Project Report on

Weather App

By

Group No: 9

4th Year, 7th Semester

Batch: 2016-2020

Sl. No.	Name of the students	Roll No
1	Ausree Ghose	34
2	Delfia Ghosh	35
3	Sukripta Ghosh	36

Department of Computer Science and Engineering

St. Thomas' College of Engineering and Technology

Affiliated to

Maulana Abul Kalam Azad University of Technology, West Bengal

Table of Contents

1. Introduction

1.1 About the Project

The app displays the weather condition of the user's current location. It is done by connecting the application to an API to retrieve the required data from the internet. API has been used for current weather as well as the weather for 5 days. Website used is "https://openweathermap.org/guide". The App shows current as well as the temperature and wind speed for upcoming 5 days at the latitude and longitude of the city the device with this app currently is. The App also gives the time of sunrise and sunset on that particular day. Apart from these details it informs the user about the latitude and longitude the user currently is in as well as the city and country and lastly the pressure and humidity in that city.

2. Main Content

2.1 Technologies Involved

- Android Studio
- Retrofit
- Weather API

2.2 Methodology Followed

- Splash Activity
  - Java
    - It contains code for displaying a page for 3 sec.
    - This is mainly an introductory page to the app.
    - Here we are also fetching the latitude and longitude of the user which has been used in the other activities.
  - Layout
    - Two TextViews have been used under Constraint Layout.
    - They are used to welcome the user to the App.
    - Background has been given an attractive image.

- Main Activity
  - Java
    - It contains several methods to display the temperature for the current day at the user's location (city) & the same for the next 5 days.
    - The methods communicate with the API to retrieve information about the temperature of current day & consecutive 5days.

        - Icons are shown beside each day depending on the temperature on that day.
        - Colour of TextView changes with respect to the temperature on the current day
    - Layout
        - One card has been designed under Relative Layout.
        - The card is used to display the Temperature of the current day.
        - The cards' background colour denotes how high the temperature is on that particular day.
        - Cards' elevation has been added to add shadows to the cards & the corners have been rounded.
        - The temperature of the next consecutive 5 days along with the icons are displayed using ListView and CustomAdapter.

- Search Activity
    - Java
        - It contains one method to display all the details of the Weather on that day for the name of the City that the user enters.
        - The methods communicate with the API to retrieve the data.
        - Icons are shown besides each information depending on the attribute displayed.
        - User enters name of City in the EditText provided.
    - Layout
        - One card has been designed under Relative Layout.
        - The card is used to take input (name of the city) from the user.
        - Cards' elevation has been added to add shadows to the cards & the corners have been rounded.
        - Then using ListView and CustomAdapter the details of weather and location of user have been displayed along with relevant icons.

- Sun Activity
    - Java
        - It contains several methods to display the sunrise & sunset time for the current day according to user's location.
        - The methods communicate with the API to retrieve information about the timestamp of sunrise & sunset.
        - The timestamps are then converted in the following way:

```
Integer sr=weatherResponse.sys.getSunrise();

Calendar cal1=Calendar.getInstance(Locale.English);

cal1.setTimeInMillis(sr*1000L);
```

String date1=DateFormat.format("dd-MM-yyyy hh:mm:ss",cal1).toString();

String[] time1=date1.split(" ");

- "sr" here is the integer type of timestamp we get after calling "getSunrise()" method defined in Sys class of WeatherResponse.java file.
- "date1" is the string containing the date & time.
- "time1" is the string type of array containing date in the first index & time of sunrise in the second index which is finally displayed to the user.
- Similarly, the timestamp of sunset is converted to date & time format & the time of sunset is extracted from it in the same way & is displayed to the user.

  - Layout
    - Two cards have been designed under Relative Layout.
    - The first card is used to display the Sunrise of the current day.
    - The second card is used to display the Sunset of the current day.
    - The cards' background colour has been set in such a way such that it appears as transparent.
    - Cards' elevation has been added to add shadows to the cards & the corners have been rounded.

- Wind Activity
  - Java
    - It contains several methods to display the wind speed for the current day according to user's location & the wind speed of the next 5days.
    - The methods communicate with the API to retrieve information about the wind speed of current day & consecutive 5days.
    - Icons are shown besides each day along with the respective wind speed.
    - Icons are changed according to the wind speed of the days.
  - Layout
    - One card has been designed under Relative Layout.
    - The card is used to display the Wind Speed of the current day.
    - The cards' background colour has been set in such a way such that it appears as transparent.
    - Cards' elevation has been added to add shadows to the cards & the corners have been rounded.
    - The wind speed of the consecutive 5 days along with the icons are displayed using ListView.

