

GeoTrainer

*Software Geográfico de Entrenamiento y
Exploración para Dispositivos Móviles*



Informe Final

Agosto de 2010

Instituto de Computación
Facultad de Ingeniería
Universidad de la República

Integrantes

*Massimiliano Menestrina
Pablo Costa Possamai*

Tutor

Ing. Sandro Moscatelli

Usuario responsable

Ing. Pablo Rebufello

Resumen

Este documento presenta, en el marco de un proyecto de grado de la carrera de Ingeniería en Computación, el desarrollo de **GeoTrainer**, un **aplicativo geográfico orientado a dispositivos móviles** para asistir a los usuarios en la realización y entrenamiento de actividades deportivas tales como *caminar, correr, ciclismo*, etc.

GeoTrainer permite el registro y almacenamiento del recorrido realizado y de los indicadores de performance (distancia, tiempo, velocidad, velocidad promedio) asociados a una actividad deportiva. Estos datos posteriormente pueden ser evaluados a través del mapa del recorrido y graficas de la evolución de los distintos indicadores a lo largo de la actividad.

Como sistema de asistencia en el entrenamiento, **GeoTrainer** ofrece la posibilidad de confeccionar una sesión de entrenamiento eligiendo un recorrido a seguir y ajustar el tiempo objetivo. Esta combinación permite al aplicativo brindar durante la realización de la actividad *feedback* al usuario acerca de su desempeño.

GeoTrainer permite la participación del usuario en competencias deportivas donde múltiples usuarios (los competidores) sincronizan su desempeño en tiempo real. El usuario tendrá entonces la posibilidad de visualizar la posición del resto de los competidores tanto en el mapa como en un listado de posiciones.

GeoTrainer presenta a su vez características de una *red social*, los usuarios podrán solicitar *amistad* a otros usuarios, los cuales podrán aceptar o rechazar dicha solicitud. Estos amigos serán luego los usuarios competidores que se visualizaran durante una competencia.

Uno de los requerimientos del proyecto era que el producto desarrollado debía contar con una interfaz de usuario (*GUI*¹) amigable e intuitiva. Gran parte del esfuerzo del equipo de proyecto estuvo dedicado a lograr un producto gráficamente atractivo, que aplicara apropiadamente conceptos de *interacción persona computador* o *HCI*² por su sigla en inglés.

GeoTrainer está destinado a dispositivos que cuentan con *Windows Mobile 6.x* como sistema operativo, con pantalla táctil (*touchscreen*) y fue desarrollado en lenguaje *C#* sobre la plataforma *Microsoft.NET*. Se utilizan además componentes de terceras partes, en particular las bibliotecas geográficas *ArcGIS Mobile* de la empresa *ESRI*³ [1].

Debido a las funcionalidades que brinda, el aplicativo está dirigido a dispositivos que cuenten con tecnología de posicionamiento geográfico *GPS*⁴ y con capacidades de conexión a internet.

Palabras clave

Software geográfico, Dispositivos móviles, GPS, Redes sociales.

¹ GUI - Graphical User Interface

² HCI - Human Computer Interaction

³ ESRI - Enviromental Systems Research Institute, empresa fundada en 1969 que actualmente desarrolla y comercializa software para Sistemas de Información Geográfica y es una de las compañías líderes en el sector a nivel mundial.

⁴ GPS – Global Positioning System

Contenido

1	Introducción	5
1.1	Antecedentes y contexto	5
1.2	Motivación	5
1.3	Definición del problema	6
1.4	Objetivos y resultados esperados	7
1.5	Conclusiones.....	7
1.6	Organización del informe	8
2	Estado del arte	9
2.1	Introducción	9
2.2	La telefonía celular y los dispositivos móviles.....	10
2.2.1	Las generaciones de la telefonía móvil	10
2.2.2	Hardware de dispositivos.....	12
2.2.3	Sistemas Operativos	14
2.3	Redes sociales	15
2.4	Desarrollo de aplicaciones	15
2.4.1	Plataformas, lenguajes de programación y entornos de desarrollo.....	15
2.4.2	Servicios y componentes de software disponibles	16
2.5	Distribución y comercialización.....	19
2.6	Aplicaciones relacionadas	20
3	Análisis y especificación de requerimientos	22
3.1	Introducción	22
3.2	Descripción general del producto y sus funcionalidades.....	22
3.2.1	Consideraciones adicionales	23
3.3	Esquema general del sistema.....	23
3.4	Especificación de requerimientos	24
3.4.1	Requerimientos funcionales.....	24
3.4.2	Requerimientos no funcionales	27
4	Diseño del sistema	28
4.1	Arquitectura general de la aplicación	28
4.2	GeoTrainer.....	28
4.2.1	Capa 1: Interfaz de usuario	28
4.2.2	Capa 2: Lógica de la aplicación.....	28
4.2.3	Capa 3: Persistencia y Comunicaciones	29

4.3	Servidor Central de GeoTrainer	29
4.3.1	GeoDatabase	29
4.3.2	WebServices: GeoTrainerWS	30
4.3.3	Servicios geográficos	31
5	Implementación	32
5.1	Infraestructura de ejecución y desarrollo	32
5.2	Lenguaje de programación y entorno de desarrollo	33
5.3	Estructura de la solución	33
5.4	GeoTrainer.....	34
5.4.1	Capas & Namespaces	34
5.4.2	Capa 1: Interfaz de Usuario	35
5.4.3	Capa 2: Lógica de la aplicación	39
5.4.4	Capa 3: Persistencia y Comunicaciones	39
5.5	GeoTrainerWS	40
5.5.1	Capa 1: Interfaz de comunicación	40
5.5.2	Capa 2: Lógica de los servicios	40
5.5.3	Capa 3: Acceso a datos.....	41
5.6	GeoTrainerDataTypes.....	41
5.7	GeoTrainerSetup	42
6	Testeo y verificación.....	43
6.1	Testeo	43
6.2	Verificación.....	44
7	Conclusiones y trabajo futuro	45
7.1	Conclusiones.....	45
7.1.1	Duración y contratiempos.....	45
7.1.2	Ajustes del alcance	46
7.1.3	Interfaz de usuario	47
7.1.4	Feedback de ICA.....	47
7.1.5	Presentación final.....	48
7.1.6	Objetivos alcanzados.....	49
7.2	Trabajo futuro	51
8	Referencias.....	53

1 Introducción

1.1 Antecedentes y contexto

Este proyecto de grado fue propuesto al *Instituto de Computación de la Facultad de Ingeniería* por la empresa **Ingenieros Consultores Asociados** [2] y consiste en la creación de un aplicativo de software geográfico orientado a dispositivos móviles.

Ingenieros Consultores Asociados (ICA) es una empresa nacional que se dedica principalmente al desarrollo y venta de soluciones basadas en *sistemas de información geográfica (SIG)* y tecnologías asociadas tales como los *sistemas de posicionamiento global* [3] (del inglés **GPS**). Además investiga y desarrolla soluciones en el área de Optimización en temáticas variadas como el ruteo de vehículos y logística de recolección y distribución de mercaderías, etc.

1.2 Motivación

Es notorio el avance que tanto a nivel de hardware como de software han tenido los distintos artefactos electrónicos que forman parte de nuestro diario vivir. Dentro de éstos, se destaca un segmento, el de los denominados dispositivos móviles y en particular los teléfonos celulares debido a la masiva aceptación y uso por parte del público.

Por otro lado, productos dentro de las áreas de trabajo de ICA como ser los SIG han dejado de ser de interés únicamente académico, empresarial o gubernamental para transformarse en productos de uso masivo. Hoy en día es común utilizar funcionalidades de SIG, como ser un mapa, la ubicación de puntos de interés o el cálculo de rutas entre dos puntos en las más variadas aplicaciones y páginas web, siendo un claro ejemplo de esto el aplicativo *Google Earth* [4].

También es interesante ver como cada vez más gente se vuelca a la práctica de deportes al aire libre como forma de llevar adelante una vida más saludable. Basta con observar la creciente concurrencia de personas en la rambla de nuestra ciudad para caminar, correr o andar en bicicleta o ver la convocatoria que año a año generan las competencias de $\frac{1}{4}$ de maratón que se llevan a cabo.

Basándose en lo anterior, ICA se plantea en el marco de un proyecto de grado, el desarrollo de **GeoTrainer**, una aplicación de naturaleza geográfica para dispositivos móviles, orientada al público en general, que permita asistir a los usuarios en la práctica de actividades deportivas como las anteriormente detalladas.

1.3 Definición del problema

GeoTrainer es un aplicativo de naturaleza geográfica orientado a la práctica de deportes y brinda una serie de funcionalidades que lo destacan de otros aplicativos existentes con similar finalidad.

GeoTrainer tiene las siguientes características:

- Permite la captura y el registro del recorrido que efectúe el usuario durante el desarrollo de la actividad.
- El mismo es visualizado junto con la posición del usuario sobre un mapa interactivo, es decir, permite la interacción con el mapa a través de herramientas como acercar, alejar, desplazar, centrar la vista sobre la posición del usuario, etc.
- Brinda al usuario información visual de *feedback* acerca de su desempeño a través de indicadores como ser el tiempo total de actividad, distancia recorrida, velocidad actual y velocidad promedio.
- Permite la realización de una actividad deportiva usando como referencia datos de otra ya existente, visualizando el trayecto a seguir en el mapa durante la actividad y simulando un *competidor virtual* con los datos de tiempos asociados a la misma. Esta modalidad se denomina **entrenamiento**.
- Varios usuarios pueden compartir un trayecto a seguir y sincronizar su posición en tiempo real durante la realización de una actividad. Esta modalidad se denomina **competencia**.
- Permite pausar y reanudar la actividad que el usuario se encuentra desarrollando, manteniendo consistentes los datos.
- Una vez finalizada una actividad los datos pueden ser almacenados en el dispositivo para su posterior consulta.
- Los datos anteriores también pueden ser **publicados** en un repositorio central denominado **Servidor Central de GeoTrainer** pudiendo ser consultados por parte de otros usuarios del aplicativo.
- Brinda herramientas de **búsqueda** de actividades tanto almacenadas en el dispositivo (actividades desarrolladas por el usuario) como publicadas en el *servidor central* (actividades propias o de otros usuarios).
- Los datos correspondientes a una actividad pueden ser visualizados mediante mapas y gráficas que permiten su evaluación.
- Permite la comparación de dos actividades una vez finalizadas.
- Permite la creación de trayectos de forma manual interactuando con el mapa.
- Ofrece al usuario una interfaz grafica amigable.

1.4 Objetivos y resultados esperados

Si bien la implementación de **GeoTrainer** es el principal objetivo buscado por el proyecto, ICA pretende como resultado directo del proceso de investigación a llevar a cabo, generar una base de conocimiento sobre los siguientes puntos:

- Usar las componentes de software *ArcGIS Mobile SDK* [5] de la empresa *ESRI* para el trabajo con información geográfica, y estudiar su potencial en aplicativos de uso masivo. Esto se debe a que el principal mercado objetivo de estos componentes es el de aplicaciones corporativas de captura y edición de información geográfica en campo.
- Estudio e implementación de un mecanismo para la comunicación y sincronización de información entre varios dispositivos móviles. Esto resulta de especial interés para ICA debido a su utilidad y aplicación en sistemas de seguimiento de vehículos, por ejemplo.
- Se desea conocer los esquemas y canales de comercialización existentes para esta clase de aplicativos (orientados al público masivo), ya que ICA en la actualidad se dedica casi exclusivamente a clientes concretos.

1.5 Conclusiones

Un aspecto a destacar es que, si bien a primera vista **GeoTrainer** puede parecer un aplicativo simple, en la etapa de investigación fue necesario abarcar una variada gama de tecnologías y áreas de estudio asociadas al mismo. Esta investigación permitió generar un amplio estudio del estado del arte, aunque quizás no con la profundidad que cada una de las temáticas merece por separado debido a su cantidad. En el capítulo nº 2 se presenta una breve reseña de éstas y el documento completo puede ser consultado en **Anexo I: Estado del Arte**.

Por otro lado cabe destacar que ciertas hipótesis realizadas por ICA acerca de la disponibilidad de software de base y bibliotecas en las que se basaba la propuesta del proyecto no se cumplieron. En particular la liberación tardía de ciertos productos de base por parte de la empresa *Microsoft* como ser *Microsoft Windows Mobile 6.5* y *Silverlight* para móviles. Debido a esto se decidió, en conjunto con ICA, invertir más tiempo en el estudio de alternativas que permitan desarrollar el aplicativo sin sacrificar algunas de las características deseadas para el mismo como era la interfaz de usuario.

A pesar de lo anterior, se logró desarrollar **GeoTrainer** abarcando un interesante conjunto de funcionalidades, cumpliendo de la mejor forma posible los requerimientos tanto funcionales como no funcionales e intentando generar conocimiento sobre los puntos de interés planteados por ICA.

Estas conclusiones se desarrollan con más profundidad en el capítulo nº 7 de este documento.

1.6 Organización del informe

Este informe está organizado en siete capítulos de la siguiente manera.

Capítulo 1: Introducción

Es el presente capítulo. En él se presentan brevemente los antecedentes y la motivación del proyecto, se realiza una descripción del problema y enumeran los objetivos y resultados esperados. Finalmente se presentan muy brevemente algunas de las conclusiones que se desarrollan en profundidad en el capítulo nº 7.

Capítulo 2: Estado del Arte

Se presenta aquí una breve reseña del estado del arte en las distintas tecnologías y temáticas involucradas con el proyecto pudiendo el lector encontrar una versión más detallada de las mismas en el **Anexo I: Estado del Arte**.

Capítulo 3: Análisis y especificación de requerimientos

En este capítulo se presenta una descripción detallada del sistema a desarrollar de acuerdo a la visión de ICA. Posteriormente se identifican de manera específica los distintos requerimientos que se desprenden del análisis. Esta especificación puede ser consultada en profundidad en el **Anexo II: Análisis y Especificación de Requerimientos**.

Capítulo 4: Diseño del sistema

Se presenta en este capítulo la arquitectura de cada uno de los componentes del sistema, se introduce de manera general la estructura de la base de datos que soportará el almacenamiento de los datos generados y se identifican los distintos servicios geográficos que deberán implementarse. Este capítulo se complementa con el **Anexo III: Diseño del Sistema**.

Capítulo 5: Implementación

Este capítulo describe de manera general las decisiones tomadas acerca de la implementación del proyecto, se presentan aquellas características relevantes y las decisiones tomadas en torno a ellas. Acompaña a este capítulo el **Anexo IV: Manual de Usuario**.

Capítulo 6: Testeo y verificación

Se presentan las características de la plataforma de testeo y los tests realizados así como los resultados obtenidos.

Capítulo 7: Conclusiones y trabajo futuro

El capítulo nº 7 presenta en detalle las conclusiones del proyecto así como aquellos aspectos que, por razones técnicas o por falta de tiempo, no se pudieron desarrollar y que a futuro se entiende puedan mejorar o complementar el aplicativo **GeoTrainer**.

2 Estado del arte

2.1 Introducción

Si bien **GeoTrainer** puede definirse simplemente como una aplicación para dispositivos móviles, dadas las distintas funcionalidades esperadas, involucra una variada gama de temáticas y tecnologías. En este capítulo se presentan aquellas que se entiende están involucradas. Un estudio más detallado de cada una de ellas se presenta en **Anexo I: Estado del Arte**.

