

Proyecto ELO-322

Redes de Computadores

Seguimiento entre Smartphones
mediante GPS

Profesor: Agustín González
Integrantes: Vanessa Pulgar
Samuel Quevedo
Fecha Entrega: 6 de Septiembre del 2013

Resumen

En los últimos años ha aumentado considerablemente la cantidad de dispositivos smartphone entre la población, lo que ha hecho posible que gran parte de la gente tenga acceso a aparatos con capacidades de posicionamiento GPS. Es recurrente la necesidad de seguimiento entre dos o más dispositivos, ya sea para transporte, monitoreo, u otros. En este trabajo se evalúa el desarrollo de una aplicación para dispositivos smartphone que satisfaga esta necesidad. Para no partir desde cero, se hizo un sondeo previo de aplicaciones que se acerquen a lo que se precisa para cumplir con lo deseado, encontrando entre otras, una aplicación de código libre, “Open GPS Tracker” que ya ha solucionado la parte más técnica de mostrar información por pantalla, así como también la de almacenamiento de “waypoints”, rutas y segmentos en GPX, un formato de intercambio GPS. Así, se ideó la posibilidad de agregar, como extensión a esta aplicación la idea de seguimiento entre múltiples dispositivos. Puesto que el seguimiento de dispositivos es un tema sensible, que toca la privacidad de las personas, se ha ideado un sistema simple de conexiones desechables con un setup minimalista, que ofrece privacidad y facilidad de uso al mismo tiempo.

Introducción

Nuestro proyecto se basa en la evaluación del desarrollo de una aplicación de seguimiento GPS, para lo que se hará una investigación de la implementación de la conexión y envío de datos entre smartphones para hacer posible la transmisión de un seguimiento obtenido en el GPS, a través de internet, creando una extensión a una aplicación ya existente, tomando en cuenta los conceptos vistos en clase sobre el envío de paquetes en la red, estructuras de conexión y servicios de transporte. Nos centraremos en la capa de transporte ya que esta es la que se preocupa de la llegada del paquete de proceso a proceso, buscando la mejor arquitectura para la aplicación.

Elección de Plataforma de Desarrollo

Para la solución al problema planteado, consideramos las tres principales Sistemas Operativos de Smartphones en la industria: iOS, Android y Windows Phone. Para ayudarnos a decidir sobre una plataforma, elaboramos una pequeña comparativa con algunas características de cada Sistema.

Sistema	Lenguaje	Documentación	Masividad	Tutoriales
Android	Java	Completa / Fácil de Navegar	Alta	Gratis / Gran Comunidad
iOS	Objective-C	Completa	Media	USD99 por año
WP7	Silverlight	Completa / Fácil de Navegar	Baja	Gratis

Referencias [1][2][3]

Con la ayuda de la comparativa, nos decidimos por el OS Google Android, principalmente por su masividad, y comunidad de desarrollo. Por lo tanto buscamos una aplicación para esta plataforma para entender cómo funciona una aplicación que se basa en el posicionamiento GPS, para enfocarnos en las Redes de Computadores asociadas a la solución del problema. Ahora mostraremos nuestra investigación acerca de Android y el programa utilizado de base, enfocándonos, como se puede notar, en la Capa de Aplicación y Transporte.

Estructura de una Aplicación de Google Android

Android es un Sistema Operativo mantenido por Google, bastante masivo, que tiene una comunidad de desarrollo muy extensa, cuyas aplicaciones tienen una estructura constituida por cuatro tipos de Componentes: **Actividades** que crean las interfaces de usuario, **Servicios** que corren en segundo plano y realizan tareas de distinta naturaleza, **Proveedores de Contenido** que corresponden a Bases de Datos a las que se puede acceder desde otras componentes, y **Escuchas de Eventos** que reciben un evento y

desarrollan una acción acorde a este.

Para enfocarnos en el análisis de la conexión entre dispositivos, la aplicación que desarrollaremos, será una versión primitiva de una solución ideal, mostrando la esencia del programa.

