librerías son *SQLiteDatabase*, *SQLiteOpenHelper* y *SQLException*. A partir de éstas podemos utilizar funciones para gestionar la base de datos. (*SQLException* funciona directamente también como función y tiene la finalidad de proporcionar información sobre errores en el acceso a la base de datos).

Tabla 8.- Funciones para la gestión de la BD

FUNCIÓN	DESCRIPCIÓN
onCreate()	Crea las tablas en la base de datos
onUpgrade()	Realiza una actualización de la base de datos, eliminando la versión antigua y creando una nueva vacía.
abrir()	Abre la base de datos y utiliza <i>SQLException</i> para comprobar si hay errores en la acción. Método de <i>SQLiteOpenHelper</i> .
cerrar()	Cierra la base de datos. Método de <i>SQLiteOpenHelper</i> .
insertarJugador()	Inserta un jugador en la Tabla Jugador. Utiliza la función insert de la base de datos.
obtenerTodosLosJugadores()	A través de una query (consulta) extrae todos los jugadores de la Tabla Jugador (con todos los campos de esta tabla).
obtenerUnJugador()	Utiliza un cursor con el que recorrer la Tabla Jugador y extrae el jugador buscado. Se hace la búsqueda por el nombre del jugador.
obtenerUnJugador()	El mecanismo es el mismo que en la anterior pero se busca el jugador por su id. Esta función tiene el mismo nombre que la anterior. Sin embargo, esto no provoca ningún problema porque el compilador sabe a qué función llamar en base al tipo de parámetros que se le pasa, esto se denomina sobrecarga de métodos en java. (Si es un entero se refiere a que busca un id y si es una cadena se llamaría a la función que busca por el nombre)
modificarJugador()	Modifica en la Tabla Jugador un jugador específico. Utiliza la función update para hacer la actualización en la base de datos.
insertarEquipo()	Agrega un nuevo equipo en la Tabla Equipo. Utiliza la función insert de la base de datos.

obtenerTodosLosEquipos()	A través de una query (consulta) extrae todos los equipos de la Tabla Equipo (con todos los campos de esta tabla).
obtenerUnEquipoByNombre()	Utiliza un cursor con el que recorrer la Tabla Equipo y extrae el equipo buscado. Se hace la búsqueda por el nombre del equipo
obtenerUnEquipoById()	Utiliza un cursor con el que recorrer la Tabla Equipo y extrae el equipo buscado. Se hace la búsqueda por el id del equipo del equipo
obtenerEstadisticaByIdJugador()	Obtiene las estadísticas de un jugador buscando por su id. Realiza la búsqueda con un cursor en la Tabla Estadística.
insertarEstadistica()	Registra las estadísticas en la Tabla Estadística. Utiliza la función <i>insert</i> de la base de datos.
modificarEstadistica()	Modifica las estadísticas recogidas en la Tabla Estadística. Utiliza la función <i>update</i> para hacer la actualización en la base de datos.
insertarHistorico()	Añade las estadísticas en la Tabla Histórico. Utiliza la función <i>insert</i> de la base de datos.
obtenerHistoricoByIdJugador()	Con la ayuda de un cursor recorre la tabla Histórico y extrae todo el histórico estadístico de ese id del jugador que se busca.

Una tarea separada ha sido la de crear el mapeo de los datos en los campos de la base de datos. Esto se ha hecho con “getters and setters”. Es decir, obtener el dato y anotarlo. Estos son necesarios en todas las funciones que quieran insertar los datos nuevos en las tablas.

Las distintas funciones creadas para el apartado de Estadística de la aplicación usarán las que se mencionan en la tabla anterior. Además de la función de bluetooth, que como se explica en su apartado correspondiente, también hace uso de ellas.

4.8 Cronómetro.

El cronómetro, como se comentó en el apartado del diseño es una herramienta muy útil para complementar a la función de los entrenamientos o para medir el tiempo de posesión y los cuartos en los partidos.

4.8.1 Layout.

El primer paso ha sido crear un layout que estuviera en consonancia con el resto de la aplicación y que sirviera de base para la representación de los números del cronómetro. Para ello se ha creado una base en la que se hacen dos subdivisiones: una capa que alberga los dos botones (“Iniciar/Continuar/Pausar” y “Stop”) y otra capa predefinida de Android que presentará los dígitos del cronómetro.

4.8.2 Funcionamiento.

El primer botón, situado a la izquierda de la pantalla, inicia el conteo desde cero si es la primera vez que se pulsa y lo detiene. Además, este mismo botón ofrece la posibilidad de continuar por donde se había detenido.

Para que esto ocurra se ha creado una variable global de estado en la que se registra en qué fase está en cada momento el cronómetro (activo, inactivo o pausado) y a partir del valor de dicha variable se habilitan las opciones del botón. Además, se utiliza la función *setBase(SystemClock.elapsedRealtime())* para que el cronómetro esté basado sobre la hora del dispositivo, es decir, con esto conseguimos que el cronómetro no se inicie según se ejecute la aplicación y siempre este referenciado a la hora. (El ejemplo contrario sería un contador de tiempo de llamada).

El segundo botón, situado a la derecha de la pantalla, se encarga de parar el cronómetro, congelarlo para poder observar o registrar el tiempo que marca y de dejarlo listo para volver a iniciarlo a cero si se pulsa el botón de “Iniciar”.

4.9 Bluetooth.

La última funcionalidad de la aplicación desarrollada sería la de poder transmitir la información registrada en la base de datos haciendo uso de una conexión de bluetooth. Para conseguirlo, se ha realizado una división en dos del trabajo.

La primera parte del trabajo se correspondería con conseguir la transmisión de las estadísticas recogidas. La otra sección sería la de recibir, importar la información. Para completar dicha sección, habría que desarrollar un apartado que se encargase de leer el archivo recibido en el dispositivo y actualizar la base de datos que utiliza la aplicación.