En primer lugar se deberá abordar el área de dispositivos móviles prestando especial atención a la evolución de los mismos en lo que refiere a especificaciones técnicas tanto a nivel de hardware como de software (sección 2.2).

En cuanto al hardware, se deberá estudiar no solo la evolución del mismo teniendo en cuenta la capacidad de cómputo y capacidad de memoria de los dispositivos, sino también la inclusión y características de los diversos accesorios que puedan estar relacionados con la temática del producto. Entre estos se destacan las capacidades gráficas de los mismos y la presencia de receptores GPS y A-GPS⁵ (sección 2.2.2).

En lo que refiere al software, como primer punto se estudiarán los distintos sistemas operativos existentes para dispositivos móviles (sección 2.2.3) y se evaluarán las distintas herramientas y entornos de desarrollo de aplicaciones también disponibles actualmente (sección 2.4).

Otro aspecto relevante es el estudio de los aplicativos disponibles en el mercado (sección 2.6) y relacionados con la temática del proyecto. En particular se evaluarán los aplicativos orientados a los deportes, que permitan la planificación, el registro y posterior evaluación de actividades y manejen los conceptos de redes sociales⁶ (sección 2.3) permitiendo la publicación e intercambio de información entre los distintos usuarios.

Un punto que fue excluido conscientemente de este estudio es el referente a los *Sistemas de Información Geográfica*. Si bien el aplicativo a desarrollar hace uso de mapas, genera y utiliza información de carácter estrictamente geográfico, se entiende que éste es un uso marginal de las funcionalidades y potencia brindadas por los SIGs. Incluir este tema en el siguiente estado del arte hubiese requerido profundizar en una variedad de temas que nada tienen que ver con el presente proyecto.

⁵ A-GPS – Sistema de posicionamiento global asistido

⁶ Red social - Son sitios basados en la web que permiten a los usuarios compartir contenido, interactuar y crear comunidades sobre intereses similares.

2.2 La telefonía celular y los dispositivos móviles

Desde sus inicios a finales de los años 70s, la telefonía celular ha evolucionado muy rápidamente, revolucionando drásticamente las actividades que se realizan diariamente. Los teléfonos celulares se han convertido en una herramienta primordial tanto para el común de la gente como en ambientes laborales [6][7].

A pesar que la telefonía celular fue concebida para la voz únicamente, debido a las limitaciones tecnológicas de esa época, la tecnología celular de hoy en día es capaz de brindar otro tipo de servicios tales como comunicación de datos, audio y video, y se ha constituido en una de las maneras más utilizadas para comunicarse entre las personas.

2.2.1 Las generaciones de la telefonía móvil

Para separar cronológicamente y tecnológicamente una etapa de la otra, la telefonía celular ha sido categorizada por generaciones. A continuación se presentan las distintas generaciones enumerando alguna de sus principales características [6][7][8].

0G: Los inicios

- Desde fines de los años 1940 hasta 1985
- Telefonía móvil previa a la era celular.
- Sistemas de radio analógicos que inicialmente utilizaban modulación en amplitud (AM) y luego modulación en frecuencia (FM)
- Dispositivos grandes y pesados, usados principalmente en vehículos.

1G: La primera generación

- Operó desde 1981 hasta 2003.
- Se introdujeron las redes celulares con múltiples estaciones de base relativamente cercanas unas de otras (celdas) y protocolos para el "traspaso" entre las bases cuando el teléfono se movía de una a otra.
- Comunicación analógica y estrictamente para voz.
- Se siguieron utilizando canales de radio FM, inicialmente en las frecuencias en torno a los 450Mhz, pasando posteriormente a frecuencias cercanas a los 900Mhz, logrando dar servicio a un mayor número de usuarios.
- Dispositivos de gran tamaño.

2G: La segunda generación

- Desde la década de los 90
- Digitalización de las comunicaciones, ofreciendo mejor calidad de voz que las comunicaciones analógicas.
- Tecnologías de bajo consumo energético permitieron mejoras en la duración de las baterías.
- Se simplifica la fabricación de los dispositivos reduciendo además su tamaño y costo.

- Se ofrecen servicios auxiliares como ser transmisión de datos y mensajería de texto (SMS⁷)
- Mercado competitivo, multitud de operadores y fabricantes.

2.5G - Generación de transición

- Se introducen las tecnologías GPRS⁸ [9] y EDGE⁹ [10] ofreciendo mayor velocidad de transferencia de datos
 - GPRS permite velocidades de datos desde 56kbps hasta 114 kbps.
 - EDGE permite velocidades de datos hasta 384 Kbps.
- Nuevos servicios de mensajería: EMS¹⁰ [11] y MMS¹¹ [12].
 - EMS es un servicio de mensajería mejorado, permite la inclusión de melodías e iconos dentro del mensaje basándose en los SMS; un EMS equivale a 3 o 4 SMS.
 - MMS consiste en mensajes que se envían mediante GPRS y permite la inserción de imágenes, sonidos, videos y texto.

3G y 3.5G - Tercera generación: Actualidad

- Nace de la necesidad de aumentar la capacidad de transmisión de datos para poder ofrecer servicios como conexión a Internet desde el móvil, videoconferencia, televisión y descarga de archivos.
- Sistema totalmente nuevo: UMTS¹² [13].
 - UMTS utiliza la tecnología CDMA¹³ [14], lo cual le hace alcanzar velocidades desde 144 Kbps hasta 7.2 Mbps.
- El sistema 3.5G es también conocido por sus siglas en inglés HSDPA¹⁴ [15] es un nuevo estándar que esencialmente ofrece velocidades de transmisión de datos aún mayores, llegando teóricamente a los 10Mbps.

4G - Cuarta generación: El futuro.

- Estándar actualmente en desarrollo por la ITU¹⁵ [16].
- Se concentra en lograr velocidades de transmisión de datos no menores a 100 Mbps en condiciones de alta movilidad y del orden de 1Gbps en escenarios estáticos.
- En la actualidad existen dos posibles candidatos a ser el estándar [17]: LTE¹⁶ [18] y WiMAX II¹⁷ [19].
- Permitirá al usuario acceder a servicios de gran consumo de ancho de banda como ser la recepción de televisión de alta definición.

⁷ SMS - Short Message Service

⁸ GPRS - General Packet Radio Service

⁹ EDGE - Enhanced Data rates for GSM Evolution

¹⁰ EMS - Enhanced Messaging Service

¹¹ MMS - Sistema de Mensajería Multimedia

¹² UMTS - Universal Mobile Telecommunications System

¹³ CDMA - Code Division Multiple Access

¹⁴ HSDPA - High-Speed Downlink Packet Access

¹⁵ ITU - International Telecommunication Union

¹⁶ LTE - Long Term Evolution

¹⁷ WIMAX - Worldwide Interoperability for Microwave Access

2.2.2 Hardware de dispositivos

Debido a que el desarrollo de una aplicación para dispositivos móviles normalmente no está destinado a un único modelo y marca de dispositivo, es de suma importancia conocer las diferentes características de hardware de los mismos.

2.2.2.1 Procesadores

Un punto de singular importancia es el que refiere al procesador del dispositivo móvil, ya que es éste quien en primera instancia define el poder de cómputo y por lo tanto las características de los aplicativos que pueda ejecutar de manera adecuada.

Los procesadores para dispositivos móviles están liderados por los chips basados en la línea *ARM*¹⁸ [20] y son fabricados por empresas tales como **Texas Instruments** [21], **Apple** [22], **NVIDIA** [23], **Samsung** [24] o **Qualcomm** [25]. Entre estos se encuentran el **A4** de **Apple** [26], **Snapdragon** de **Qualcomm** [27], **Tegra2** de **NVIDIA** [28] y el **Marvell Armada** de **Samsung** [29], cada uno de ellos con una frecuencia de reloj de 1GHz.

También **Intel** [30] compite en este segmento con el desarrollo de su nuevo procesador denominado **Moorestown Z6xx Atom** que funcionará a frecuencias de entre 1.5GHz y 1.9GHz [31].

2.2.2.2 Capacidades gráficas

Las capacidades gráficas de los dispositivos móviles se han incrementado al mismo ritmo que la capacidad de procesamiento de los procesadores de los mismos.

Es posible encontrar en la actualidad dispositivos capaces de reproducir video de alta definición e incluso con soporte para *OpenGL*¹⁹ permitiendo ejecutar los últimos juegos y aplicaciones 3D [32][33].

2.2.2.3 Touch y Multi Touch Screen

A las pantallas que mediante un contacto directo sobre su superficie permiten la entrada de datos y órdenes al dispositivo se les denomina pantallas *touchscreen* (pantallas táctiles). [34][35][36].

Existen a su vez las llamadas *Multi Touch-Screen*, y se diferencian de las simples en la capacidad de reconocer simultáneamente múltiples puntos de contacto y también interpretar dichas interacciones simultáneas en el software asociado.

Existen diferentes tecnologías en la implementación de las pantallas táctiles. Las más utilizadas en dispositivos móviles son:

- Resistiva
- De onda acústica superficial
- Capacitivas

¹⁸ ARM - Advanced RISC Machine.

¹⁹ OpenGL - Open Graphics Library, es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D.

2.2.2.4 GPS y A-GPS

El *Global Position System (GPS)* [37] o Sistema de Posicionamiento Global es un Sistema Global de Navegación por Satélite (*GNSS*²⁰) que permite determinar en todo el mundo la posición de un objeto, habitualmente con una precisión de unos pocos metros, aunque es posible mediante el uso de ciertas técnicas obtener precisiones del orden de los centímetros.

El sistema GPS funciona mediante una red de 27 satélites (24 operativos y 3 de respaldo) en órbita sobre la Tierra, a 20.200 km con trayectorias sincronizadas para cubrir toda su superficie.

Cuando se desea determinar la posición, el receptor GPS localiza como mínimo tres satélites de la red, de los que recibe la posición y ciertas señales que le permiten calcular la distancia a cada uno de ellos. Una vez conocidos estos datos, se calcula por triangulación²¹ la posición del receptor.

Los receptores GPS convencionales presentan dificultades a la hora de proporcionar posiciones precisas en condiciones de baja señal. Por ejemplo, cuando se está rodeado de edificios altos (como consecuencia de la recepción de múltiples señales rebotadas) o cuando la señal del satélite se ve atenuada por obstáculos, dentro de edificios o debajo de árboles, por ejemplo.

Para mejorar la precisión de la posición se utilizan los sistemas A-GPS (GPS Asistido), los cuales complementan los cálculos mediante el uso de información extra proveniente o bien de servidores externos (información acerca de los satélites) o bien de las propias antenas de la red celular.

2.2.2.5 Autonomía y Alimentación

Un factor que hay que tener en cuenta al momento de desarrollar una aplicación para un dispositivo móvil es la capacidad de autonomía y de alimentación que tiene el mismo.

Por ejemplo, si se desea desarrollar una aplicación que hace uso intensivo del procesador del dispositivo, utiliza GPS y necesita usar la red 3G para conectarse a Internet, probablemente se necesite un dispositivo con grandes prestaciones de autonomía.

Existen varias tecnologías con respecto a las baterías de los dispositivos móviles, las actuales son [38]:

- NiMH
- Litio
- Baterías de Polímeros de Litio

Actualmente se están estudiando y desarrollando:

- Baterías solares
- Baterías de alcohol

²⁰ GNSS – Global Navigation Satellite System

²¹ La triangulación, en geometría, es el uso de la trigonometría de triángulos para determinar posiciones de puntos, medidas de distancias o áreas de figuras.

2.2.3 Sistemas Operativos

Algo a tener en cuenta al momento de desarrollar una aplicación para un dispositivo móvil es el sistema operativo [39], ya que éste provee funcionalidades y características que pueden hacer que tanto la aplicación como su desarrollo sean viables o no. El sistema operativo determina en muchos casos las herramientas y entornos de desarrollo disponibles.

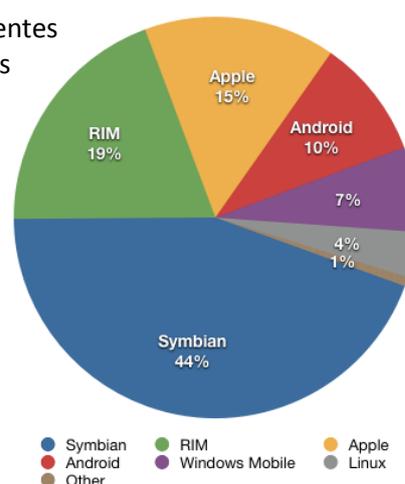


Algunos de los puntos a tener en cuenta al evaluar el desarrollo para cierto sistema operativo son:

- Funcionalidades brindadas por el núcleo (*kernel*²²) del sistema operativo
- Interacción con el usuario
 - Soporte para pantallas *Touch* o *Multi Touch*
 - Uso de *stylus* (lápiz) o uso de las manos directamente (tecnología de la pantalla)
 - Teclado físico o virtual (esto también depende del modelo del dispositivo)
 - Reconocimiento de gestos.
- Esquemas de personalización de la interfaz (p. ej.: tamaño de texto)
- Multitasking.
- SDKs²³ de desarrollo y documentación apropiada.
- Esquemas de comercialización de aplicaciones.

En la actualidad son seis los principales sistemas operativos presentes en dispositivos móviles que comparten fracciones muy distintas del mercado en el primer trimestre de 2010 [40] como se aprecia en la siguiente figura.

- Symbian OS [41]
- RIM Blackberry [42]
- Apple iPhone OS [43]
- Android [44]
- Windows Mobile [45]
- Linux [46]



²² Kernel - Es el software responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma más básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema.

²³ SDK - Software Development Kit (Kit de desarrollo de software)

2.3 Redes sociales

En general, una red social [47][48][49][50][51][52][53] es una estructura social que se representa en forma de uno o varios grafos en el cual los nodos representan individuos y las aristas relaciones entre ellos.

También se definen *Redes Sociales basadas en la Web* a los servicios que permiten que los individuos construyan grupos públicos o semi-públicos dentro de un sistema limitado, articular una lista de otros usuarios con los que comparten una conexión y ver y recorrer su lista de conexiones y las hechas por otros dentro del sistema.

En las redes sociales en Internet se tiene la posibilidad de interactuar con otras personas aunque no se conozcan, el sistema es abierto y se va construyendo con lo que cada suscriptor a la red aporta, cada nuevo miembro que ingresa transforma al grupo en otro nuevo.

Entre el 2001 y 2002 surgen los primeros sitios que fomentan las redes de amigos. Hacia 2003 se hacen populares con la aparición de sitios tales como *Friendster* [54], *Tribe* [55] y *Myspace* [56].

El relativamente nuevo concepto de las redes sociales en la informática es la base de la denominada Web 2.0, en la que, a diferencia de la Web 1.0, el contenido es generado por los propios usuarios a demás del generado por los publicadores de los mismos.

En la actualidad las redes sociales más conocidas son: *Facebook* [57], *Twitter* [58] e incluso *YouTube* [59].

2.4 Desarrollo de aplicaciones

2.4.1 Plataformas, lenguajes de programación y entornos de desarrollo.

La siguiente tabla presenta un resumen de las distintas opciones para el desarrollo de aplicaciones para los principales sistemas operativos [60][61][62][63][64][65][66][67][68][69][70][71].