- Details Activity
  - Java
    - It contains several methods to display the other details like City, Country, Longitude, Latitude, Pressure & Humidity for the current day according to user's location.
    - The methods communicate with the API to retrieve information about the details.
    - The details are extracted using different getters & setters that have been defined in the WeatherResponse.java file.
  - Layout
    - Three cards has been designed under Relative Layout.
    - The first card is used to display the Longitude & Latitude according to user's current location.
    - The second card is used to display the City & Country according to user's current location.
    - The third card is used to display the Pressure & Humidity according to user's current location.
    - The cards' background colour has been set in such a way such that it appears as transparent.
    - Cards' elevation has been added to add shadows to the cards & the corners have been rounded.

- Retrofit
  - Retrofit turns HTTP API into a Java Interface.

```
public interface WeatherService {
   @GET("data/2.5/weather?")
   Call<WeatherResponse> getCurrentWeatherData(@Query("lat")
String lat, @Query("lon") String lon, @Query("APPID") String
APPID);

   @GET("data/2.5/weather?")
   Call<WeatherResponse> getCurrentWeatherDataC(@Query("q")
String city, @Query("units") String unit, @Query("APPID") String
APPID);

   @GET("data/2.5/forecast?")
   Call<WeatherData> getFutureWeatherData(@Query("lat") String lat,
@Query("lon") String lon, @Query("APPID") String APPID);
}
```

  - The Retrofit class generates and implementation of the WeatherService interface.

```
Retrofit retrofit=new Retrofit.Builder()
        .baseUrl(BaseUrl)
        .addConverterFactory(GsonConverterFactory.create())
```

```
        .build();
    WeatherService service=retrofit.create(WeatherService.class);
```

- Each call from the created WeatherService can make a synchronous or asynchronous HTTP request to remote web server.

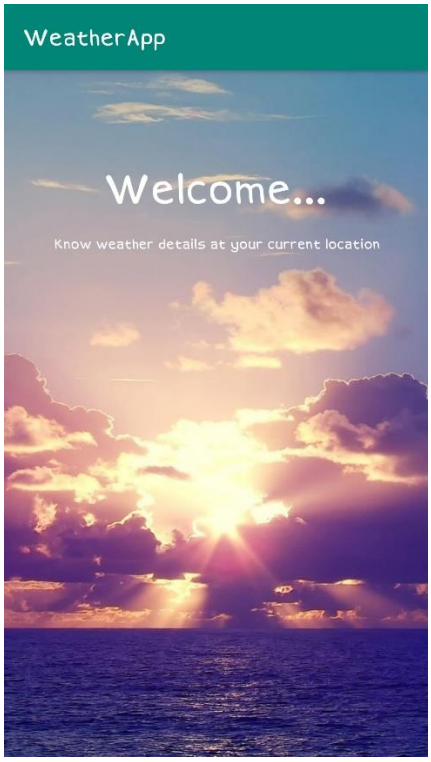
```
Call<WeatherData>
call1=service.getFutureWeatherData(lat,lon,Appid);
```

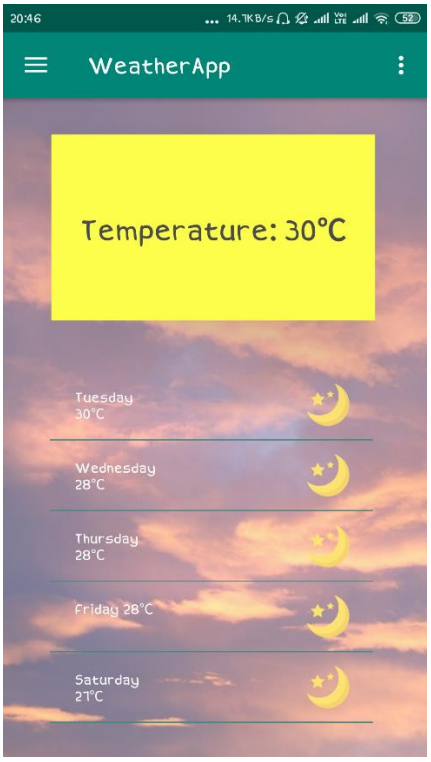
- On Android, callbacks will be executed on the main thread.
- On the JVM, callbacks will happenon the same thread that executed the HTTP request.
- Retrofit is the class through which the API interfaces are turned into callable objects. By default, Retrofit will give sane defaults for the platform but it allows for customization.