Ambos segmentos se han implementado en el “MainActivity”, es decir, van a estar en la actividad principal o núcleo de la herramienta con lo que se permite que las dos funciones (transmitir y recibir/importar) se encuentren siempre accesibles. (El layout definido establece de forma fija el acceso a las opciones del bluetooth en la zona superior derecha de la pantalla).

4.9.1 Transmitir.

Para poder enviar los datos estadísticos que se hayan guardado en un dispositivo hay que seguir una serie de pasos.

1. Cargar los datos a enviar y guardarlos en un fichero (txt) en la sdcard. (Función crearFileTransferir).
2. Almacenar el fichero en la memoria externa del dispositivo.
3. Crear un “intent” (mecanismo para invocar componentes) de envío.
4. Se define que el archivo a enviar será un texto plano.
5. Marcar en el intent el componente bluetooth para el envío.
6. Asociar el fichero con la información al “intent”.
7. Hacer el envío (comenzar la actividad definida por el “intent”). En este apartado el dispositivo ofrece la opción de a qué otro, de los sincronizados por bluetooth, se enviará el archivo.

Tabla 9.- Función base de transmisión bluetooth

Función	Sub-funciones	Descripción
crearFileTransferir()	Crear una lista	Se crea una lista que será la que recopile los diferentes datos
	Abrir base de datos	Se abre una base de datos tipo
	Crear un cursor	Se crea un cursor que se utilizará para recorrer la base de datos e ir obteniendo los datos
	Recorrer la base de datos	Se recorre la base de datos desde el primero hasta el último elemento (en un bucle) extrayendo todos los parámetros estadísticos y almacenándolos en una variable específica para cada uno de ellos.
	Generar el fichero	Se crea el fichero que almacenará los datos estadísticos
	Completar el fichero	Se traspasa la información recopilada en la lista al fichero

4.9.2 Recibir.

El primer paso, para poder recibir en la base de datos la información proveniente de otro dispositivo, es aceptar el archivo que está siendo enviado. Una vez que se ha aceptado el archivo se almacena en la tarjeta externa y es necesario extraer los datos y actualizar la base de datos con ellos. Las fases que se deben realizar son las siguientes:

1. Definir el camino hasta dónde se encuentra el archivo.
2. Procesar los datos del fichero por cada línea.
 - 2.1 Dentro de cada línea deben estar todos los parámetros estadísticos.
 - 2.2 Cada parámetro de la línea es almacenado en una variable.
 - 2.3 El primer campo es el jugador. Con ese identificador y mediante un cursor, se recorre la base de datos. Entonces existen tres posibilidades.
 - a) El jugador no existe en la base de datos.

Se introduce el jugador en la base de datos y se comprueba si también hay que introducir el campo equipo en el caso de que no se encuentre ya registrado (al que pertenece el jugador). Por último se agregan los datos referentes a dicho jugador.
 - b) El jugador ya existe en la base de datos.

Se verifica si está en el mismo equipo (campo segundo).

 - En caso afirmativo hay que comprobar también si juega en la misma posición (campo tercero). Si también coincide se actualizan las estadísticas anteriores, de lo contrario se añaden al histórico del jugador.
 - En caso negativo, se consulta si ese equipo se encuentra en la base de datos. Si es así se hace la actualización y si no existe se agrega y se guardan los nuevos datos.

Tabla 10.- Funciones para importar la información recibida

Función	Descripción
obtenerUnJugador()	Extrae el campo primero de la línea del fichero y lo busca con un cursor en la base de datos. En caso de encontrarlo obtiene su posición y el equipo para hacer las futuras comparaciones y toma de decisiones.
getIdEquipo()	Obtiene el identificador del equipo. Éste es el campo segundo de la línea.
getPosition()	Obtiene el valor de la posición. Éste es el campo tercero de la línea.
insertarJugador()	Agrega un jugador a la base de datos con los parámetros identificativos (nombre, posición y equipo).
insertarEstadistica()	Introduce en la base de datos todos los datos estadísticos correspondientes a un jugador.
insertarHistorico()	Inserta los nuevos datos en el histórico del jugador.

4.9.3 Modificación.

Uno de los problemas que hubo que solucionar consistía en lo siguiente: si en un dispositivo ya se había recibido un primer fichero con la información, el siguiente que se recibía lo almacenaba con un nombre distinto al que buscaba la aplicación. Esto provocaba que el final de la ruta marcada en la programación fuera diferente y por tanto, la aplicación volvía a importar unos datos que ya se habían cargado anteriormente. La solución fue la de borrar el fichero una vez que se hubieran importado los datos en la base de datos de la aplicación.

4.10 Idioma.

Como se mencionó anteriormente, la aplicación la pueden utilizar personas de habla inglesa o española. Es decir, con la introducción de un idioma más se busca conseguir un abanico de usuarios internacionales.

En el directorio del proyecto existe una carpeta que almacena los "Strings", palabras y textos que se muestran en la aplicación. Para cada idioma hay una carpeta de este tipo. El sistema Android se dirigirá a una determinada carpeta para obtener los textos dependiendo del idioma del dispositivo electrónico. Por tanto, para poder ofrecer distintos idiomas se deben realizar las traducciones pertinentes y situar los textos a mostrar en las carpetas que correspondan.

Una parte más específica y en la que se han tenido que hacer cambios fue la forma de presentación de los entrenamientos.

