

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

Ingeniería de Telecomunicación

**ESTUDIO SOBRE LA UTILIZACIÓN DEL GPS DE
TELÉFONOS INTELIGENTES PARA EL ANÁLISIS DE
DEPORTE AL AIRE LIBRE**

Sergio Zurita Eiranova

Septiembre 2016

Estudio sobre la utilización del GPS de teléfonos inteligentes para el análisis de deporte al aire libre

AUTOR: Sergio Zurita Eiranova

TUTOR: Eduardo Boemo Scalvinoni

DSLab

Dpto. de Tecnología Electrónica y Comunicaciones

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Septiembre 2016

AGRADECIMIENTOS

Ha sido complicado manejar el trabajo personal y el PFC al mismo tiempo pero creo que con el apoyo de mi familia al final ha salido un muy buen resultado. Gracias por la preocupación a mi madre, mi padre, mis dos hermanas, mi hermano, Manolo, Claudette, Manolo, Juan y Emma.

Agradecimientos a Ana siempre por todo, a Luis, Hugo, Fer, Marta, Silvio y a Eduardo Boemo por darme la oportunidad de desarrollar este proyecto.

RESUMEN

Este proyecto consiste en desarrollar una aplicación que haga el seguimiento del ejercicio físico por parte del usuario y lo muestre por pantalla de manera que se pueda estudiar la actividad física realizada.

Esto se ha conseguido mediante el uso del GPS, una herramienta a la orden del día que tiene mucho potencial ya que los usuarios valoran aplicaciones que le hagan la vida más sencilla, así puede monitorizar la localización y ofrecer servicios cercanos, aportar datos al instante según la zona en la que se encuentre el usuario o medir el ejercicio que realiza.

Una actividad física a realizar por parte del usuario podría ser jugar un partido de fútbol de manera que se puedan recoger los datos y gestionarlos de manera que el usuario que utiliza la aplicación, vea por pantalla características útiles sobre ese ejercicio que ha realizado y está realizando. Por ello, se ha programado una aplicación muy completa creando un módulo que gestiona los datos de un partido de fútbol como los goles y el tiempo, esto sería portable a usarlo con cualquier ejercicio físico.

Además contiene una serie de entrenamientos que son indispensables a la hora de realizar ejercicio.

Se ha conseguido desarrollar una aplicación completa que, para un usuario que le guste mucho el fútbol, un mercado muy extenso, se le da la oportunidad de monitorizar sus ejercicios y llevar un histórico de resultados y estadísticas. Por supuesto es válido para ejercicios alejados del fútbol en los que se quiera medir la distancia recorrida así como la velocidad media.

PALABRAS CLAVE

Móvil, tablet, Android, iOS, GPS, ejercicio, fútbol.

ABSTRACT

This Project consists in developing an application to track physical exercise and to show it in the screen.

This has been achieved through the use of the GPS, nowadays a tool up to date that has much potential. It collects and manages the data so that the user, who uses the application, is able to see on the display useful characteristics about the exercise is doing or has done.

It has been created a module that manages the football match data as the goals or the timer; this would be also applied to another kind of exercise

It also includes a list of trainings that are very important when doing sport.

KEY WORDS

Mobile, tablet, Android, iOS, GPS, exercise, soccer.

ÍNDICE DE CONTENIDOS

1	Introducción.....	13
1.1	Motivación.....	13
1.2	Objetivos.....	13
1.3	Organización de la memoria.....	14
2	Estado del arte	15
2.1	Introducción.....	15
2.2	Android como plataforma de código abierto.....	15
2.2.1	Android, cambios en el sistema operativo y fragmentación.....	17
2.3	iOS y la unión del HW y el SW	18
2.3.1	iOS, cambios en el sistema operativo y fragmentación.....	19
2.4	Comparación entre Android e iOS	20
2.4.1	Bajo nivel.....	20
2.4.2	Desarrolladores	20
2.4.3	Interacción e interfaz	21
2.5	Localización del dispositivo	22
2.5.1	Localización por red	22
2.5.2	Localización por componentes del terminal	24
2.6	Aplicaciones y uso de la localización.....	27
3	Diseño.....	29
3.1	Introducción y propósito.....	29
3.2	Enfoque y Herramientas utilizadas.....	29
3.2.1	Enfoque.....	29
3.2.2	Herramientas utilizadas	30
3.3	Diagrama de estados	33
4	Desarrollo	35
4.1	Introducción.....	35
4.2	Pantallas de la aplicación.....	35
4.2.1	Pantalla Equipos	35
4.2.2	Pantalla Estadísticas	37
4.2.3	Pantalla Entrenamiento.....	38
4.2.4	Pantalla Info.....	39

4.2.5 Pantalla Partido.....	40
4.2.6 Desarrollo a bajo nivel – Explicación técnica	46
5 Test de la aplicación	51
5.1 Dispositivo elegido para las pruebas y sus características	51
5.2 Aplicaciones elegidas para la prueba.....	52
5.3 Prueba básica de la aplicación desarrollada jugando un partido de fútbol.....	53
5.3.1 Condiciones iniciales.....	53
5.3.2 Prueba, dinámica de utilización de la aplicación y resultados	53
5.3.3 Conclusiones.....	54
5.4 Prueba de precisión frente a la competencia	54
5.4.1 Prueba de la aplicación en un recorrido medio-largo de 7-8 km.....	54
5.4.2 Prueba de la aplicación en un recorrido corto de 100 metros.....	57
5.4.3 Conclusión de las pruebas de precisión realizadas.....	60
5.5 Prueba de batería frente a la competencia	62
5.5.1 Consumo de batería en 2 recorridos distintos.....	62
5.5.2 Conclusiones de las pruebas de batería realizadas	63
6 Conclusiones del proyecto y trabajo futuro.....	65
6.1 Conclusiones.....	65
6.2 Trabajo futuro	66
Referencias	69
Glosario	71
Anexos.....	73

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Arquitectura Android	16
Ilustración 2: Fragmentación Android [24]	17
Ilustración 3: Arquitectura iOS [36]	18
Ilustración 4: Fragmentación iOS [35]	19
Ilustración 5: Aplicaciones vs ingresos de Apple Store [37].....	20
Ilustración 6: Conectividades de un Smartphone [25].....	22
Ilustración 7: Cell ID [26]	22
Ilustración 8: eCell ID [27].....	23
Ilustración 9: Ángulo de llegada [28]	23
Ilustración 10: Huella multitrayecto [29]	23
Ilustración 11: Constelación NAVSTAR [30].....	24
Ilustración 12: Triangulación de señales para localizar un punto [31].....	25
Ilustración 13: Diferencias entre captación de señal de los satélites [32]	26
Ilustración 14: La localización ayuda en la agricultura [33]	27
Ilustración 15: Enfoque del proyecto.....	29
Ilustración 16: Frameworks usados	31
Ilustración 17: Meteor [1].....	31
Ilustración 18: Angular [2]	32
Ilustración 19: Cordova [3]	32
Ilustración 20: Ionic [4].....	32
Ilustración 21: Diagrama de estados.....	33
Ilustración 22: Barra fija.....	35
Ilustración 23: Pantalla Equipos	36
Ilustración 24: Pantalla estadísticas	37
Ilustración 25: Pantalla entrenamiento	38
Ilustración 26: Pantalla Info	39
Ilustración 27: Insertar Equipos.....	40
Ilustración 28: Pantalla jugando el partido	40
Ilustración 29: Pantalla Partido	41
Ilustración 30: Pantalla de los permisos del GPS	42
Ilustración 31: La Tierra como elipsoide o como esfera [38].....	43
Ilustración 32: Aplicación de la ley de los cosenos esféricos y Haversine para tres puntos dados en una esfera [34]	43
Ilustración 33: Diagrama del cálculo de la distancia iterando Haversine.....	45
Ilustración 34: Estructura de ficheros	47
Ilustración 35: Recorrido realizado	55
Ilustración 36: Excel con puntos de localización	60
Ilustración 37: Diferencia entre tomar más o menos puntos de localización [39]	61
Ilustración 38: Diferencia de trazados registrados en distintas aplicaciones.....	61

ÍNDICE DE TABLAS

Tabla 2: Distancia medida por otras aplicaciones en 7.28km	55
Tabla 3: Distancia medida por la aplicación desarrollada en 7.28km	55
Tabla 4: Desviación máxima de la distancia recorrida en 7.28km	56
Tabla 5: Desviación en la distancia recorrida de la aplicación desarrollada en 7.28km	56
Tabla 6: Velocidad media medida por otras aplicaciones en 7.28km	56
Tabla 7: Velocidad media medida por la aplicación desarrollada en 7.28km	57
Tabla 8: Desviación máxima de la velocidad media medida en 7.28km	57
Tabla 9: Desviación en la velocidad media medida en la aplicación desarrollada en 7.28km	57
Tabla 10: Distancia medida por otras aplicaciones en un recorrido teórico de 100 m...	58
Tabla 11: Distancia medida por la aplicación desarrollada en 100m	58
Tabla 12: Desviación máxima de la distancia recorrida en 100m	58
Tabla 13: Desviación en la distancia recorrida de la aplicación desarrollada en 100m .	58
Tabla 14: Velocidad media medida por otras aplicaciones en 100m	59
Tabla 15: Velocidad media medida por la aplicación desarrollada en 100m	59
Tabla 16: Desviación máxima de la velocidad medida en 100m	59
Tabla 17: Desviación en la velocidad media medida en la aplicación desarrollada en 100m	60
Tabla 18: Análisis de consumo de batería en 43 minutos	62
Tabla 19: Análisis de consumo de batería en 12 minutos	63
Tabla 20: Consumo medio de batería de la aplicación desarrollada	63

1 Introducción

1.1 Motivación

Este proyecto principalmente nace con la motivación de aprender el sistema operativo más usado en el mundo: Android.

Sobre todo lo que motiva es pensar en el uso de un dispositivo con Android, diferenciar en la persona que esté usando ese dispositivo. El perfil de la persona es clave: la edad, el tipo de vida, su nivel socioeconómico, sus necesidades, sus deseos, sus gustos... Todos estos factores han de calificarse en su correcta medida para poder valorar el uso de cada dispositivo.

Android es un gran mundo que abarca muchos nichos, desde la interfaz visual más básica, hasta la programación a bajo nivel. Todo lo que hay bajo ese gran paraguas es lo que ha motivado este proyecto: interfaces, módulos, uso, funcionalidades, diseño... Al fin y al cabo este proyecto refleja el interés por entender a bajo nivel cómo funcionan los dispositivos que llevan millones de personas en su bolsillo.

La llegada de los dispositivos móviles inteligentes (a partir de ahora, *smartphones*), produjo que se integrara en su hardware módulos de geolocalización como es el GPS así como conexiones de red móvil para aprovecharse de la continua conexión a Internet.

Concretamente este documento aún dos elementos que las personas tienen muy presentes en su día a día: el deporte y el dispositivo móvil.

El usuario tiene la oportunidad de monitorizar su estilo de vida, sus hábitos, su rutina de ejercicio. Ahí nace la idea de desarrollar este proyecto.

1.2 Objetivos

Este proyecto tiene por objetivo estudiar la utilización de un GPS estándar de teléfonos móviles para analizar deportes; en particular el fútbol.

El objetivo es aprender a programar aplicaciones para dispositivos Android y sobre todo, lenguaje Javascript.

Desde el punto de vista educativo el objetivo del proyecto es adquirir los siguientes conocimientos:

- Programación orientada a objetos: Javascript.
- Lenguaje XML.
- Conocimiento de la plataforma Android (SDK).
- Uso del GPS en Android

La aplicación será válida tanto como para móviles como para tablets y sus distintos tamaños de pantalla, así como para distintos sistemas operativos.

1.3 Organización de la memoria

La memoria se compone de los siguientes capítulos:

- Capítulo 1: Introducción
Se explican los motivos y objetivos del proyecto.
- Capítulo 2: Estado del arte
Se estudia los principales sistemas operativos móviles así como los procesos para conseguir la localización.
- Capítulo 3: Diseño del proyecto
Se analiza cómo se enfocó el proyecto desde el principio, las herramientas utilizadas y el diagrama de estados
- Capítulo 4: Desarrollo del proyecto
Se muestra el proyecto completo, las distintas pantallas por las que pasa el usuario y cómo se implementaron las funcionalidades más importantes.
- Capítulo 5: Test de la aplicación
Se realiza en análisis de una serie de pruebas de campo que miden el comportamiento de la aplicación.
- Capítulo 6: Conclusiones
Se sacan las conclusiones tras la finalización del proyecto así como futuras implementaciones en el mismo

2 Estado del arte

2.1 Introducción

Desde el lanzamiento del sistema operativo Android por parte de Google, y de iOS por parte de Apple, han sido muchas las funcionalidades que se han ido añadiendo y también muchos los cambios que se han añadido a los dispositivos que usan estos sistemas operativos.

En las próximas líneas se realiza un análisis de la arquitectura de los dos sistemas operativos móviles más usados, la fragmentación actual que nos servirá para tener una visión global del proyecto, así como el componente hardware que utilizan estos sistemas operativos: el GPS.

2.2 Android como plataforma de código abierto

El 5 de noviembre de 2007, se fundó la Open Handset Alliance (OHA), liderada por Google con otros 34 miembros entre los que se incluían fabricantes de dispositivos móviles, desarrolladores de aplicaciones, algunos operadores de comunicaciones y fabricantes de chips.

Con la formación de la Open Handset Alliance, se presentó Android, una plataforma de código libre para teléfonos móviles basada en el núcleo operativo Linux.

Android se caracteriza por ser un sistema operativo de código abierto, es decir, está abierto a ser implementado en casi cualquier hardware y además admite modificaciones sobre su propio código para añadir funcionalidades extra. Todo esto le aporta una gran globalidad y que la comunidad de desarrolladores encuentre muchas facilidades a la hora de programar con este sistema operativo.