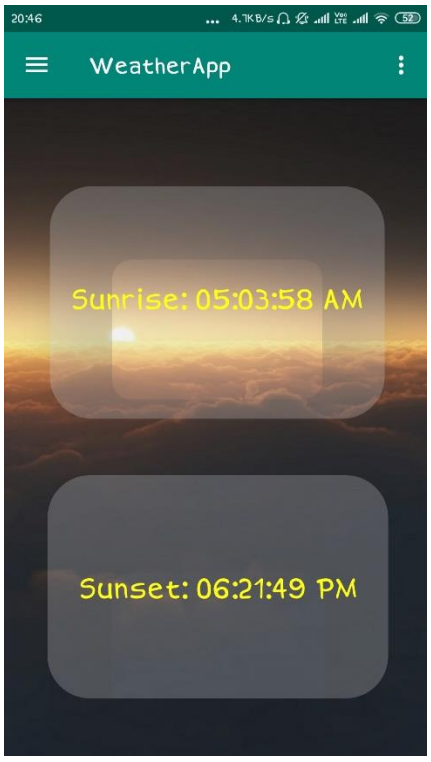
3. Contribution

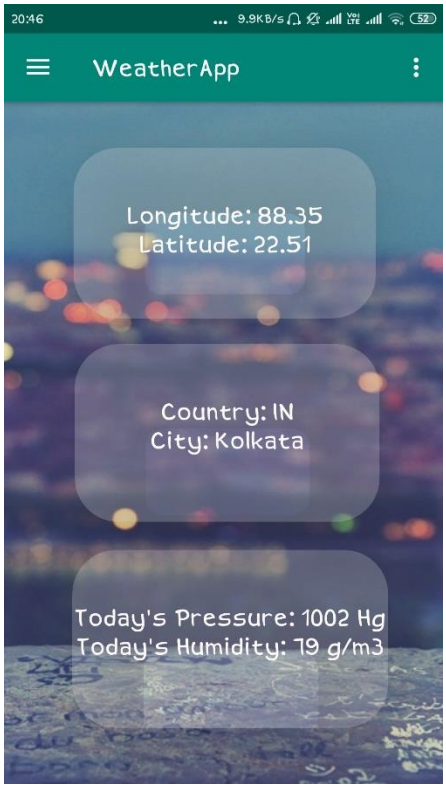
Roll Number	Name	Contribution
34	Ausree Ghose	▪ Nav Drawer (For all activities) ▪ Search Activity (Layout + Java) ▪ Temp Activity (Java) ▪ Splash Activity (Layout + Java) ▪ Project Report
35	Delfia Ghosh	▪ Temp Activity (Layout) ▪ Sun Activity (Layout + Java) ▪ Details Activity (Layout + Java) ▪ Wind Activity (Layout + Java) ▪ Custom Adapter (Java) ▪ Project Report
36	Sukripta Ghosh	▪ Document Printing