Plataforma	Symbian OS, Linux	RIM Blackberry	Apple	Android	Windows Mobile
Entorno de ejecución	Java Micro Edition (JavaME)	Java Micro Edition (JavaME)	iPhoneOS	Android OS	Microsoft .NET Compact Framework
Entorno de desarrollo	NetBeans IDE Eclipse	Eclipse	XCode IDE	Eclipse	Microsoft Visual Studio .NET
Lenguaje de desarrollo	Java	Java	Objective-C	Java	C#, Visual Basic

2.4.2 Servicios y componentes de software disponibles

Los componentes de software ya existentes que puedan ser integrados a las distintas aplicaciones son un aspecto relevante para este proyecto.

Teniendo en cuenta la temática del proyecto, se destacan dos grandes grupos. Por un lado se encuentran las **infraestructuras de datos y servicios geográficos** y por otro las **bibliotecas** o **APIs** que permitan acceder a las primeras.

2.4.2.1 Infraestructuras de datos y servicios

Los siguientes son algunos de los principales proveedores de información y servicios geográficos.

ArcGIS Online Services

- *ArcGIS Online Services* [72] brinda cartografía, imágenes satelitales, globos 3D y funciones geográficas (Ruteo, Geocodificación²⁴, Búsquedas, etc.) de todo el mundo a través de Internet.
- Existen 2 tipos de licenciamiento: *Estándar* y *Premium*
 - *Estándar*: sin costo, permite uso privado y público NO comercial.
 - *Premium*: suscripción anual, acceso a mayor información (ej.: Imágenes satelitales de mejor resolución)

Google Maps

- *Google Maps* [73] ofrece cartografía e imágenes satelitales mundiales
- Permite la generación de mapas de varios estilos: de calles, satelitales, híbridos (combinación de ambos), topográfico, etc.
- Licenciamiento *Royalty-Free*²⁵
- Brinda también servicios de ruteo y geocodificación.

Yahoo! Maps

- *Yahoo! Maps* [74] es análogo a *Google Maps*

²⁴ Geocodificación - Proceso de asignar coordenadas geográficas (e.g. latitud-longitud) a puntos del mapa (direcciones, puntos de interés, etc.)

²⁵ Royalty-Free - Se refiere a un tipo de contrato entre dos entidades (el licenciante y el licenciatario), significa que una vez que el contenido está disponible bajo un conjunto de directrices, el licenciatario está normalmente libre de usarlo a perpetuidad sin pagar derechos adicionales.

OpenStreetMap

- *OpenStreetMap* [75] ofrece mapas de calles a escala mundial **gratuito**
- *Colaborativo*: la edición de la información está a cargo de la comunidad de usuarios.
- *Desventajas*: existen zonas donde hay poca información

2.4.2.2 Bibliotecas de software

A continuación se presentan algunas de las principales bibliotecas disponibles para el desarrollo de aplicaciones geográficas.

ArcGIS Mobile SDK

ArcGIS Mobile SDK [76][77] es un kit de desarrollo diseñado por la empresa ESRI, que brinda funcionalidades básicas geográficas a las aplicaciones. Permiten la visualización de mapas, edición de datos y sincronización de los mismos con un servidor y brindan soporte para el manejo de dispositivos GPS.

Está construido sobre *Microsoft .NET Framework*, se integra a *Microsoft Visual Studio .NET* y comprende una serie de clases y controles diseñados específicamente para las plataformas móviles Windows (*Smartphone*, *Pocket PC* y *Tablet PC*).

Google Maps APIs

Las *API de Google Maps* [78][79] proporcionan a los desarrolladores diversas formas de insertar datos provenientes de *Google Maps* en aplicaciones.

- *Google Maps JavaScript API*
- *Google Maps API for Flash*
- *Google Static Maps API*

Yahoo! Maps Web Services

De manera prácticamente análoga a *Google Maps APIs*, *Yahoo! Maps Web Services* [80][81] permite embeber mapas dinámicos y estáticos en aplicaciones.

- *Yahoo Maps Flash API*
- *Yahoo Maps Ajax API*
- *Yahoo Maps Image API*

CloudMade

CloudMade [82] brinda una serie de APIs que ofrecen distintos servicios basados en la información de *OpenStreetMap*. Algunos de estos servicios cubren el cálculo de rutas (*Routing*), la búsqueda de direcciones (*Geocoding*) y la generación de mapas (*iPhone Maps Library* y *Mobile Maps Library*)

MGMaps Lib SDK

MGMaps Lib SDK [83] es una biblioteca de mapeo liviana para utilizar en aplicaciones *Java ME* y trabaja por defecto sobre la información del proyecto *OpenStreetMap*, aunque otros proveedores pueden ser configurados.

Si bien el kit de desarrollo tiene costo, es una librería de licenciamiento *Royalty-Free* y se encuentran disponibles versiones de evaluación.

Fire Eagle

Fire Eagle [84] es una plataforma de servicios para el manejo de ubicaciones geográficas. Brinda funciones para que usuarios de una cierta aplicación intercambien sus posiciones geográficas. Las coordenadas son almacenadas en el servidor y pueden ser consultadas por otros usuarios a los que se les permite el acceso con el nivel de precisión que el publicador configure (*posición exacta, código postal, ciudad, país, etc.*)

Google Latitude API

Google Latitude API [85] ofrece servicios similares a los de *Fire Eagle*. Permite agregar a las aplicaciones funcionalidades para publicar la posición geográfica del usuario y consultar la de otros usuarios.

2.5 Distribución y comercialización

El modo en que se comercializan las aplicaciones para dispositivos móviles es muy distinto al de las aplicaciones para computadores personales. Su costo incluso es muy inferior. Éstas en su mayoría, o bien son gratis o su costo es mínimo (normalmente entre 1 y 10 dólares) y es en la masividad de las mismas que los desarrolladores obtienen las ganancias, aunque hay excepciones.

Dependiendo de la plataforma destino del aplicativo, existen a través de internet tiendas virtuales donde las aplicaciones son comercializadas electrónicamente. Éstas pueden ser descargadas a un computador personal para luego ser instaladas o directamente en el mismo dispositivo.

Básicamente todas comparten el mismo esquema de comercialización, donde un 70% del costo de la aplicación es destinado al dueño de la misma y el 30% restante corresponde al administrador de la tienda [86][87][88][89][90].

Plataforma	Tienda
iPhone iPod Touch	Apple AppStore [86] http://www.apple.com/iphone/apps-for-iphone/
Windows Mobile Windows Phone 7	Windows Marketplace for Mobile [87] http://marketplace.windowsphone.com/Default.aspx
Android	Android Market [90] http://www.android.com/market

2.6 Aplicaciones relacionadas

Si bien actualmente hay varias aplicaciones disponibles que son similares a **GeoTrainer**, ninguna ofrece la gama de funcionalidades de éste. El común denominador de estas aplicaciones se centra en la captura de una actividad deportiva usando GPS y el registro de valores asociados como velocidad, tiempo, etc.

La mayoría de ellas a su vez trabaja en conjunto con un servidor central en el cual se publican estas actividades y que posteriormente pueden ser evaluadas, incluso con mejor detalle, por otros usuarios mediante una interfaz web. Algunas permiten además compartir los resultados obtenidos a través de redes sociales como *Facebook* o *Twitter*.

Algunas de las aplicaciones disponibles son:



MotionX-GPS [91]



SportyPal [92]



Google My Tracks [93]



Nokia Sports Tracker [94]



Sony Ericsson Tracker [95]



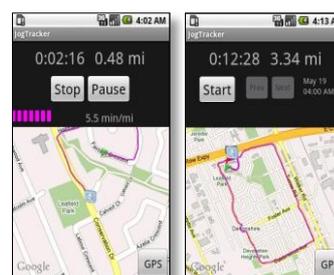
Endomondo Sports Tracker [96]



RunKeeper [97]



iMapMyRun [98]



JogTracker [99]

A continuación se presenta una tabla comparativa de las funcionalidades ofrecidas por algunas de las aplicaciones evaluadas y **GeoTrainer**.

	GeoTrainer	Motion X-GPS [91]	SportyPal [92]	Google My Tracks [93]	Nokia Sports Tracker [94]	Sony Ericsson Tracker [95]	Endomondo Sports Tracker [96]	RunKeeper [97]	iMapMyRun [98]	Jog Tracker [99]
Uso de GPS para el registro	●	●	●	●	●	●	●	●	●	●
Realización de entrenamientos en base a datos previos	●					●				
Competencias con múltiples usuarios concurrentes	●						●			
Graficas de evaluación	●		●			●				
Mapa interactivo	●	●		●	●			●	●	●
Mapa estático	●		●				●			
Servidor central para la publicación de actividades	●		●	●	●		●	●		●
Portal web dedicado			●		●		●	●		●
Sistemas operativos										
Windows Mobile	●		●				●			
iPhoneOS		●	●				●	●		
Android			●	●			●	●	●	●
RIM BlackBerry			●				●			
Symbian OS			●		●		●			
Sony Ericsson			●			●	●			
Integración con redes sociales										
Facebook			●							
Twitter				●					●	

3 Análisis y especificación de requerimientos

3.1 Introducción

En este capítulo se presenta una descripción general del aplicativo a desarrollar, la cual es producto de la compilación de datos aportados por ICA a través de una serie de entrevistas y de documentos intercambiados. Es una expresión de todo aquello que se desea que la aplicación cumpla.

Como resultado de su evaluación se desprenden, en primer lugar, una serie de consideraciones que plantean ciertas restricciones que deberán ser atendidas. En segundo lugar se identifican los requerimientos tanto funcionales como no funcionales del sistema a construir.

3.2 Descripción general del producto y sus funcionalidades

El principal objetivo del sistema a desarrollar es el de brindar a los deportistas una herramienta con la cual puedan registrar y analizar sus sesiones de entrenamiento de deportes tales como correr, ciclismo, caminatas, etc. incluyendo tanto los tiempos y distancias realizados como los trayectos propiamente dicho con sus coordenadas geográficas.

En base a lo propuesto por ICA, el aplicativo deberá dar gran importancia a la información geográfica y la principal forma de interacción con el usuario durante las actividades deportivas será mediante la visualización en el dispositivo del mapa del recorrido de la actividad. Este recorrido podrá estar pre cargado sobre el mapa (recorrido a seguir), o se irá registrando y desplegando durante el desarrollo de la actividad.

La captura de los datos asociados a los recorridos de las actividades se hará mediante el uso del receptor GPS con las que deberá contar el dispositivo.

Se permitirá a su vez la comparación de estas actividades con otras realizadas anteriormente por el propio usuario o por otros. Esta comparativa podrá ser realizada tanto luego de haber finalizado la actividad deportiva (*comparativa off-line*), así como durante el propio desarrollo de la misma (*comparativa en tiempo real*).

En este último caso, el aplicativo desplegará no solo los datos correspondientes al usuario que se encuentra realizando la actividad, sino también los correspondientes a la actividad contra la que se desea realizar la comparación. La manera en que se efectuará la comparativa será simulando con estos datos un segundo competidor, virtual en este caso, el cual será visualizado sobre el mapa al igual que sus datos (velocidad, distancia recorrida, promedios, etc.). Se incluirán además aquellos valores que permitan rápidamente evaluar esta comparativa, por ejemplo la distancia entre el usuario y el competidor virtual.

Será posible realizar actividades asociadas a eventos deportivos y/o entrenamientos, por ejemplo una carrera 10k, o una competencia ciclística. En este caso, un esquema de comparación similar al anteriormente detallado (*comparativa en tiempo real*) se utilizará en particular comparando otros usuarios también competidores. Para esto será necesario poder contar con comunicación con el servidor central, ya que será a través de éste que se sincronizará la información entre los distintos usuarios. Cada usuario comunicará su posición en intervalos regulares de tiempo, obteniendo a su vez la de los otros competidores. En esta modalidad se acotará la cantidad de competidores para evitar el exceso de información el cual puede resultar contraproducente y llevar a confusiones por parte del usuario.

El almacenamiento de estos datos se hará en un servidor central, por lo cual los distintos usuarios deberán contar con alguna forma de identificación ante éste (usuario y contraseña).

3.2.1 Consideraciones adicionales

De lo evaluado en reuniones con el responsable del proyecto por parte de ICA se desprenden ciertas restricciones a tener en cuenta, tanto para el diseño como para la implementación del aplicativo.

La más importante de esas restricciones sin duda está dada por la necesidad de que el aplicativo se desarrolle utilizando los componentes *ArcGIS Mobile SDK* presentados en el capítulo anterior. Esto se debe a que ICA es representante de la empresa ESRI, desarrolladora del SDK.

Esta elección de desarrollo determina a su vez la plataforma de ejecución del aplicativo; estos componentes están disponibles únicamente para la plataforma *Windows Mobile*.

A su vez, queda determinado el entorno de desarrollo a utilizar, estos componentes están diseñados para el desarrollo de aplicaciones utilizando *Microsoft Compact Framework .NET* y se integran por defecto en *Microsoft Visual Studio*.

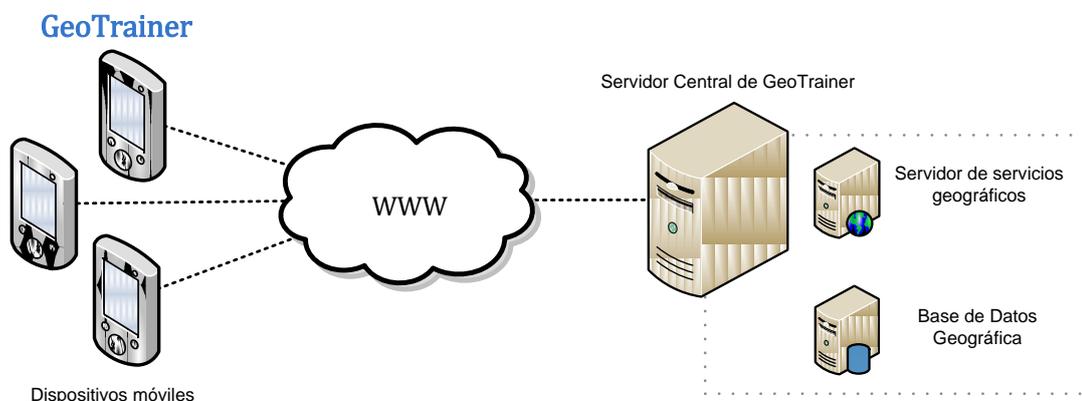
Otro punto a tener en cuenta, que se debe también al uso de los componentes *ArcGIS Mobile SDK*, es que la información geográfica que despliegue el aplicativo, es decir el mapa, sea proveniente de un servidor geográfico *ArcGIS Server*, producto también de la empresa ESRI.

El proyecto debe resolver, además del desarrollo del aplicativo móvil, el diseño de la base de datos que almacenará los datos publicados y la implementación de servicios del lado del servidor para la consulta y el mantenimiento de dichos datos.

3.3 Esquema general del sistema

Se desprende de lo mencionado anteriormente que **GeoTrainer** será, además del propio aplicativo, un sistema más amplio. Éste involucra también un Servidor Central que se encargará del almacenamiento de la información provista por los distintos usuarios y también será el encargado del intercambio de información entre los dispositivos en el escenario de las competencias deportivas.

La siguiente figura muestra un esquema de los distintos componentes del sistema.



Por un lado se encuentran los dispositivos móviles con el aplicativo **GeoTrainer** instalado, los cuales harán uso a través de internet, de distintos servicios provistos por un servidor central.