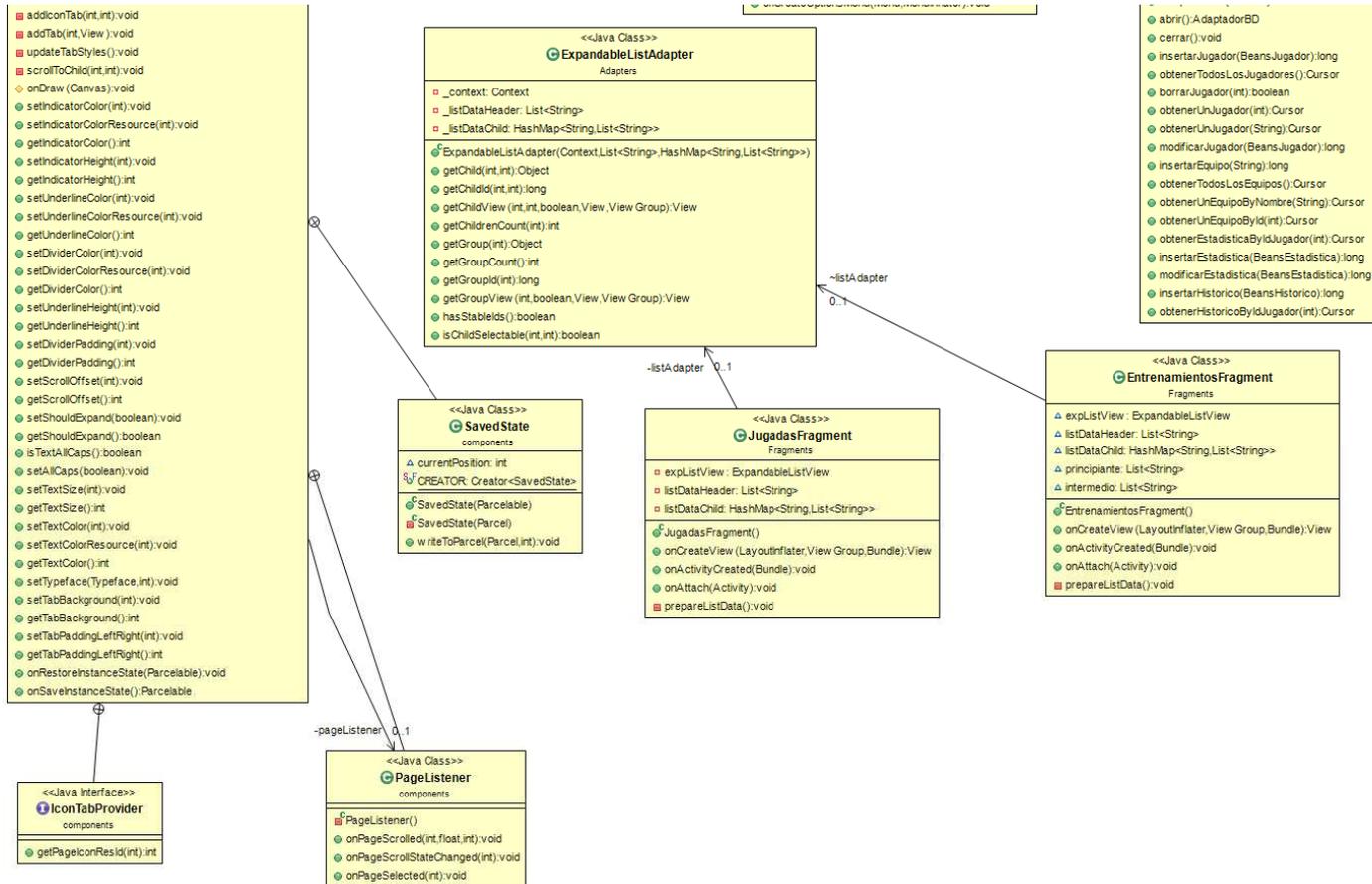
Primero, los textos de los entrenamientos que se mostraban tenían un formato genérico que no mostraba de forma adecuada algunos caracteres como son las palabras con acentos. La mejora que se hizo fue cambiar el tipo a "UTF-8".

Segundo, los entrenamientos se mostraban directamente a partir de una activity pero para poder mostrarlos también en inglés ha habido que ubicarlos en la carpetas mencionadas anteriormente, en español en la carpeta "values-es" y en inglés en la carpeta "values-en".

4.11 Diagrama de clases.

Si se instala el plugin de UML2 para Eclipse, se puede extraer un diagrama de clases que da una visión general de todas las clases, métodos y variables de la aplicación.

Figura 42.- Diagrama de clases, 2



5 Conclusiones

El final del proyecto ha sido cerrado cumpliéndose todas las propuestas que se habían realizado y además se ha conseguido añadir funcionalidades extra.

Como se comentó en el apartado de objetivos, situado en la introducción, el principal fin de la aplicación era dar soporte para poder registrar diferentes datos estadísticos durante un partido. También se propuso que la aplicación sirviera como fuente de información para nuevos practicantes y entrenadores del deporte tratado. Para ello hay recopilados una serie de entrenamientos y de jugadas.

Funcionalidades que ofrece la aplicación:

- ❖ Registro de jugadores añadiendo información sobre el equipo y la posición en la que juega.
- ❖ Registro estadístico. Se anotan diferentes datos estadísticos que suceden durante un partido y se agregan a un jugador, haciendo diferencia de si suceden en un equipo o en una posición nueva. Es decir, un jugador puede tener datos correspondientes a distintas situaciones (diferentes equipos y/o posiciones) y estar todos ellos guardados de forma independiente.
- ❖ Consulta de las estadísticas almacenadas.
- ❖ Transmisión y recepción de los datos estadísticos a través de bluetooth.
- ❖ Recopilación de entrenamientos divididos por niveles de dificultad e intensidad.
- ❖ Función de cronómetro. Muy útil para complementar los entrenamientos y tomas de tiempo durante los partidos.
- ❖ Representación de diez jugadas con textos explicativos.
- ❖ Se ofrece la aplicación en dos idiomas: castellano e inglés.

Por otro lado, todo el trabajo que fuera desempeñado para crear la aplicación tendría un valor altamente didáctico.