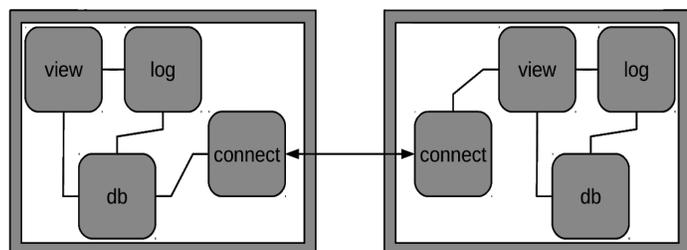
GPS

El Sistema de Posicionamiento Global (GPS por sus siglas en Inglés) consta de una constelación de 24 o más satélites, que permiten la triangulación de un dispositivo, gracias a que están constantemente transmitiendo paquetes de información con la hora, que permite calcular la distancia a cuatro o más satélites, calculando entonces su posición en la tierra. Esta tecnología se ha masificado enormemente en los dispositivos de comunicación, siendo ya cotidiano su uso. (Referencia [4])

Open-GPS Tracker ¿Cómo funciona una aplicación GPS?

En este caso, para entender cómo funciona una aplicación GPS nos centramos en la aplicación para android “Open-GPS Tracker”, por su código abierto, además de su buen funcionamiento. Esta aplicación consta de 3 componentes: **logger** que es la que captura los datos que entrega el chip GPS y los entrega a la base de datos , **db** es la base de datos, la que guarda la información que le entrega logger , y **viewer** que es el que toma la información desde la base de datos y la muestra en pantalla. Esta aplicación traza una ruta, y mide la velocidad a la que va, proceso que hace viewer usando los paquetes que ofrecen una interfaz para trabajar con los mapas de Google, que vienen integrados en Android, por lo que una posible solución a nuestro problema podría ser una extensión a una aplicación de esta naturaleza, implementando una cuarta componente del programa

que sea la encargada de tomar los datos desde la base de datos y enviarlos al destinatario. La forma en que esto puede ser hecho será discutida a continuación.

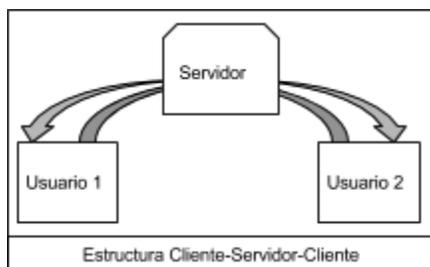
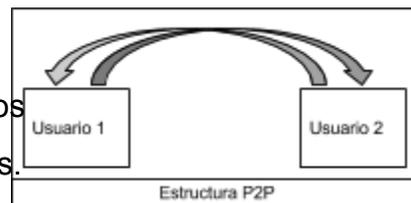


Tipos de Conexión

Dada la frecuencia con que se adquieren los datos de posición mediante GPS, consideramos que en la comunicación que necesitamos prima la rapidez por sobre la confiabilidad. Es por esto, que consideramos que el protocolo que conviene implementar es el de UDP.

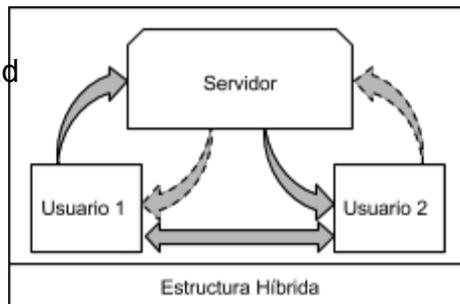
Sobre la arquitectura a utilizar, buscamos los pros y contras de Peer to Peer (P2P) Y Cliente-Servidor.