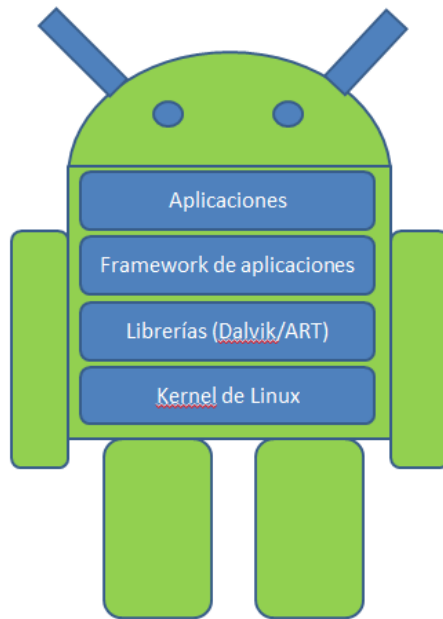


Ilustración 1: Arquitectura Android

Al hablar de Linux, se debe diferenciar entre sus distintas distribuciones y su kernel o núcleo. Al arrancar Android, se carga el kernel de Linux al igual que lo haría en una distribución de Linux aunque muchas partes del software son diferentes (por ejemplo, Android no incluye todas las bibliotecas GNU).

Para conformar por completo el sistema operativo Android es necesario agregar capas a este núcleo:

- **Kernel de Linux:** contiene los drivers necesarios de cualquier componente hardware de manera que se pueda acceder a este con las llamadas correspondientes.
- **Librerías:** escritas con C/C++, proporcionan las principales características a Android: bases de datos, fuentes, códecs multimedia, navegadores, motores gráficos...
- **Framework y entorno de ejecución:** herramientas para desarrollar aplicaciones. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, utilizan el mismo conjunto de APIs.

El componente principal del entorno de ejecución de Android es la máquina virtual Dalvik, componente que ejecuta todas y cada una de las aplicaciones no nativas de Android. Las aplicaciones se codifican normalmente en Java y son compiladas en un formato específico para la máquina virtual Dalvik, que es la que las ejecuta. Esto permite compilar una única vez las aplicaciones y distribuirlas ya compiladas con la total garantía de que podrán ejecutarse en cualquier dispositivo Android que disponga de la versión mínima del sistema operativo que requiera cada aplicación.

- **Aplicaciones:** sean nativas de Android o diseñadas por terceros utilizan los servicios, las API y librerías de los niveles anteriores.

2.2.1 Android, cambios en el sistema operativo y fragmentación

Desde el nacimiento de Android, han sido muchas las versiones que han ido saliendo a los usuarios. Las actualizaciones del sistema operativo son muy importantes porque solucionan fallos, añaden parches de seguridad para adaptarse a la época e incluyen módulos y código que hacen que sea más estable y fluido. Además de mejorar por dentro el sistema operativo, estas actualizaciones añaden nuevas funcionalidades y mejoran o cambian la interfaz del dispositivo de cara al usuario.

Con las primeras versiones había muchos más cambios relevantes, en las últimas se está puliendo aún más el sistema operativo y cambiando mucho la interfaz. Fue con Ice Cream Sandwich (Android 4.0) cuando se llegó a ver el gran cambio: multitarea, notificaciones, personalización e interactividad

Version	Codename	API	Distribution
2.2	Froyo	8	0.2%
2.3.3 - 2.3.7	Gingerbread	10	3.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	2.7%
4.1.x	Jelly Bean	16	9.0%
4.2.x		17	12.2%
4.3		18	3.5%
4.4	KitKat	19	36.1%
5.0	Lollipop	21	16.9%
5.1		22	15.7%
6.0	Marshmallow	23	0.7%

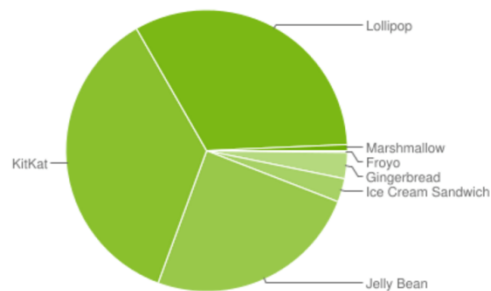


Ilustración 2: Fragmentación Android [24]

Con las nuevas versiones llega el soporte híbrido tanto para terminales con baja calidad de hardware como para los dispositivos de tope gama. Por supuesto, también la compatibilidad con los wearables, un mercado con un gran crecimiento actualmente.

2.3 iOS y la unión del HW y el SW

iOS es el sistema operativo propietario de la compañía estadounidense Apple. A diferencia de Android, este sistema operativo explotó ligado a potentes productos hardware como son el iPhone y el iPad.

La característica principal de este sistema operativo es la prohibición de instalarlo en otros dispositivos que no sean de la marca de la manzana. Esto da una señal de exclusividad, los usuarios tienen que comprarse productos que no ofrecen muchas alternativas en cuanto a precio o especificaciones. Además el aunar el propio SW y diseñar el propio HW, aporta una gran capacidad de optimización del sistema. Al no ser de código abierto aporta más seguridad.

Destaca su interfaz limpia y fluida, con mucha importancia al diseño y no tanta al contenido o a la funcionalidad. Pantallas simples y llamativas, tonos blancos, y menús poco intrusivos con poca personalización para el usuario, la intención clara es la sencillez.

A diferencia de Android, iOS está basado en XNU(X no es Unix) desde la capa que está a más bajo nivel. A continuación se muestran las capas que constituyen el sistema operativo iOS de Apple:

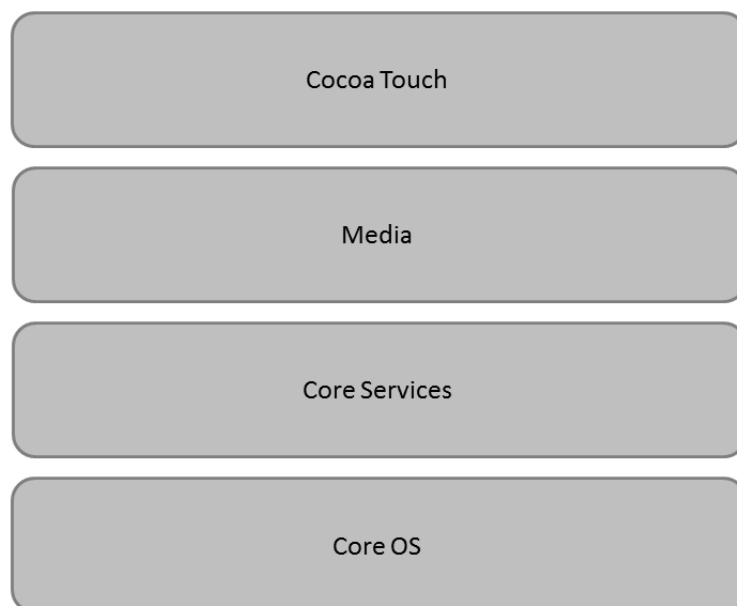


Ilustración 3: Arquitectura iOS [36]

- **Core OS:** contiene las funcionalidades a más bajo nivel, define métodos y funciones de interacción con un HW externo: acceso a memoria, seguridad, autenticación, manejo de ficheros, drivers...

- **Core Services:** contiene los principales servicios que requieren las aplicaciones. Éstas actualmente son muy completas y necesitan de multitud de servicios como el acceso a red, acceso a la base de datos, protección de datos, soporte XML...
- **Media:** esta capa contiene las tecnologías multimedia. Tanto las APIs para los componentes gráficos así como el procesamiento de audio son características esenciales que se transfieren a la capa superior.
- **Cocoa Touch:** es la parte visible del sistema operativo. En esta capa tiene lugar la interacción del usuario con el sistema operativo.

2.3.1 iOS, cambios en el sistema operativo y fragmentación

Como se comentaba anteriormente, Apple aún a HW y SW en sus productos, esto les aporta gran capacidad de optimización. Desde los inicios el sistema iOS fue muy fluido, la latencia en la interacción de los usuarios con sus productos era muy baja con lo que la satisfacción de la calidad aumentaba.

Cada año Apple pone a disposición de casi todos sus productos una nueva versión del sistema operativo móvil iOS. Cada actualización incluye mejoras de estabilidad, seguridad e incluye nuevas funcionalidades. En este gráfico vemos la fragmentación actual que existe en los dispositivos estadounidenses:

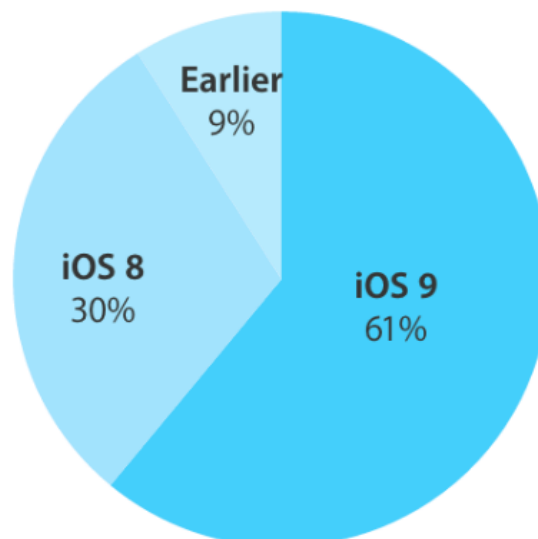


Ilustración 4: Fragmentación iOS [35]

Se observa como la última actualización iOS 9 abarca la mayor parte del mercado. Esto es una buena práctica ya que se consigue poca fragmentación, y con ello la resolución de problemas y poder dar mejor soporte a los usuarios. Además se consigue que las nuevas funcionalidades estén disponibles para un mayor número de personas y así tener un mayor nivel de fidelidad entre los compradores.

2.4 Comparación entre Android e iOS

A continuación se muestra un análisis de las principales características de los 2 sistemas operativos:

2.4.1 Bajo nivel

Una de las mayores diferencias es el lenguaje de programación, en el caso de Android, utiliza Java mientras que en el caso de iOS usa uno propietario: Objective-C/Swift. Ambos son lenguajes de orientación a objetos. Según estilos de programación, los desarrolladores prefieren uno u otro, Swift es más nuevo pero tiene menos soporte, sin embargo Java está ya muy maduro y tiene un gran soporte y documentación.

2.4.2 Desarrolladores

A la hora de que un desarrollador se decida por un sistema operativo y otro, entran varios factores en escena. El más importante sería el propósito de la aplicación, si el desarrollador quiere lucrarse de poner a disposición de los usuarios su aplicación, si le interesa más la interfaz de Apple o prefiere elegir Android para poner a disposición de otros programadores su aplicación y así mejorarla.

Sirva de referencia el panorama de descargas enfrentado al de ingresos, la tienda de aplicaciones de Google registra un mayor número de descargas que la de Apple, sin embargo los ingresos son mayores en el caso de la empresa de la manzana:



Ilustración 5: Aplicaciones vs ingresos de Apple Store [37]

Aunque Play Store registre un mayor número de descargas, hay que tener en cuenta que tiene muchas más aplicaciones gratuitas que Apple Store.

Algo que valoran mucho los desarrolladores y usuarios de Android es la llegada de novedades antes que la competencia. Esto se produce ya que los que programan en Android tienen más libertad, en cambio sus competidores tienen más barreras ya sea por control, seguridad o por una simple decisión. A pesar de las restricciones, muchas aplicaciones llegan antes a iOS que a Android debido a la posible mayor recaudación económica.

Como veremos más adelante, existen herramientas a disposición de los desarrolladores para conseguir programar un código y adaptarlo a distintos sistemas operativos

2.4.3 Interacción e interfaz

Con el paso de los años cada sistema operativo ha ido puliendo su interfaz cambiando transiciones, colores y funcionalidades para llamar la atención del consumidor. Es clave aquí conseguir la mejor experiencia de usuario posible.

iOS se basa casi exclusivamente en el entorno táctil al contar con un solo botón central, sin embargo Android al ser usado por distintos fabricantes tiene distintas variantes: LG sólo incluye botones físicos atrás, Samsung y Huawei tiene un botón central más dos táctiles a su lado. Las transiciones entre pantallas han ido mejorando en Android, ya son muy fluidas, algo que siempre ha caracterizado a Apple.

Lo más destacable en este apartado son sin duda las capas de personalización.. Al ser iOS un sistema exclusivamente dedicado a sus productos, todos estos tienen la misma apariencia, no tienen modificaciones visuales según qué dispositivo se tenga. Ocurre totalmente lo contrario en la compañía de Google, generan un sistema operativo que incluyen en sus productos (Nexus), pero lo ponen a disposición de fabricantes y desarrolladores para que lo modifiquen a su antojo. Transición entre ventanas, iconos, menús, barra de notificaciones, nuevas funcionalidades...de este modo cada fabricante ha ido creando su propia capa de personalización, es decir, su identidad. Los más conocidos son Sense de HTC, Touchwiz de Samsung y MIUI de Huawei.

2.5 Localización del dispositivo

Al mirar la hoja de especificaciones de un dispositivo inteligente, normalmente vemos las características referentes a la capacidad de localización:

COMMS	WLAN	Wi-Fi 802.11 a/b/g/n/ac, dual-band, Wi-Fi Direct, hotspot
	Bluetooth	v4.2, A2DP, LE, aptX
	GPS	Yes, with A-GPS, GLONASS, BDS
	NFC	Yes
	Radio	No
	USB	microUSB v2.0, USB Host

Ilustración 6: Conectividades de un Smartphone [25]

La geo-localización o geo-referenciación, determina la ubicación geográfica a través de un dispositivo móvil con cualquiera de los medios disponibles para lograrlo. Hay varias formas de conseguir la ubicación, usar la red del operador, un router o un componente HW dentro del dispositivo como un receptor GPS. Sin duda la forma más usada en la actualidad es el dispositivo móvil, principalmente debido a su portabilidad. Con la irrupción de los teléfonos inteligentes, que llevan integrados receptores GPS, podemos ubicarnos en cualquier punto del mundo, gracias a una red de satélite que rodea al globo terráqueo. Además, podemos conseguir también la geolocalización a través de una conexión a Internet, aunque con una precisión menor que la ofrecida por el GPS.

2.5.1 Localización por red

- **Identificación de celda:** para la localización se utiliza la celda (antena) a la que el teléfono está conectado.

