COMP3018 COURSEWORK 2 — REPORT

Student Name: Yixin Qi
Student ID: 16522057
Word Count: 1472 words

1 Introduction

The main functionality of this application is to track users' movements and locations. It will document the total distance of users' movements while categorizing the distances into running and walking section based on the speed of the movements. All the movements will be stored into database by dates which enables users to inspect their movement (exercise) history by week, month, or specific date. Additionally, the users can locate themselves by using an google map that is implemented into the application. The detailed design and functionalities of the application will be introduced below.
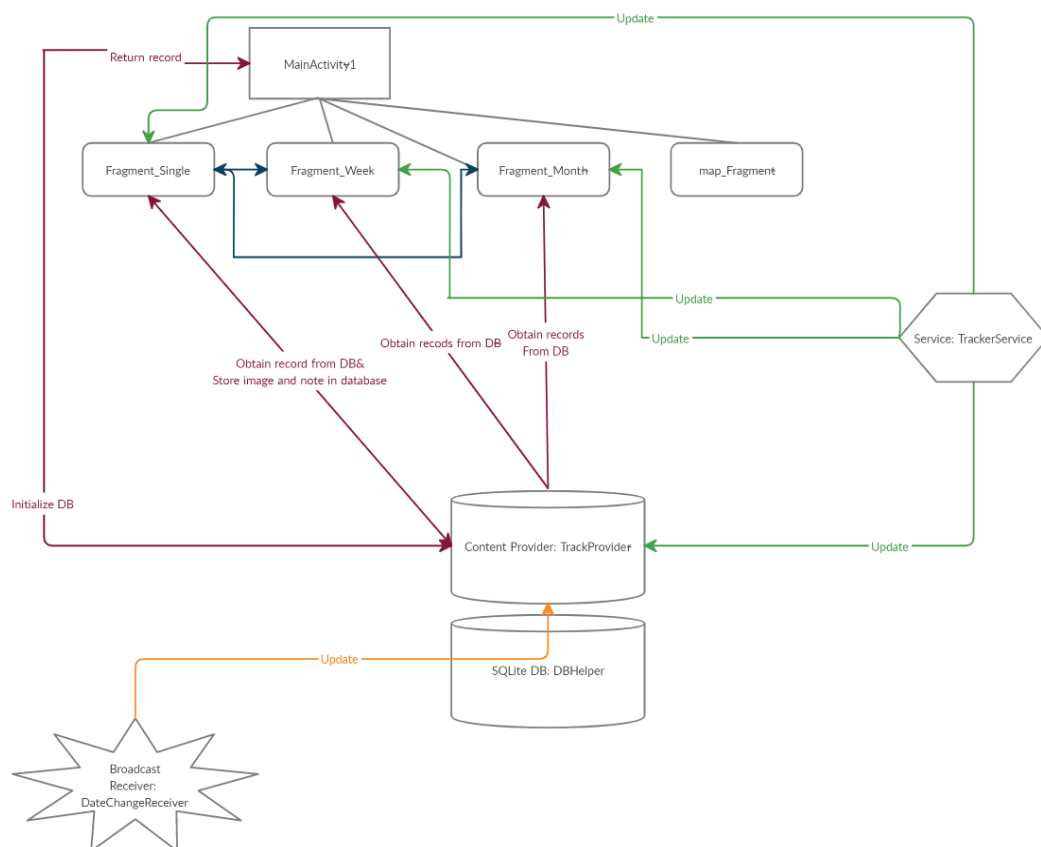
2 Design and Architecture

Fig 1

The design and architecture of the four components (Activity, Service, Broadcast Receiver, and Content Provider) has been graphically demonstrated in *Fig 1*.

As shown in the graph, the main activity (i.e. MainActivity1) has four fragments: Fragment_Single, Fragment_Week, Fragment_Month and map_fragment which are responsible for exhibiting the movement records of a single day, week, month and userss current location respectively. Fragment_Week and Fragment_Month can switch to the Fragment_Single when users demand to see a detailed record of a specific date (the blue arrow line in the graph). For Fragment_Single, Fragment_Week, Fragment_Month, they need to obtain related data (e.g. distance of movement, note, image etc.) from database when they are created (the red lines in graph among fragments and Content Provider). When the application first launched, the main activity (i.e. MaincActivity1) is responsible for creating the first row in database. And if users require to see the detailed movement record of a specific date by using search button on the main activity, the database will be queried by main activity to offer related data (the red line between MainActivity1 and Content Provider).

Since the distance of movement will change following the updates of users' location, the data shown on the Fragment_Single, Fragment_Week, Fragment_Month will need to be updated accordingly. This is guaranteed by Service (TrackerService). In Service, there is a real-time update mechanism for users' movement implemented by LocationManager and and LocationListener. Every time the distance has updated, the service will update the corresponding data stored in data base as well as the data shown on the interfaces (i.e. Fragment_Single, Fragment_Week, Fragment_Month). (The green lines shown in the graph) There is also an AlarmManager deployed in Service that is initiated when the service is created to send an intent to Broadcast Receiver (DateChangeReceiver) to update the date every 24 hours. The BoradcastReceiver will accordingly update the database after receiving the intent by adding an new row in the table to represent the new date (the yellow line shown in the graph).