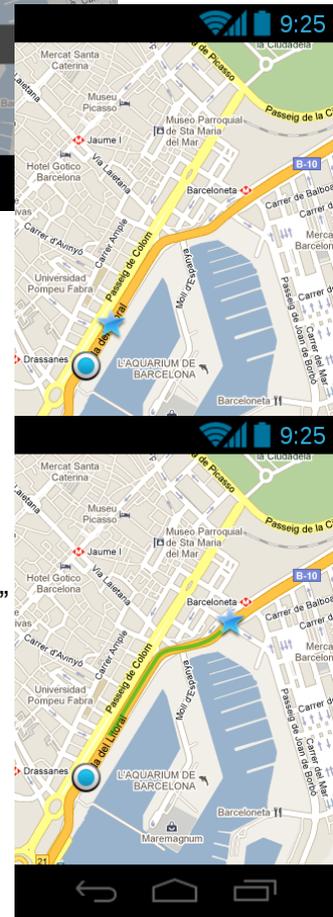
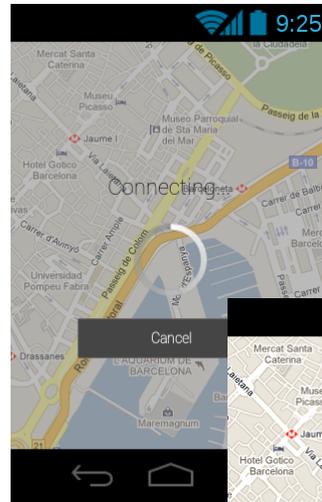
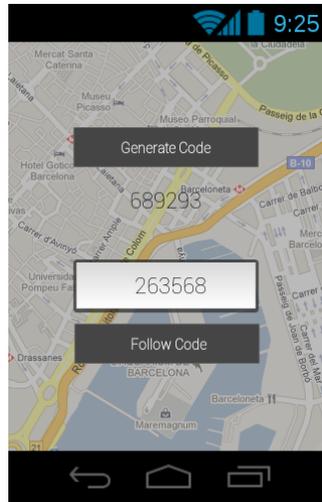
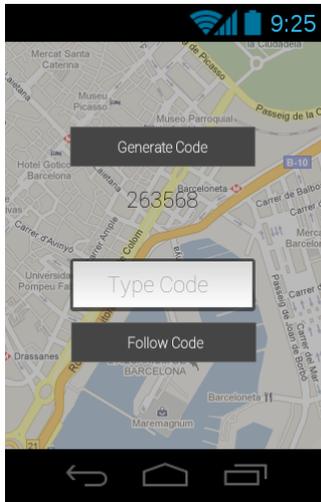
Como vimos en clases, las ventajas de una arquitectura P2P son la inmediatez de la comunicación, al eliminarse los intermediarios que puedan introducir retardos significativos.



Por el otro lado, la arquitectura Cliente-Servidor-Cliente nos permite simplificar la transmisión de datos, pues el Servidor tendría una dirección IP fija, que sería más fácil de configurar.

También está la posibilidad de implementar un híbrido entre ambas arquitecturas, aprovechando la simplicidad de C-S-C y la rapidez de P2P. Por esto se consideró como más adecuada la opción de arquitectura híbrida, pues simplifica enormemente la conexión inicial.





Apoyándonos en las maquetas presentadas [5] se elaboró un protocolo de establecimiento de conexión en la capa de aplicación:

- Cada terminal genera un código.
- Este código se ingresa en el terminal remoto
- Al generar un código, se envía al servidor, que lo almacena junto con la dirección correspondiente
- Servidor espera parejas de códigos y responde a ambos.
- Se verifica la validez de los datos y se realiza un “handshake” entre los terminales
- Se muestra la ubicación de ambos terminales, y se dibuja el recorrido del terminal remoto.
- Se muestra la ubicación de ambos terminales, y se dibuja el recorrido del terminal remoto.

En anexos se presenta un diagrama temporal.

Parte del código para la conexión a utilizar será similar a los ejemplos vistos en la tarea 4, basada en los sockets de conexión, como ejemplo al establecer la conexión entre los dispositivos después de la autenticación con el servidor:

```
String messageStr=/* mensaje de handshake con información para la conexión */;
int server_port = 5896;
DatagramSocket s = new DatagramSocket();
InetAddress peer = "/* Dirección ip entregada por el servidor */";
int msg_length=messageStr.length();
byte[] message = messageStr.getBytes();
DatagramPacket p = new DatagramPacket(message, msg_length,peer,server_port);
s.send(p);
```

Conclusión

El seguimiento entre dispositivos celulares es un tema que puede ser abordado desde distintos puntos, dependiendo de la aplicación y los requerimientos de cada circunstancia. En este trabajo se abordó el tema enfocado a usuarios casuales. La implementación se propone en función de la teoría de redes de computadores vista en el ramo. Así se dio una orientación en el sentido de las dos capas más superficiales del modelo OSI, capa de aplicación y de transporte. Distintos aspectos de la implementación de nuestra solución propuesta podrían seguir siendo analizadas en pos de distintos escenarios que requieran de nuevas capacidades.

Referencias

- [1] <http://www.microsoft.com/silverlight/resources/>
- [2] <https://developer.apple.com/library/ios/navigation/>
- [3] <http://developer.android.com/>
- [4] <http://www.gps.gov/multimedia/poster/>
- [5] <https://www.fluidui.com/>

Anexo

1.- Diagrama temporal de autenticación capa de aplicación