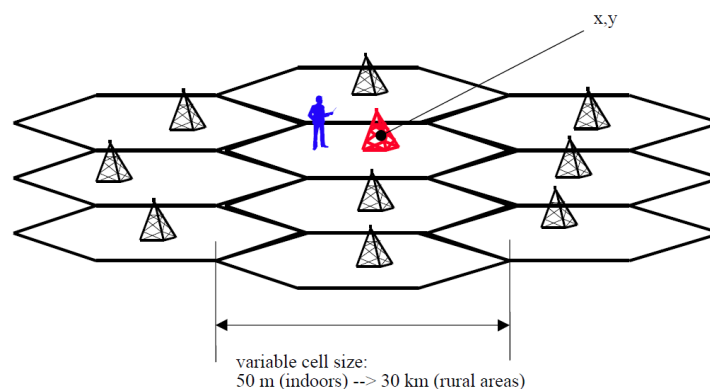


Ilustración 7: Cell ID [26]

- **Identificación mejorada de celda:** además de la celda conectada, se utiliza el TA (Time of Arrival), que estima la distancia entre la estación base de la celda y el terminal.

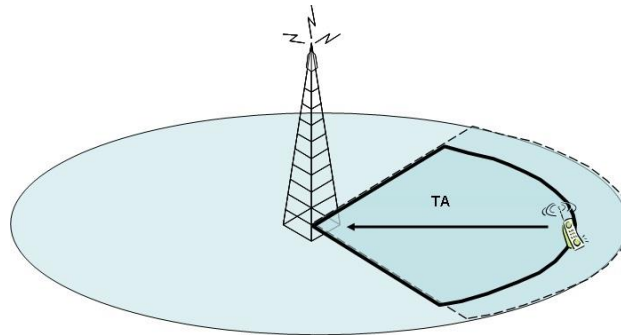


Ilustración 8: eCell ID [27]

- **Ángulo de llegada:** utiliza antenas multiarrays situadas en la estación base para determinar el ángulo de la señal recibida. Se necesitan al menos 2 estaciones para medir ambos ángulos y conseguir la localización.

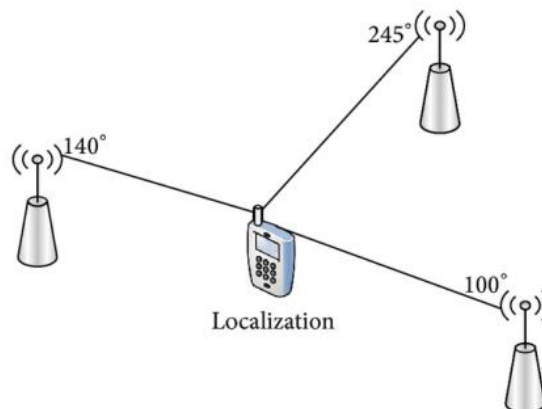


Ilustración 9: Ángulo de llegada [28]

- **Huella multitrayecto:** esta técnica caracteriza las señales que llegan desde diferentes localizaciones, las denominadas señales multitrayecto, que son reflejadas por diferentes obstáculos y que llegan a la estación base.

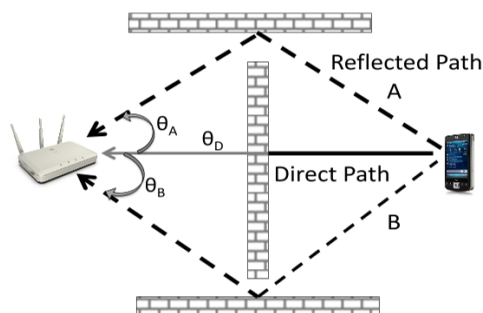


Ilustración 10: Huella multitrayecto [29]

2.5.2 Localización por componentes del terminal

Al usar sólo las redes móviles, la precisión que se consigue no es muy precisa. De ahí nació la inclusión del GPS en los dispositivos:

- **GPS:** El GPS es un sistema de posicionamiento global que se inauguró en 1995. Fue creado principalmente como una herramienta militar, pero dada la gran cantidad de aplicaciones civiles que se desarrollaron, se acabó de liberar para usos no sólo militares. Fueron Texas Instruments y Trimble, los primeros receptores que lograban la ubicación de una persona con cierta precisión. 24 satélites (más 3 de emergencia) orbitan a más de 20.000 km de altura, de este modo, en todo instante cualquier punto de la superficie terrestre está siendo observado por uno o más de estos satélites. En la Tierra hay 5 estaciones repartidas que controlan que los satélites no se salgan de su órbita y verifican su correcto funcionamiento.



Ilustración 11: Constelación NAVSTAR [30]

Cada satélite emite señales en dirección a la tierra, señales gracias a las que un receptor GPS es capaz de calcular la posición exacta, pero para hacer eso, no basta un único satélite, sino que hace falta recibir la señal de al menos tres de estos aparatos para así poder triangular la posición del receptor. Con tres satélites obtendríamos la latitud y la longitud para lograr la ubicación. Si se conecta con un cuarto satélite, será posible obtener la altura sobre el mar.

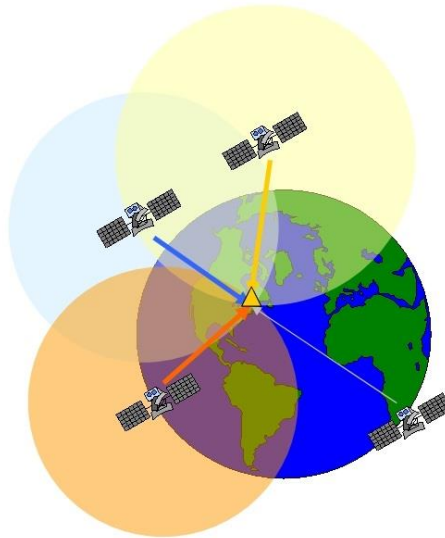


Ilustración 12: Triangulación de señales para localizar un punto [31]

Las señales que envían son de dos tipos y emiten a distintas frecuencias por seguridad:

- Tipo P (Precisión): señal de uso militar que utiliza las frecuencias 1575.42 MHz y 1226.7 MHz. Proporciona la máxima precisión, de ahí que por seguridad se prohíbe en usos civiles.
- Tipo SPS (Servicio de posición estándar): esta señal está reservada a usos civiles. Utiliza la frecuencia de 1575.42 MHz y no tiene tanta precisión como las señales de tipo P.

Las señales emitidas por los satélites indican la posición del mismo, además de enviar un reloj que permitirá calcular la distancia más adelante.

A partir de la señal de cada uno de los satélites, el receptor GPS, es capaz de calcular una serie de puntos posibles en los que se encuentra haciendo uso del retardo de la señal (tiempo que tarda en llegar del satélite a la tierra) y de la posición del satélite, de manera que si se reciben señales de más de un satélite, se puede triangular la posición, es decir disminuir progresivamente el número de posibles puntos en los que se encuentra el receptor.

A continuación se muestra una imagen que ilustra muy bien el comportamiento del GPS. Según nuestra posición y la de los satélites, tendremos un ratio de señal ruido mejor o peor de manera que conseguiremos posicionarnos de una manera más o menos precisa:



Ilustración 13: Diferencias entre captación de señal de los satélites [32]

No todo son ventajas en el GPS ya que no funciona en interiores y además es necesario tener el sistema activado, con el consiguiente gasto de batería en el dispositivo. Además se debe de tener en cuenta que el sistema GPS tiene un proceso inicial, denominado TTFF (Time To First Fix) o tiempo para la primera localización o posicionamiento, que suele ser muy largo en general, del orden de incluso minutos.

- **A-GPS:** Para minimizar este inconveniente los fabricantes de dispositivos inteligentes han desarrollado el sistema A-GPS o Sistema de Posicionamiento Global Asistido. De esta manera, en cuanto encendamos el teléfono móvil, se enviará a un servidor externo la identificación de la antena o célula GSM en la que estamos y el teléfono obtendrá como respuesta los satélites que tiene encima y su posición (están almacenados en el servidor externo, y según nuestra posición nos dará los datos de unos satélites u otros). De esta manera el cálculo de la posición por parte del equipo GPS del móvil se acorta en gran medida.
- **GLONASS:** es el homólogo ruso del GPS. Es un conjunto de 31 satélites en órbita, colocados a unos 19.000 km de altitud. No solo se centra en Rusia sino que puede ofrecer su servicio a todo el mundo sin problema. En los últimos años hemos visto cada vez más inclusión de este servicio en terminales. Esto, en cierto modo, es debido a que por parte del gobierno ruso se tomaron medidas drásticas para que su servicio de posicionamiento se popularizase, amenazando con subir los impuestos a los terminales que no incluyesen su sistema de geoposicionamiento.

- **Beidou:** es el homólogo chino del GPS. A diferencia de los sistemas GPS y GLONASS, que utilizan satélites en órbitas bajas y ofrecen servicio global, Beidou usa satélites en órbita geoestacionaria. Esto limita su cobertura en cuanto a la visión de satélites al país chino y sus vecinos, pero una gran ventaja es que no requiere una gran constelación de satélites. Otra gran diferencia de Beidou es que calcula las coordenadas únicamente con dos satélites y una estación en tierra, esto implica la necesidad de enviar una señal desde el dispositivo remoto, algo que no es necesaria con GPS o GLONASS.

2.6 Aplicaciones y uso de la localización

Son muchas las aplicaciones que hacen uso de la localización. Unas porque la propia aplicación se basa en la localización del terminal y otras que lo ofrecen como un servicio de valor añadido.

Desde que los smartphones tienen componentes más avanzados, los desarrolladores han querido añadir la localización para dar más valor a sus aplicaciones. Buscar la gasolinera más barata cerca de una ubicación, el mejor restaurante, controlar el seguimiento de los pasos, de la actividad física, vender y comprar artículos a personas cercanas...un sinfín de servicios que quieren hacer la vida más fácil al usuario.

- Una de las aplicaciones más importantes es la de ayudar a las personas ciegas o con discapacidad visual. Mediante el GPS y la navegación asistida, se puede guiar a las personas de manera que puedan dirigirse a su destino de manera autosuficiente.
- Debido a su gran amplitud, el GPS de precisión está siendo cada vez más utilizado en la agricultura. Esta tecnología permite acoplar y manejar eficientemente la gran cantidad de datos obtenidos en tiempo real. Tiene grandes beneficios como el levantamiento geográfico, el muestreo de suelo, mapas de rendimiento, posición exacta de plagas, insectos y malezas.

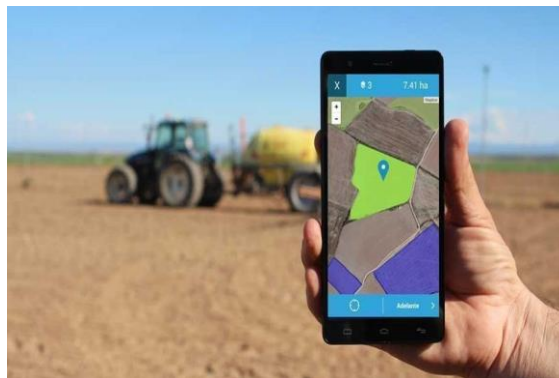


Ilustración 14: La localización ayuda en la agricultura [33]

También sirven para las avionetas de fumigación sin necesidad de señalizadores humanos, mejorar el uso de la tierra y el agua y mejorar el uso de fertilizantes.

- El GPS permite trabajar en condiciones de baja visibilidad, aunque haya lluvia, polvo, niebla o penumbra, sigue funcionando.
- La localización ha traído muchos beneficios medioambientales: zonas más impenetrables para evaluar flora y fauna, pronosticar y comprender cambios en el medio ambiente, rastrear las costumbre migratorias de algunas especies en peligro de extinción, rastrear zonas desérticas, erosionadas, deforestadas...
- Beneficia la cartografía y la modelización del mundo físico-desde montañas y ríos, hasta calles, edificios, cables y tuberías de los servicios públicos y otros recursos.
- La localización nació para beneficios militares. Por supuesto que localizar posiciones es crucial en este ámbito.
- La salud es otro de los temas bajo el que se encuentra el paraguas de la localización.

3 Diseño

3.1 Introducción y propósito

En este apartado se dará una visión global del proyecto, las distintas visualizaciones y por qué pantallas se pasa para ver el flujo general. Se informará de las herramientas utilizadas.

3.2 Enfoque y Herramientas utilizadas

3.2.1 Enfoque

Uno de los puntos más complicados del proyecto fue elegir con qué trabajar. Esta decisión fue muy importante porque desde que se toma, se condiciona todo el trabajo.

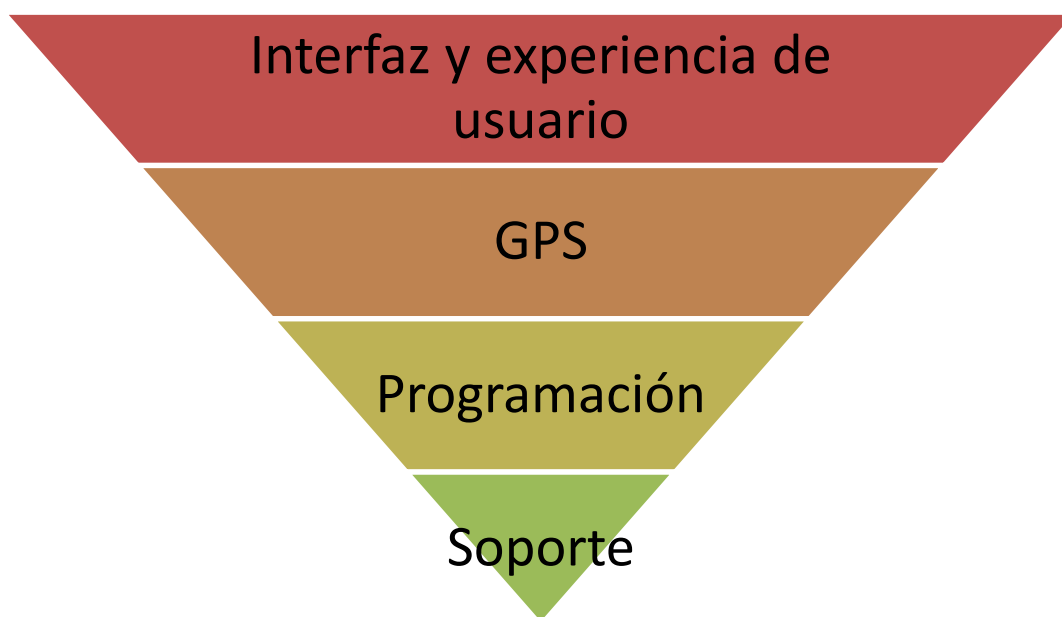


Ilustración 15: Enfoque del proyecto

Inicialmente en la hoja de ruta se contaba con valorar los aspectos:

- Interfaz, rapidez y visuales
- GPS
- Programación
- Soporte

Según la herramienta que se use, es más o menos fácil desarrollar unas **interfaces** con una buena apariencia de cara al usuario. Las transiciones y mejorar la experiencia del usuario son puntos clave al desarrollar una aplicación.

