Specific Design Document

Team Name: The Leftovers

Team Members: Michael Ginn, Tim Horner, Tinh Lam, Viet Le, Joseph Adirzone, Christopher Curreri, Abigail Palen.

Project name: Street Marco Polo

SUMMARY: We are building a mobile Android application that uses GPS function to track another user in a game setting. The application is a game that can be played between two people. One player will be a seeker and the other will be a hider. The seeker must use the GPS tracking to their advantage to find the person that is hiding. The GPS will not always be available for the seeker - the hider's location will toggle between visible and not visible to give the hider a fair advantage.

1. SCREEN/VISUAL SECTION:

Image of Google Map Tracking View

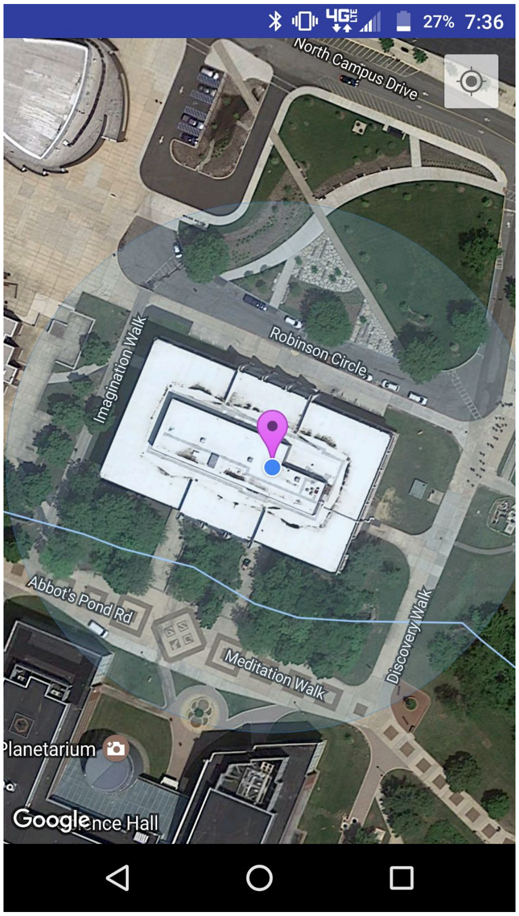
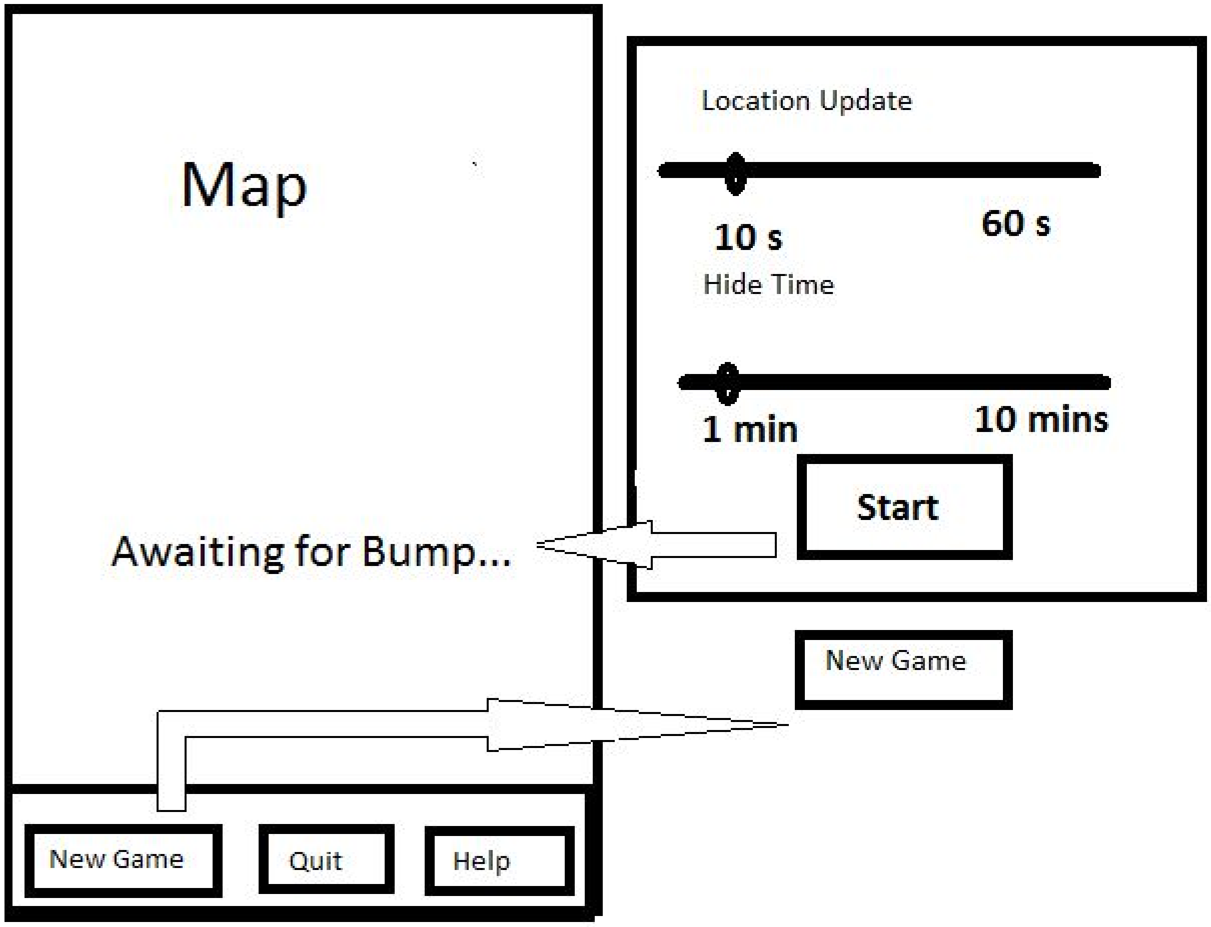