Habilidades obtenidas o mejoradas en el transcurso del desarrollo del proyecto:

- ❖ Búsqueda de información y de argumentos para tomar una decisión como ha sido la de utilizar la plataforma Android.
- ❖ Diseñar una aplicación móvil basándose en cubrir posibles demandas de los usuarios destinatarios e intentando crear una interfaz sencilla y directa.
- ❖ Capacidad de programar en un lenguaje que antes no se había hecho como es java y xml.
- ❖ Conocimiento de la arquitectura Android.
- ❖ Comprensión y control de base de datos SQL.
- ❖ Aprender a diferenciar las fases de un proyecto, a la vez que conseguir crear un buen reporte del trabajo realizado.

6 Trabajo futuro

En el transcurso de la creación de la herramienta desarrollada se han ido tomando diferentes caminos y además, el haber ido probando la aplicación durante su realización ha provocado que se manifestaran posibles mejoras para el proyecto.

Habría diferentes líneas de mejora:

❖ Aspectos técnicos.

1. Opción de utilizar un servidor externo para almacenar los datos estadísticos.
2. Actualización de los entrenamientos. A partir de un servidor externo se podrían consultar nuevos entrenamientos y de esta forma tener un amplio abanico. Habría que llevar un registro de fechas de los entrenamientos.
3. Ampliación del libro de jugadas de la misma forma que la comentada para los entrenamientos.
4. Adaptación de la aplicación a otras plataformas como IOS de Apple.
5. Añadir una funcionalidad de pizarra en la que se puedan realizar dibujos de jugadas o explicaciones para distintos ejercicios en los entrenamientos.
6. Completar la función del cronómetro con la opción de hacer una cuenta atrás.
7. Ampliar los idiomas que se ofrecen.
8. Agregar a la base de datos un campo donde el jugador pueda insertar una foto suya a modo de identificación.
9. Aumentar las funcionalidades de la aplicación como la de llevar un registro estadístico a nivel de equipo. Por ejemplo: resultados de los encuentros, partidos perdidos como local o como visitante, partidos ganados como local o como visitante, puntos en la liga, etc.

❖ Aspectos de la interfaz de usuario.

1. Mejorar el diseño gráfico. Por ejemplo, representar las jugadas en formato vídeo con varias secuencias de imágenes.
2. Añadir sonidos a la aplicación. Este sería el caso de cuando se pulsan los botones o de si se diseñase una cuenta atrás que avisase de los últimos segundos.

❖ Aspectos económicos y comerciales.

1. Darse a conocer en las redes sociales y medios deportivos como periódicos o foros.
2. Añadir publicidad para obtener más ingresos.
3. Una vez que la aplicación fuera más famosa se podría ofrecer una versión de pago más completa.

7 Manual de Usuario

El primer paso es descargar el archivo .apk para poder instalar la aplicación. Una vez que se haya instalado ejecutamos la aplicación seleccionando el icono que la identifica.

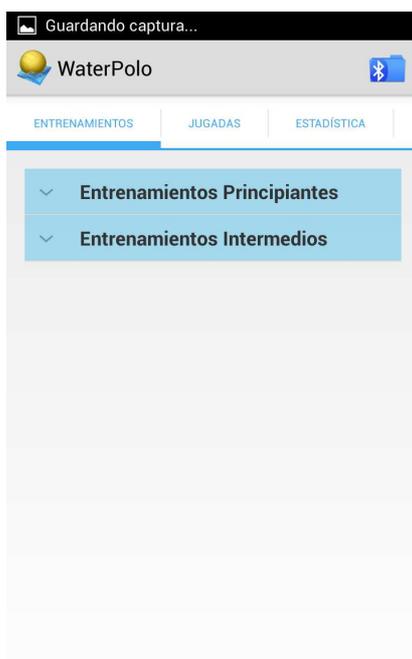
La imagen del Splash Screen será lo primero que aparezca y ocupará toda la pantalla.

Figura 43.- Imagen de carga de aplicación



Después de unos segundos aparecerá la primera pestaña abierta que es la de Entrenamientos. A través de esta pestaña podemos ir a las demás o bien pulsando sobre la etiqueta de cada una de ellas o bien desplazando el dedo por la pantalla en línea horizontal.

Figura 44.-Menú Inicial



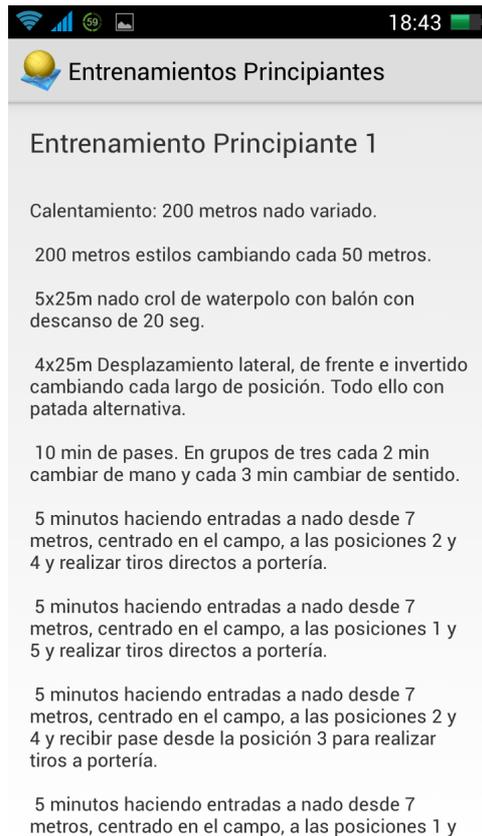
ENTRENAMIENTOS

Como se ha comentado justo antes la primera pestaña es la de entrenamientos y en la imagen anterior podemos ver que hay dos niveles de entrenamientos: uno para principiantes y otro para Intermedios. Los botones correspondientes a dichos niveles son desplegables que albergan los enlaces a los entrenamientos que hay almacenados.