Tratar en cómo se iba a implementar la funcionalidad del **GPS** tenía complejidad. Los conocimientos previos eran muy básicos, muestreos, almacenarlos, hacer cuentas con esos muestreos, como conseguir los permisos de acceso al GPS...

Aunque se puede programar con distintos lenguajes, el conocimiento de **Java** es un punto a favor a la hora de realizar el proyecto.

El soporte de la comunidad de desarrolladores es vital, toda la **documentación** a la que se pueda acceder agiliza los trámites de solución de problemas e innovación a la hora de diseñar los distintos módulos de la aplicación.

3.2.2 Herramientas utilizadas

En un principio todo se decantaba por usar el SDK de Android con su correspondiente simulador y el entorno de programación Eclipse. Las ventajas que ofrecen estas herramientas son múltiples y el soporte de la comunidad de desarrolladores mayor aún. Al usar Eclipse, se cuenta con muchas distribuciones y soporta un gran número de lenguajes (Java, Python, PHP, C/C++, HTML, JavaScript...). Tiene plugins para prácticamente todo así como un sencillo editor de código. Lo malo es que es muy pesado y lento en la ejecución.

Con otros métodos se consigue que el servidor soporte toda la carga de la generación de páginas. Este proceso además de costoso en recursos es lento e impide utilizar nuestras aplicaciones cuando no hay conexión.

De ahí el cambio de decisión, los navegadores modernos soportan javascript de forma muy eficiente. Ejecutan javascript de forma rápida y gracias a las APIs de HTML5 son capaces de hasta guardar datos para su consulta en modo offline.

Para trabajar de forma ágil se decidió usar una serie de frameworks que aprovechan la potencia de los navegadores para hacer muchas de las operaciones, que antes se desarrollaban en el servidor, en el navegador del cliente dejando a los servidores trabajar lo mínimo imprescindible.

El uso de Meteor permite reflejar en tiempo real los cambios recibidos tanto en el servidor como en el cliente, de manera que automáticamente se refrescan los datos y la vista generada en el cliente, si fuera necesario. Esto permite que si la aplicación estuviera abierta en distintos terminales para desarrollar una aplicación multi-plataforma, se podría visualizar los cambios efectuados. Esto es una alternativa favorable ya que si no habría que establecer una serie de herramientas para el intercambio de información.

El uso de Angular proporciona una inyección de dependencias ya a la hora de desarrollar, facilita y optimiza tanto el desarrollo como la ejecución de la aplicación.

Ionic y Cordova, como los anteriores, se explicarán en detalle a continuación:

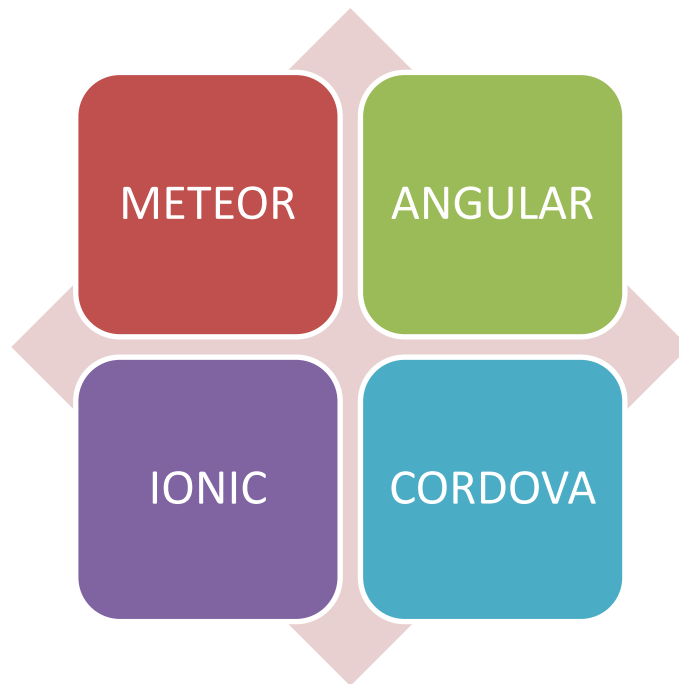


Ilustración 16: Frameworks usados

Meteor: es una plataforma, que nos provee tanto el cliente como el servidor para que podamos crear aplicaciones completas sólo con Javascript, las bases de datos las maneja con mongodb a través de colecciones y utiliza un tipo de bases de datos en el cliente llamada minimongo la cual es una copia exacta de la base de datos en el servidor y se sincroniza de forma automática cuando se ejecuta cualquier operación contra la misma.



Ilustración 17: Meteor [1]

- Sencillez para programadores con experiencia no muy extensa.
- Framework desarrollado para correr la aplicación en tiempo real: el hecho de realizar un cambio de código y verlo casi instantáneamente reflejado en la interfaz aporta una agilidad notable a la hora de programar.
- Lenguaje: uniformidad a la hora de escribir código con Javascript
- Soporte de los desarrolladores.

Angular: framework que trabaja en el cliente el cual nos permite desarrollar aplicaciones en versión web y móvil utilizando javascript.



Ilustración 18: Angular [2]

- El desarrollo de aplicaciones se hace mucho más sencillo y nos permite hacer cosas complejas con muy poco código.
- Framework algo pesado
- Módulo de diseño.

Cordova: framework de licencia libre que cuenta con muchas APIs de diversos dispositivos móviles para desarrollar aplicaciones nativas dentro de un smartphone. Para el desarrollo de las aplicaciones se utilizan las tecnologías web HTML, CSS y JavaScript.



Ilustración 19: Cordova [3]

- Desarrollo para iOS, Android y demás sistemas operativos sin la necesidad de programar en sus lenguajes nativos.
- Las aplicaciones programadas en Cordova comparten la misma interfaz mientras que por debajo sigue estando el código nativo de cada sistema operativo.
- Capaz de crear un puente de interacción vía JavaScript que permite interactuar con las funcionalidades del hardware del dispositivo (en el caso de este proyecto, el GPS).

Ionic: framework que permite crear aplicaciones multi-plataforma utilizando HTML5 y componentes Javascript muy ligados de Angular.



Ilustración 20: Ionic [4]

- Alto rendimiento y velocidad.
- Trabaja perfectamente con Angular.
- Se inspira en las SDK de desarrollo móviles nativos más populares, por lo que es fácil de entender para cualquier persona que ha construido una aplicación nativa para iOS o Android.

3.3 Diagrama de estados

Una de las características más importante de una aplicación es la experiencia de usuario. Desde que alguien abre la aplicación, ve la interfaz, recorre e interactúa por los distintos menús, la experiencia del usuario es lo que acaba decantando la valoración de una aplicación. Por supuesto entran en juego la funcionalidad, los componentes extra y el grado de complejidad de la misma.

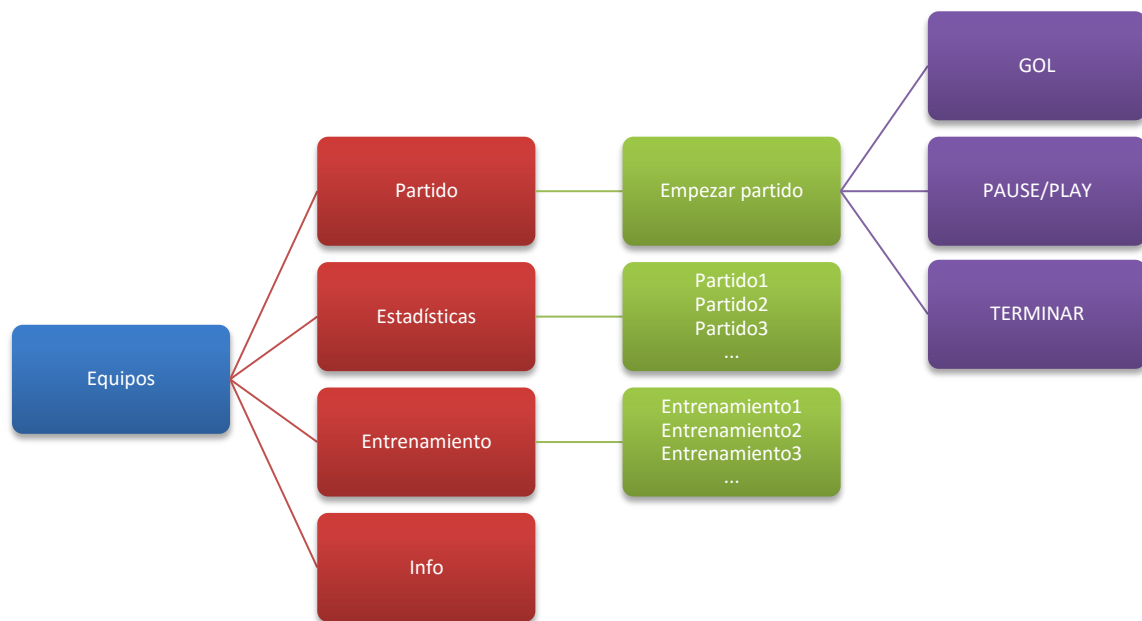


Ilustración 21: Diagrama de estados

La aplicación se inicia con la pantalla principal, se ha incluido una barra fija en la parte superior desde la cual se pueden acceder a las demás pantallas principales de la aplicación: Equipos, Partido, Estadísticas, Entrenamiento e Info.

En el siguiente apartado se ahonda en la aplicación, el diseño y en cómo se han programado los diferentes módulos.

4 Desarrollo

4.1 Introducción

En este apartado se detalla cómo se ha implementado el diseño de la aplicación con las herramientas mencionadas anteriormente. Se explicará por pantallas para saber localizar en todo momento de qué módulo, interfaz o funcionalidad se está hablando.

Además de ilustraciones de la propia aplicación, se profundizará a bajo nivel en cómo se han diseñado sus módulos así como las interfaces.

4.2 Pantallas de la aplicación

Las pantallas y secciones principales son “Equipos”, “Partido”, “Estadísticas”, “Entrenamiento” e “Info”.

La interfaz es muy sencilla e intuitiva, muy poco intrusiva para el usuario, se da mucha importancia al contenido. A las pantallas principales se accede desde una barra fija en la parte superior, este es un diseño muy visual y funcional:



Ilustración 22: Barra fija

4.2.1 Pantalla Equipos

Para no hacer perder el tiempo al usuario, la aplicación se inicia en una de las pantallas principales y funcionales llamada “Equipos”.

Ésta es una de las pantallas más importantes y en la que se basa la aplicación. Desde esta misma se tiene acceso a las demás pero también podemos empezar a interactuar con

la aplicación. De este modo se consigue que con pocos toques, el usuario acceda a toda la información de una manera rápida y cómoda.

Contiene la siguiente estructura:

- Nombre de la pantalla
- Barra fija desde la que se accede a las distintas pantallas
- Texto con información clara y concisa
- Introducción de los textos de los equipos que jugarán el partido
- Botón de “Empezar partido”

Esta es la pantalla “Equipos”:

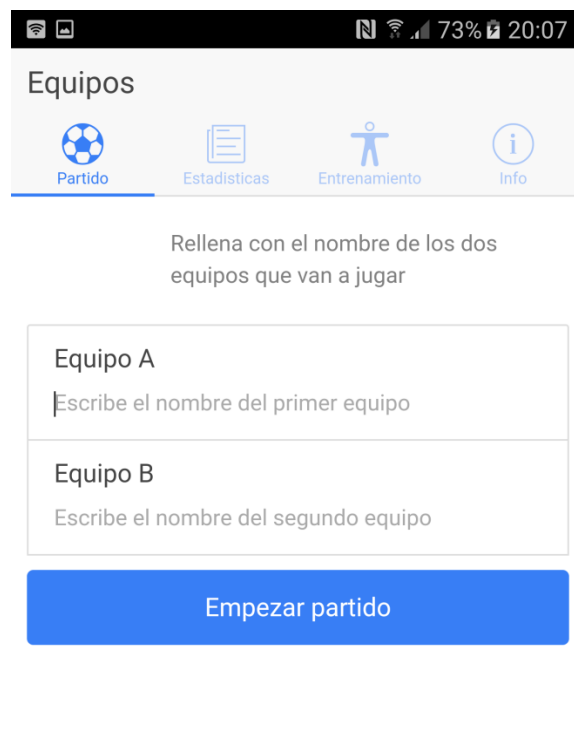


Ilustración 23: Pantalla Equipos

Más adelante ahondaremos en esta pantalla ya que es la principal de la aplicación y necesita de explicación extensa.

4.2.2 Pantalla Estadísticas

En esta pantalla se muestran los partidos que se han jugado y guardado en la base de datos. Se ha elegido una interfaz muy simple y accesible de manera que el usuario de un vistazo pueda acceder a ver todos los datos de los partidos que ha guardado en su terminal.

Contiene la siguiente estructura:

- Nombre de la pantalla “Estadísticas”
- Barra fija desde la que se accede a las distintas pantallas (“Estadísticas aparece seleccionado”)
- Lista de partidos en cascada y deslizable:
 - Partido1
 - Equipos y resultado
 - Velocidad de media
 - Distancia recorrida
 - Partido2
 - Equipos y resultado
 - Velocidad de media
 - Distancia recorrida
 -

Esta es la pantalla “Estadísticas”:



Ilustración 24: Pantalla estadísticas

4.2.3 Pantalla Entrenamiento

En esta pantalla se muestran una serie de entrenamientos que son beneficiosos para el jugador que vaya a jugar o haya jugado el partido.

Es una sección que aporta valor a la aplicación ya que no se quería caer en la simplicidad de una aplicación de medición, el objetivo es que sea completa y que aporte al usuario las necesidades que puede llegar a tener al practicar deporte.

Contiene la siguiente estructura:

- Nombre de la pantalla “Entrenamiento”
- Barra fija desde la que se accede a las distintas pantallas (“Entrenamiento aparece seleccionado”)
- Lista de entrenamientos en cascada y deslizable con ilustraciones y texto:
 - Entrenamiento1
 - Entrenamiento2
 - Entrenamiento3

Esta es la pantalla Entrenamiento:



Ilustración 25: Pantalla entrenamiento

4.2.4 Pantalla Info

En esta pantalla se muestra información básica de la aplicación. Fue una de las primeras en programar ya que era la más fácil, incluía imágenes lo que ya proporcionaba conocimientos de manejo de recursos y además se empezaba a ver la estructuración de los distintos módulos texto-imagen dentro de una misma pantalla.