3.4 Especificación de requerimientos

A continuación se presentan los distintos requerimientos funcionales y no funcionales identificados como parte del análisis anterior. Esta presentación se hará de manera general, adjuntando una breve descripción. Una versión más detallada puede ser consultada en el **Anexo II: Análisis y Especificación de Requerimientos**.

3.4.1 Requerimientos funcionales

3.4.1.1 Manejo de información personal del usuario

En primer lugar la aplicación a desarrollar deberá hacer uso de servicios sobre Internet y brindar funcionalidades de red social, por lo cual se entiende necesario al menos el registro de los usuarios con su correspondiente identificación y clave, etc.

El usuario deberá poder acceder a una interfaz que le permita el ingreso de estos valores.

3.4.1.2 Información geográfica: Mapas y Navegación

Debido a la naturaleza y objetivo de la aplicación gran parte de las funcionalidades brindadas deberán ser presentadas en forma geográfica, esto implica la visualización de un mapa y las distintas herramientas de navegación habituales (acercar/alejar, desplazar, etc.).

3.4.1.3 Información geográfica: Posicionamiento

Una vez desplegado un mapa, se podrá visualizar la posición del usuario sobre el mismo y para esto se deberá contar con las coordenadas geográficas de la posición del usuario. En este caso se utilizarán los datos provenientes del dispositivo GPS con el que debe contar el dispositivo móvil.

3.4.1.4 Actividad deportiva: Exploración

El modo de exploración permitirá al usuario el registro de una actividad deportiva sin ninguna planificación previa. Se entiende por planificación previa, el seguimiento de algún recorrido previo.

Durante el registro de la actividad se irá desplegando el trayecto recorrido sobre el mapa.

El usuario deberá poder pausar y reanudar el desarrollo de la actividad.

3.4.1.5 Actividad deportiva: Entrenamiento/Competencia virtual

Esta modalidad permitirá la realización de una actividad teniendo como referencia los datos de otra previa.

Se deberá visualizar el recorrido a seguir sobre el mapa y, en base a los tiempos asociados, se simulará un usuario virtual (competidor virtual) el cual avanzará conforme avance el tiempo de actividad.

Durante el registro de la actividad se irá desplegando el trayecto recorrido sobre el mapa.

El usuario podrá pausar y reanudar el desarrollo de la actividad.

3.4.1.6 Actividad deportiva: Competencia en Tiempo Real

Esta modalidad será similar a la anterior en el sentido que se utilizará un recorrido de referencia. Pero, a diferencia del entrenamiento, el aplicativo comunicará en intervalos regulares la posición del usuario durante el desarrollo de la actividad al **Servidor Central de GeoTrainer**. A su vez, se desplegarán las posiciones de otros usuarios que en ese momento estén participando de la misma actividad.

Durante el registro de la actividad se irá desplegando el trayecto recorrido sobre el mapa.

No se permitirá una vez iniciada la actividad, pausar y reanudar el registro de la actividad en esta modalidad.

3.4.1.7 Creación manual de trayectos

Se deberá brindar la posibilidad de editar manualmente sobre el mapa trayectos. Éstos podrán posteriormente ser utilizados como referencia en las modalidades *Entrenamiento* y *Competencia en Tiempo Real*.

3.4.1.8 Almacenamiento de actividades

Las actividades una vez finalizadas deben de poder ser almacenadas en el dispositivo para su posterior evaluación y comparación con otras o ser utilizadas como referencia para entrenamientos.

3.4.1.9 Publicación de una actividad deportiva

Las actividades realizadas se podrán publicar en el **Servidor Central de GeoTrainer**. Se deben comunicar al servidor datos como ser, la fecha de la actividad, el deporte realizado, trayecto recorrido, tiempos, etc.

3.4.1.10 Creación de un evento deportivo

Se deberán poder crear y publicar en el servidor eventos deportivos para ser utilizados como referencia en la modalidad *Competencia en Tiempo Real*. Esto requiere al menos los siguientes datos: nombre del evento, trayecto a recorrer, fecha y hora de inicio, deporte.

3.4.1.11 Búsqueda de actividades y eventos

El aplicativo debe permitir la búsqueda de actividades y eventos deportivos, tanto almacenados en el dispositivo como publicados en el servidor. Esta búsqueda deberá poder realizarse filtrando valores como el deporte realizado, fecha, nombre del evento, distancia del recorrido, etc.

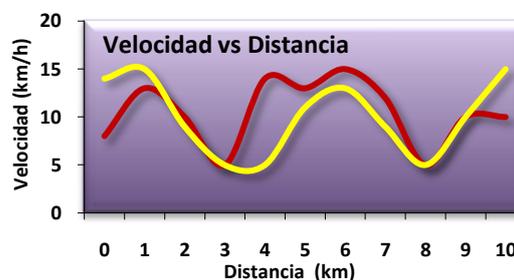
3.4.1.12 Evaluación de rendimiento y comparativas

Los usuarios deberán poder evaluar los datos de cualquier actividad realizada visualizando los datos asociados. Para ello se desplegará el mapa del trayecto realizado, la distancia recorrida, el tiempo y velocidad promedio obtenidos. También se deben mostrar gráficas que permitan rápidamente evaluar los indicadores de la actividad durante su desarrollo. Entre estas graficas se destacan la de Velocidad vs Distancia y la de Altura del terreno vs Distancia (perfil del terreno, ver sección 3.4.1.14).

3.4.1.13 Comparación de actividades

Se deberá poder comparar al menos dos actividades. Una alternativa factible para la comparación es desplegar la información de las actividades de manera análoga al punto anterior pero combinando los datos de ambas. De esta manera se visualizará un mapa conteniendo ambos trayectos y las gráficas combinadas de *Velocidad vs Distancia*, *Distancia vs Tiempo* y *Altura del terreno vs Distancia*.

Ej.: *Velocidad vs Distancia*
En este caso se muestra comparativamente como varió la velocidad de dos competidores a lo largo del recorrido



3.4.1.14 Servicios geográficos: Perfil de terreno

Una de las graficas que resulta de suma importancia para la evaluación de una actividad es la que muestra la variación de alturas del terreno a lo largo del recorrido.

Los valores de la altura del terreno serán obtenidos a través de un servicio que deberá brindar el **Servidor Central de GeoTrainer**.

3.4.2 Requerimientos no funcionales

3.4.2.1 Componentes de software a utilizar

De la etapa de análisis previo se desprende que deben utilizarse los componentes *ArcGIS Mobile SDK*.

3.4.2.2 Información geográfica

El requerimiento anterior condiciona a que el mapa a desplegarse deberá provenir de un servidor dedicado *ArcGIS Server*. El mapa deberá basarse en datos locales a la red donde se encuentre el servidor ya que servicios de datos compatibles con *ArcGIS Server* como ser *ArcGIS Online Services* (sección 2.4.2.1) no son compatibles con los componentes móviles *ArcGIS Mobile SDK*.

3.4.2.3 Plataforma de los dispositivos móviles

El uso de los componentes *ArcGIS Mobile SDK* determina a su vez que el dispositivo móvil debe utilizar *Microsoft Windows Mobile* como sistema operativo.

3.4.2.4 Interfaz de usuario

Si bien el aplicativo no será una versión comercial definitiva, tampoco será un prototipo cuya única función sea la de testear funcionamiento y tecnologías. Se espera que el mismo provea una interfaz de usuario atractiva pero amigable al mismo tiempo, que saque provecho de las capacidades gráficas del dispositivo evitando los clásicos formularios de las plataformas Windows.

Se sugiere en este sentido ofrecer al usuario una interacción a través de la pantalla táctil del dispositivo, mediante el uso de controles que permitan su accionar utilizando los dedos directamente y tratando de evitar el uso del *stylus*.

También, de manera de ofrecer una experiencia más rica visualmente, se propone el uso de algunos efectos gráficos como son las transparencias o reflejos.

4 Diseño del sistema

4.1 Introducción

En este capítulo se presenta una breve descripción de la arquitectura del sistema en general y también de cada uno de sus componentes.

Una presentación más detallada de estos puntos se puede consultar en el **Anexo III: Diseño del Sistema**.

4.2 Arquitectura general de la aplicación

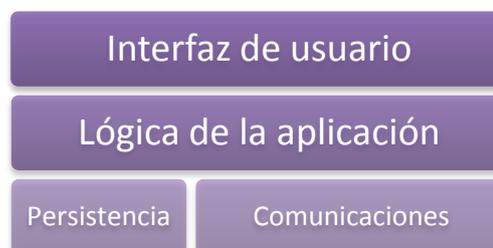
A nivel general, la aplicación se separa en dos subsistemas bien definidos, el aplicativo **GeoTrainer** que se ejecutará sobre los dispositivos móviles y el **Servidor Central de GeoTrainer**.

La comunicación entre estos subsistemas se realizará mediante la invocación a servicios web **GeoTrainerWS** brindados por el servidor.



4.3 GeoTrainer

El diseño del aplicativo **GeoTrainer** se realizó según el modelo de arquitectura en capas y fue desarrollado bajo el paradigma de orientación a objetos, siguiendo los criterios de alta cohesión y bajo acoplamiento de componentes.



4.3.1 Capa 1: Interfaz de usuario

Si bien en esta capa simplemente se resuelve lo relacionado con las distintas pantallas que el usuario visualiza, debido a las características deseadas del aplicativo en lo que refiere a interfaces e interacción, la misma contiene una variedad de componentes gráficos que el *framework* de desarrollo no proveía de forma nativa y debieron ser desarrollados.

4.3.2 Capa 2: Lógica de la aplicación

Ésta es sin dudas la capa que cuenta con mayor carga de funcionalidades ya que resuelve todo lo relativo al funcionamiento del aplicativo.

4.3.3 Capa 3: Persistencia y Comunicaciones

Esta tercera capa conjuga dos funcionalidades, la persistencia propiamente dicha de los datos en el dispositivo y la comunicación hacia el servidor central.

Los datos que se persisten en el dispositivo son los asociados a las actividades desarrolladas, los datos de identificación del usuario y los distintos parámetros del sistema.

En cuanto a las comunicaciones, esta capa centraliza el diálogo con los servicios web del servidor.

4.4 Servidor Central de GeoTrainer

El servidor central cumple varios papeles importantes, por un lado alberga a través del software de base *ArcSDE* la *GeoDatabase*²⁶, que es el soporte de todos los datos del sistema.

Por otro lado, publica un *WebService*²⁷ que brinda una serie de funciones a través de las cuales se canaliza la mayor parte de la comunicación entre el aplicativo y el servidor. La restante parte de la comunicación entre ambos es la relacionada al mapa de referencia que utiliza **GeoTrainer**. El acceso a éste es resuelto por los componentes *ArcGIS Mobile SDK* utilizados.

Tanto el *WebService* como el mapa publicado por el servidor acceden a los datos a través de dos **servicios geográficos** brindados mediante el software *ArcGIS Server*. Uno de ellos brinda el mapa de referencia mencionado anteriormente y el otro es utilizado por el *WebService* y permite el almacenamiento y lectura de todos los datos generados por el aplicativo (*Actividades, Competencias, Información de usuarios, Amistades, etc.*).

4.4.1 GeoDatabase

La *GeoDatabase* soportará la persistencia de los datos de todos los usuarios de **GeoTrainer**. En ella se guardarán los siguientes datos:

- Registro de usuarios
- Actividades realizadas
- Competencias deportivas
- Amistades
- Participantes de competencias

Para almacenar estos elementos, la *GeoDatabase* consta de tres capas geográficas y dos tablas de datos descriptivos

²⁶ GeoDatabase – Base de Datos Geográfica.

²⁷ WebService – Servicio Web.

4.4.1.1 Capas geográficas

Las capas geográficas en la *GeoDatabase* estarán agrupadas en un *FeatureDataset*²⁸, llamado **GeoTrainerData** y su sistema de coordenadas será el estándar WGS84²⁹, compatible con las coordenadas que transmiten los dispositivos GPS.

Las capas son:

ActivityPaths

Esta capa almacenará los datos asociados a las distintas actividades realizadas y publicadas por los usuarios. Sus elementos tendrán geometría de tipo **POLYLINE**.

Events

Esta capa almacenará los datos asociados a las competencias deportivas que ofrecerá **GeoTrainer** a los usuarios. Sus elementos tendrán geometría de tipo **POLYLINE**.

OnlineUsers

Esta capa será la encargada de almacenar las posiciones de los usuarios cuando éstos están participando en una competencia. Los elementos en ella almacenados serán de tipo **POINT**.

4.4.1.2 Tablas

Users

Esta tabla almacenará los datos básicos del registro de usuarios como ser el nombre de usuario, clave y *nickname*.

UserFriends

Esta tabla mantendrá todo lo relacionado a los amigos y solicitudes de amistad de un usuario.

4.4.2 WebServices: GeoTrainerWS

Los servicios web del servidor **GeoTrainerWS**, fueron estructurados de manera análoga al aplicativo siguiendo el modelo de capas.

Interfaz de comunicación

Lógica de los servicios

Acceso a datos

²⁸ Un *FeatureDataset* es una colección de capas geográficas que comparten un mismo sistema de coordenadas.

²⁹ WGS84 - World Geodetic System 84 (Sistema Geodésico Mundial 1984) es un sistema de coordenadas cartográficas mundial que permite localizar cualquier punto de la Tierra.

4.4.2.1 *Capa 1: Interfaz de comunicación*

Esta es una capa muy delgada que simplemente se encarga de la recepción de las peticiones que realiza el aplicativo y el traslado de las mismas hacia la capa de lógica. Los servicios brindados se resumen en los siguientes grupos funcionales:

- Gestión de usuarios
- Manejo de actividades
- Competencias deportivas
- Administración de amistades
- Cálculos

4.4.2.2 *Capa 2: Lógica de los servicios*

Esta capa lleva a cabo la lógica de los distintos servicios, la misma es muy sencilla debido a que las distintas operaciones se pueden traducir en operaciones básicas de bases de datos (altas, bajas, modificaciones y consultas).

4.4.2.3 *Capa 3 – Acceso a datos*

Aquí únicamente se hace efectivo el acceso al repositorio de datos.

4.4.3 *Servicios geográficos*

Los servicios geográficos que se publican en el servidor son los siguientes:

4.4.3.1 *Mapa de referencia: GeoTrainer*

Este servicio es accedido de manera transparente por el aplicativo **GeoTrainer** a través de los componentes de *ArcGIS Mobile SDK* y ofrece el mapa de la ciudad de Montevideo en formato de imagen a varias escalas.

La razón para publicar el mapa en formato de imagen y no vectorial es, por un lado evitar el manejo y compaginación del mapa a partir de grandes volúmenes de datos por parte del dispositivo, modalidad que fue probada y en la práctica demostró ser un cuello de botella, y por otro lado, quizás la razón más importante, para evitar la descarga de los datos vectoriales al propio dispositivo, lo cual significa un riesgo de seguridad ya que ICA hace uso comercial de ellos.

4.4.3.2 *Datos de GeoTrainer: GeoTrainerData*

A través de este servicio se realiza el acceso a todos los datos almacenados en la *GeoDatabase* de los usuarios. Este servicio es utilizado por los servicios web **GeoTrainerWS**.