Solo es necesario pulsar sobre el botón que identifica un entrenamiento para que se abra el texto con el detalle.

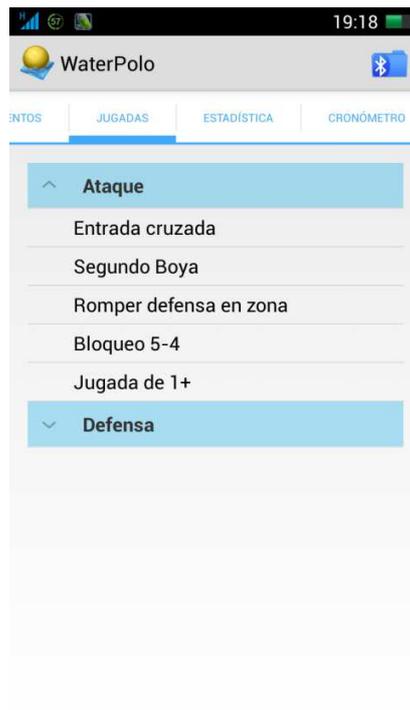
Figura 46.- Ejemplo de Entrenamiento



JUGADAS

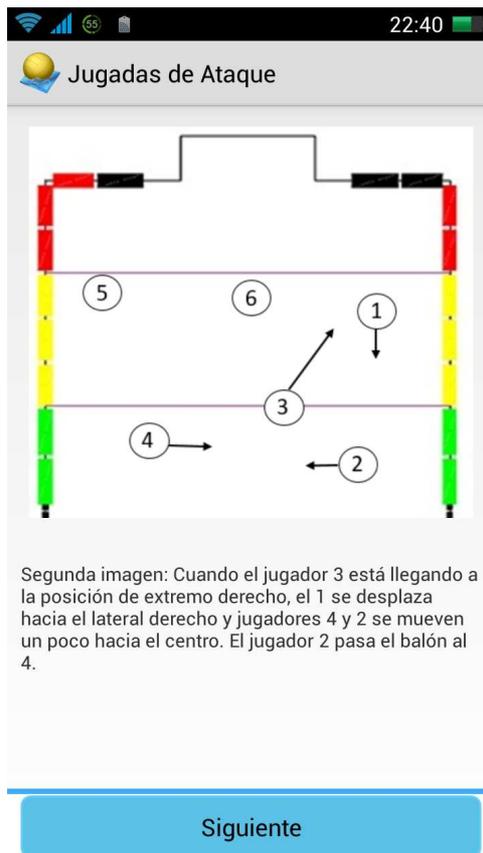
La segunda pestaña contiene la funcionalidad de las Jugadas. Al igual que los entrenamientos, las jugadas están recogidas en dos bloques que si se pulsa sobre sus botones identificativos abrirá un desplegable. Estos botones hacen referencia a Ataque y a Defensa. Dentro cada bloque tenemos cinco jugadas diferentes identificadas por su nombre. Sólo será necesario pulsar sobre la que queramos consultar.

Figura 47.- Menú Jugadas



Una vez se haya selecciona una jugada se abrirá una nueva pantalla en la que habrá que pulsar el botón de “Inicio” para que comience la representación de la jugada. A partir de la primera jugada este botón estará identificado como “Siguiete” para indicar la siguiente fase de la jugada. En cada fase se puede ver el texto explicativo de la jugada.

Figura 48.- Jugada ejemplo

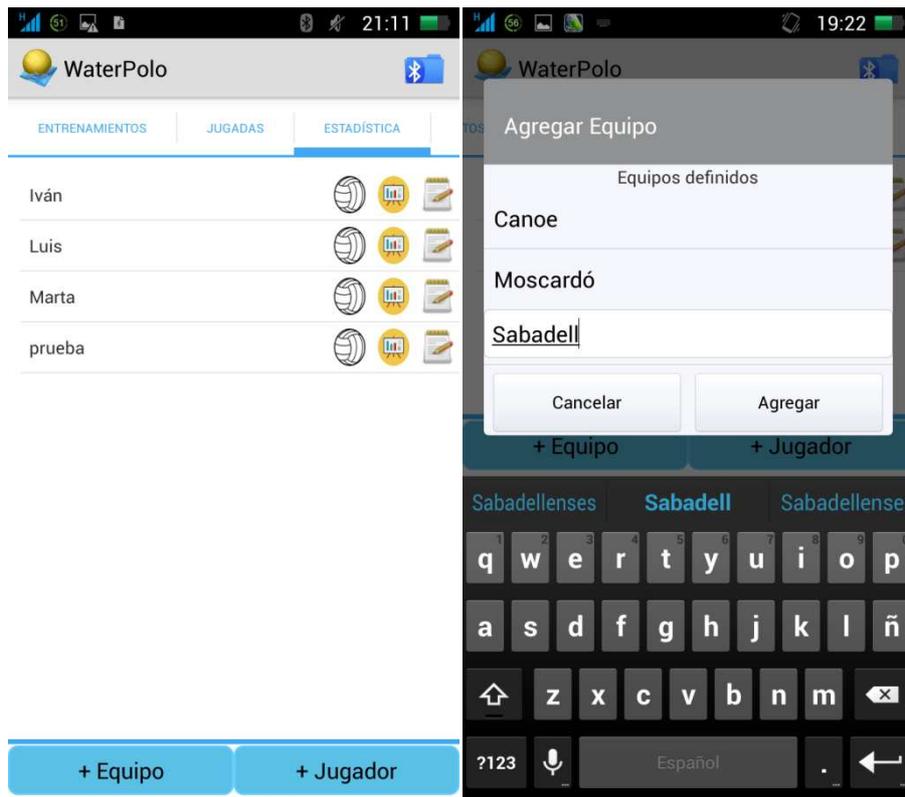


ESTADÍSTICA

La mejor forma de explicar cómo funciona esta funcionalidad es poner un ejemplo.

