Bike Share Finder

Author: German Shalnev

Email: gshalnev@myseneca.ca

List of features

- Reading the list of stations on startup and on request

- Filtering stations based on a radius around the user's location

- Customizing the filtering radius

- Sorting stations list based on distance from user's location and number of bikes available

- Updating distance information regularly when the user's location changes

- Map showing all stations around Toronto using colored markers reflecting the bike availability

- Displaying extended station information such as station number, intersection name, and time of information update

Unimplemented features

- Once in a while, the app crashes on startup. I suspect that there is a problem at the point where the app reads the string of text from the JSON file from the server. Unfortunately, I did not record what type of exception the error's throwing. I don't know how to trigger it manually in order to test how it's being handled.

I've included some logic that catches two types of exceptions from the JSON reader/parser function and reports a general error to the Station List activity's UI. Whether that fixes the problem remains to be seen. I have ran the app many times after that, but I haven't encountered that error again. The fact that the failsafe logic I've mentioned above is not triggered means that the error doesn't occur anymore.

- The user's location polling is not as power-saving as I'd hope it would be. I wanted to include some logic that selects an appropriate location provider based on user's current situation. However, I don't know how to organize it, i.e. which circumstances should trigger provider change, which provider would work best in these circumstances, or whether switching providers too much would impact power consumption.

I've tried to alleviate this problem: I've set the criteria for finding the best provider to be as accurate and power-saving as possible. This code is in the MainActivity.java, onCreate function, lines 76-81. I've also thought of giving the user extensive control over how to update current location, but I think it would be too much for a regular user to handle. There is also the fact that which providers are enabled is determined by the device's location settings, so it ultimately comes down to what the device allows the app to do.