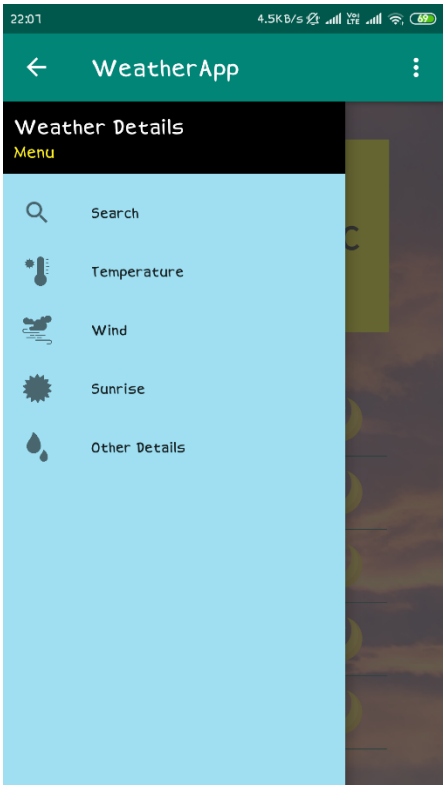
4. Screenshots

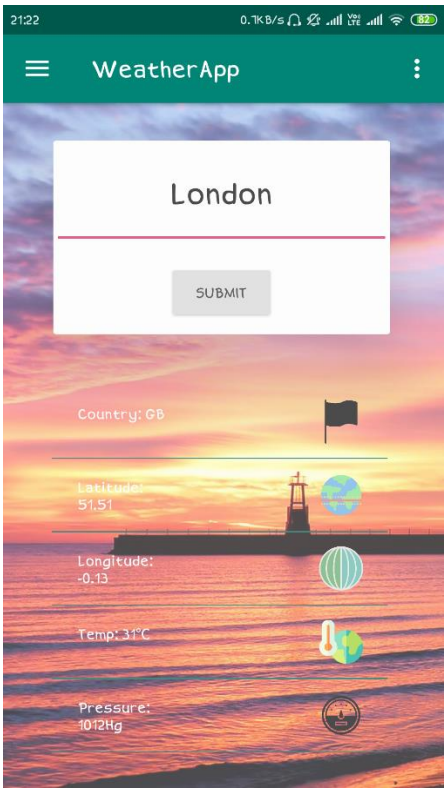


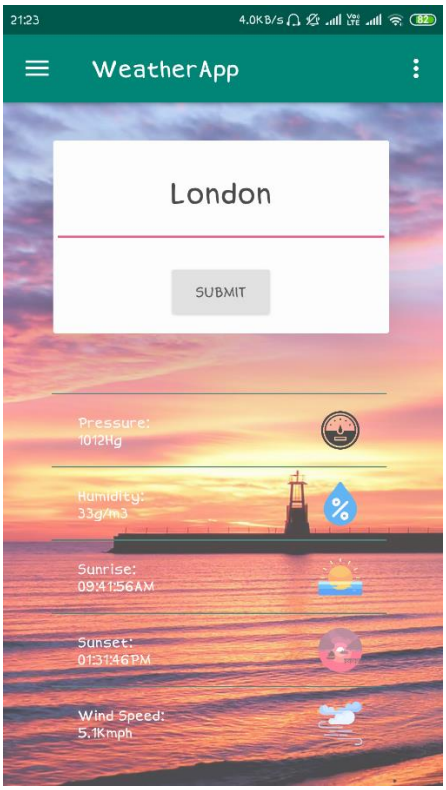


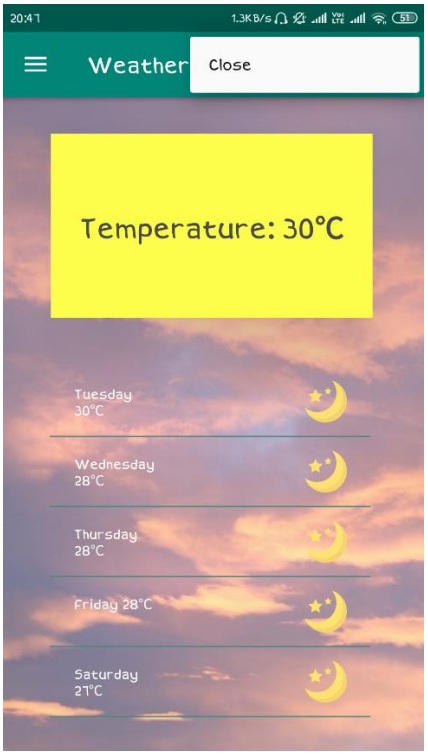



20:46
9.9KB/s
WeatherApp
Longitude: 88.35
Latitude: 22.51
Country: IN
City: Kolkata
Today's Pressure: 1002 Hg
Today's Humidity: 79 g/m3


22:07
4.5KB/s
WeatherApp
Weather Details
Menu
Search
Temperature
Wind
Sunrise
Other Details


21:22
0.7KB/s
WeatherApp
London
SUBMIT
Country: GB
Latitude:
51.51
Longitude:
-0.13
Temp: 31°C
Pressure:
1012Hg


21:23
4.0KB/s
WeatherApp
London
SUBMIT
Pressure:
1012Hg
Humidity:
33g/m3
Sunrise:
09:41:56AM
Sunset:
01:31:46PM
Wind Speed:
5.1Kmph


20:47
1.3KB/s
Weather
Close
Temperature: 30°C
Tuesday
30°C
Wednesday
28°C
Thursday
28°C
Friday 28°C
Saturday
27°C

5. Conclusion

Android is a truly open, free development platform. Handset makers can use & customize the platform without paying any royalty.
A component-based architecture inspired by Internet mash-ups. Parts of one application can be used in other ways not originally envisioned by the developer. Can even replace built-in components with own improvised versions.
This unleashes a new round of creativity in the mobile space.
Here, the Weather App that has been discussed about, fulfils certain purposes required by the user such as knowing the temperature, wind speed, humidity, pressure, sunset, sunrise, latitude, longitude, city & country based on the current location of the user.
It also provides a section where the user can search & find all the above-mentioned details of a place, he/she is looking for.
Besides all these, the app has got interesting & nice-looking user interface that might be a pleasure to the eyes at many a times.
So, it can be concluded by mentioning that the Weather App as useful as it'll ever be with all it is features, will help the users to be aware about the weather conditions at every step.