3 Functionalities

The application's main functionality is to enable users to track their own movements (exercises) per day. By default, the application will automatically track users' movement in the background after launching and receiving users' permission for accessing information about their locations without users explicitly starting the tracking functionality. However, users still have the choice to turn off the tracking functionality easily on the user interface as well. The details about the functionalities of the application and associated interface will be demonstrated in this section.

When the application is launched for the first time, it will ask users for their permissions to access information about their locations (Fig 9). If users grant the permissions to the application, the tracking functionality will begin automatically. If users reject the application's request for accessing their location information, the application will not track users' movement. Users therefore can only use the application to view the history record of movement stored in the application if there is any. In the case of users refusing to grant permission, when the users

launch the application again, the application will pop up a dialog explaining the usage of the permission and asking for users' permission again (*Fig 10 & Fig 11*). If users reject the request again and choosing "Deny & don't Ask Again" in *Fig 11*, the request will not pop up again. The users then can only grand the permission to the application through the settings of the application.
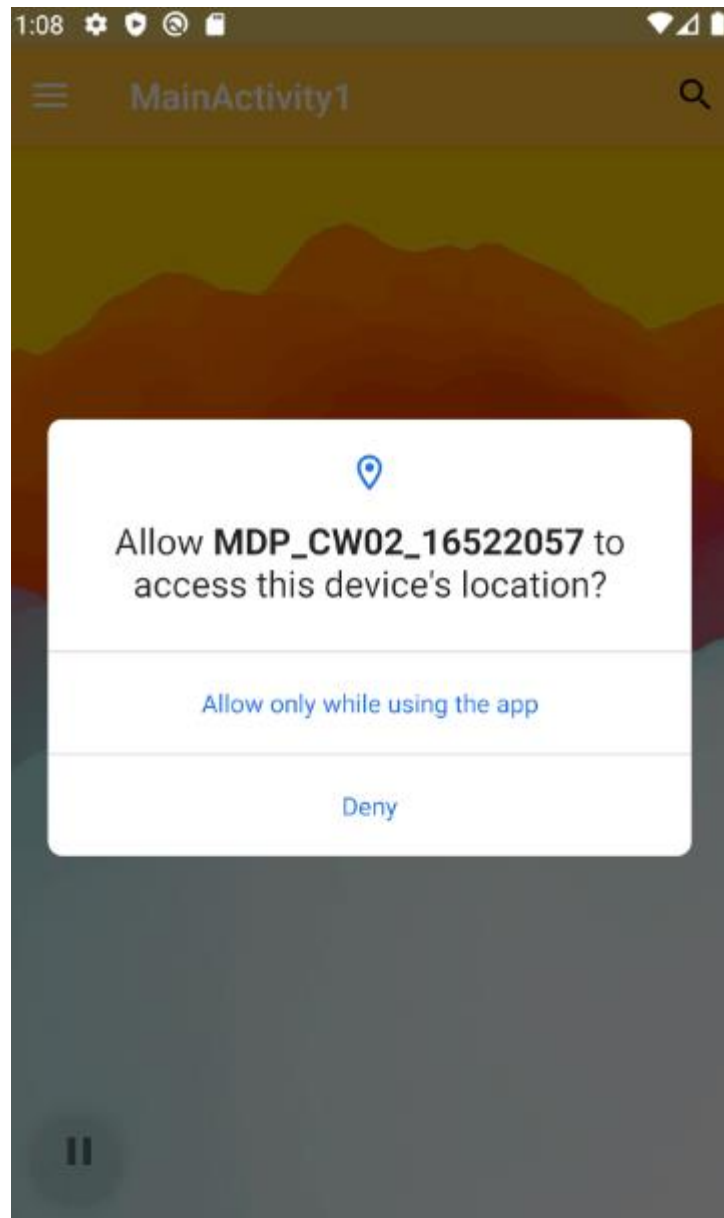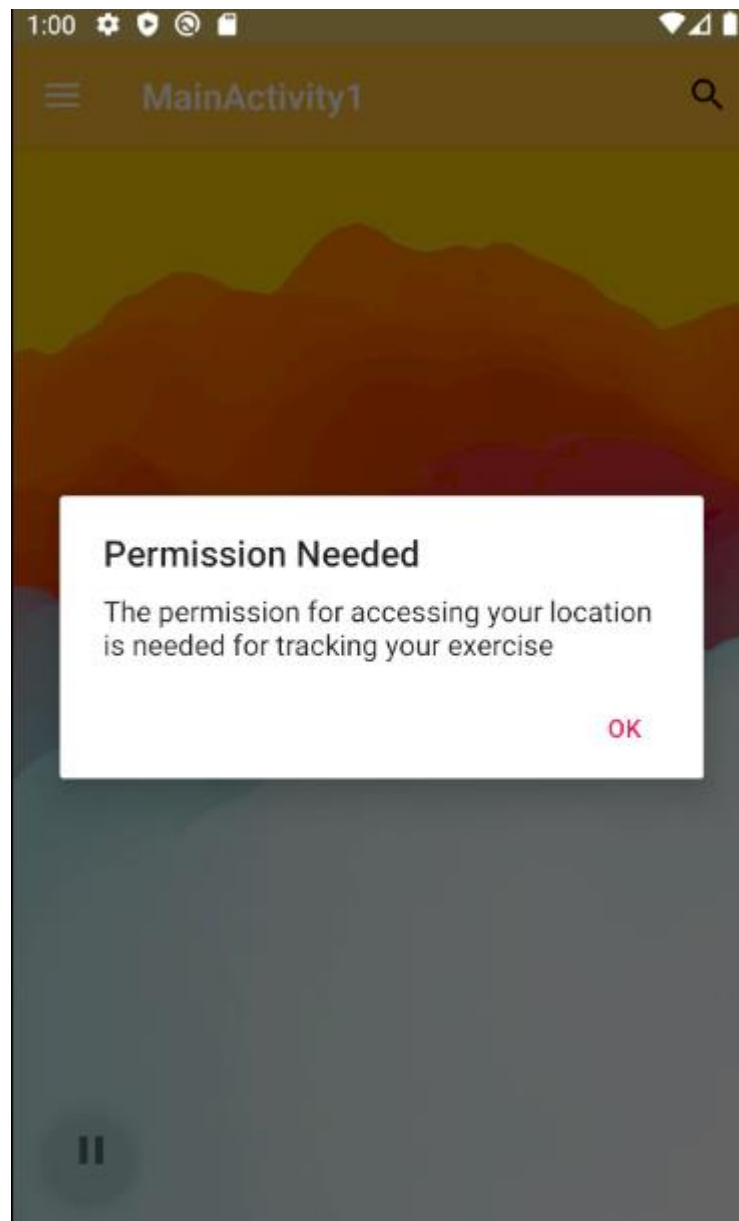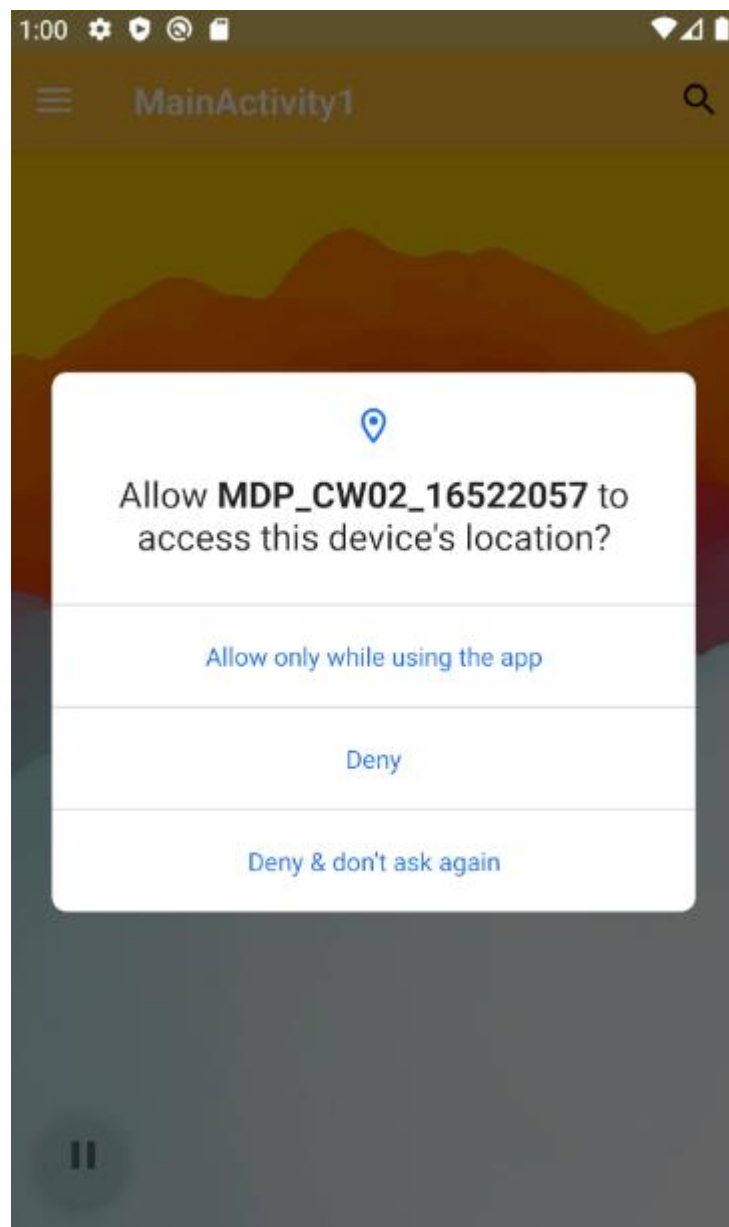