5 Implementación

5.1 Introducción

En este capítulo se describe todo lo relacionado con la etapa de implementación de **GeoTrainer**, la cual involucró no solo el desarrollo del aplicativo y de los servicios que el **Servidor Central de GeoTrainer** debía ofrecer, sino también toda la puesta en marcha (instalación y configuración) de dicho servidor.

5.2 Infraestructura de ejecución y desarrollo

La puesta en marcha del **Servidor Central de GeoTrainer** involucró la creación de una máquina virtual ya que ICA no tenía la posibilidad de ofrecer un servidor dedicado para este proyecto. A su vez se debieron instalar y configurar los siguientes aplicativos y software provistos por ICA.

Software de base

- Sistema Operativo Windows Server 2008 Enterprise Edition [100]
- Microsoft SQL Server 2005 [101]

Software específico de Sistemas de Información Geográfica

- ArcGIS Server 9.3 [102]
- ArcSDE 9.3 [103]
- ArcGIS Desktop 9.3.1 [104]

Software de desarrollo y gestión de código

- Microsoft Visual Studio 2008 Team Foundation Server [105]
- Microsoft Windows Sharepoint Services 3.0 [106]

Software de diseño gráfico y edición de imágenes

- Paint.NET [107]
- GIMP [108]

Además, mediante el uso de los servicios de **DynDNS.com**³⁰ [109], se registró de manera gratuita el dominio web **gtserver.selfip.org** y se lo vinculó con el servidor instalado en la máquina virtual para acceder de manera más amigable, evitando el uso de la dirección IP del mismo.

³⁰ DynDNS (Dynamic Network Services, Inc.) es una compañía de Internet dedicada a soluciones de DNS (Domain Name System) en direcciones IP dinámicas. Ofrece servicios gratuitos de redirección a IP de subdominios de una gran lista de nombres disponibles.

5.3 Lenguaje de programación y entorno de desarrollo

GeoTrainer fue desarrollado en lenguaje *C#.NET* utilizando *Visual Studio 2008* como entorno de desarrollo.

Se utilizó además la infraestructura de *Team Foundation Server* de *Microsoft* para realizar la gestión de código y permitir, de manera segura y ordenada, la implementación en paralelo por parte de los integrantes del grupo.

5.4 Estructura de la solución

La implementación de **GeoTrainer** se estructuró en una única solución *Visual Studio* que referencia a cuatro proyectos distintos.

- **GeoTrainer**
Este es un proyecto del tipo *Smart Device Application*³¹ y corresponde a la implementación del aplicativo móvil.
- **GeoTrainerWS**
Este proyecto, de tipo *ASP.NET Web Service Application*³², implementa los servicios web del servidor central.
- **GeoTrainerDataTypes**
Proyecto de tipo *Class Library*³³ donde se definen todas las estructuras de datos que se utilizan los proyectos anteriores.
- **GeoTrainerSetup**
Este último proyecto es de tipo *Smart Device CAB Project*³⁴ y es el que genera el instalador del aplicativo móvil.

³¹ Smart Device Application – Aplicación para dispositivos móviles

³² ASP.NET Web Service Application – Aplicación que publica funciones y servicios a través de Internet.

³³ Class Library – Librería de clases

³⁴ Smart Device CAB Project – Proyecto para la instalación de un aplicativo

5.5 GeoTrainer

A pesar de que la última versión del *Compact Framework .NET* es la 3.5, debido a restricciones por parte de las componentes *ArcGIS Mobile SDK*, este proyecto es desarrollado para la versión 2.0 del mismo. Sin embargo este condicionamiento no supuso dificultades extras al desarrollo del aplicativo.

5.5.1 Capas & Namespaces

Siguiendo las especificaciones de diseño, este proyecto se estructuró en una serie de *namespaces* y clases de manera de respetar lo mejor posible las responsabilidades de cada capa.



Jerarquía de namespaces del proyecto GeoTrainer



Asociación de los namespaces a las capas de la aplicación

5.5.2 Capa 1: Interfaz de Usuario

Namespaces

```
GeoTrainer.UI
GeoTrainer.UI.Components
GeoTrainer.UI.Components.AlphaImage
GeoTrainer.UI.Components.CharManager
GeoTrainer.UI.Components.ScrollList
GeoTrainer.UI.Components.ScrollList.ListItems
GeoTrainer.UI.Components.ScrollList.ScrollPanels
```

Tal como lo requería el proyecto, el aplicativo debía contar con una interfaz de usuario novedosa, que resultara atractiva a los usuarios en general.

Por eso mismo se tuvieron en cuenta varios factores y sugerencias al momento de diseñar las pantallas o visualizaciones dentro del aplicativo, basados principalmente en los principios y buenas prácticas que sugiere el área de HCI³⁵ [110]

A continuación se enumeran algunas características que se tuvieron en cuenta al momento de diseñar la interfaz.

- A partir del estudio de las tecnologías con las que se desarrolla el aplicativo y aunque las mismas presenten algunas restricciones, se decide construir las interfaces gráficas mediante el uso de mapas de bits (*bitmaps*) ya que no se cuenta con otro tipo de funcionalidades gráficas de alto nivel.
- Dadas las dimensiones pequeñas de la pantalla del dispositivo y dado que el usuario va a utilizar el aplicativo mayoritariamente mientras realiza alguna actividad deportiva, se optó por crear botones y controles en general, de tamaño lo suficientemente grande como para poder hacer uso de ellos utilizando los dedos sin necesidad de utilizar un lápiz o *stylus*.
- Por las mismas razones se decidió mostrar los contadores y valores durante la actividad en un tamaño que sea fácilmente visible por el usuario en el momento de la misma.
- Se utilizaron transparencias en los paneles y botones que se despliegan sobre el mapa, para que éste ocupe la mayor porción de la pantalla del dispositivo y se pueda visualizar a pesar de estar por debajo de algún control o panel.
- Dado que la mayoría de los dispositivos con pantalla táctil carecen de teclados físicos, el ingreso de texto es todo un desafío. Para facilitar en este aspecto se implementó un mecanismo de entrada de texto especial mediante un control único, ajustable al *teclado virtual (teclado en pantalla que proporciona la plataforma)*, para los casos en que no se podía prescindir de ello (*ingreso de texto libre*) y utilizar otro tipo de controles como los botones en los casos que se pudiera suplantar este método (*selección del deporte o ajuste del tiempo para una competencia, por ejemplo*), reduciendo de esta manera errores y mecanismos de control para el ingreso de información inconsistente (*letras cuando se esperan números, por ejemplo*).
- Para optimizar el tamaño de la pantalla y para poder mostrar más información, se implementaron controles desplazables (*scroll*) para los listados y los detalles de actividades y eventos.

³⁵ HCI – Human Computer Interaction

- Se asignaron colores a las distintas funcionalidades del aplicativo y se diseñaron las pantallas en base a ellos de manera de brindar información de contexto rápida al usuario.
- Se utilizó una iconografía que representa los conceptos de forma intuitiva y consistente en todas las interfaces del dispositivo. (Ver sección 9 Iconografía, del **Anexo IV: Manual de Usuario**)
- Se agruparon las funcionalidades del aplicativo y se les asignó un color correspondiente a cada una para que el usuario, de esta forma, pueda asociar los colores a conceptos y tipos de actividad mientras navega por las distintas pantallas.



Colores asociados a los distintos conceptos o funcionalidades del aplicativo

Una explicación más detallada de estos puntos y ejemplos de los mismos pueden consultarse en los anexos: **Anexo III: Diseño del Sistema** y **Anexo IV: Manual de Usuario**.

GeoTrainer.UI

En este *namespace* se definen todos los formularios (pantallas) que conforman la interfaz del aplicativo.

La solución implementada involucró la utilización de métodos de programación gráfica como ser el uso de *OffScreen Buffers* (o *Double Buffers*) [111]. Este método es comúnmente utilizado en aplicaciones gráficas como los juegos para lograr pantallas con animaciones suaves y fluidas evitando lo más posible efectos como el parpadeo. Como el armado de cada cuadro de pantalla puede consistir en el manejo e impresión de muchos objetos (personajes de un video juego, botones, paneles, imágenes, etc.) este método consiste básicamente en confeccionar cada cuadro de imagen completo en memoria evitando hacerlo directamente sobre la pantalla. Una vez armada, mediante una única operación se 'copia' ese cuadro completo a pantalla. Esto se debe a que las operaciones de modificación de pantalla son más lentas que sobre memoria.

GeoTrainer.UI.Components

Aquí se implementan una serie de componentes gráficos que son utilizados en los formularios anteriormente mencionados.

Se desarrollaron los siguientes componentes:

- *MessageBox* - Dialogo de mensaje. Permite desplegar notificaciones o realizar preguntas al usuario en pantalla y aguardar por una respuesta. Se desarrollaron dos tipos distintos de *MessageBox*. Uno que simplemente notifica y brinda un botón de OK, y otro que permite al usuario contestar por SI o por NO a una pregunta.
- *InputBox* - Dialogo para la entrada de datos. Es el componente utilizado para el ingreso de texto y trabaja en conjunto con un teclado virtual en pantalla provisto por el sistema operativo.
- *WaitMessage* - Mensaje de espera. Componente utilizado para notificar al usuario que una operación está siendo realizada. El mismo no requiere intervención por parte del usuario, ocultándose cuando la operación finaliza.



GeoTrainer.UI.Components.AlphaImage

Debido a que el *Compact Framework .NET* no ofrece de manera nativa componentes para el correcto manejo de *bitmaps* con características tales como canal de transparencia (*alpha channel*³⁶), fue necesario desarrollarlos.

Este *namespace* publica una única clase de igual nombre, ***AlphaImage***, que representa una imagen con canal de transparencia proveniente de un archivo. Estas imágenes son las utilizadas para la creación de toda la interfaz de usuario.

Las demás clases y estructuras contenidas en este *namespace* permiten referenciar a bajo nivel a las bibliotecas del sistema operativo para la realizar la carga de estos archivos.

En el proyecto se utilizaron imágenes en formato PNG32³⁷.

³⁶ Alpha channel, refiere a tipo especial de información almacenada en un archivo gráfico que representa transparencia.

³⁷ PNG32 - Portable Network Graphic de 32 bits. Es un formato grafico basado en mapas de bits (bitmaps) en el cual se utilizan 32 bits por pixel, 24 se utilizan para la codificación RGB (Red, Green y Blue) y 8 para indicar un nivel de transparencia del pixel (256 valores posibles)

GeoTrainer.UI.Components.ChartManager

Este *namespace* brinda un componente para la generación de gráficas de líneas, de hasta dos series de datos.

GeoTrainer.UI.Components.ScrollList

Se implementan aquí las clases de base para el manejo de listas con desplazamiento (*ScrollLists*). Estas permiten el manejo de listas de varios elementos (*ScrollItems*) o de un solo elemento de proporciones mayores al espacio visible y permiten además su desplazamiento (*ScrollPanels*).

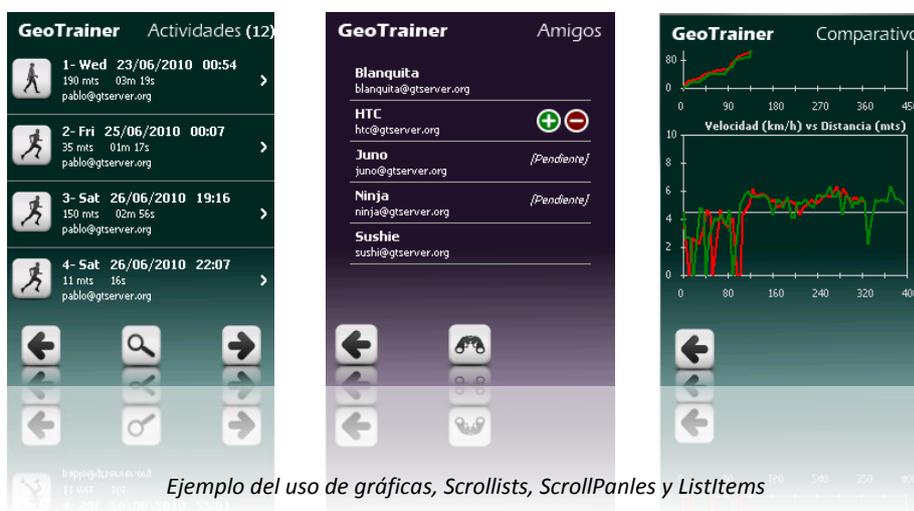
Estas listas implementan movimiento inercial desacelerando al 'soltar' el control en movimiento.

GeoTrainer.UI.Components.ScrollList.ListItems

Aquí se implementan distintos *scroll items* para utilizar como elementos de un *ScrollList*. Permiten soportar listas de actividades, amistades y competidores.

GeoTrainer.UI.Components.ScrollList.ScrollPanels

Se implementan paneles con desplazamiento para el despliegue de los detalles de las actividades y eventos deportivos y para la comparación de actividades deportivas.



5.5.3 Capa 2: Lógica de la aplicación

Namespaces

GeoTrainer.AppLogic

GeoTrainer.AppLogic

Se implementaron distintas clases siguiendo el patrón *Singleton* [112] haciendo a cada una de ellas responsable de un aspecto de la lógica del aplicativo como ser, estado de la aplicación, conjunto de amigos, configuración, datos de la actividad actual, etc.

5.5.4 Capa 3: Persistencia y Comunicaciones

Namespaces

GeoTrainer.Communication

GeoTrainer.Communication.GoogleUtils

GeoTrainer.Communication

En este *namespace* se implementan las clases que se encargan de la entrada y salida de datos del aplicativo.

Por un lado, se implementó también bajo el patrón *Singleton* una clase que centraliza todas las invocaciones a los servicios web publicados por el servidor central. Un punto a destacar es que los llamados a los servicios web se realizan en su totalidad de forma asincrónica, evitando así que el aplicativo, a los ojos del usuario, sufra paralizaciones temporales.

Y por otro lado, enmascarando un componente provisto por la biblioteca *ArcGIS Mobile SDK*, se implementó una clase que recibe y notifica los eventos provenientes del dispositivo GPS del dispositivo.

GeoTrainer.Communication.GoogleUtils

Como al inicio del proceso de implementación no se contaba con acceso a un mapa de buena calidad y compatible con las bibliotecas utilizadas, se implementó, para verificar los datos capturados un componente que consulta el servicio de *Google Static Maps API* (sección 2.4.2.2).

Este componente genera mapas estáticos en formato PNG que no permiten interacción alguna. Es utilizado en la versión final del aplicativo únicamente para presentar el recorrido de una actividad ya realizada al visualizar sus detalles o al configurar una sesión de entrenamiento en base a ella.

5.6 GeoTrainerWS

Si bien el desarrollo de este proyecto es independiente del desarrollo del aplicativo **GeoTrainer**, se decidió con el único fin de mantener la uniformidad de los componentes, desarrollarlo sobre la versión 2.0 del *.NET Framework* de *Microsoft*, aunque bien podría haberse hecho sobre una versión más actual, 3.5 por ejemplo.

Siguiendo las especificaciones de diseño, el desarrollo de los servicios web se estructuró en tres clases donde cada una de ellas implementa una capa correspondientemente.