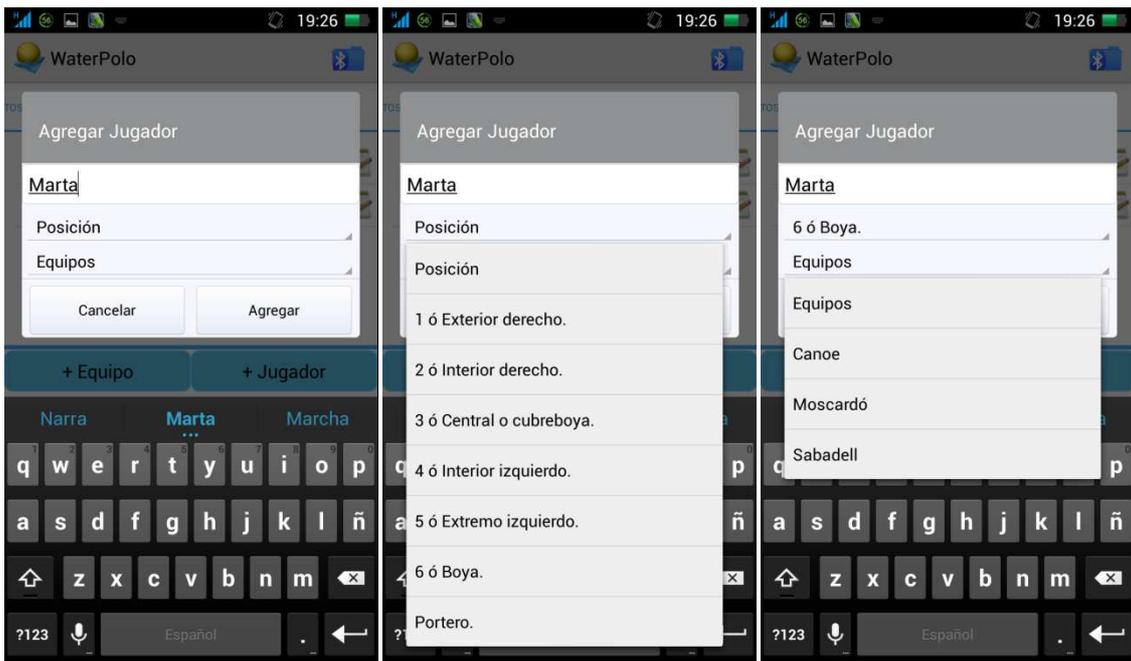
Lo primero será crear un equipo, sino hay alguno ya registrado, para que se puedan agregar jugadores a él. Creamos el equipo llamado Sabadell. Para ello solo hay que pulsar el botón de añadir equipo e introducir el nombre del equipo.

Figura 49.- Agregar un Equipo



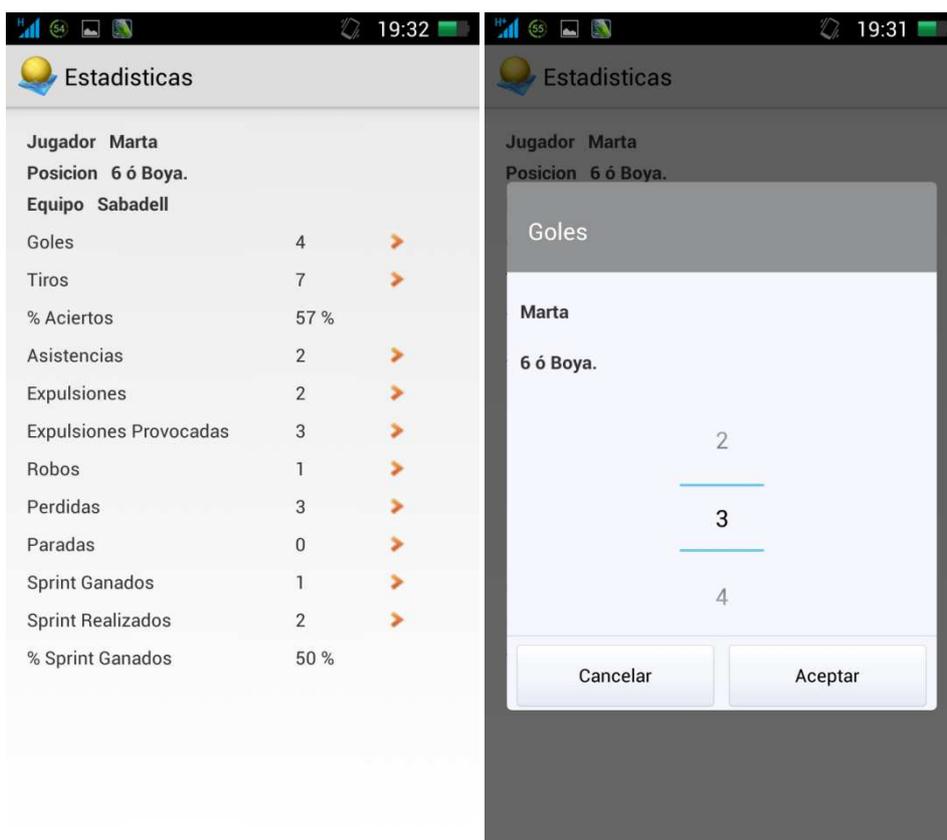
Después ingresamos el nombre del jugador. Pulsar el botón de añadir jugador e introducimos el nombre, la posición (sale un desplegable para elegir la posición) y el equipo (sale un desplegable para elegir entre los equipos que están en la base de datos). Introducimos el nombre de Marta, que juega en la posición 6 o boyas.

