Design Document

Linux – Android GPS

26.03.2019

Jeffrey Choy

Kiaan Castillo

Jenny Ly

Ben Zhang

Contents

Preface

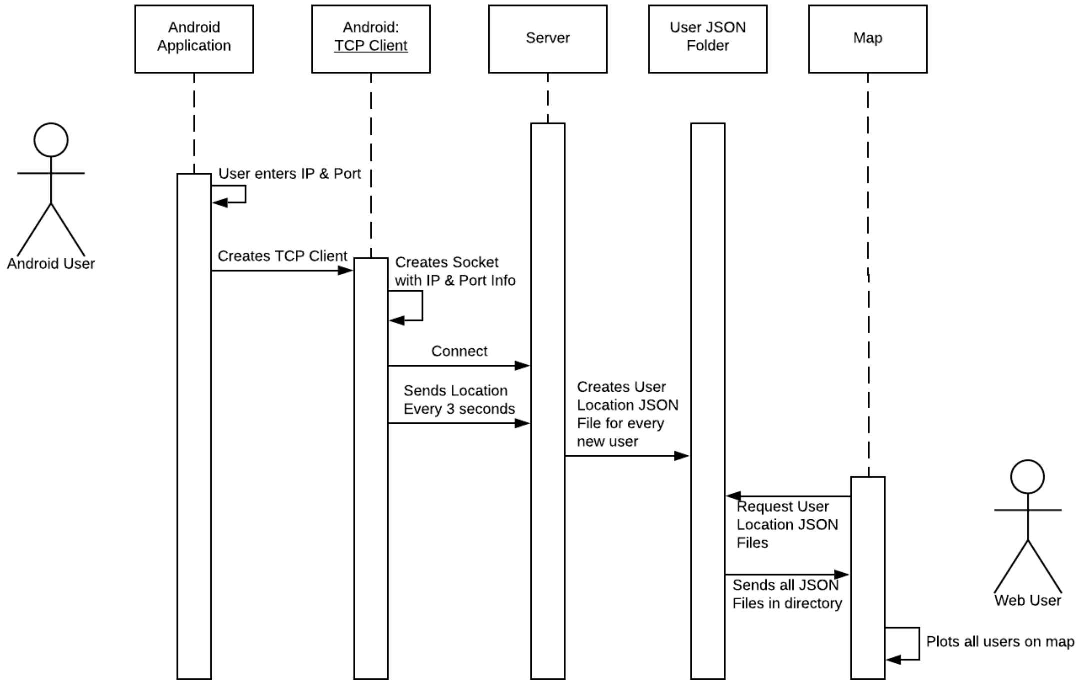
The Android GPS project is a Client and Server application suite, with an Android mobile application as the client, and a node script as the server.

The Android application shall connect to the server via TCP ports and send its current location through either Wi-Fi or mobile data.

The server receives the location as JSON objects and stores each users' locations separately.

Finally, a website with authentication will allow administrators to view the locations of each user.

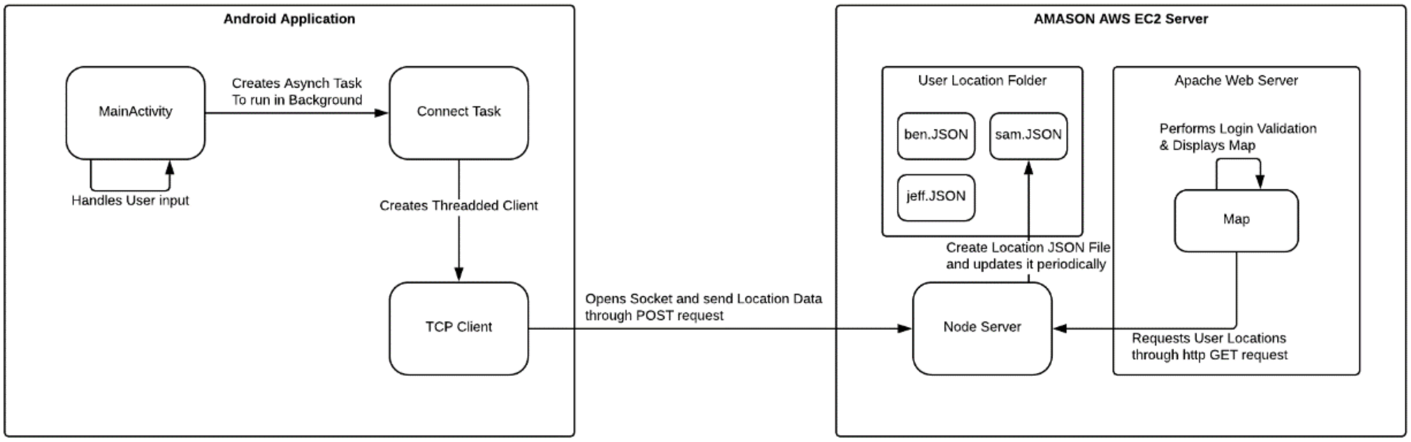
Sequence Diagram


Android Application
Android: TCP Client
Server
User JSON Folder
Map
Android User
User enters IP & Port
Creates TCP Client
Creates Socket with IP & Port Info
Connect
Sends Location Every 3 seconds
Creates User Location JSON File for every new user
Request User Location JSON Files
Sends all JSON Files in directory
Web User
Plots all users on map

The above diagram depicts the typical flow of using the application suite.

1. Start the Server
2. Start the Android Application
3. User enters the server IP & Port
4. The application creates a threaded TCP Client
5. The TCP Client handles connection to server's socket
6. The application resends their latest location data periodically
7. Server saves received data into a folder
8. Web user/administrators open the map
9. The map requests user locations from the server and displays them

Deployment Diagram



There are mainly two deployment modules:

1. Client

The Client module is an Android application that is either ran on an emulator on Android Studio, or natively as an APK on an android phone.

2. Server

All components of the server are stored in an AWS EC2 Ubuntu instance.

Apache is used to host the map html to view the locations of the users.

A Node server is running in the same directory as the map and the user locations folder.

The Node server acts as a middleman for both the client-side android application as well as the server side map webpage.

}

Pseudocode

Android

```
MainActivity {
    Create Async connection task;
    Create TCP Client; (External class)

    OnCreate() {
        Create google location service client;
        Set listener for connect button;
    }

    OnClickConnectButton() {
        Get device location via google location service client;
        Set location as JSON format;
        Edit TCP Client’s IP and Port fields to be user defined values;
        Execute Async connect task;
        Puts the connect task on a repeated 3 seconds cycle;
    }
}

TCP Client {
    Run () {
        Create new socket with user defined server address and port;
        Creates a new thread to send the location JSON to the server;
    }
}
```

Server

```
Server.js {
        Import net module;
        Create net server to handle TCP requests;
        Set server to listen on local host with port 3000;
        Set server to receive connections and messages;
        Converts messages to JSON file and store in users folder;

        Import express module;
        Create express application;
        Use JSON parser and allows foreign IP to connect to the application;
        Set application to handles GET requests by reading all JSON files in the users directory and
        sending them to the client as a JSON array;
}
```

Map

```
Map.html {
<script>
        Initialize Google Maps;
        Set a timed request to the Node server for a list of user location JSON objects;
        Displays the locations using map markers;
</script>
}
```