Se informa sobre la aplicación, los programadores, la versión o datos de contacto.

Contiene la siguiente estructura:

- Nombre de la pantalla “Info”
- Barra fija desde la que se accede a las distintas pantallas (“Info aparece seleccionado”)
 - Información sobre el laboratorio
 - Información de la aplicación
 - Información sobre los programadores
 - Direcciones de contacto

Esta es la pantalla Info:



Ilustración 26: Pantalla Info

4.2.5 Pantalla Partido

Para acceder a esta pantalla tenemos que rellenar los datos de los equipos de la pantalla “Equipos” y pulsar “Empezar partido”

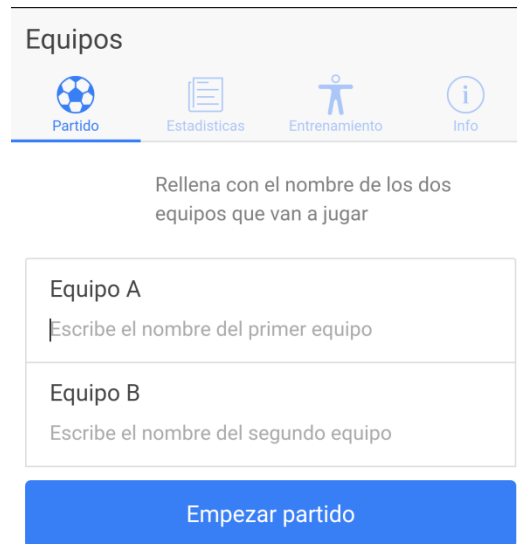


Ilustración 27: Insertar Equipos

Al pulsar “Empezar partido” habremos guardado ya en la base de datos los nombres de Equipo A y Equipo B y habremos llegado a la pantalla más importante de la aplicación. Esta pantalla “Partido” se presenta con una interfaz clara y no intrusiva en la cual se pueden ver los datos del partido que se está jugando e interactuar.



Ilustración 28: Pantalla jugando el partido

Se puede añadir goles al marcador tanto de un equipo como de otro. Se visualiza la parte que se está jugando así como el reloj corriendo.

Si hay algún contratiempo en el partido, se puede pausar y reanudar la cuenta. Este módulo se ha diseñado para poder hacer una pausa durante el ejercicio.

Además hay un botón “Terminar” que finalizaría el ejercicio/partido y guardaría en una base de datos las estadísticas del partido.

Los datos del ejercicio se mostrarán en tiempo real, en este caso mostraremos:

- Distancia recorrida
- Velocidad Media

A continuación se muestra la pantalla “Partido”:

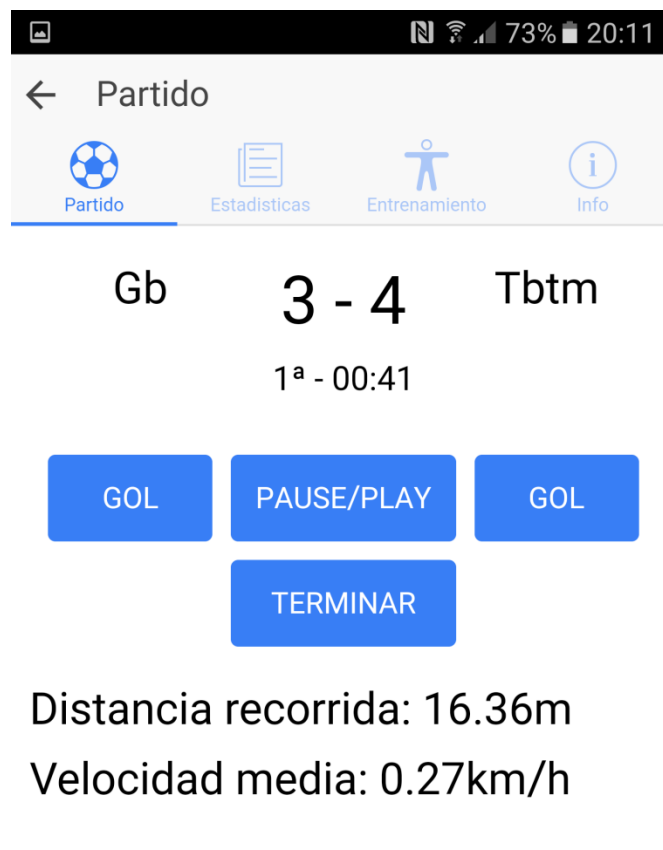


Ilustración 29: Pantalla Partido

4.2.5.1 GPS

La implementación del GPS en la aplicación se antojaba uno de los puntos más costosos del proyecto. Desde conseguir los permisos de acceso al HW del terminal, obtener los datos, tratar esos datos y finalmente mostrarlos al usuario.

Antes de nada hay que tener en cuenta que coleccionar y usar datos de geo-localización supone tener mucho cuidado con la privacidad. Hay que gestionar cómo la aplicación usa los datos recogidos, si los comparte y el nivel de precisión de los datos. Son datos muy sensibles porque revelan la posición del usuario y, si se almacena, el historial de sus recorridos. Así se ha considerado que el teléfono pregunte al usuario si permite a esta aplicación acceder a la ubicación de su dispositivo:

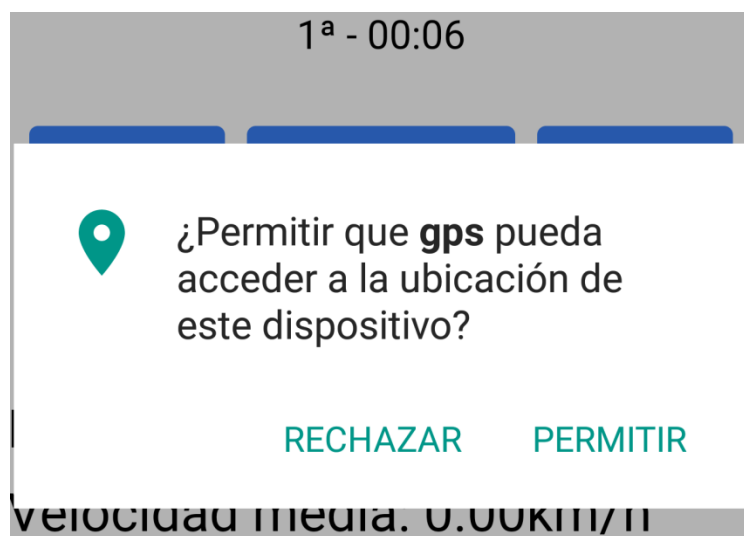


Ilustración 30: Pantalla de los permisos del GPS

Al usar el framework Meteor se tiene la ventaja de poder implementar módulos de otros frameworks. Indagando se encontró el complemento:

Cordova-plugin-geolocation

Este plugin contiene una serie de métodos que nos pueden proporcionar:

- Latitud(en grados)
- Longitud(en grados)
- Altitud (en metros)
- Velocidad (en m/s)
- Timestamp: Momento de la captura seleccionado en cada segundo (en ms)

Estos datos de latitud y longitud que obtenemos, se ubican sobre la forma terrestre, sin embargo lo que interesa es calcular la distancia lineal entre dos posiciones dadas: posición1 (timestamp) y posición2 (timestamp+1).

Método para hallar la distancia recorrida

Calcular la distancia recorrida fue uno de los puntos críticos del proyecto. Hay que tener en cuenta que calcular la distancia entre dos puntos sobre un plano se complica porque en el cálculo de la distancia entre ambas posiciones debemos contemplar la curvatura terrestre. Los cálculos matemáticos que hay que realizar dependen de cómo se considere la forma de la Tierra, si es un elipsoide o si es una esfera.

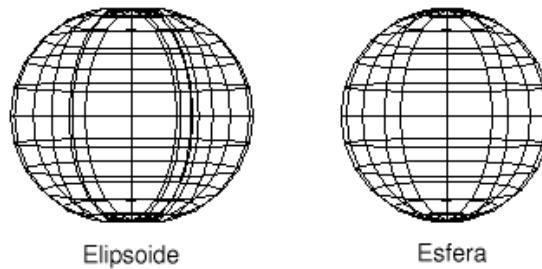


Ilustración 31: La Tierra como elipsoide o como esfera [38]

Si se considera que la Tierra tiene forma de elipsoide, los cálculos que hay que realizar para calcular la distancia entre dos puntos se basan en las **Fórmulas de Vincenty**^[40]. Estas fórmulas proporcionan gran precisión pero requieren una gran cantidad de código y además no tiene solución para dos puntos casi antipodales.

Si se considera que la Tierra es una aproximación cercana a una esfera, es matemáticamente mucho más fácil de manejar, de ahí la elección de esta forma para realizar los cálculos. En navegación y astronomía también se decantan por esta opción. Dada la Tierra esférica tenemos dos métodos para calcular la distancia entre dos puntos: la ley de los cosenos esféricos y la fórmula de Haversine. Aunque ambas son matemáticamente equivalentes, a continuación veremos como la fórmula de Haversine acaba siendo por precisión el mejor método para hallar la distancia entre dos puntos.

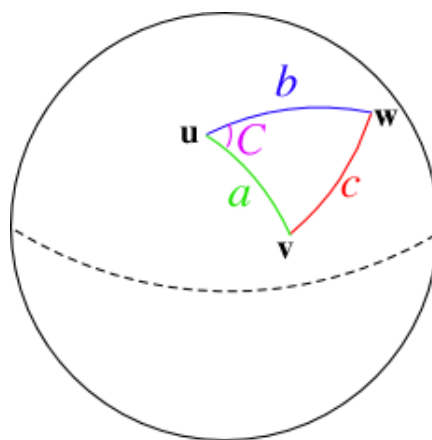


Ilustración 32: Aplicación de la ley de los cosenos esféricos y Haversine para tres puntos dados en una esfera [34]

Al considerar una esfera se cuenta en estos cálculos con 3 puntos “u”, “v” y “w”, en el que se considera w en el polo norte. Para estos cálculos, además de las dos posiciones de latitud y longitud, necesitamos el radio de la Tierra. Este valor es relativo a la latitud, pues al no ser la Tierra perfectamente redonda, el valor del radio ecuatorial es de 6378 km mientras que el polar es de 6357 km. El radio equivolumen es de 6371 km. En la aplicación hemos usado el radio equivolumen de la Tierra en metros.

Los 2 métodos para hallar la distancia recorrida son:

- **Ley de los cosenos esféricos**^[34]:

$$a = \sin(\text{lat1}) * \sin(\text{lat2})$$

$$b = \cos(\text{lat1}) * \cos(\text{lat2}) * \cos(\text{lon2} - \text{lon1})$$

$$c = \arccos(a + b)$$

$$\text{distancia} = R * c$$

Aunque esta fórmula es matemáticamente exacta a la que viene a continuación, no es precisa para pequeñas distancias porque el inverso del coseno aporta datos indiferenciables con 7 decimales:

$$\cos(5 \text{ grados}) = 0.9961946981$$

$$\cos(1 \text{ grado}) = 0.9998476952$$

$$\cos(1 \text{ minuto}) = 0.9999999577$$

$$\cos(1 \text{ segundo}) = 0.999999999882$$

$$\cos(0.05 \text{ segundos}) = 0.99999999999971$$

- **Fórmula de Haversine**^[34]: esta fórmula es la que se ha implementado en el código, se expresa en funciones de Haversine o sus múltiplos:

$$\text{Haversine} \left(\frac{d}{R} \right) = \text{haversine}(\text{lat1} - \text{lat2}) + \cos(\text{lat1}) * \cos(\text{lat2}) * \text{haversine}(\text{long2} - \text{long1})$$

Dónde:

d: es la distancia recorrida

R: es el radio de la Tierra

lat1 y long1: latitud y longitud del punto 1

lat2 y long2: latitud y longitud del punto 2

$$\text{haversine}(x) = \sin^2\left(\frac{x}{2}\right)$$

Al despejar la distancia recorrida, queda el código utilizado en la aplicación:

$R = \text{radio de la Tierra}$

$\Delta lat = lat2 - lat1$

$\Delta long = long2 - long1$

$$a = \sin^2(\Delta lat/2) + \cos(lat1) \cdot \cos(lat2) \cdot \sin^2(\Delta long/2)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R \cdot c$$

De esta manera, para hallar la distancia recorrida, vamos iterando Haversine cada segundo y así se obtendría la distancia total recorrida.

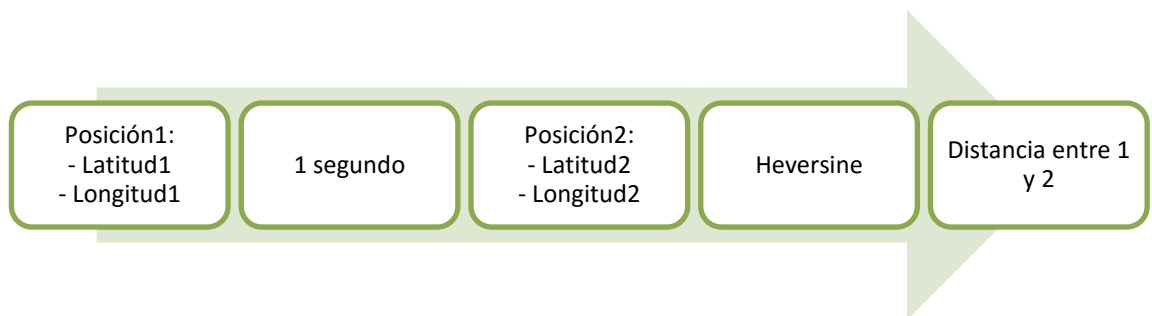


Ilustración 33: Diagrama del cálculo de la distancia iterando Haversine

4.2.5.2 Error, exactitud y precisión del GPS

El GPS es un sistema muy complicado en el que intervienen muchos factores. Entre satélites, envío de datos, estaciones de control de Tierra, hardware...hay que tener en cuenta que la precisión del GPS puede variar y puede dar lugar a errores de propagación.