Fig 9

Fig 10

Fig 11

In the case of users granting the permission to the application, the application will automatically start tracking the movement. Then the users can view today's movement in the following interface:
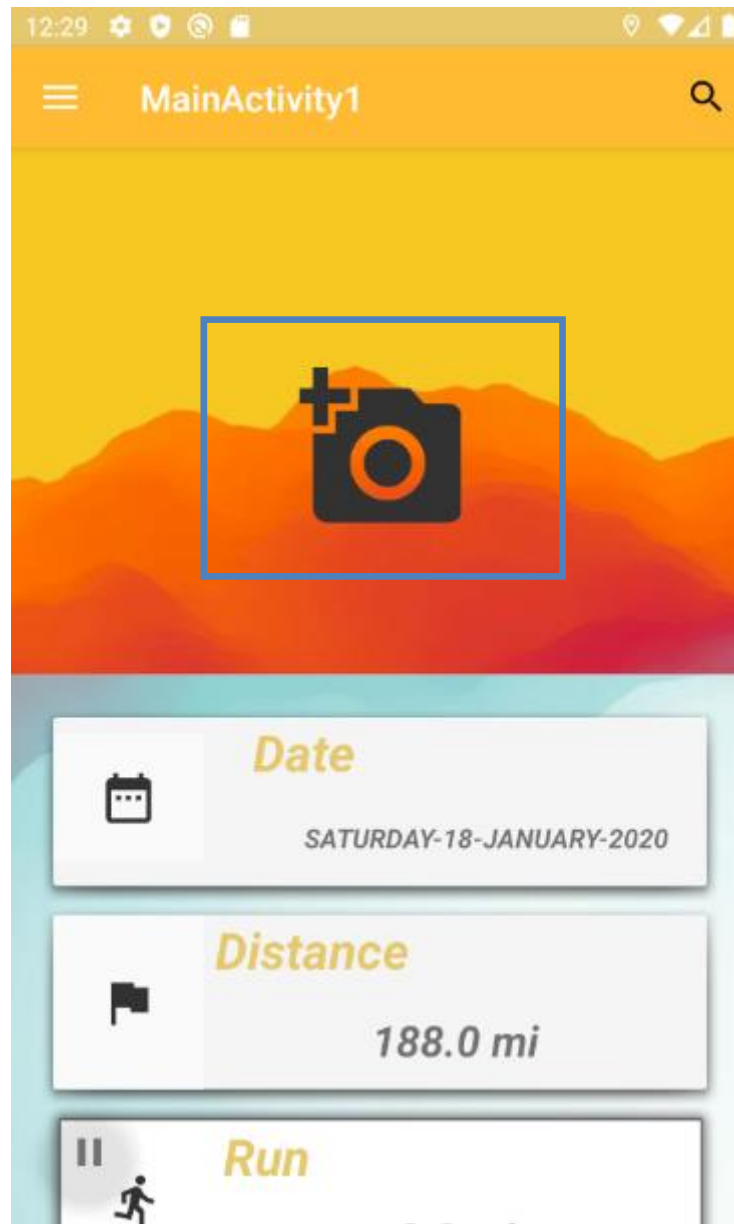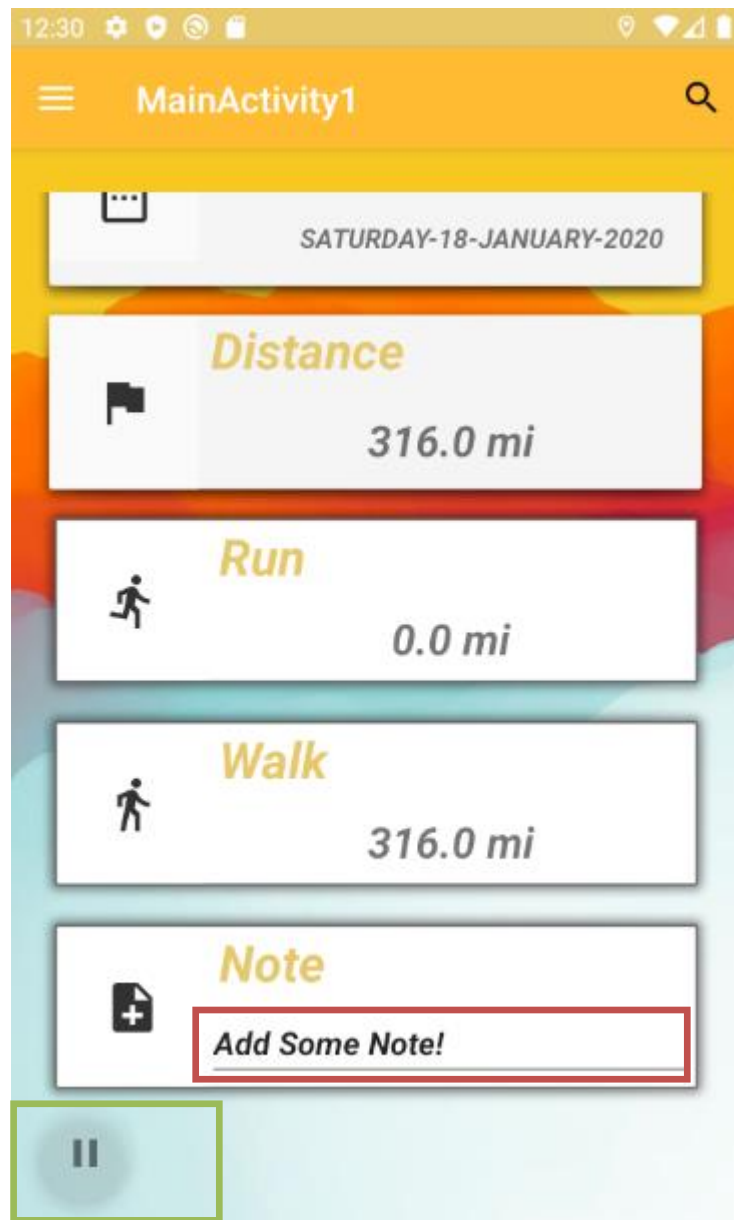
Fig 2

