Mobile App Final Project Report

Team Name : Android 8.0: Oreo

Team Members: Andrew Chan, Omar Ouf

App Name: Swiss Army App

Github Repo: https://github.com/RutgersMobileAppClass/LocationProximityTask

Work Responsibilities:

- Andrew:
  - Created Login/Registration UI
  - Created GPSService
    - Scans every 3 Seconds for changed location
    - Checks if current location is within 100 meters
      - If location changed and within range, broadcasts appropriate data
  - Created Reminder Task and corresponding code
    - Created AddReminderActivity
      - Activity serves as UI to add a String reminder and associate it with location
    - Added Reminder code in MainActivity
      - If called, shows an AlertDialog with the reminder set by the user
  - Created PowerPoint presentation
- Omar:
  - Integrated Firebase login system
  - Receiver of the proximity broadcast from the GPS Service.
  - Created Navigation Task and corresponding code
    - Activity serves as UI to add 2 locations and will get Google Maps directions from A to B when triggered/called
  - Created Location Related Activities.
    - MapFragment
      - Responsible for displaying the map. And selecting / creating new location via the capture button.
    - ListLocation
      - Responsible for Displaying the locations acquired by the user.
        - On a Normal Click . u navigate to that exact location in the map.
        - On a Long Click . Pops up ModifyContact activity.
    - ModifyContact
      - Responsible for modifying the Location name or Deleting the Location from the Database.
  - Created SQLHelper
    - Create / Update / Delete LocationBased Entries to a LocalDatabase.

Omar:

Since this was a location based application in its core. I started by implementing the MapActivity.

MapActivity basically displays a given location on the map . and then when capture button is clicked. A Marker is added to the center of the Map . That Marker Data. Like .. Lat,Lat are also added to the Local Location Database.

At this point i started working on Modifying the SQLhelper that i previously adapter for my Assignment 3 HW. What the SQLhelper does is provide a means of accessing all entries , adding entries, , deleting and updating entries. This is our main Method of making sure that data across all app is consistent.

The Purpose of List Location Activity is to provide the means to the user of modifying / deleting the locations. It also allows the user to jump to a specific location as long as it has been previously added to the Database using Capture.. A Normal Click would result in triggering the MapActivity with a specific location at its origin. And this basically allows the user to preview the previously added locations.

A Long Click would take u to modify contact activity which is supposed to display all of the data belonging to the Clicked on location. It allows the user to modify an existing location by changing its name or deleting it entirely

At This point we had to detect the Location in a background service as not to disrupt or block the UI thread.

To make our tasks simple, we did not include any specific UI for displaying them. For my task. Navigation . When it gets triggered. It opens a google map with navigation from current location (location triggered proximity on list) and destination which is saved to a shared pref file belonging to that task.

Andrew:

So I created the GPS Service which is responsible for getting location updates and comparing it against the Local Location Database. If Current Location is found to be within 100 meters, the saved location is broadcasted to the MainActivity.

The receiver is then responsible for tracking down all tasks belonging to the location in proximity if the Task has not been executed within the last 24 hours (can be changed, we just set it to 24 hours to prevent spamming Task Triggers). The task is then launched; tasks are stored in the SharedPrefs. For the Reminder Task. it displays a Reminder Dialog that was previously saved when Task was created. It also vibrates and creates a lock-screen notification if the phone is locked.

We used FireBase for Authentication and to store the Login / Registration info. We chose to use Firebase since we had some experience with it, using it in one of the homework assignments.

We chose to use Local Database to Store User's Locations as that was the easiest form of storage for us to use for our desired purpose. And SharedPref was used to store tasks belonging to the saved location.

As stated in our original project proposal, we were planning on using Bluetooth Beacons to trigger proximity instead of GPS coordinates, but we could not get the hardware (I, Andrew, had a beacon, but it broke and the model I am familiar with is too expensive to replace).

However, I am glad with the results and how our app came out. The UI is kind of sloppy, but we did not have a lot of time to fix it. I initially had a lot of trouble making the GPSService since I was not sure what functions to use. I initially wanted to use a LocationManager ProximityAlert, but it did not seem like the best solution for what we wanted to do. I ended up using a Location Listener since that was what I was most familiar with, and having the location check every 3 seconds. The listener would then check if the location has changed, and if so, compare it to any saved location. I decided to do the location checking/comparison in the service so I wouldn't have to broadcast every time the location changed, instead it only broadcasted once the locations matched.