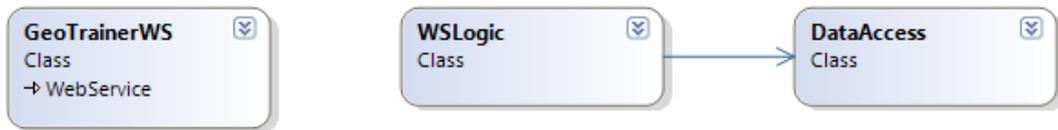


Diagrama de clases del proyecto GeoTrainerWS



Grupos funcionales y funciones ofrecidas por el servicio GeoTrainerWS

A continuación se detallan cada una de las tres capas de diseño y sus correspondientes implementaciones.

5.6.1 Capa 1: Interfaz de comunicación

La clase **GeoTrainerWS** es la responsable de la interfaz entre las funciones brindadas por el servicio y las peticiones de los dispositivos móviles. Básicamente deriva dichas peticiones a la clase **WSLogic** encargada de la lógica de cada uno de las funciones.

5.6.2 Capa 2: Lógica de los servicios

Debido a que cada una de las funciones ofrecidas por el servicio web son de cierta manera operaciones básicas de acceso a base de datos (altas, bajas, modificaciones y consultas), la lógica de las distintas funciones base de **WSLogic** es muy sencilla.

Sin embargo, debido al diseño de la capa de acceso a datos, es necesario realizar una preparación tanto de los parámetros de entrada como los de salida para poder invocar sus funciones.

5.6.3 Capa 3: Acceso a datos

DataAccess es la clase responsable de esta funcionalidad, resuelve de manera genérica el almacenamiento y lectura de los datos en la *GeoDatabase* a través de *ArcGIS Server* y del servicio de datos geográfico *GeoTrainerData* presentado con anterioridad.

El diseño es genérico en el sentido que sus funciones no tienen en cuenta sobre cual tabla o capa se realiza una operación. La tabla o capa objetivo es especificada como uno de los parámetros de la operación.

Debido a la naturaleza *stateless*³⁸ de los servicios web, cada función se encarga de la conexión y desconexión del servidor.

5.7 GeoTrainerDataTypes

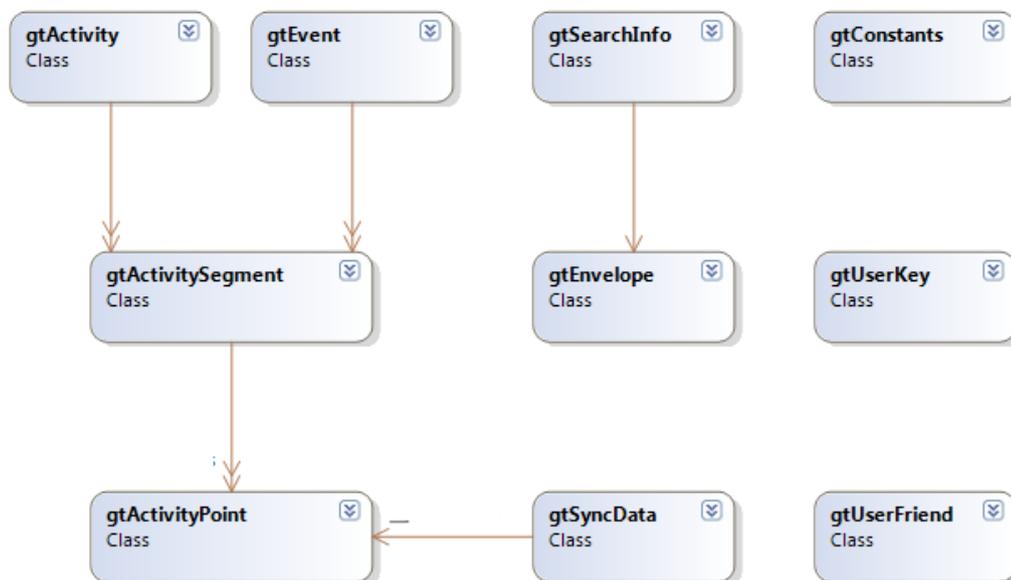
Este proyecto es una biblioteca de tipos de datos para su uso compartido tanto por parte del proyecto *GeoTrainer* como por *GeoTrainerWS*.

Tipos de datos implementados:

- **gtActivity**. Define una actividad deportiva, la cual además de datos generales como la fecha y hora de realización, el usuario o el deporte entre otros, conforma su recorrido como una lista ordenada de segmentos.
- **gtActivitySegment**. Define un segmento de un recorrido, es una simple colección de puntos.
- **gtActivityPoint**. Define un punto geográfico. Este punto mantiene, además de sus coordenadas geográficas, todos aquellos valores relevantes como ser la altura del terreno, la velocidad en ese punto, la dirección (0 a 360 grados), la distancia y el tiempo acumulado.
- **gtConstants**. Conjunto de constantes utilizadas para la codificación de distintos valores.
- **gtEnvelope**. Define una extensión geográfica rectangular.
- **gtEvent**. Análogo a una actividad, define todo lo relacionado a una competencia deportiva. Mantiene datos generales como el nombre de la competencia, la fecha y hora de realización, el deporte y la distancia entre otros. Define también su recorrido como una lista ordenada de segmentos.
- **gtSearchInfo**. Es una estructura para el pasaje de los parámetros utilizados en los servicios de consulta. Contiene distintos valores que son tenidos en cuenta para crear los criterios de filtrado de la consulta.
- **gtSyncData**. Estructura utilizada para la sincronización de las posiciones de los usuarios durante una competencia. Contiene el identificador del usuario y la posición del mismo.
- **gtUserFriend**. Define una relación de amistad, mantiene el usuario amigo y el estado de la relación (Confirmada, Ignorada, Autorización requerida, Autorización pendiente).

³⁸ Stateless – Sin estado. Hace referencia a servicios web donde no se mantiene en el servidor, ningún tipo de estado entre distintas peticiones de un mismo usuario.

- **gtUserKey**. Estructura de control que viaja en todas las solicitudes entre el aplicativo y los servicios web. Básicamente es la dupla *username-password*, pero contiene como dato adicional el *nickname* del usuario.



5.8 GeoTrainerSetup

Este proyecto simplemente genera el instalador para el aplicativo móvil. Toma la salida del compilador de **GeoTrainer** y lo empaqueta junto con todos aquellos archivos de recursos necesarios para su ejecución (imágenes, sonidos, archivos de configuración, etc.)

6 Testeo y verificación

6.1 Testeo

En primer lugar se destaca que todas las pruebas llevadas a cabo fueron de índole funcional y no fue posible realizar tests de stress del sistema. Esto se debió a limitaciones propias de la infraestructura utilizada para la puesta en marcha del sistema, principalmente en lo que respecta al servidor central y a la no disponibilidad de múltiples dispositivos móviles con los que realizar pruebas reales.

Por un lado ICA no disponía de una máquina exclusiva para este proyecto, por lo tanto el servidor central fue implementado mediante una máquina virtual, que fue alojada en un servidor compartido con otras máquinas virtuales y servicios. Esto de por sí implica una degradación importante en el desempeño del servidor. A lo anterior se le suman las exigentes características de hardware requeridas por los distintos softwares instalados, desde el sistema operativo, manejador de base de datos, software geográfico hasta los componentes de desarrollo y gestión de código.

Por otro lado, el servidor fue puesto en marcha sobre la infraestructura de servidores de ICA y fue habilitado el acceso desde internet hacia él a través de servicio ADSL con el que ICA cuenta y cuyo ancho de banda es compartido con todas las aplicaciones que ésta provee a distintos clientes y por el uso diario del personal de la empresa.

Se entiende en principio que las condiciones anteriormente detalladas no son las ideales para que el servidor pueda responder de forma aceptable en caso de recibir un gran número de peticiones simultáneas. De todos modos y a pesar de ello fue posible realizar pruebas en profundidad de todas las funcionalidades del aplicativo en sí.

Por un lado el grupo de trabajo contaba con un dispositivo *Windows Mobile*, con plan de conexión a Internet con el cual se pudo probar el aplicativo en situaciones lo más reales posibles.

Además *ArcGIS Mobile SDK* incorpora componentes de simulación de dispositivos GPS a partir de datos almacenados en archivos. Esto, en combinación con el emulador de dispositivos provisto por el entorno de desarrollo *Visual Studio*, permitió la ejecución de múltiples instancias del aplicativo, situación que resultó especialmente útil para la realización de competencias.

En cuanto a la realización de actividades deportivas, si bien por una razón de practicidad no se efectuaron muchas actividades de gran distancia, algunas fueron registradas para descartar inconvenientes tanto en la captura de los datos como en el almacenamiento y recuperación de los mismos.

Para el caso de las competencias deportivas, se realizaron test que involucraron simultáneamente hasta cinco competidores, un dispositivo físico y cuatro emuladores, compartiendo su avance a lo largo de una competencia.

En este caso se encontró aceptable el tiempo de respuesta del sistema, entendiendo como aceptable que la duración del intervalo entre sincronizaciones con el servidor era muy superior al tiempo requerido por cada petición. El software se configuró con intervalos de sincronización de 60 segundos y el tiempo de respuesta promedio se mantuvo en 5 segundos aproximadamente, nunca sobrepasando los 15 segundos.

Otro aspecto importante es que debido a la invocación de los distintos servicios web de manera asincrónica, la interfaz del aplicativo respondía de manera adecuada sin 'congelarse' temporalmente, se mantenía el registro del tiempo y de la posición del usuario a través del GPS de manera consistente y continua.

6.2 Verificación

En líneas generales el aplicativo respondió de la manera esperada no encontrándose en su versión final *bugs* aparentes. Todos aquellos *bugs* detectados se corrigieron durante el proceso de desarrollo y testeo.

7 Conclusiones y trabajo futuro

7.1 Conclusiones

El proceso llevado a cabo abarcó el estudio de numerosas áreas tecnológicas, estudio que se plasmó resumidamente en el capítulo nº 2 del presente documento y más detalladamente en el **Anexo I: Estado del Arte**.

Luego se realizó una etapa de análisis, identificación y especificación de los requerimientos planteados por ICA que culminó con el documento **Anexo II: Análisis y Especificación de Requerimientos** y que se resume en el capítulo nº 3 de este documento.

El desarrollo del presente proyecto se vio afectado por una serie de contratiempos que se detallan en 7.1.1 y que hicieron que el mismo se extendiera más allá de los plazos originalmente planificados. Esto derivó en un proceso de negociación del alcance de los requerimientos identificados. En algunos casos significó la simplificación del requerimiento, en otros incluso la no implementación del mismo.

La etapa de implementación presentó nuevos desafíos, tanto a nivel de las herramientas que hubo que estudiar, como de decisiones de implementación que debieron ser tomadas debido principalmente a los contratiempos mencionados anteriormente.

Finalmente y como principal conclusión se desea destacar que el proyecto finalizó con la implementación exitosa de un sistema completo, incluyendo no sólo al aplicativo móvil **GeoTrainer** sino también a todos los servicios de base que conforman al **Servidor Central de GeoTrainer**. El aplicativo fue presentado a personal de ICA en dos ocasiones, recibiendo en la última una muy buena aceptación.

Cómo segunda conclusión del trabajo, se plantean respuestas a los objetivos secundarios perseguidos por ICA, que se mencionan en **1.4 Objetivos y resultados esperados** y que se resumen en:

- Estudio de *ArcGIS Mobile SDK* en aplicativos de uso masivo.
- Estudio e implementación de un mecanismo para la comunicación y sincronización de información entre varios dispositivos móviles.
- Esquemas y canales de comercialización aplicativos para dispositivos móviles.

En las siguientes secciones se detalla cada uno de los puntos presentados anteriormente.

7.1.1 Duración y contratiempos

En primer lugar, al momento de la propuesta del proyecto, fue asumido por parte de ICA que ciertos componentes de software estarían disponibles para ser utilizados en la implementación del aplicativo. En particular, se contaba con poder utilizar la versión para dispositivos móviles de la plataforma *Silverlight* [113] de Microsoft que formaría parte del sistema operativo próximo a lanzarse, *Windows Mobile 6.5*. El uso de esta tecnología era fundamental para el diseño de interfaces gráficas ya que brinda componentes para el manejo de gráficos vectoriales, efectos visuales, animaciones y funcionalidades multimedia.

Lamentablemente, no sólo el lanzamiento de la versión 6.5 de *Windows* para dispositivos móviles sufrió gran retraso siendo liberada recién a fines del año 2009 [114][115], sino que finalmente *Silverlight* para móviles no formó parte del sistema. Éste último fue elegido para

formar parte de la siguiente versión del sistema operativo, *Windows Phone 7* [116] y que será lanzada recién entre octubre y noviembre de 2010 [117].

En segundo lugar, si bien el desarrollo de los servicios que brindaría el **Servidor Central de GeoTrainer** formaba parte del proyecto, fue necesario además realizar la instalación de todos los componentes de base del servidor, incluidos el sistema operativo, los aplicativos geográficos de base y el entorno de desarrollo y gestión de código, que en un principio serían provistos por ICA.

Debido a estos contratiempos se debieron tomar ciertas decisiones al momento de comenzar con la implementación que se detallan a continuación. Algunas significaron la modificación de algunos requerimientos y otras el estudio de alternativas para lograr los objetivos planteados.

7.1.2 Ajustes del alcance

Como consecuencia directa de los contratiempos presentados en el punto anterior, fue necesario revisar el alcance de algunos de los requerimientos especificados. En algunos casos significó la no implementación del requerimiento, en otros una simplificación del mismo. En contrapartida se incorporaron otras características al aplicativo cuyo desarrollo no implicaba complejidad y se entendía que brindaban valor agregado.

- **Requerimientos modificados**

Parámetros del sistema. En este caso, si bien se cuenta con una serie de parámetros almacenados en un archivo de configuración, el aplicativo simplemente los lee al inicio pero no brinda una interfaz apropiada para su ajuste.

Competidor virtual más realista. En el caso de un entrenamiento a partir de una actividad previa, el competidor virtual debería ser simulado siguiendo los datos precisos de esa actividad de referencia, respetando por ejemplo, las variaciones de velocidad. Se decidió, para simplificar en parte esta funcionalidad, que el avance del competidor virtual se realice a velocidad constante a lo largo del recorrido de referencia de acuerdo al tiempo objetivo marcado.

- **Requerimientos no implementados**

Información de referencia. Este requerimiento no se implementó. Consistía en permitir durante la realización de una actividad en modalidad *Exploración*, el ingreso de datos auxiliares que complementen un recorrido. Por ejemplo, permitir tomar una fotografía, ingresar algún comentario escrito o de voz y asociarlo a algún punto sobre el recorrido realizado, para luego ser visualizados durante la planificación de un entrenamiento o competencia.

Cálculo de calorías consumidas. Esta funcionalidad no formaba parte original de los requerimientos propuestos por ICA, sin embargo fue propuesto por el grupo de desarrollo entendiendo que aportaba al producto. Fue incorporada al análisis inicial presentado en el **Anexo III: Análisis y especificación de requerimientos**, aunque finalmente no fue implementada debido a la falta de información disponible para realizar un cálculo preciso y no una simple relación entre el deporte y tiempo de actividad.

- **Requerimiento agregado**

Feedback sonoro. Si bien no era parte de los requerimientos iniciales, ésta fue una funcionalidad que el grupo de desarrollo decidió incorporar al aplicativo a poco tiempo de finalizar la implementación. Fue muy bien vista por parte de ICA, en el entendido que realmente aporta al funcionamiento y distingue a **GeoTrainer** de muchos de los aplicativos similares actualmente disponibles. Consiste en el uso de distintas alertas sonoras para reportar al usuario, en intervalos regulares de tiempo acerca de su desempeño con respecto a los objetivos de entrenamiento o competencia planteados. Por ejemplo, si está avanzando por debajo o por arriba del tiempo esperado, si mejoró o empeoró con respecto al último reporte recibido.