Fig 3

The total distance of the movement will be categorized as running movement and walking movement based on the speed. Users can stop the tracking functionality by pressing the pause icon (highlighted in green square in *Fig 3*). Users can also add pictures from gallery to the record by pressing the photo icon (highlighted in blue square in *Fig 2*). When users try to choose picture from gallery for the first time, another dialog will pop up to ask for users' permissions for accessing the gallery (*Fig 12*). Users can only choose picture from gallery if they agree to grand the permission. If users reject the request the first time and press the photo icon again, similar to the case of accessing location permission, the users will be provided with an explanation and a second chance to grand the permission. Additionally, the users can also add notes in the record by clicking on the "Add Some Note!" line (highlighted in red square in *Fig 3*). When they finish writing the notes, they need to press the green check icon (highlighted in blue square in *Fig 13*)

on the keyboard to store the information, otherwise, the information will not be stored. After the image and notes are stored, the system will pop up a toast to inform users that the image/notes are stored.
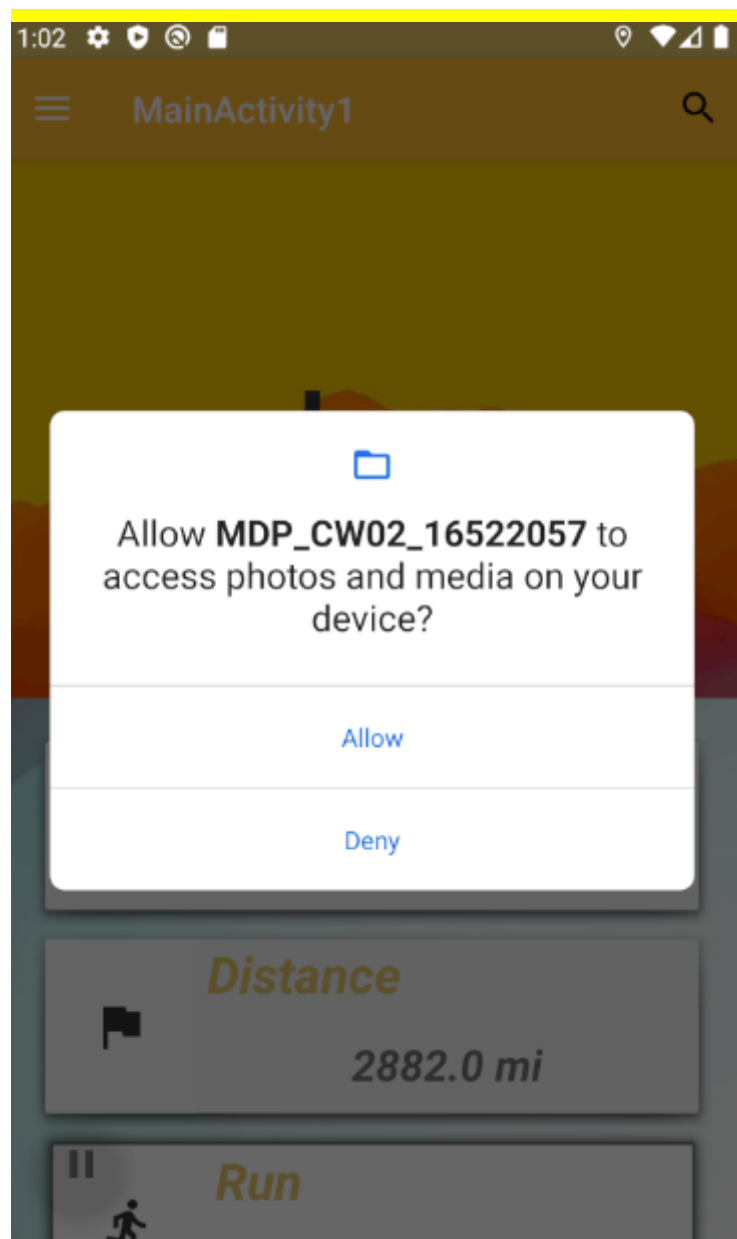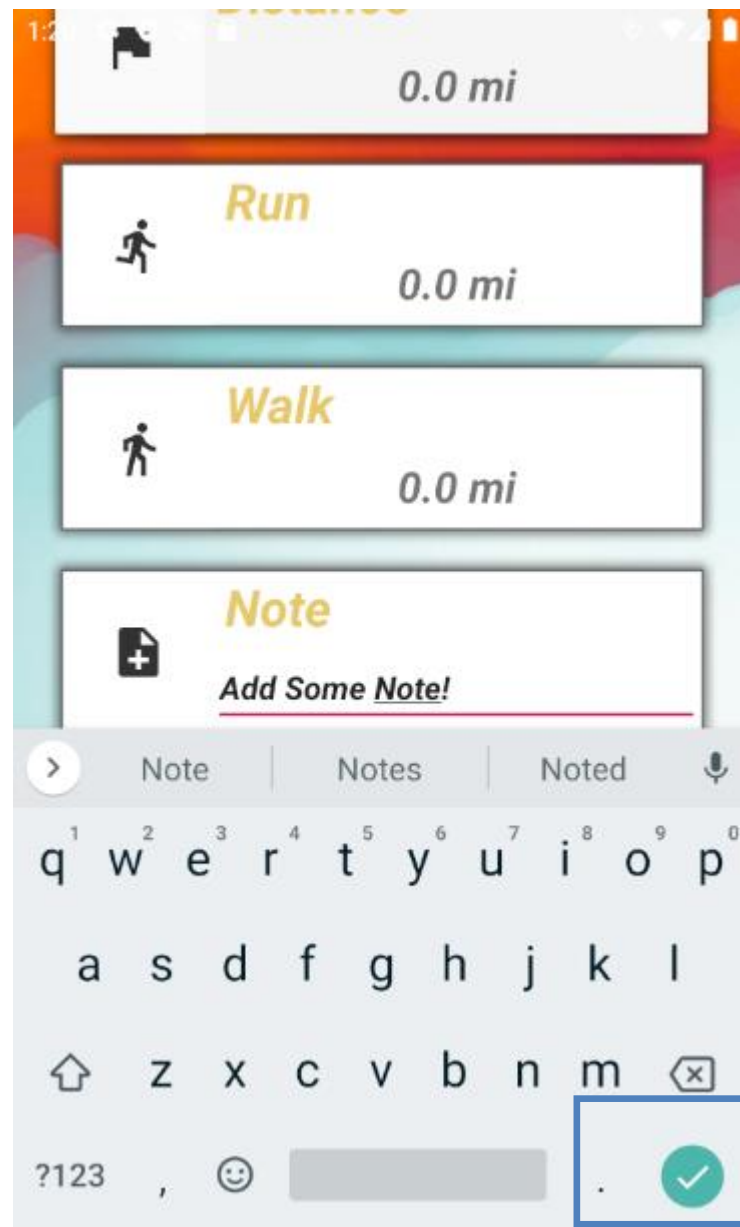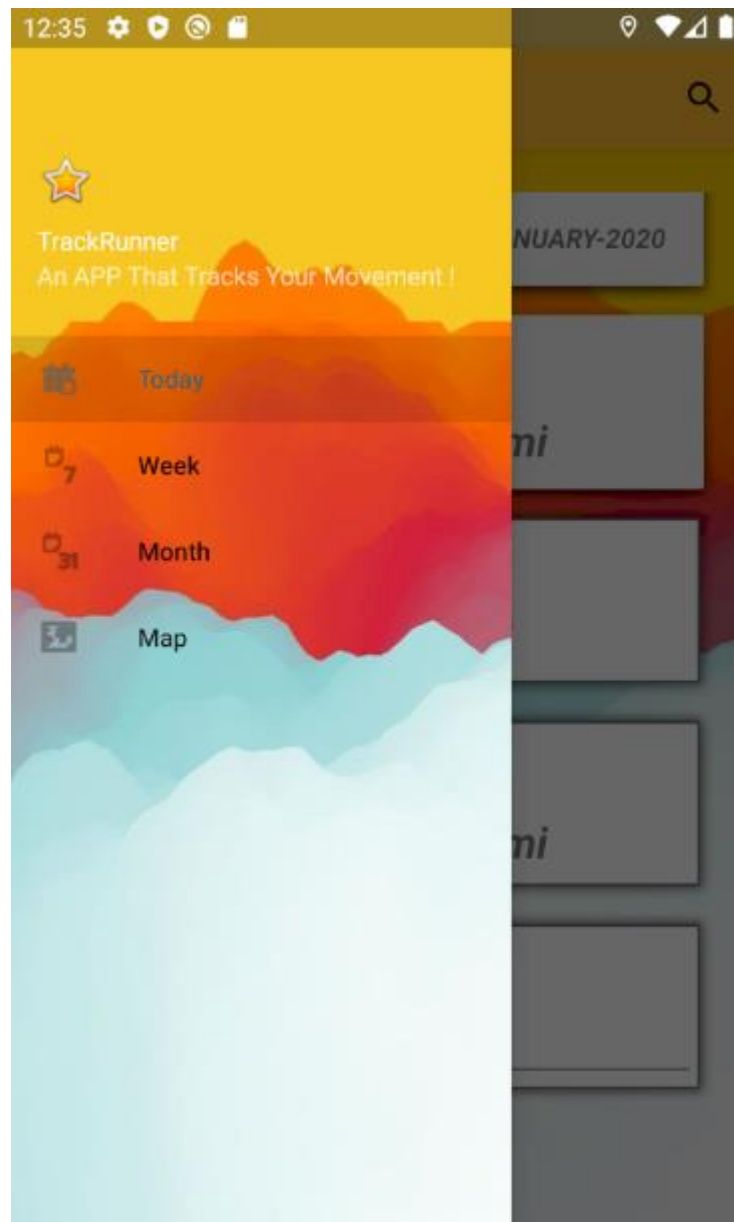

Fig 12

Fig 13

The users can choose other functionalities of the application from the Navigation Menu shown below:

Fig 4

In the Week section, the users can view their movement records for the week (*Fig 5*).Every Sunday, the record of last week will be eliminated. But they can be still retrieved by using searching icon or the month fragment that will be illustrated later. In *Fig 5* users will only view the total distance of certain day of the week. If they want to view more detailed record like the one in *Fig 3 & Fig 2,* they can click on the day that they want to view and if there is any record for that day, a detailed record will be shown. When the users click on the day that do not have any record yet, the system will pop up a toast and notify users. Additionally, if the users are moving, the total distance of corresponding day on the week fragment will update accordingly.
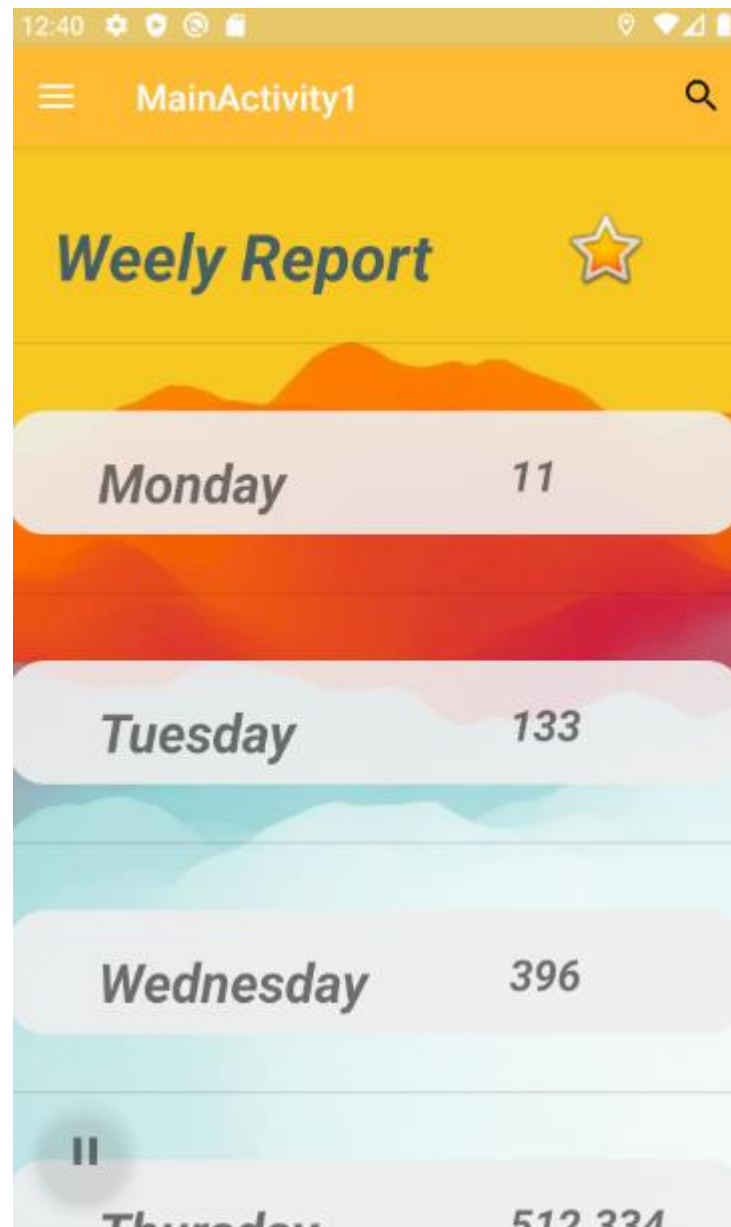
Fig 5

The users can also view the monthly record of their movements in the Month section (*Fig 6*). This section will only show the days that have movements record in the current month. Similar to Week section, users can only view the total distance of associated day. If they want to view more detailed record, they just need to press the corresponding day and a detailed record like the one in *Fig 2 and Fig 3* will be shown.

Fig 6

In the Map section, users can use a google map to locate themselves. (*Fig 7*) They can press the button that is highlighted in red square in *Fig 7* to locate themselves. And they can zoom in or zoom out the map by using the two buttons highlighted in blue square in *Fig 7.*
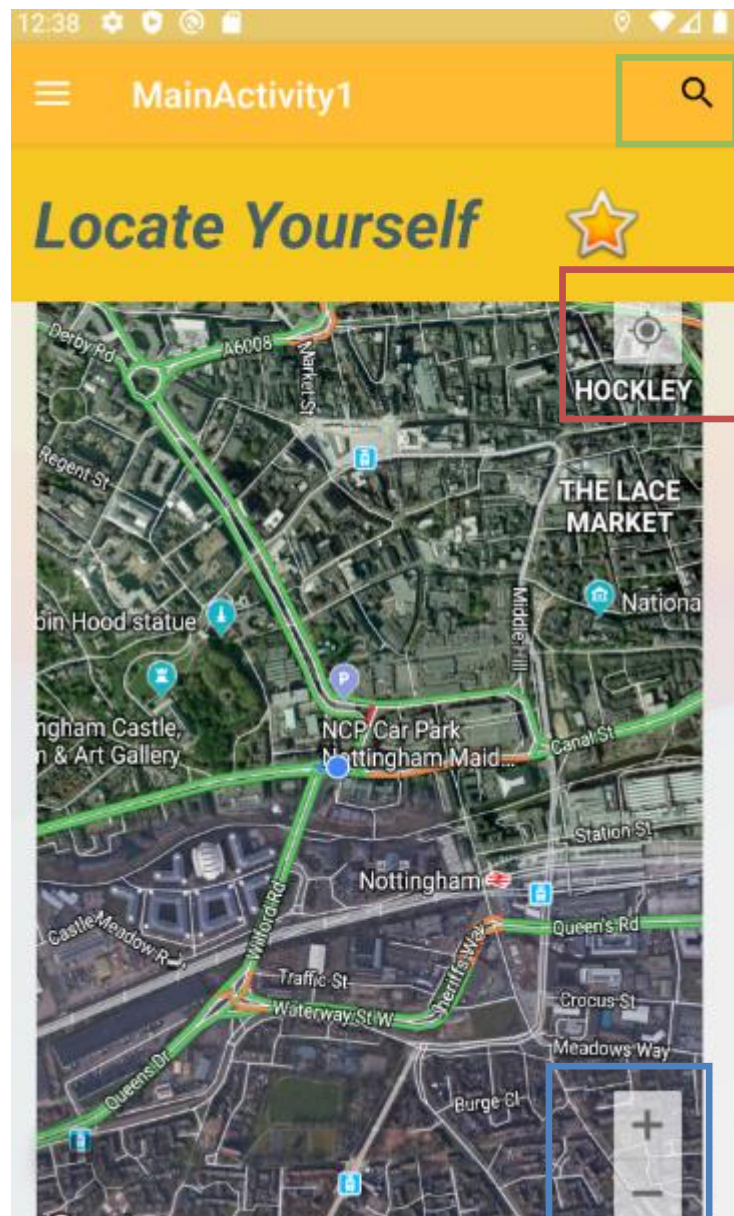
Fig 7