Los satélites son muy precisos enviando señales porque contienen un reloj atómico de gran precisión. Esta señal es reconocida por los receptores en Tierra y calcula la distancia que le separa de él. Cuando se conecta con 4 satélites ya se obtendría la posición. Sin embargo los receptores GPS no cuentan con estos relojes por lo que aplican distintas mediciones para paliar esa menor precisión.

Se debe de tener en cuenta^[7]:

- Los errores de los relojes de los satélites pueden originar errores de precisión de un metro. El posible cambio de velocidad de los satélites desincroniza el canal de comunicación con las estaciones de Tierra que tienen que estar permanentemente verificando sus comportamientos.
- Los errores en efemérides pueden producir 1 metro de error. Retardos troposféricos, 1 metro. La troposfera es la capa más baja de la atmósfera (desde la superficie hasta entre 8 y 13 Km), esta capa está afectada

por cambios de temperatura, presión y humedad asociados a cambios meteorológicos.

- Retardos por la ionosfera y por la atmósfera. Estos errores pueden producir un desajuste de 10 metros. Este error depende de la altura a la que esté el satélite (cuanto más bajo, más le afecta), de la densidad de la ionosfera que depende de la incidencia de la luz del Sol (por la noche no tiene casi efecto y la señal no disminuye su velocidad) y del vapor de agua (el efecto se reduce con modelos atmosféricos).
- Multicamino: ½ metro. Se produce por la reflexión de las señales en superficies próximas al receptor y pueden interferir o producir errores en las señales que llegan directamente desde los satélites al receptor. El error por multicamino es muy difícil de detectar y en ocasiones es imposible de evitar.
- Las espurias pueden producir errores de cientos de kilómetros.
- Errores en el segmento de control debido a fallos humanos o de computación pueden causar errores desde un metro a cientos de kilómetros.
- Ruido y desvío de relojes, pueden resultar en errores de alrededor de los 15 metros para cada satélite utilizado en el cálculo de la posición.

4.2.6 Desarrollo a bajo nivel – Explicación técnica

El desarrollo de esta aplicación ha tenido varias complicaciones. Una de ellas ha sido la **implementación de los 4 frameworks** en un solo proyecto de manera que se consiga un código robusto que es capaz de correr en varios sistemas operativos.

Meteor, fue elegido el entorno de programación debido a su convergencia con los otros 3 frameworks. La estructura de archivos al programar en Meteor es la siguiente:

- client – Estará disponible en el lado de cliente. Usará código Angular e Ionic
- server - Estará disponible en el lado de servidor.
- public – Estos ficheros sirven como recursos para el cliente como imágenes, fuentes...
- lib – Los archivos llamados lib de cualquier jerarquía se ejecutarán los primeros.

Estructura de ficheros

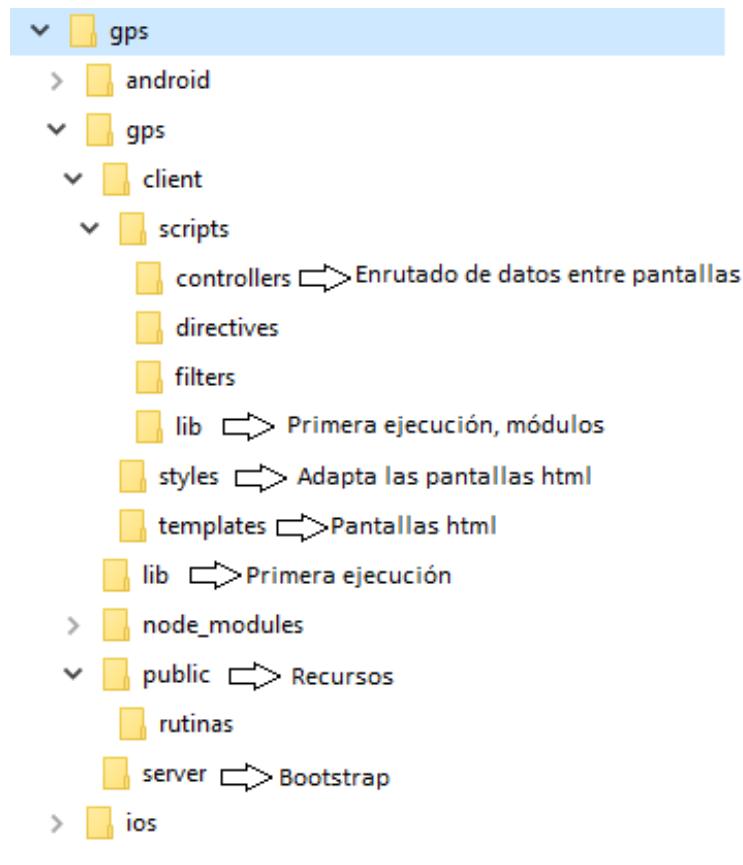


Ilustración 34: Estructura de ficheros

Una de las cosas más importantes que destaca la aplicación es el **diseño adaptativo**, adapta la interfaz del sitio web (en nuestro caso index.html) al tamaño del dispositivo en el que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de un ordenador, una tablet u otro dispositivo. Esto se consigue con el framework bootstrap y se tiene que ejecutar lo primero por lo que se incluye en lib:

```
function onReady() {  
    Angular.bootstrap(document, [App]);  
}
```

Para configurar la **barra fija pero seleccionable en la parte superior** de la aplicación se ha usado Meteor e Ionic en el archivo html “index” con 2 funciones para crear la barra y navegar por ella:

```
</ion-nav-bar>  
<ion-nav-view></ion-nav-view>
```

El **enrutado de datos entre pantallas** se diseñó importando el fichero “routes.js” y usando los controladores, a continuación mostrando los con las distintas pantallas llamada “templates”. Una de las cosas más complicadas fue crear ese **servidor donde se almacenarán los datos** así como conectar el servidor con cliente. Para ello se contó con la colección llamada “mongo” que está disponible tanto en servidor como en cliente para compartir los datos de medidas, estadísticas, partidos...

En este caso ponemos el ejemplo de la declaración de la colección medidas, que recoge los datos del GPS:

```
export const Medidas = new Mongo.Collection('medidas');
```

Para llevar al servidor la colección de datos “medidas” que entra en cliente usamos los controladores:

```
import { Settings, Medidas, Estadisticas, Partidos } from
'../../../../../lib/collections';
```

En el controlador se produce la lógica que opera con los datos.

Una de las operaciones más importantes es la de muestrear los datos que se recogen del GPS.

```
navigator.geolocation.getCurrentPosition((posicion)=>{
    var nuevaMedida = {
        latitud : posicion.coords.latitude,
        longitud : posicion.coords.longitude,
        altitud : posicion.coords.altitude,
        velocidad : posicion.coords.speed,
        timestamp : posicion.timestamp, }
}
```

Con esta función obtenemos en un tiempo timestamp determinado, la latitud, longitud, altitud y velocidad.

Avanzamos en el tiempo incrementando el timestamp desde la lógica de la carpeta lib en la que se encuentra “method”:

```
actualizarEstadisticas(nuevaMedida) {
    var ultimaMedida = Medidas.findOne({}, {sort: {timestamp:-1}})
    var estadisticas = Estadisticas.findOne();
    if (!ultimaMedida) {
        Estadisticas.update(estadisticas._id, {'$inc':
{'muestras': 1}});
        Medidas.insert(nuevaMedida);
        return;
    }
}
```


Y si utilizamos un bucle que utilice la fórmula de Haversine antes mencionada:

```
var calcularDistanciaRecorrida = function(nuevaMedida,
ultimaMedida) {
  var lat1 = ultimaMedida.latitud;
  var lat2 = nuevaMedida.latitud;

  var lon1 = ultimaMedida.longitud;
  var lon2 = nuevaMedida.longitud;

  var R = 6371e3; // metros
  var ï†1 = lat1.toRadians();
  var ï†2 = lat2.toRadians();
  var Î"ï† = (lat2-lat1).toRadians();
  var Î"Î» = (lon2-lon1).toRadians();

  var a = Math.sin(Î"ï†/2) * Math.sin(Î"ï†/2) +
Math.cos(ï†1) * Math.cos(ï†2) *
Math.sin(Î"Î»/2) * Math.sin(Î"Î»/2);
  var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));

  var d = R * c;

  return d;
}
```

Aplicando Haversine, muestreando cada segundo, se obtiene la distancia recorrida.

5 Test de la aplicación

Una vez conocidos los detalles de la aplicación, se realizaron una serie de pruebas para comprobar su funcionamiento en real. Se eligió un dispositivo y se realizaron pruebas de precisión y batería.

5.1 Dispositivo elegido para las pruebas y sus características

El dispositivo elegido para hacer las pruebas es el Samsung Galaxy Note 7 (SM-N930F). Este teléfono es el último presentado por el fabricante surcoreano que sigue la línea de los anteriores Note: incluye de las mejores especificaciones HW que hay en el mercado, pantalla grande y un lápiz llamado S-Pen para interactuar con la pantalla.

Características físicas

El terminal destaca por una pantalla de 5.7", curva por los dos laterales. Pese a ser una pantalla de grandes pulgadas, el cuerpo del teléfono no es muy voluminoso y tiene una relación pantalla-tamaño del teléfono muy destacable.

La resolución de la pantalla es de 2560 x 1440 pixels, por lo que con 5.7" se traduce en 518 pixels por pulgada.

A la hora de usar la aplicación una ventaja clara es su visionado, las letras, menús, transiciones y en general la interfaz es de mucha calidad. Sin embargo tener la pantalla encendida mientras se realiza el ejercicio puede penalizar la batería. Juega en contra también el tamaño del terminal, ajustarlo a un brazalete puede llegar a ser incómodo mientras se realiza el ejercicio.

Conectividad

El teléfono cuenta con conectividad 2G, 3G y 4G. La tecnología 4G es de categoría 9 por lo que podría alcanzar velocidades de descarga de datos de 450 Mbps y de subida de 50 Mbps. Esto sin duda sirve de ayuda cuando se utiliza A-GPS, ya que se conseguiría descargar más rápido la información de los servidores con las posiciones de los satélites y reducir el tiempo de primera conexión con los mismos.

Además el dispositivo cuenta con los sistemas de posicionamiento GPS, GLONASS y Beidou y además A-GPS. La aplicación de este proyecto se beneficia de estas conectividades de localización, y podría ser usada en cualquier parte del mundo.

Una conectividad muy destacable es ANT + que, unido a Bluetooth v4.2, nos permite conectar el teléfono a otros dispositivos como monitores de ritmo cardiaco, de glucosa en sangre, temperatura...El usar estos dispositivos e implementar su uso en la aplicación la haría muy completa.

Batería

Este es un punto vital a la hora de utilizar aplicaciones que hagan uso de la localización. El hecho de tener el teléfono conectado a los satélites conlleva el intercambio de mucha información y de esto modo un consumo elevado de la batería. El Samsung Galaxy Note 7 tiene una batería de 3500 miliamperios por hora. Es una batería con mucha capacidad aunque no hay que olvidar que tiene que alimentar una pantalla de gran tamaño y resolución.

5.2 Aplicaciones elegidas para la prueba

Las aplicaciones escogidas son las principales que hay de monitorización del ejercicio en la Play Store de Google. La mayoría tiene capacidad de registrar datos como velocidad media, velocidad máxima, distancia recorrida, calorías quemadas:

- **Runtastic:** una de las más populares y descargadas. Tiene la ventaja de que se pueden diferenciar distintos ejercicios como el buceo, crossfit, escalada, ski...
- **Endomondo:** muy parecida a la anterior. También permite interactuar con las redes sociales así como consultar rutas cercanas. No tiene versión de pago al contrario que Runtastic.
- **Nike + Running:** la aplicación de la famosa marca de ropa se basa exclusivamente en el ejercicio de correr y no soporta otros ejercicios. Además cuenta gratuitamente con un entrenador virtual de manera que el usuario se puede preparar para eventos como una carrera.
- **Sports Tracker:** esta es una de las aplicaciones más enfocadas a la búsqueda de actividades compartidas por otros usuarios. A parte de la monitorización que hacen las demás aplicaciones, esta ofrece un mapa intuitivo en el que se puede ver la localización y los detalles de los distintos ejercicios que han compartido otros usuarios de la aplicación.
- **S Health:** aplicación precargada en los Smartphone de gama media y alta de Samsung. Está más orientada a monitorizar la salud del usuario. Además de registrar pasos y carreras, controla el peso, la glucosa, la cantidad de agua que se bebe, la cafeína o las horas de sueño. Además se beneficia del sensor de ritmo cardiaco que incluyen algunos de los smartphones surcoreanos en los que con tan sólo colocar la yema del dedo en el sensor y con el cambio de pigmentación de la piel, es capaz de medir la frecuencia cardiaca o el nivel de estrés.

5.3 Prueba básica de la aplicación desarrollada jugando un partido de fútbol

5.3.1 Condiciones iniciales

Esta prueba fue la primera que se realizó tras tener la aplicación desarrollada. Con el móvil dentro de un brazalete anclado al brazo se jugó un partido de fútbol 7.

Los partidos de fútbol 7 lo juegan 14 jugadores en un campo de:

- Largo: 65 a 50 metros
- Ancho: 45 a 30 metros

El campo donde se realizó la prueba es de 50 metros de largo por 30 de ancho. La superficie es importante de cara a saber si beneficia o perjudica a la pisada, en este caso era césped sintético, con un corte no muy largo por lo que las carreras se podían realizar sin dificultad.

El tiempo total que del ejercicio fue de 60 minutos, dos partes de 30 minutos cada una y con 10 minutos de descanso.

5.3.2 Prueba, dinámica de utilización de la aplicación y resultados

Inicialmente se introdujo el nombre de los dos equipos en la pantalla “Equipos” de la aplicación para, posteriormente, pulsar en el botón “Empezar Partido” y comenzar con la monitorización del ejercicio.

Al acabar la primera parte, y por tanto transcurridos los primeros 30 minutos, se pausó la monitorización del ejercicio con el botón “Pause”. Tras 10 minutos el descanso finalizó y de nuevo se reanudó la aplicación pulsando en el botón “Play”.