7.1.3 Interfaz de usuario

La interfaz de usuario era uno de los desafíos del presente proyecto y por lo tanto un punto que **ICA no quería sacrificar** a pesar de no contar para el desarrollo con las herramientas de *Silverlight*.

Se decidió entonces de común acuerdo con ICA invertir más tiempo en el estudio de otras alternativas que permitiesen obtener un aplicativo que, por su riqueza gráfica, resulte atractivo al público en general.

En este sentido, se llegó a una solución que requirió de bastante estudio y trabajo debido a las restricciones presentadas por la plataforma utilizada. Involucró el uso de funciones de bajo nivel de la API del sistema operativo del dispositivo, funciones que normalmente no son utilizadas cuando se trabaja con un *framework* de más alto nivel como es el *Compact Framework .NET*.

Se implementaron componentes de bajo nivel para la carga y el uso de archivos gráficos en formato PNG32 teniendo en cuenta la información de transparencia incluida en la imagen.

Además se desarrollaron componentes gráficos de alto nivel para sustituir los controles nativos provistos por el *framework* de desarrollo como ser *ScrollLists*, *Gráficos de líneas*, *MessageBox*, *InputBox* y otros, presentados en el capítulo nº 5 de este documento.

La solución implementada involucró la utilización de métodos de programación gráfica como ser el uso de *OffScreen Buffers* (o *Double Buffers*) para lograr pantallas con animaciones suaves y fluidas evitando lo más posible efectos como el parpadeo.

7.1.4 Feedback de ICA

Cuando la etapa de implementación se encontraba suficientemente avanzada, el aplicativo fue presentado a ICA, estando presentes el tutor del proyecto y el responsable por parte de la empresa.

En esta ocasión, si bien se plantearon detalles a corregir, el trabajo realizado fue muy bien visto y se solicitó al grupo la posibilidad de realizar una nueva presentación del aplicativo a otras personas una vez finalizado el desarrollo del mismo.

7.1.5 Presentación final

Una vez ajustados los detalles pendientes y las observaciones relevadas durante la primer reunión se realizó una nueva presentación del aplicativo. Esta vez, además de estar presentes los asistentes anteriores, participaron personal de marketing de ICA, personal del departamento de desarrollo y se invitó a un representante de la empresa *Microsoft* debido a la relación comercial existente entre ambas empresas y el uso de productos de la misma en el desarrollo.

El aplicativo recibió una muy buena aceptación por parte de los asistentes.

Algunas de las observaciones, gratificantes por sobre todo, fueron:

- La conformidad por parte de ICA con el producto y resultados obtenidos.
- El reconocimiento por parte del representante de *Microsoft* sobre el manejo de bajo nivel realizado en lo visual, debido a las limitantes que en esta área presentaba la plataforma 6.x de *Windows Mobile*.

7.1.6 Objetivos alcanzados

Tal como se mencionó anteriormente, el primer objetivo alcanzado fue la implementación completa del sistema **GeoTrainer** el cual involucra al aplicativo para dispositivos móviles y a los servicios del servidor central. Esta implementación se describe con profundidad a lo largo de este documento y se complementa con los anexos que lo acompañan.

En segundo lugar se plantean respuestas a los objetivos secundarios perseguidos por ICA.

Utilización de ArcGIS Mobile SDK en aplicativos de uso masivo

Este punto representa quizás la conclusión más importante dentro del conjunto de objetivos perseguidos.

A pesar de que el SDK utilizado resuelve de manera transparente para el desarrollador un conjunto importante de funcionalidades como ser el consumo de mapas y el manejo del dispositivo GPS, se entiende que no es éste el tipo de aplicativos en los cuales se puede sacar mayor provecho de él.

En primer lugar el SDK presenta un punto en contra. Requiere para su funcionamiento acceder a datos y mapas almacenados en un servidor *ArcGIS Server*, un software comercial de alto costo.

Este SDK está principalmente orientado a aplicaciones geográficas corporativas, para la edición de grandes volúmenes de datos en campo y maneja nativamente los datos en formato vectorial y separados en distintas capas de información, modalidad que no fue ni requerida ni aprovechada en un aplicativo como **GeoTrainer**. En particular **GeoTrainer** utiliza el mapa como un elemento exclusivamente de referencia.

A su vez y apuntando a la masividad esperada de un aplicativo como **GeoTrainer**, los mapas a visualizar en el dispositivo deberían ser globales, es decir deberían cubrir todo el mundo. Debido a la imposibilidad de utilizar los servicios *ArcGIS Online Services* que proveen información a nivel mundial, que si bien son compatibles con el servidor geográfico *ArcGIS Server* no lo son con los componentes móviles *ArcGIS Mobile SDK*; en este proyecto **se cubrió únicamente la ciudad de Montevideo** con un mapa de calles y nueve niveles de escala implicando el almacenamiento de cerca de **1.26Gb de información en formato comprimido**. Estos valores hacen pensar que de orientar **GeoTrainer** al público general y querer cubrir el planeta con información geográfica, tanto a nivel de calles como utilizando imágenes satelitales, los requerimientos de almacenamiento ser harían inmanejables.

Existiendo en el mercado opciones gratuitas como ser *Google Maps*, *Yahoo! Maps* o *Bing Maps* [118], que brindan acceso a información geográfica a nivel mundial, con mapas cartográficamente diseñados, incluso de imágenes satelitales y que no requieren almacenamiento extra, se entiende que bien vale el esfuerzo de desarrollar aquellas funcionalidades que facilita el SDK evaluado y prescindir del mismo. Estas funcionalidades involucran entre otras: el manejo dinámico del mapa permitiendo la interacción con el mismo y la captura de la información proveniente de los dispositivos GPS.

Por otro lado, el uso de este SDK impone algunas restricciones al proceso de desarrollo. Sin duda la mas importante de ellas es la imposición de plataforma *Windows Mobile* ya que la misma no solo cubre una parte menor, un **7%** del mercado actual de dispositivos móviles (ver **2.2.3 Sistemas Operativos**), sino que en este momento está siendo sustituida por la nueva versión del sistema operativo denominada *Windows Phone 7* con la que el SDK no es compatible.

El sólo hecho de independizarse de la plataforma y apuntar a cubrir otras como ser los móviles *iPhone* o los basados en *Google Android* implicaría abarcar un **25%** adicional del mercado sin tener en cuenta el tentador **44%** acaparado por los móviles basados en el sistema operativo *SymbianOS*.

Sincronización de información entre múltiples dispositivos móviles

Se diseñó un esquema para la sincronización de manera que sea mínimo el tamaño de los mensajes entre los usuarios y el servidor con el fin de aprovechar de la mejor manera posible el canal de comunicación y además disminuir en parte las limitantes impuestas por la infraestructura utilizada, es decir la implementación del servidor central mediante una máquina virtual sobre hardware y conexión a internet compartida.

Se entiende que esta forma de implementar los mensajes no solo saca provecho de las condiciones en las que se desarrolló el actual proyecto sino también de una configuración del servidor orientada a la puesta en producción del sistema, es decir evitando un ambiente virtualizado, mejores características de hardware y mejores condiciones de ancho de banda.

Canales de comercialización aplicativos para dispositivos móviles

En cuanto a los canales de comercialización, no son muchas las alternativas. Si bien siempre es posible comercializar el aplicativo por los medios tradicionales, es decir que sea ofrecido por la propia empresa desarrolladora a través de su sitio web o mediante el uso de propaganda, se encuentra muy favorable el uso de los canales estándar para este tipo de aplicaciones.

Estos canales o tiendas en línea dirigidos cada uno a una plataforma específica, resuelven no solo la distribución del aplicativo sino quizás lo más importante, brindan los servicios para el proceso completo de venta y cobranza de las aplicaciones.

En contrapartida tienen costo. Generalmente el esquema utilizado es el del cobro de un arancel fijo anual que habilita la publicación de uno o varios aplicativos en la tienda y de la distribución de la ganancia generada por las ventas, comúnmente un **70%** corresponde al desarrollador y un **30%** a la tienda como se presentó en el capítulo nº 2 de este documento y en el **Anexo I: Estado del Arte**.

Las principales tiendas en línea de aplicaciones móviles son:

- Apple AppStore
- Android Market
- Windows Marketplace for Mobile

7.2 Trabajo futuro

Múltiples son las modificaciones que a futuro pueden realizarse sobre el producto desarrollado. Algunas involucran funcionalidades totalmente nuevas, otras significan mejoras sustanciales a las actualmente presentes.

A continuación se enumeran algunas de las que se han visto como más interesantes y que se anexan a aquellos puntos presentados anteriormente en **7.1.2 Ajuste del alcance** y que no se pudieron desarrollar o no se desarrollaron completamente por falta de tiempo.

- **No depender de ArcGIS Mobile SDK.** El uso de estos componentes geográficos fue uno de los requerimientos planteados por ICA. Tiene algunas ventajas ya que enmascaran el manejo del dispositivo GPS y facilitan el despliegue y consulta de mapas dentro de la aplicación. Sin embargo, tienen la desventaja que implican el uso de un servidor de mapas propietario, como es *ArcGIS Server* no pudiendo consumir servicios de otros proveedores gratuitos como ser *Google Maps*, *Yahoo! Maps* o *Bing Maps*.
- **Extensión a otras plataformas móviles.** *ArcGIS Mobile SDK* sólo funciona en dispositivos Windows Mobile. Resolver el punto anterior, habilitaría la posibilidad de migrar el aplicativo de manera de abarcar otras plataformas móviles. **GeoTrainer** fue desarrollado en lenguaje C# y debido a la similitud sintáctica de éste con otros lenguajes como ser *Java* o *C*, resulta tentador la reutilización del código para extender el uso de **GeoTrainer** a dispositivos como ser el *iPhone* o aquellos basados en el sistema operativo *Google Android*.
- **Portal Web.** La mayoría de los aplicativos similares relevados cuentan de alguna manera u otra con una solución de este tipo. En estos casos el portal involucra un lugar centralizado donde el usuario puede consultar y visualizar de manera más cómoda sus actividades y que permite obviar en ciertas circunstancias las limitantes físicas de los dispositivos móviles, sobre todo el tamaño de la pantalla. Aprovechando que es a través del **Servidor Central de GeoTrainer** que se realiza la sincronización de datos entre los usuarios que se encuentran participando de una competencia, una funcionalidad exclusiva que podría brindar un **Portal de GeoTrainer** es la de ofrecer a los usuarios y al público en general un monitor del desarrollo de las competencias en tiempo real, desplegando sobre un mapa el avance de los usuarios a medida que reportan su posición.
- **Otros idiomas.** Poder elegir dentro de la configuración de **GeoTrainer** el idioma en que se desea visualizar su interfaz.
- **Otros deportes.** La versión actual de **GeoTrainer** maneja un conjunto de cuatro deportes, sería deseable que maneje un número arbitrario de ellos.
- **Integración con redes sociales existentes.** Al igual que otros productos similares, resulta interesante ofrecer al usuario la posibilidad de compartir los resultados del desarrollo de una actividad deportiva a través de redes sociales como *Facebook* o *Twitter*. También, la integración con estas redes sociales podría intervenir en la funcionalidad de lista de amigos. Avisar al usuario que ciertos amigos en *Facebook* también usan **GeoTrainer**, por ejemplo.

- **Servicios de mensajería.** Poder configurar al momento de realizar una competencia o un entrenamiento, el envío automático de mensajes SMS en intervalos regulares de tiempo o distancia con información acerca de lo realizado al momento. Esto puede resultar especialmente útil en competencias de largo recorrido como puede ser una maratón o una actividad de ciclismo para mantener el contacto con familiares, amigos o compañeros de equipo.
- **Incorporar el manejo de otros sensores de actividad.** Si bien el diseño actual de la base de datos que alberga la información de las actividades lo soporta, en esta versión de **GeoTrainer** el dispositivo GPS es el único (e imprescindible) sensor del que se obtiene información para asociar a cada punto de un recorrido. Mediante el esquema de *plugins*, podría ser factible la incorporación de la lectura de otros sensores que brinden información relevante como ser un sensor del ritmo cardíaco o sensores de cadencia de pedaleo (en caso de actividades ciclísticas).

8 Referencias

- [1] ESRI - Environmental Systems Research Institute. (Último acceso: Julio 2010)
<http://www.esri.com>
- [2] ICA – Ingenieros Consultores Asociados (Último acceso: Julio 2010)
<http://www.ica.com.uy>
- [3] GPS –Global Positioning System (Último acceso: Julio 2010)
<http://www.gps.gov/>
- [4] Google Earth. (Último acceso: Julio 2010)
<http://earth.google.com>
- [5] ArcGIS Mobile | Overview. (Último acceso: Julio 2010)
<http://www.esri.com/software/arcgis/arcgismobile/index.html>
- [6] A Brief Overview of GSM, 1994. (Último acceso: Mayo 2009)
<http://user.cs.tu-berlin.de/~jutta/gsm/js-intro.html>
- [7] Generaciones de la Telefonía Celular. (Último acceso: Julio 2010)
[http://www.cabinas.net/monografias/tecnologia/generaciones de la telefonía celular.asp](http://www.cabinas.net/monografias/tecnologia/generaciones%20de%20la%20telefon%C3%ADa%20celular.asp)
- [8] Cómo funcionan los teléfonos celulares. (Último acceso: Julio 2010)
<http://www.monografias.com/trabajos14/celularhist/celularhist.shtml>
- [9] What is GPRS (General Packet Radio Service)? (Último acceso: Agosto 2010)
<http://www.mobile-phones-uk.org.uk/gprs.htm>
- [10] EDGE (Tasas de datos mejoradas para la evolución de GSM), 2008. (Último acceso: Julio 2010)
<http://es.kioskea.net/contents/telephonie-mobile/edge.php3>
- [11] EMS (Enhanced Messaging Service). (Último acceso: Julio 2010)
<http://www.cellular.co.za/technologies/ems/ems.htm>
- [12] What is Multimedia Messaging Service (MMS)? (Último acceso: Julio 2010)
<http://www.mobile-phones-uk.org.uk/mms.htm>
- [13] What is UMTS™? (Último acceso: Julio 2010)
<http://www.cellular.co.za/umts.htm>
- [14] What is CDMA? (Último acceso: Julio 2010)
<http://www.cellular.co.za/cdma.htm>
- [15] HSDPA: Llega el ADSL sin cables, 2006. (Último acceso: Mayo 2009)
[http://www.idg.es/pcworldtech/mostrararticulo.asp?id=177533&seccion=movilidad, 2006.](http://www.idg.es/pcworldtech/mostrararticulo.asp?id=177533&seccion=movilidad,2006)
- [16] ITU: Committed to connecting the world. (Último acceso: Julio 2010)
<http://www.itu.int>
- [17] LTE vs WiMAX: A Little 4G Sibling Rivalry. (Último acceso: Julio 2010)
<http://gigaom.com/2008/03/05/a-little-4g-sibling-rivalry/>
- [18] LTE, Long Term Evolution, Technology. (Último acceso: Julio 2010)
<http://www.3gamericas.org/index.cfm?fuseaction=page§ionid=249>
- [19] WiMAX Technology. (Último acceso: Mayo 2009)
<http://www.intel.com/technology/wimax/>
- [20] Procesadores para teléfonos móviles. (Último acceso: Julio 2010)
<http://www.consumer.es/web/es/tecnologia/hardware/2010/04/21/192285.php>
- [21] Texas Instruments. (Último acceso: Julio 2010)
<http://www.ti.com/>
- [22] Apple. (Último acceso: Julio 2010)
<http://www.apple.com/>