When users want to retrieve the record of a certain date, they can use the searching functionality by pressing the search icon on the right corner of the interface (highlighted in green square in *Fig 7*). When the button is pressed, a datepiker dialog will pop up to enable users to choose a specific date that they want to retrieve (*Fig 8*). After the users picked up a date, if there is a record in database for the date, a detailed record like the one in *Fig 2& Fig 3* will show up, otherwise, the system will inform users that there is no records for the chosen date by using a toast.

Finally, the application will automatically update the date by using an AlarmManager. As explained in the design section, the date will be updated when the Broadcast Receiver receive the intent sent from AlarmManager and then the interface will be updated accordingly. For example, while testing, if changing the system date manually in settings, the record shown in

Today section will change and the data in Week and Month section will update as well. However, one thing needs to notice is that, sometimes, the intent will be received a while after the change of date. So, sometimes there will be a delay in updating the interfaces.
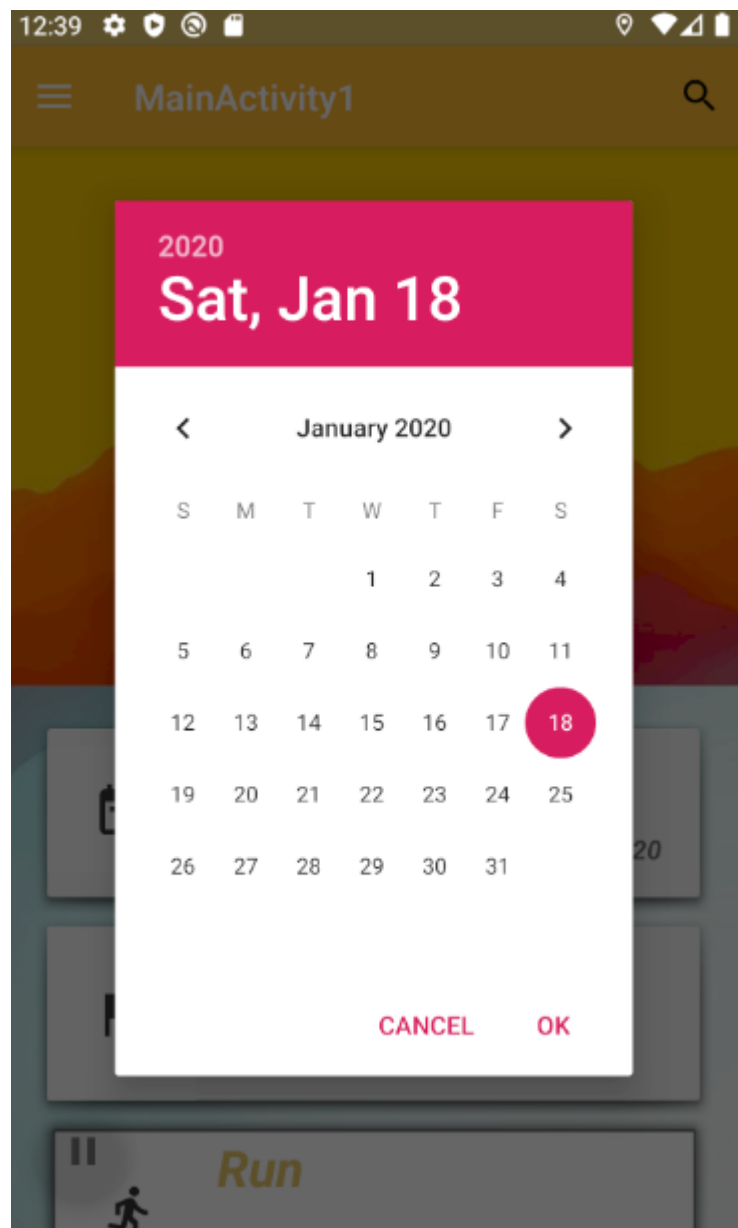

Fig 8