Para finalizar el ejercicio tras 60 minutos de monitorización, se pulsó el botón “Terminar” y los datos quedaron de este modo registrados en la pantalla “Estadísticas”. Los resultados fueron:

- Distancia recorrida: 12.32 km
- Velocidad media medida: 8.60 km/h

5.3.3 Conclusiones

Tras los resultados, se concluye que la aplicación mide en exceso tanto la distancia recorrida como la velocidad media. Si la mayoría del partido se estuvo casi parado, la distancia media se muestra muy elevada. Igual ocurre con la distancia recorrida, la aplicación, al no usar el acelerómetro, sigue midiendo aunque el jugador esté parado. Esta medición lleva ya un error de precisión y no detecta que el jugador este quieto, por lo que se considera excesiva.

Sin duda alguna la logística de la prueba no es del todo ideal. Jugar un partido con un móvil anclado al brazo es incómodo e ilegal en el fútbol profesional.

Sin embargo la aplicación puede ser muy beneficiosa para aficionados que quieran monitorizar sus ejercicios. Además para el árbitro es una buena manera de cuantificar su ejercicio además de controlar el marcador y tiempo del partido.

Una mejora clara de la aplicación en el futuro es el diseño de unos módulos funcionales para los árbitros: registros de tarjetas amarillas y rojas, alineaciones con nombres y números, sección de anotaciones, actas de partidos...

5.4 Prueba de precisión frente a la competencia

Para probar la precisión de la aplicación, se decidió analizarla junto a otras aplicaciones que monitorizan el ejercicio físico, se decidió comprobarlo en una distancia de entre 7 y 8 kilómetros y en una distancia de 100 metros.

5.4.1 Prueba de la aplicación en un recorrido medio-largo de 7-8 km

La prueba que se realizó fue una carrera por asfalto. El móvil, como se mencionaba en el análisis anteriormente, fue un Samsung Galaxy Note 7 anclado al brazo con un brazaletes.

Se abrieron las cinco aplicaciones y sin movimiento se iniciaron todas, al no haber movimiento, el tiempo de inicio entre unas y otras no debería de ser importante a la hora de hacer cálculos como la distancia recorrida o la velocidad media. Lo mismo se aplica para el final del ejercicio, es decir, todas las aplicaciones se iniciaron y se pararon en los mismos puntos.

El recorrido que se realizó fue el siguiente:



Ilustración 35: Recorrido realizado

Distancia teórica: 7.28 km.

La herramienta usada para calcular la distancia recorrida fue Google maps.

A continuación se muestran los resultados de las distintas aplicaciones al acabar el ejercicio:

Estudio de la distancia recorrida

APLICACIÓN	DISTANCIA
Runtastic	7.03 km
Endomondo	7.41 km
Nike + Running	7.88 km
Sports Tracker	7.62 km
S Health	7.81 km

Tabla 1: Distancia medida por otras aplicaciones en 7.28km

Como se aprecia, usando el mismo dispositivo y en igualdad de condiciones, las aplicaciones devolvieron unos resultados muy dispares. La diferencia de resultados entre distintas aplicaciones se analizará más adelante.

La aplicación desarrollada en este proyecto dio como resultado una distancia recorrida de:

Aplicación PFC desarrollada	7.85 km
-----------------------------	---------

Tabla 2: Distancia medida por la aplicación desarrollada en 7.28km

Analizando estos resultados tenemos:

DISTANCIA MEDIA ENTRE LAS APPS USADAS	7.60 km
DESVIACIÓN MÁXIMA	0.85 km
% DESVIACIÓN MÁXIMA	11.18%

Tabla 3: Desviación máxima de la distancia recorrida en 7.28km

Se observa que para la misma distancia recorrida, el mismo dispositivo y las mismas condiciones, se obtiene una desviación máxima de 11.18%. Al hablar de distancias medias como son 7.60 km de distancia media medida, es bastante considerable. Sabemos cómo está la aplicación de este proyecto desarrollada pero desconocemos como registran datos las demás aplicaciones.

Centrándonos en la aplicación desarrollada, arrojó los siguientes datos:

DATOS DE LA APP PFC DESARROLLADA	
Distancia recorrida medida en la app PFC desarrollada	7.85 km
Distancia media entre las apps usadas	7.60 km
Desviación de la aplicación desarrollada vs la media	0.25 km
% Desviación de la aplicación desarrollada vs la media	3.28%

Tabla 4: Desviación en la distancia recorrida de la aplicación desarrollada en 7.28km

Estudiaremos más adelante el porqué de la desviación de las distintas aplicaciones.

Estudio de la velocidad media

APLICACIÓN	VELOCIDAD MEDIA
Runtastic	9.8 km/h
Endomondo	9.82 km/h
Nike + Running	No soportado
Sports Tracker	10.50 km/h
S Health	10.2 km/h

Tabla 5: Velocidad media medida por otras aplicaciones en 7.28km

En el caso de la velocidad media, la aplicación de Nike no muestra el resultado para informar al usuario.

La aplicación desarrollada en este proyecto dio como resultado una velocidad media de:

Aplicación PFC desarrollada	10.63 km/h
------------------------------------	------------

Tabla 6: Velocidad media medida por la aplicación desarrollada en 7.28km

Analizando estos resultados tenemos:

VELOCIDAD MEDIA ENTRE LAS APPS USADAS	10.19 km/h
DESVIACIÓN MÁXIMA	0.83 km/h
% DESVIACIÓN MÁXIMA	8.14%

Tabla 7: Desviación máxima de la velocidad media medida en 7.28km

En esta ocasión la desviación que se observa en la medición de la velocidad media es menor que en la de la distancia recorrida.

Centrándonos en la aplicación desarrollada, arrojó los siguientes datos:

DATOS DE LA APP PFC DESARROLLADA	
Velocidad media	10.63 km/h
Velocidad media entre las apps usadas	10.19 km/h
Desviación de la aplicación desarrollada vs la media	0.44 km/h
% Desviación de la aplicación desarrollada vs la media	3.28%

Tabla 8: Desviación en la velocidad media medida en la aplicación desarrollada en 7.28km

5.4.2 Prueba de la aplicación en un recorrido corto de 100 metros

Además de recorridos de larga distancia, la aplicación también se probó en recorridas de corta distancia. En este caso, se realizó un recorrido de 100 metros.

Distancia teórica: 100 metros.

Antes de hacer el recorrido, se contaba ya con el conocimiento de que la aplicación desarrollada en este proyecto no hacía uso del acelerómetro, en cambio las demás aplicaciones ya están optimizadas y cuentan con mayor tiempo de vida, lo que significa que han tenido mucho más tiempo de depuración de errores y mejoras de precisión.

Estudio de la distancia recorrida

APLICACIÓN	DISTANCIA
Runtastic	0.10 km
Endomondo	0.11 km
Nike + Running	0.13 km
Sports Tracker	0.10 km
S Health	0.12 km

Tabla 9: Distancia medida por otras aplicaciones en un recorrido teórico de 100 m

Como se aprecia, igual que para recorridos de largas distancias analizado en el apartado anterior, las aplicaciones devolvieron unos resultados dispares.

La aplicación desarrollada en este proyecto dio como resultado una distancia recorrida de:

Aplicación PFC desarrollada	0.34 km
-----------------------------	---------

Tabla 10: Distancia medida por la aplicación desarrollada en 100m

Si analizamos los resultados:

DISTANCIA MEDIA ENTRE LAS APPS USADAS	0.15 km
DESVIACIÓN MÁXIMA	0.24 km
% DESVIACIÓN MÁXIMA	160%

Tabla 11: Desviación máxima de la distancia recorrida en 100m

Centrándonos en la aplicación desarrollada, arrojó los siguientes datos:

DATOS DE LA APP PFC DESARROLLADA	
Distancia recorrida medida en la app PFC desarrollada	0.34 km
Distancia media entre las apps usadas	0.15 km
Desviación de la aplicación desarrollada vs la media	0.20 km
% Desviación de la aplicación desarrollada vs la media	133.33%

Tabla 12: Desviación en la distancia recorrida de la aplicación desarrollada en 100m

Claramente resalta cómo a distancias muy cortas la aplicación desarrollada no tiene buena respuesta. Las aplicaciones de monitorización de ejercicio suelen tener un mínimo de tiempo o distancia para dejar registrado el ejercicio, es decir, hay que andar un mínimo de 100 metros por ejemplo, o tienen que pasar 30 segundos para que se dé el ejercicio como iniciado para no dar lugar a errores o posibles pulsaciones erróneas por parte del usuario. La sincronización entre la precisión y los puntos de localización es crítica a la hora de medir distancias cortas. Si además se le añade el proceso de primera conexión con los satélites, es complicado medir distancias cortas, hay que optimizar mucho la aplicación.

Estudio de la velocidad media

APLICACIÓN	VELOCIDAD MEDIA
Runtastic	3.6 km/h
Endomondo	4.4 km/h
Nike + Running	No soportado
Sports Tracker	3.8 km/h
S Health	5 km/h

Tabla 13: Velocidad media medida por otras aplicaciones en 100m

En el caso de la velocidad media, la aplicación de Nike no muestra el resultado para informar al usuario.

La aplicación desarrollada en este proyecto dio como resultado una velocidad media de:

Aplicación PFC desarrollada	6.2 km/h
-----------------------------	----------

Tabla 14: Velocidad media medida por la aplicación desarrollada en 100m

Analizando estos resultados tenemos:

VELOCIDAD MEDIA ENTRE LAS APPS USADAS	4.6 km/h
DESVIACIÓN MÁXIMO-MÍNIMO	2.6 km/h
% DESVIACIÓN MÁXIMA	56.52%

Tabla 15: Desviación máxima de la velocidad medida en 100m

De nuevo la desviación que se observa en la medición de la velocidad media es menor que en la de la distancia recorrida.

Centrándonos en la aplicación desarrollada, arrojó los siguientes datos:

DATOS DE LA APP PFC DESARROLLADA	
Velocidad media	6.2 km/h
Velocidad media entre las apps usadas	4.6 km/h
Desviación de la aplicación desarrollada vs la media	1.6 km/h
% Desviación de la aplicación desarrollada vs la media	34.78%

Tabla 16: Desviación en la velocidad media medida en la aplicación desarrollada en 100m

5.4.3 Conclusión de las pruebas de precisión realizadas

Como se ha desarrollado esta aplicación, se sabe que va registrando cada segundo la posición del dispositivo. Esto genera tantos **puntos de localización** como tiempo esté la aplicación activada en el modo Partido. Sin embargo, de las otras aplicaciones desconocemos los criterios que implementan para el registro de puntos de localización y el procesado de los datos. El estudio de la obtención de estos puntos es clave a la hora de entender las diferencias tan marcadas entre las distintas aplicaciones de monitorización del ejercicio:

1	lon	lat	ns1:ele
2	-3,639881849	40,52536011	732
3	-3,639904737	40,52547073	778
4	-3,639957666	40,5255127	775
5	-3,63999939	40,52546692	773
6	-3,63999939	40,52546692	773
7	-3,640055895	40,52539063	763
8	-3,64015007	40,5253334	768
9	-3,640203238	40,52529144	768
10	-3,640203238	40,52529144	768
11	-3,640203238	40,52529144	768
12	-3,640203238	40,52529144	768
13	-3,640253544	40,5252037	765
14	-3,640343428	40,52515793	765
15	-3,640404701	40,52511215	764
16	-3,640496731	40,52507019	764
17	-3,640557289	40,52501297	763
18	-3,640557289	40,52501297	763
19	-3,640485764	40,52503586	761
20	-3,639531136	40,526474	751
21	-3,639653683	40,52664185	743
22	-3,639731169	40,52664185	744
23	-3,639731169	40,52664185	744
24	-3,639787912	40,52658844	748
25	-3,639787912	40,52658844	748

Ilustración 36: Excel con puntos de localización

Como indica la ilustración de la página anterior, las aplicaciones que hacen uso del GPS registran los datos encapsulados en etiquetas XML. Los datos que guardan son la hora, latitud, longitud y altura, entre otros. En la mayoría de las aplicaciones es posible acceder a esta información que se presenta en un archivo con formato *.gpx*, este tipo de archivos son muy útiles para la transferencia de datos de localización entre aplicaciones.

Si ponemos el ejemplo de una rotonda la cual se recorre de manera circular, algunas aplicaciones dibujarían el movimiento circular sobre el mapa si registran muchos puntos de localización (recorrido rojo en la ilustración que se muestra a continuación) y otras que registran menos puntos dibujarían un recorrido más rectilíneo (en verde):



Ilustración 37: Diferencia entre tomar más o menos puntos de localización [39]

No sólo influye el número de puntos que vaya registrando cada aplicación sino que además entra en juego la **precisión con la que se tomen estos puntos de localización**.

Para muestra, este es el recorrido que han mostrado algunas de las aplicaciones durante la prueba:

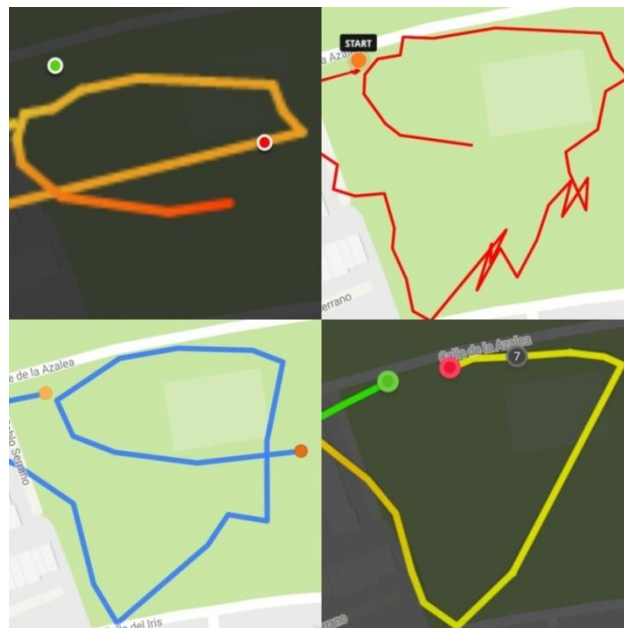


Ilustración 38: Diferencia de trazados registrados en distintas aplicaciones

Como se puede apreciar, pese a hacer el mismo recorrido, las aplicaciones muestran sendas muy distintas, esto se ve reflejado claramente en los cálculos que se realizan.

Si a la diferencia de registro de puntos de localización y a la diferencia de precisión añadimos la distancia total recorrida, el error se va propagando y las diferencias aumentando.