-
- [23] Welcome to NVIDIA - World Leader in Visual Computing Technologies. (Último acceso: Julio 2010)
<http://www.nvidia.com>
- [24] SAMSUNG Semiconductor - Products - Mobile SoC - Application Processor. (Último acceso: Julio 2010)
http://www.samsung.com/global/business/semiconductor/products/mobilesoc/Products_ApplicationProcessor.html
- [25] Qualcomm. (Último acceso: Julio 2010)
<http://www.qualcomm.com/>
- [26] El procesador del iPad, Apple A4. (Último acceso: Julio 2010)
<http://www.applesfera.com/ipad/el-procesador-el-ipad-apple-a4>
- [27] Qualcomm Products and Services - The Snapdragon™ Platform. (Último acceso: Julio 2010)
http://www.qualcomm.com/products_services/chipsets/snapdragon.html
- [28] Next Generation NVIDIA Tegra. (Último acceso: Julio 2010)
http://www.nvidia.com/object/tegra_250.html
- [29] Marvell's ARMADA: Custom Designed ARM SoCs Break 1GHz – AnandTech. (Último acceso: Julio 2010)
<http://www.anandtech.com/show/2860>
- [30] Intel. (Último acceso: Julio 2010)
<http://www.intel.com>
- [31] Procesadores Intel® Atom™: Dispositivos inteligentes en todas partes. (Último acceso: Julio 2010)
<http://www.intel.com/espanol/technology/atom/index.htm>
- [32] ARM vows Xbox-level 3D on phones late 2009, 2009. (Último acceso: Junio 2009)
<http://www.electronista.com/articles/09/03/23/arm.mali.200.and.400/>
- [33] Las capacidades gráficas del iPhone 3GS, son mejores que las del PSP, 2009. (Último acceso: Junio 2009)
<http://www.appleismo.com/las-capacidades-graficas-del-iphone-3g-s-son-mejores-que-las-del-psp/>
- [34] Tendencias en las pantallas de los dispositivos móviles, 2008. (Último acceso: Mayo 2009)
<http://www.xatakamovil.com/multimedia/tendencias-en-las-pantallas-de-los-dispositivos-moviles>
- [35] Latest Touch Screen Cell Phones, 2009. (Último acceso: Junio 2009)
<http://www.touchscreencellphones.net/>
- [36] New Touchscreen Cell Phones: iPhone Killers?, 2007. (Último acceso: Mayo 2009)
<http://blogs.pcworld.com/staffblog/archives/003700.html>
- [37] Global Positioning System. (Último acceso: Julio 2010)
<http://www.gps.gov/systems/gps/>
- [38] Conoce más sobre la batería del celular, 2007. (Último acceso: Mayo 2009)
<http://celularesamil.blogspot.com/2007/12/la-bateria-de-tu-celular.html>
- [39] Comparativa entre sistemas operativos móviles, 2009. (Último acceso: Mayo 2009)
<http://diarioandroid.com/2009/05/06/comparativa-entre-sistemas-operativos-moviles/>
- [40] Gartner Says Worldwide Mobile Phone Sales Grew 17 Per Cent in First Quarter 2010. (Último acceso: Julio 2010)
<http://www.gartner.com/it/page.jsp?id=1372013>
- [41] Symbian OS. (Último acceso: Julio 2010)
<http://www.symbianos.org>
- [42] RIM BlackBerry. (Último acceso: Julio 2010)
<http://www.rim.com/>
- [43] Apple iPhone OS. (Último acceso: Julio 2010)
<http://www.apple.com/iphone/>
- [44] Android. (Último acceso: Julio 2010)
<http://www.android.com/>
-

-
- [45] Windows Mobile. (Último acceso: Julio 2010)
<http://www.microsoft.com/windowsmobile/es-es/default.aspx>
- [46] Linux. (Último acceso: Julio 2010)
<http://www.linuxfordevices.com/c/a/Linux-For-Devices-Articles/Linux-Mobile-Phones/>
- [47] Redes sociales móviles, 2008. (Último acceso: Mayo 2009)
<http://www.revista-ays.com/DocsNum18/MundoMovil/Dacil.pdf>
- [48] Las Redes sociales online llegan poco a poco al móvil, 2007. (Último acceso: Julio 2010)
<http://sociedaddelainformacion.telefonica.es/jsp/articulos/detalle.jsp?elem=3780>
- [49] Ipoki y las redes sociales móviles sobre GPS, 2008. (Último acceso: Mayo 2009)
http://www.soitu.es/soitu/2008/07/30/pieldigital/1217427842_695169.html
- [50] Redes sociales + localización, 2008. (Último acceso: Mayo 2009)
<http://www.celularis.com/opinion/redes-sociales-localizacion.php>
- [51] REDES- Revista hispana para el análisis de redes sociales. Vol.3,#2, sept-nov. 2002. (Último acceso: Mayo 2009)
http://revista-redes.rediris.es/html-vol3/vol3_2.htm
- [52] Qué son y cómo funcionan las Redes Sociales. (Último acceso: Mayo 2009)
http://www.microsoft.com/spain/empresas/rppp/redes_sociales_intro.aspx
- [53] Danah M. Boyd & Nicole B. Ellison. Social Network Sites: Definition, History, and Scholarship, 2007. (Último acceso: Mayo 2009)
<http://jcmc.indiana.edu/vol13/issue1/boyd.ellison.html>
- [54] Friendster. (Último acceso: Julio 2010)
<http://www.friendster.com/>
- [55] Tribe. (Último acceso: Julio 2010)
<http://www.tribe.net/>
- [56] Myspace. (Último acceso: Julio 2010)
<http://www.myspace.com/>
- [57] Facebook. (Último acceso: Julio 2010)
<http://www.facebook.com>
- [58] Twitter. (Último acceso: Julio 2010)
<http://www.twitter.com>
- [59] YouTube. (Último acceso: Julio 2010)
<http://www.youtube.com>
- [60] Consideraciones para el Desarrollo de Aplicaciones Móviles, 2009. (Último acceso: Mayo 2009)
<http://www.slideshare.net/soreygarcia/consideraciones-basicas-para-el-desarrollo-de-aplicaciones-mviles>
- [61] Desarrollo de Aplicaciones Móviles. Elementos a Considerar, 2007. (Último acceso: Mayo 2009)
<http://www.sg.com.mx/content/view/572>
- [62] Java ME Technology. (Último acceso: Junio 2009)
<http://java.sun.com/javame/technology/index.jsp>
- [63] Eclipse Galileo. (Último acceso: Junio 2009)
<http://www.eclipse.org/galileo>
- [64] Eclipse Pulsar (Último acceso: Junio 2009)
<http://www.eclipse.org/pulsar>
- [65] Eclipse Pulsar Platform: Uniting Mobile Manufacturers With a Single Development Platform, 2009. (Último acceso: Junio 2009)
<http://ostatic.com/blog/eclipse-pulsar-platform-uniting-mobile-manufacturers-with-a-single-development-platform>
- [66] Eclipse Pulsar Simplifies Mobile Development, 2009. (Último acceso: Junio 2009)
<http://www.internetnews.com/mobility/article.php/3809676>
- [67] Eclipse Pulsar seeks mobile app dev unity, 2009. (Último acceso: Junio 2009)
<http://www.infoworld.com/d/developer-world/eclipse-pulsar-seeks-mobile-app-dev-unity-891>
-

-
- [68] .NET Compact Framework, 2007. (Último acceso: Junio 2009)
<http://msdn.microsoft.com/es-es/library/f44bbwa1.aspx>
- [69] Arquitectura de .NET Compact Framework, 2007. (Último acceso: Junio 2009)
<http://msdn.microsoft.com/es-es/library/9s7k7ce5.aspx>
- [70] Visual Studio y .NET Compact Framework, 2007. (Último acceso: Junio 2009)
<http://msdn.microsoft.com/es-es/library/ms172551.aspx>
- [71] Dominios de aplicación de .NET Compact Framework, 2007. (Último acceso: Junio 2009)
<http://msdn.microsoft.com/es-es/library/18adbxba.aspx>
- [72] ArcGIS Online. (Último acceso: Junio 2009)
<http://www.esri.com/software/arcgis/arcgisonline/index.html>
- [73] Google Maps. (Último acceso: Julio 2010)
<http://maps.google.com>
- [74] Yahoo! Maps, Driving Directions, and Traffic. (Último acceso: Julio 2010)
<http://maps.yahoo.com/>
- [75] OpenStreetMap. (Último acceso: Junio 2009)
<http://www.openstreetmap.org/>
- [76] ArcGIS Mobile. (Último acceso: Junio 2009)
http://www.esri.com/software/arcgis/arcgismobile/key_features.html
- [77] Developing Mobile Applications. (Último acceso: Junio 2009)
http://resources.esri.com/help/9.3/ArcGISmobile/adf/mobile_start.htm
- [78] API de Google Maps, 2009. (Último acceso: Junio 2009)
<http://code.google.com/intl/es/apis/maps/faq.html#mapsformobile>
- [79] API de Google Static Maps, 2009. (Último acceso: Junio 2009)
<http://code.google.com/intl/es/apis/maps/documentation/staticmaps/>
- [80] Yahoo! Maps Web Services, 2009. (Último acceso: Junio 2009)
<http://developer.yahoo.com/maps/>
- [81] Yahoo! Maps Web Services - Map Image API, 2009. (Último acceso: Junio 2009)
<http://developer.yahoo.com/maps/rest/V1/>
- [82] CLOUDMADE -Makes Maps Differently, 2009. (Último acceso: Junio 2009)
<http://cloudmade.com/>
- [83] MGMaps Lib SDK. (Último acceso: Junio 2009)
<http://www.nutiteq.com/mgmapslib.html>
- [84] Introduction to Fire Eagle, 2009. (Último acceso: Junio 2009)
<http://fireeagle.yahoo.net/developer/documentation>
- [85] API Documentation - Google Latitude API - Google Code. (Último acceso: Julio 2010)
<http://www.google.com/intl/en-us/mobile/latitude/>
- [86] Apps for Everything, 2009. (Último acceso: Junio 2009)
<http://www.apple.com/iphone/apps-for-iphone/>
- [87] Windows Marketplace for Mobile: Shop Apps (Último acceso: Julio 2010)
<http://marketplace.windowsphone.com/Default.aspx>
- [88] Windows Phone for Developers, 2009. (Último acceso: Junio 2009)
<http://developer.windowsmobile.com/Marketplace.aspx>
- [89] Preguntas más frecuentes de Windows Marketplace, 2009. (Último acceso: Junio 2009)
<http://developer.windowsmobile.com/Help.aspx>
- [90] Android.com – Market. (Último acceso: Julio 2010)
<http://www.android.com/market>

-
- [91] MotionX-GPS 7.1 for the iPhone, 2009. (Último acceso: Junio 2009)
<http://news.motionx.com/category/motionx-gps/>
- [92] SportyPal. (Último acceso: Julio 2010)
<http://www.sportypal.com/>
- [93] Google My Tracks. (Último acceso: Julio 2010)
<http://mytracks.appspot.com/>
- [94] Nokia Sports Tracker Beta, 2008. (Último acceso: Junio 2009)
<http://sportstracker.nokia.com>
- [95] Sony Ericsson Tracker. (Último acceso: Julio 2010)
https://www.sonyericsson.com/cws/download/1/487/431/1213169968/W760_GSG_Tracker_EN_1211_2835_1.pdf
- [96] Endomondo. (Último acceso: Julio 2010)
<http://www.endomondo.com>
- [97] RunKeeper. (Último acceso: Julio 2010)
<http://runkeeper.com/>
- [98] iMapMyRun. (Último acceso: Julio 2010)
<http://www.imapmy.com/>
- [99] JogTracker. (Último acceso: Julio 2010)
<http://www.jogtracker.com/>
- [100] Windows Server 2008 R2 Operating System. (Último acceso: Julio 2010)
<http://www.microsoft.com/windowsserver2008/en/us/default.aspx>
- [101] Microsoft SQL Server 2005. (Último acceso: Julio 2010)
<http://www.microsoft.com/sqlserver/2005/en/us/default.aspx>
- [102] ArcGIS Server | GIS Web Server Software | Web Mapping Server. (Último acceso: Julio 2010)
<http://www.esri.com/software/arcgis/arcgisserver/index.html>
- [103] ArcSDE Technology. (Último acceso: Julio 2010)
<http://www.esri.com/software/arcgis/arcscde/index.html>
- [104] Esri Products | A Complete GIS and Mapping Software System. (Último acceso: Julio 2010)
http://www.esri.com/products/index.html#desktop_gis_panel
- [105] Team Foundation Server. (Último acceso: Julio 2010)
<http://msdn.microsoft.com/en-us/vstudio/ff637362.aspx>
- [106] Microsoft Windows Sharepoint Services 3.0. (Último acceso: Julio 2010)
<http://technet.microsoft.com/en-us/windowsserver/sharepoint/default.aspx#tab=1>
- [107] Paint.NET - Free Software for Digital Photo Editing. (Último acceso: Julio 2010)
<http://www.getpaint.net/>
- [108] GIMP - The GNU Image Manipulation Program. (Último acceso: Julio 2010)
<http://www.gimp.org>
- [109] DynDNS.com - Free Domain Name, Managed DNS, Email Services. (Último acceso: Julio 2010)
<http://www.dyndns.com/>
- [110] Curso de Interacción Persona-Computadora, FING, (Último acceso: Julio 2010)
<http://www.fing.edu.uy/inco/cursos/inpercom>
- [111] Game Programming Patterns / Sequencing Patterns / Double Buffer. (Último acceso: Julio 2010)
<http://gameprogrammingpatterns.com/double-buffer.html>
- [112] Larman, Craig. Applying UML and patterns: An introduction to object-oriented analysis and design. 1997. Editorial Prentice Hall. ISBN 0-13-748880-7.
- [113] What Is Silverlight? | Microsoft Silverlight, (Último acceso: Julio 2010)
<http://www.microsoft.com/silverlight/what-is-silverlight/>
-

- [114] Microsoft announces Windows Mobile 6.5. (Último acceso: Julio 2010)
<http://www.engadget.com/2009/02/16/microsoft-announces-windows-mobile-6-5/>
- [115] Windows Mobile 6.5 review. (Último acceso: Julio 2010)
<http://www.engadget.com/2009/10/06/windows-mobile-6-5-review/>
- [116] Silverlight for Windows Phone : The Official Microsoft Silverlight Site, (Último acceso: Julio 2010)
<http://www.silverlight.net/getstarted/devices/windows-phone/>
- [117] Windows Phone 7 Release Dates, November For US, October For Europe. (Último acceso: Julio 2010)
<http://www.geeky-gadgets.com/windows-phone-7-release-dates-november-for-us-october-for-europe-02-08-2010/>
- [118] BING MAPS PLATFORM - Bing Maps – Create and Connect with Maps APIs from Microsoft. (Último acceso: Julio 2010)
<http://www.microsoft.com/maps/>