Figura 1 50.- Agregar un Jugador



El siguiente paso es el de introducir las estadísticas. Para ello se debe pulsar el icono de las barras estadísticas. En la pantalla que se abre se pueden ver todos los campos que se pueden registrar y en la parte alta el nombre del jugador, su posición y el equipo en el que juega. Para anotar los valores solamente hay que pulsar sobre el icono de la flecha y se abrirá un contador en el que moviendo el dedo en línea vertical se puede seleccionar el valor buscado.

Figura 51.- Anotar datos estadísticos



Una vez introducidas las estadísticas éstas pueden ser consultadas si pulsamos sobre el icono del balón.

Figura 52.- Consulta de Estadísticas



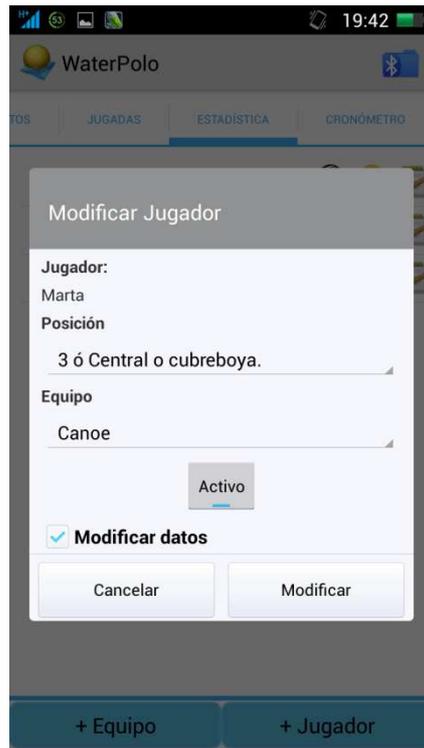
The screenshot shows a mobile application interface for viewing player statistics. At the top, there is a status bar with signal strength, 4G, and battery icons, and a time of 19:33. Below the status bar is a header with a globe icon and the text 'Ver Estadística'. The main content area displays the following information:

Jugador	Marta
Posicion	6 ó Boya.
Equipo	Sabadell
Goles	4
Tiros	7
% Aciertos	57 %
Asistencias	2
Expulsiones	2
Expulsiones Provocadas	3
Robos	1
Perdidas	3
Paradas	0
Sprint Ganados	1
Sprint Realizados	2
% Sprint Ganados	50 %

At the bottom of the screen, there are two blue buttons: 'Anterior' and 'Próximo'.

En el caso de que Marta cambie de equipo o de posición podemos realizar esa modificación en la aplicación y seguir guardando datos estadísticos de esa jugadora. Para reflejar dicha modificación en nuestra herramienta deberemos pulsar el icono de la libreta con el lápiz. En la pantalla que nos sale de Modificar Jugador seleccionamos Modificar datos y cambiamos la posición y el equipo. Finalmente pulsamos modificar.

Figura 53.- Modificación de Jugador



Ahora completaremos los datos estadísticos de la misma forma que antes para la misma jugadora pero en distinta posición y en distinto equipo.

Figura 54.- Anotación de estadísticas

The screenshot shows the 'Estadísticas' screen for player Marta. The player's name 'Marta' is at the top. Below it, the position '3 ó Central o cubreboya.' and team 'Canoe' are listed. A table follows with various statistics and their values, each with an orange arrow pointing right.

Jugador Marta		
Posicion 3 ó Central o cubreboya.		
Equipo Canoe		
Goles	1	>
Tiros	3	>
% Aciertos	33 %	
Asistencias	8	>
Expulsiones	1	>
Expulsiones Provocadas	1	>
Robos	6	>
Perdidas	2	>
Paradas	0	>
Sprint Ganados	2	>
Sprint Realizados	3	>
% Sprint Ganados	66 %	

Finalmente si volvemos a consultar los datos de la jugadora veríamos que están recogidos todos los que se han ido anotando. Además se muestra la suma de todos los datos estadísticos, es decir, de las diferentes situaciones de la jugadora.

Figura 55.- Consulta de Estadísticas Ampliadas

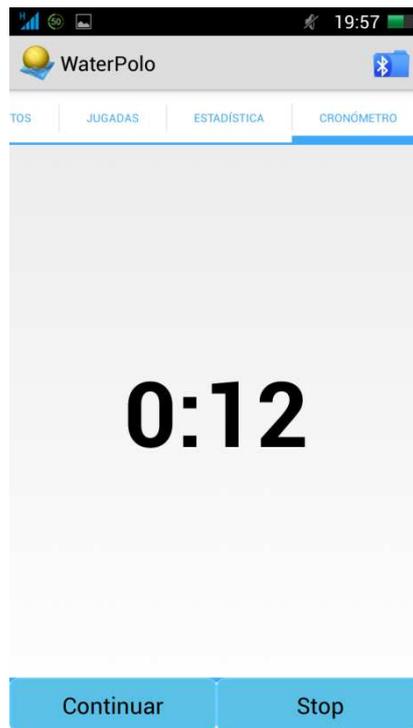
Ver Estadística	
Jugador	Marta
Posicion	3 ó Central o cubreboya.
Equipo	Canoe
Goles	1
Tiros	3
% Aciertos	33 %
Asistencias	8
Expulsiones	1
Expulsiones Provocadas	1
Robos	6
Perdidas	2
Paradas	0
Sprint Ganados	2
Sprint Realizados	3
% Sprint Ganados	66 %

Ver Estadística	
Jugador	Marta
Posicion	Todas las posiciones
Equipo	Todos los equipos
Goles	5
Tiros	10
% Aciertos	50 %
Asistencias	15
Expulsiones	3
Expulsiones Provocadas	3
Robos	9
Perdidas	3
Paradas	3
Sprint Ganados	2
Sprint Realizados	4
% Sprint Ganados	50 %

CRONÓMETRO

El funcionamiento de esta opción es bastante sencillo. Simplemente hay que pulsar el botón de la izquierda para iniciar, pausar y continuar la cuenta del tiempo y el botón de la derecha, para detenerlo y que la siguiente vez que empiece a contar sea desde cero.