5.5 Prueba de batería frente a la competencia

5.5.1 Consumo de batería en 2 recorridos distintos

Tener el GPS activado, es una de las funcionalidades que más consume batería en los dispositivos móviles. Además de activar el GPS, muchas aplicaciones incluyen funcionalidades extra que las hacen ser más completas pero que también penalizan el consumo de batería:

- Tener la pantalla encendida.
- Emisión de mensajes de audio para informar al usuario de los detalles del ejercicio que está realizando.
- Estar conectado a Bluetooth
- Tener los datos activados
- Reproducción de música

Para cada aplicación, antes de realizar el ejercicio, el dispositivo se había reiniciado previamente y se habían cerrado las aplicaciones abiertas, sólo los procesos del sistema y la aplicación correspondiente estaban corriendo.

Se analizó la batería en dos recorridos:

- Recorrido de alrededor de **7.5 kilómetros en 43 minutos** analizado en el apartado 4.3. Por el largo del recorrido, se analizaron sólo 2 aplicaciones, por lo que el recorrido se realizó 2 veces, cada uno con una sola aplicación activada. El consumo de batería que arrojaron cada una de las dos aplicaciones fue de:

DISTANCIA DE 7.5 KM – TIEMPO 43 MINUTOS	
APLICACIÓN	BATERÍA CONSUMIDA
Runtastic	16%
Aplicación PFC desarrollada	15%

Tabla 17: Análisis de consumo de batería en 43 minutos

- Recorrido de **1 kilómetro en 12 minutos** con las distintas aplicaciones. Este recorrido se realizó 6 veces, cada vez con una sola aplicación activada para medir el consumo individual:

DISTANCIA DE 1 KM – TIEMPO 12 MINUTOS	
APLICACIÓN	BATERÍA CONSUMIDA
Runtastic	2%
Endomondo	2%
Nike + Running	2%
Sports Tracker	2%
S Health	2%
Aplicación PFC desarrollada	2%

Tabla 18: Análisis de consumo de batería en 12 minutos

5.5.2 Conclusiones de las pruebas de batería realizadas

CONSUMO DE BATERÍA CADA 12 MINUTOS DE LA APLICACIÓN PFC DESARROLLADA	2%
% DESVIACIÓN DE LA APLICACIÓN DESARROLLADA RESPECTO A LAS DEMÁS APLICACIONES	0%

Tabla 19: Consumo medio de batería de la aplicación desarrollada

De estos resultados se observa que por cada 12 minutos, todas las aplicaciones consumen alrededor de un 2% de la batería del dispositivo elegido. La aplicación desarrollada en este proyecto tiene el mismo consumo de batería que las demás aplicaciones de su mismo estilo.

Junto con la pantalla, tener funcionando el GPS, contactándose así con los satélites e intercambiando mensajes, es de las funcionalidades de un dispositivo móvil que más consume batería. Si se desea realizar un ejercicio de larga distancia como en el primer recorrido de 45 minutos, supone que la batería se drene demasiado y la correspondiente insatisfacción en el usuario. En menos de una hora se ha perdido más del 15% de la batería total.

La mano del desarrollador de la aplicación es crítica en el comportamiento de la batería: requerir puntos de localización cada poco tiempo, un error que complique la aplicación y drene la batería, incompatibilidad con algún proceso del sistema operativo o con otras

aplicaciones...Sin embargo el propio proceso de localización supone un gran gasto de batería, hay que alimentar al receptor GPS, esto supone un gran gasto ya que la comunicación entre los satélites y estos receptores es de sólo 50 bits por segundo por lo que el tiempo de suministro debe de ser mayor. Es peor incluso el drenaje de batería al intentar la primera conexión con los satélites, de ahí que la implementación del A-GPS sea una gran mejora ya que en vez de GPS, los datos los proporcionarían la conexión de datos o de WiFi.

Hay que puntualizar que tener activo el GPS en los dispositivos móviles, no significa que se esté haciendo uso del mismo por lo que no consume batería. Al activarlo damos permiso a que el dispositivo utilice el GPS para conseguir localizarnos en el momento en el que alguna aplicación lo solicite o sea necesario.

6 Conclusiones del proyecto y trabajo futuro

6.1 Conclusiones

El resultado final ha sido muy positivo, se ha aprovechado las últimas tecnologías disponibles hoy en día para desarrollar una aplicación muy funcional.

1. Se ha hecho uso de herramientas de trabajo que aportan gran valor al proyecto (Meteor, Angular, Ionic y Cordova), se han buscado soluciones eficientes y se acabado usando un conglomerado de herramientas que sobre todo permiten la programación multiplataforma. A partir de los conocimientos obtenidos en este proyecto se consigue programar en dos de los sistemas operativos más usados en el mundo (Android e iOS) así como en entorno web. Esto quiere decir que se tienen la capacidad para desarrollar un proyecto personal a gran escala o de terceros y poder adaptar el trabajo para que sea global y accesible por todos.
2. Ya se tenían conocimientos de programación de anteriores años pero, sin duda, esta aplicación ha servido para aprender Javascript, consolidar conceptos y aprender algunos nuevos.
3. Sin duda la aplicación ha servido para entender de geolocalización y cartografía digital. El funcionamiento del GPS tiene pocos misterios una vez realizado este proyecto, y ha resultado muy interesante saber su funcionamiento a tan bajo nivel. Algo tan común que usamos día a día, es muy gratificante saber cómo funciona y entenderé por qué en ocasiones tarda en posicionarte.
4. Se ha probado la aplicación, los resultados han sido relativamente buenos si pensamos en que la mayoría de los usuarios realizarán ejercicios de largo recorrido. Al igual que otras aplicaciones, esta aplicación no es muy precisa en cortas distancias.

Como se proponía se ha hecho hincapié en 3 aspectos:

- Facilidad para ampliar la aplicación de manera paramétrica: tiene un futuro prometedor y sobre todo fácil de implementar nuevos módulos y funcionalidades.
- Robustez de la misma respecto a versiones de sistema operativo, modelos de terminales, resoluciones y tamaños de pantalla: no sólo eso, si no que se ha conseguido que sea compatible con distintos sistemas operativos.

- Pruebas de campo para verificar su funcionamiento: se realizaron satisfactoriamente con un consumo de batería estándar para aplicaciones de seguimiento de ejercicio.

Las principales funcionalidades y tareas que se han desarrollado en la aplicación son:

- Introducción de equipos por pantalla
- Posibilidad de añadir goles al partido
- Reloj que se puede pausar y reanudar
- Utilización del GPS y tratamiento de los datos para dar en tiempo real datos de la velocidad media y de la distancia recorrida
- Guardado de la velocidad media y la distancia recorrida total en base de datos
- Mostrar los datos guardados por pantalla

Desde el punto de vista formativo, se han aprendido conceptos como:

- Javascript
- Meteor
- Angular
- Ionic
- Cordova
- Geolocalización y cartografía
- Diseño visual
- HTML5
- MongoDB

6.2 Trabajo futuro

A lo largo del diseño del proyecto saltaron muchas ideas de complementos que añadir a la aplicación. Tiene muchos puntos de mejora pero también mucho potencial.

Algunas de las ideas que surgieron durante la realización del proyecto para desarrollar en el futuro son:

- Adaptarla a varios deportes y no sólo al fútbol.
- Incluir la fecha de los partidos para guardarla en la base de datos
- Calcular las Calorías, Ritmo promedio, temperatura, tiempo, estado físico...
- Añadir un mapa para ver por dónde se ha realizado la actividad
- Añadir más calentamientos para antes del ejercicio y estiramientos para después.
- Incluir un temporizador para cada ejercicio del entrenamiento.
- Posibilidad de modificar la base de datos de la pestaña “Estadísticas”
- Poder mandar el ejercicio por email
- Crear una red social de manera que todos los que tengan la aplicación puedan compartir sus ejercicios y que se integre sus movimientos con los registrados en tu dispositivo cuando se realice el ejercicio a la vez.

- Incluir compatibilidad con el acelerómetro de manera que el GPS sea más preciso.
- Incluir una pizarra virtual para poder mostrar jugadas o apuntes.

Referencias

- [1] Meteor website oficial, www.meteor.com
- [2] Angular website oficial, www.angularjs.com
- [3] Cordova website oficial, www.cordova.apache.org
- [4] Ionic website oficial, <http://ionicframework.com/>
- [5] <http://stackoverflow.com/>
- [6] <http://www.genbetadev.com/>
- [7] <http://www.alsitel.com/tecnico/gps/errores.htm>
- [8] Norbert Márkus, András Arató, Zoltán Juhász, Gábor Bognár, and László Késmárki, MOST-NNG: An Accessible GPS Navigation Application Integrated into the MOBILE SLATE TALKER (MOST) for the Blind, 2010
- [9] <https://movilfacil.files.wordpress.com>
- [10] www.researchgate.net
- [11] www.pocketgpsworld.com%2Fhowgpsworks.php&psig=AFQjCNHbM9I2IMt2FGUjV0Utze-kIDxR8Q&ust=1468515209595175
- [12] <http://www.sxbluegps.com/wp-content/uploads/2012/12/gps-triangulation.jpg>
- [13] Android Zone, “Historia de Android: la Evolución a lo largo de sus versiones”, <http://androidzone.org/wp-content/uploads/2013/05/Historia-de-Android-y-sus-versiones1.jpg>
- [14] Jose Antonio E García Álvarez, “Así funciona el GPS”, http://www.asifunciona.com/electronica/af_gps/af_gps_9.htm
- [15] <http://blog.twonav.es/>
- [16] <https://github.com/meteor/meteor/wiki/Meteor-Cordova-Phonegap-integration>
- [17] <https://www.discovermeteor.com/blog/meteor-cordova-famous-the-chill-way-to-build-apps/>
- [18] <http://ngcordova.com/>
- [19] José Jesús Pérez Rivas, “Qué es y cómo empèzar con Ionic framework”, <http://www.phonegapspain.com/que-es-y-como-empezar-con-ionic-framework/>
- [20] <https://www.angular-meteor.com/tutorials/socially>
- [21] <https://www.codementor.io/meteor/tutorial/why-you-should-use-angular-meteor-library>
- [22] Willy Tut Si, “GPS, Tipos y usos”, http://www.academia.edu/5248668/GPS_Tipos_y_usos
- [23] <https://www.agroptima.com/wp-content/uploads/2015/12/gps-agricola-beneficios-para-el-agricultor.jpg>
- [24] <http://unwire.pro/2016/01/07/android-marshmallow-market-share-less-than-1-percent/news/>
- [25] <http://www.gsmarena.com/>
- [26] https://movilfacil.files.wordpress.com/2011/03/cell_id_ta.jpg
- [27] https://movilfacil.files.wordpress.com/2011/03/cell_id.jpg
- [28] www.researchgate.net/publication/258390460/figure/fig9/AS:328441036591115@1455317854047/Angle-of-arrival-positioning-method.png

- [29] <https://ai2-s2-public.s3.amazonaws.com/figures/2016-03-25/a7016075b373b572b3ced9f6712f0d565c608c7c/1-Figure1-1.png>
- [30] www.pocketgpsworld.com%2Fhowgpsworks.php&psig=AFQjCNHbM9I2IMt2FGujV0Utze-kIDxR8Q&ust=1468515209595175
- [31] <http://www.sxbluegps.com/wp-content/uploads/2012/12/gps-triangulation.jpg>
- [32] <http://www.arcxsites.shh.net/LGG2GPS1.jpg>
- [33] <https://www.agroptima.com/wp-content/uploads/2015/12/gps-agricola-beneficios-para-el-agricultor.jpg>
- [34] <https://upload.wikimedia.org/wikipedia/commons/thumb/3/38/Law-of-haversines.svg/220px-Law-of-haversines.svg.png>
- [35] <http://www.imore.com/sites/imore.com/files/styles/medium/public/field/image/2015/10/ios-9-midoctober-adoption.png?itok=5WTvcPNr>
- [36] <http://blog.ticsandroll.es/ios-y-su-arquitectura-interna-en-4-capas/>
- [37] <http://www.androidauthority.com/google-play-store-vs-the-apple-app-store-601836/>
- [38] <http://encorda2.com/2012/11/05/orientacion-basica-la-tierra-estrella-polar-y-la-cruz-del-sur/>
- [39] <http://blog.zitasport.com/2591/2014/10/09/6-apps-la-vez-la-explicacion/>
- [40] https://es.wikipedia.org/wiki/F%C3%B3rmulas_de_Vincenty

Glosario

API: Application Programming Interface

A-GPS: Assisted Global Position System

ART: Android Run Time

GPS: Global Positioning System

DsLab: Digital System Laboratory

HTML: HyperText Markup Language

KM: Kilómetro

OHA: Open Handset Alliance

PDF: Portable Document Format

PFC: Proyecto Fin de Carrera

SDK: Software Development Kit

SQL: Structured Query Language

SO: Sistema Operativo

UAM: Universidad Autónoma de Madrid

XML: eXtensible Markup Language

Anexos

PRESUPUESTO

1) EJECUCIÓN MATERIAL DURANTE 6 MESES DE PROYECTO	
Amortización de ordenador y SW.....	92 €
Móvil Android.....	42 €
Tablet Android.....	34 €
Material de oficina.....	50 €
Total de ejecución material.....	218 €
<small>*2 años de amortización</small>	
2) GASTOS GENERALES	
16 % sobre la Ejecución material.....	34.88 €
3) BENEFICIO INDUSTRIAL	
6% sobre la Ejecución Material.....	13.08 €
4) HONORARIOS DEL PROYECTO	
500 horas a 15€/hora.....	7500 €
5) MATERIAL FUNGIBLE	
Gastos de impresión.....	55 €
Encuadernación.....	40 €
6) SUBTOTAL DEL PRESUPUESTO	
Subtotal del presupuesto.....	7860.46 €
7) I.V.A aplicable	
21% del Subtotal del Presupuesto.....	1650.70 €
8) Total del presupuesto	
Total del Presupuesto.....	9511.16 €

Madrid, Septiembre de 2016
El Ingeniero Jefe de Proyecto
Fdo.: Sergio Zurita Eiranova
Ingeniero de Telecomunicación

PLIEGO DE CONDICIONES

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un ESTUDIO SOBRE LA UTILIZACIÓN DEL GPS DE TELÉFONOS INTELIGENTES PARA EL ANÁLISIS DE DEPORTE AL AIRE LIBRE. 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 partida alzada 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.

