Orkun Krand

Dr. Shubham Jain

CS541 - App Development for Smart Devices

April 12, 2018

Location

I am glad I saved my 72 hour extension for this assignment. Not because it was overly complicated, but because setting up the database according to the assignment requirements took longer than expected. I initially set my database to hold each check-in as one entry in the database but after the discussion in class, I made some adjustments as per your request.

Using the tutorials available on Google's website made this assignment easy and left me flabbergasted at how efficient Google's documentation is. The database uses a string to hold information on multiple check-ins as per the class discussion. Overriding the equals function of my custom class to check if two locations are within 30 meters of each other made life a little easier when I was checking to see which location a check-in belongs to. I had to create another custom class that basically holds the same information as the main one, but holds each check in as one instance of the class in order to easily print a list of check-ins (not each location). The data access object (dao) for my database comes with a built-in update function so I didn't have to write one myself. Just calling the update function with the updated values let Room update the correct entry with ease.

Using the tutorial online made creating the map much easier than I expected. I got it to immediately go to my current location, and show my check ins as well. Enabling the features like zoom and compass were not a big deal at all. After finishing the app, I realized I mixed the order of operations. In my app, user adds the location name first, then picks the location of the marker by tapping on the map.

Overall, I think this was a good assignment to successfully teach us how location works. Both using the raw location information, and using the API. Most apps rely on Google Maps API for their location info so

1

knowing the how to work with the raw info would put us a step ahead of everyone else.

After playing with the locations for a day, I noticed that the network location is received less often, at about once every 20 seconds and the outdoor accuracy is worse than gps location. I can set the GPS to collect much more frequently. I opted for once every 2 seconds as it was flooding the Logcat otherwise. The accuracy reported for GPS is usually around 3-4 while the accuracy from network is always above 20, usually around 25. Comparing the accuracy reported to the accuracy I calculated on a map shows that Android's calculation is almost dead on as I am usually about 25-30 meters away from the latitude and longitude provided by the network. For GPS, I notice that the accuracy shown goes from 3-5 to 9-11 once I walk indoors. Once indoors, GPS' calculated accuracy compared to the my calculation are vastly different. For instance, sitting in the northeast corner of Borjo Coffeehouse, I get the following lat-lons from GPS: 36.885038,-76.300122 with an accuracy of 11. Putting the coordinates on Google Maps puts me just outside Mojo Bones which is more than 25 meters away. On WiFi, network puts me much closer to where I actually am indoors, even though it shows an accuracy of 25, I'm more like 8-10 meters away.

When I sat just outside the Batten Arts building, real accuracy for GPS vastly improved. It showed exactly where I'm sitting as I write this. The accuracy received from the system improved as well. Network accuracy remained the same as inside, probably because the WiFi network on campus is pretty strong.

At the library, non-WiFi network thinks I'm actually inside the library while I'm sitting outside. Turning WiFi off reduced real accuracy while accuracy from the system increased minimally. GPS accuracy remained about the same. Showing 3-4 while the real accuracy is about 10 meters or so.

Unfortunately, I had no time to plot the points. But you can find some screenshots of location coordinates I received from both GPS and network in the submission folder.