Figura 56.- Cronómetro



BLUETOOTH

Por último, existe la posibilidad de enviar y de recibir los datos estadísticos almacenados en la base de datos. Para ello primero deben estar los dos dispositivos con el bluetooth conectado y vinculados.

El procedimiento para enviar o recibir sería pulsar sobre el icono de bluetooth en la parte superior derecha. Una vez presionado aparecen dos opciones: Exportar e Importar.

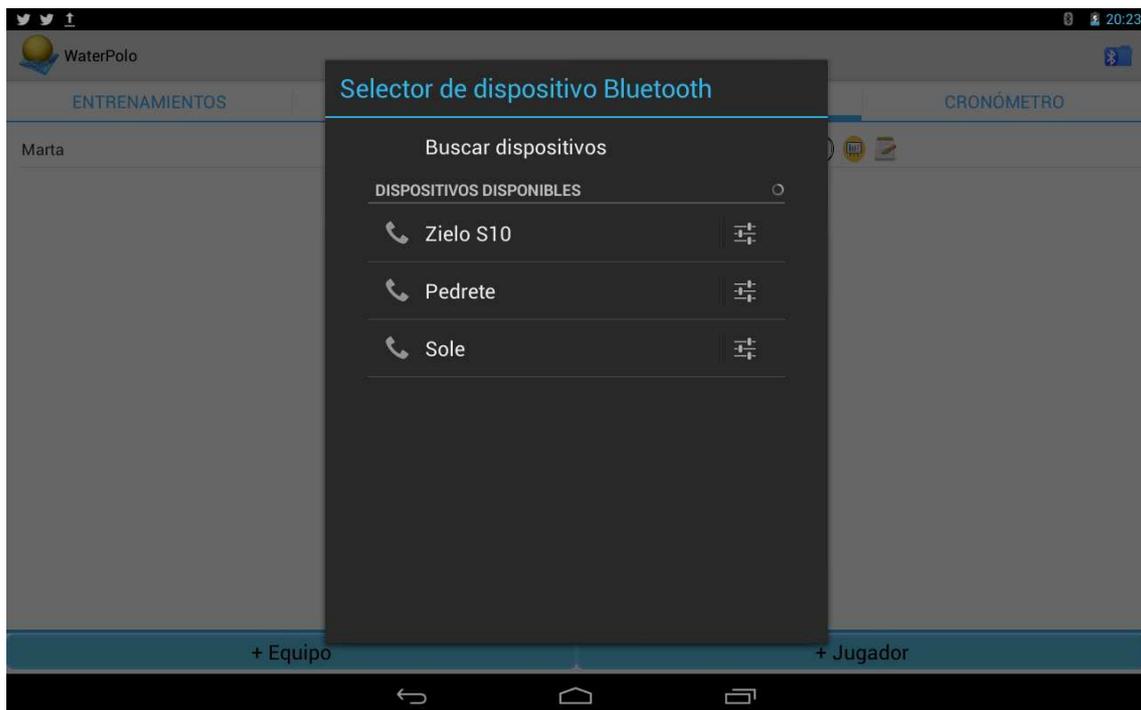
Pulsaremos Exportar en el caso de querer enviar los datos e Importar en el caso de querer cargar en la aplicación y en la base de datos la información recibida.

Una vez que se selecciona Exportar aparece un mensaje en el que informa que los datos han sido correctamente grabados. Es decir, el archivo para enviar está preparado. El siguiente paso es seleccionar el dispositivo destinatario.

Figura 57.- Funciones de Bluetooth



Figura 58.- Selección del dispositivo receptor (Bluetooth)



La importación se hace de forma automática. Solamente es necesario aceptar el archivo que se recibe por bluetooth.

Referencias

- [1] Entrenadores de waterpolo.
- [2] Página Oficial de la Real Federación Española de Natación.
- [3] Marko Gargenta. "Learning Android".
- [4] <http://www.startcapps.com/blog/>
- [5] <https://www.wikipedia.org/>
- [6] <http://blog.solbyte.com/>
- [7] <http://www.lancetalent.com/blog/tipos-de-aplicaciones-moviles-ventajas-inconvenientes/>
- [8] <http://www.xatakandroid.com/>
- [9] <http://es.slideshare.net/javacasm/21-android-cep-jaen-2014-estructura-de-aplicacin>
- [10] <http://www.cubrid.org/>
- [11] <http://help.eclipse.org/luna/index.jsp>
- [12] <http://androideity.com/>
- [13] <http://jarroba.com>
- [14] <http://developer.android.com/reference/android/app/Activity.html>
- [15] <http://www.stackoverflow.com>
- [16] <http://www.developereconomics.com/>
- [17] <http://www.androidauthority.com/>

PRESUPUESTO

1) Ejecución Material

- Compra de ordenador personal (Software incluido)..... 1500 €
- Alquiler de impresora láser durante 6 meses 50 €
- Material de oficina 100 €
- Móvil con SO Android 350 €
- Tablet con SO Android 300 €
- Total de ejecución material 2.300 €

2) Gastos generales

- 16 % sobre Ejecución Material..... 368 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material..... 138 €

4) Honorarios Proyecto

- 664 horas a 15 € / hora..... 9.960 €

5) Material fungible

- Gastos de impresión..... 60 €
- Encuadernación..... 200 €

6) Subtotal del presupuesto

- Subtotal Presupuesto 13.026 €

7) I.V.A. aplicable

- 21% Subtotal Presupuesto..... 2735,5 €

8) Total presupuesto

- Total Presupuesto 15.761,5€

Madrid, Febrero de 2015

El Ingeniero Jefe de Proyecto

Fdo.: Iván del Castillo Gallardo

Ingeniero de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de la siguiente aplicación: “Waterpolo Improvement: aplicación Android para la mejora en dicho deporte y realización de estudios estadísticos”. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la

misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.