SOFE4640U: Mobile Application Development

Assignment 2: LocationFinder App

Raj Parikh, 100655911

Github Repo: https://github.com/RajP99/Mobile-Assignment-2

Overview

In this report I will explain how the app which I designed uses databases and location services, as well as show screenshots of the various deliverables. I populated the database with 50 latitude and longitude values from the following link: https://simplemaps.com/data/world-cities

Databases

I utilized databases extensively throughout my application. SQLite database functionality was used to create tables, populate tables with data, query the tables, update rows within the tables, and delete rows within the tables. The table which I used stores the address, latitude, longitude, and a unique id for each location. My app initially begins by creating a Location table and filling it with data if the table is empty. It then uses a search bar to listen for queries. When a user enters a location in the search bar, the text is taken and then used to query the database. The SQL query I used finds addresses which contain the user's location query. Another instance of database functionality is when the user attempts to insert a new location to the database. First, I used queries to ensure that the location is not already in the database. After verifying this, I used another query to insert the user values into the database, which then also updates the values in the main activity, to show the newly added row. I also used database functionality for allowing users to update/delete rows. When a user wants to update a location, the database is first queried to make sure the new location doesn't already exist. After this, the id of the location is found through a query, and this id is used to update the address, latitude, and longitude. For the delete functionality, a query is used to delete the row in the table that contains the id of the location the page is currently on. Since I made the id a primary key column in the table, all the ids are unique.

Location Services

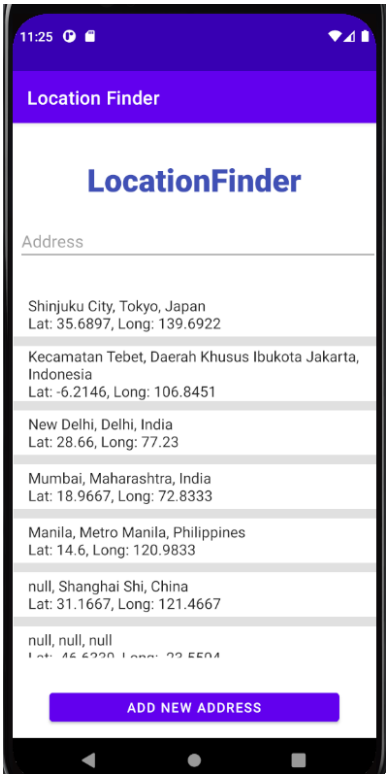
I used location services in my app with respect to forward and reverse geocoding. Forward geocoding refers to getting the latitude and longitude coordinates from an address, whereas reverse geocoding refers to getting the address from a coordinate pair. I began the code by hard-coding 50 latitude and longitude coordinate pairs into my code, these pairs refer to 50 cities. I then used reverse geocoding to determine the address of the coordinate pairs. From the address, I decided to store the city, province, and country into a database by concatenating the fields. I used forward geocoding for adding new locations and update locations. The user is prompted to enter a location/address in the search bar. I then use geocoding to determine the latitude and longitude coordinate pairs from the address. Below are two screenshots showing location service use, the first one is reverse geocoding and the second one is forward geocoding.

```
List<Address> ls = geocoder.getFromLocationName(this.address,  maxResults: 1);
```

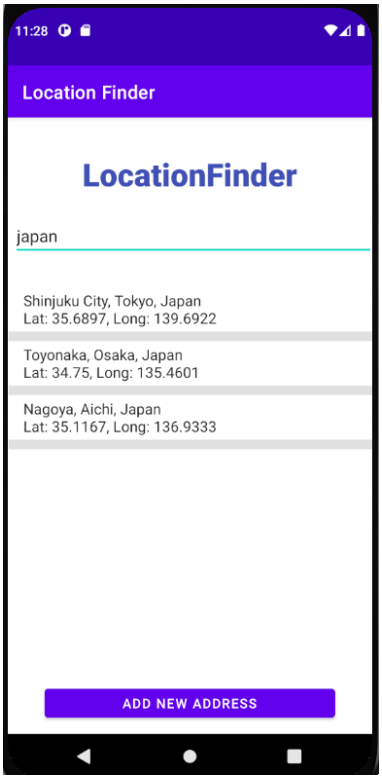
```
List<Address> ls = geocoder.getFromLocation(latitude, longitude,  maxResults: 1);
```

App Functionality

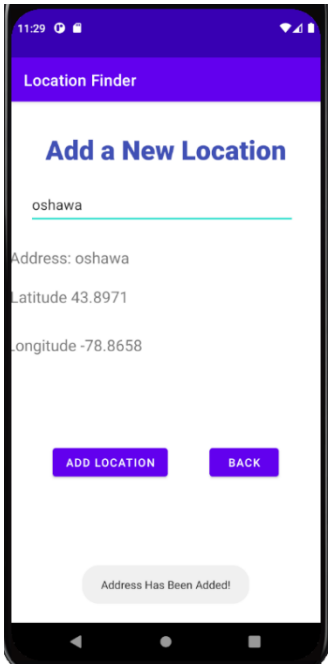
Showing address, latitude, and longitude of 50 locations (required scrolling down)

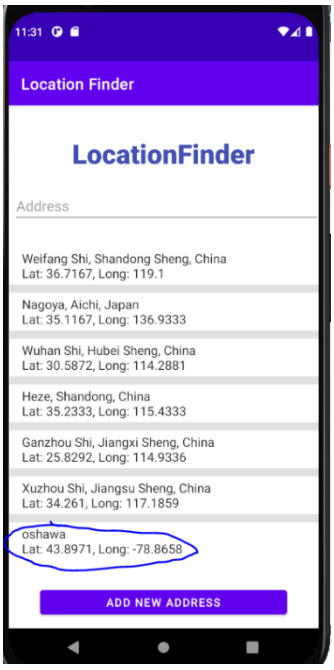


Query feature to display latitude and longitude (plus full address) for a given address



Adding locations





Update/Delete Page (not enough space to show all functionality)