Image of GUI layout draft:



2. GOOGLE MAP API : Android Studio provides a downloadable Google Maps API that has built-in functions useful for manipulating the GPS on the map.

- *CheckLocationPermission* - checks the permissions for using the GPS. Must be run first before getting a user's location.
- *OnLocationChanged* - listens for movement and updates every time the user moves to keep track of where the user is going. Gets latitude and longitude of the user when the location changes and puts a marker on the map where the user's current location is.
- *setMyLocationEnabled* - allows for a user's location to be seen. Setting this to true makes the location visible and setting it to false makes the location invisible.
- *setMapType* - sets up the Google map with all locations in the world labeled and includes images of locations.

3. ReSTFUL SERVICE:
Our ReSTFUL Service will be fairly simple and only be tasked with storing and retrieving a player's location via their unique ID (phone number).

Endpoint: POST /players/: phone number
Purpose: Creates a new player
Inputs: phone number: path parameter
Output: None
Implementation: Create a new entry in the players table using the phone number as the unique identifier and null as the location field

Endpoint: PUT /players/'phone number'/: location
Purpose: Update location of player
Inputs: String containing coordinates for location: path parameter
Output: None
Implementation: Find the entry for player using the phone number and update the location field

Endpoint: GET /players/'phone number'/
Purpose: Get the specified player's entry from the table
Inputs: none
Output: JSON format for player
player-id: String
location: String
Implementation: Retrieve the entry in the table identified by the given phone number and return its contents in JSON format.

Endpoint: DELETE /players/'phone number'/
Purpose: Remove specified entry from players table. Ensures that upon starting a new game, the POST command to create player does not create multiple entries for the same phone number.
Inputs: none
Output: none
Implementation: Find entry for player using phone number and remove its information from the table.

4. DATABASE SECTION: We will be using 1 table to store each player's unique ID and a string containing their most recent location. There will be 2 columns, the first will be the unique ID, a String containing the phone number, of the player and the second column will be a String containing the coordinates of the most recent location of the player.

5. NFC: To establish the connection between the two devices, we will pass the phone number of the inviting player to the other using NFC. The person initiating the NFC Bump will be the Hider in our game. In essence, the Hider will start the game and broadcast the NFC message and the Seeker will receive the message and store their phone number for use with the ReSTFUL Service. Now that the phone number is stored, the Seeker is able to use this to retrieve the location of the Hider using the ReSTFUL Service. Using NFC will also allow us to ask for permission to open the app on the other user’s phone, and if they do not have the app installed it can directly link to the GooglePlay Store for download.

6. IMPLEMENTATION PLAN SECTION: In the first two weeks of the implementation plan we plan to enable the Google Maps API with GPS location tracking. When the user opens the app they will be able to track their location as they move and see the marker identifying their location. We will also create the GUI. Buttons that rely on unimplemented features will be present, but not functional at the end of the first two weeks of implementation. We will have NFC communication functioning. It will be used to pass the NDEF message containing the phone number. We will also create the ReSTFUL API and endpoints needed for our app.

Individual Assignments:

ReSTFUL API: Tim Horner, Michael Ginn
Database: Tim Horner, Michael Ginn, Jay Adirizone
NFC: Joseph Adirzone, Tim Horner.
GPS: Michael Ginn, Tinh Lam, Viet Le.
GUI: Michael Ginn, Chris Curreri.
Debug/Testing: Everyone.
Github: Everyone.

7. TECHNOLOGY STACK:

1. Android Studio
2. Eclipse IDE
3. Android.nfc Package
4. Google Maps API for GPS
5. ReSTFUL API for Network Communication
6. ‘Apache Tomcat’ for hosting web server
7. ‘Jersey’ for writing backend of ReSTFUL service in Java