EXC1081

Open Source Development for Google Applications

Project: Weather forecast app using the given API

Name: Pallav Gupta

Reg. No. : 16BCE0941

INDEX

Topic **Page no**

ABSTRACT

If you search on Google Play Store, you'll find most weather apps either full of ads or require too many permissions, or includes unwanted features that most of us never use.
So, instead, wouldn't it be great to offer a simple weather app of your own eliminating those unwanted features that are irrelevant?

The aim of the Weather Forecasting App is to display the weather for one particular day and should also show the forecast of the weather over the next 5 days. API 1: https://developer.accuweather.com/ was used to display the forecast. This API allows the user to only access the data 50 times in a day.

INTRODUCTION

This App is called The WeatherApp. This app uses the API from ñaccuweatherò

To display the forecast over a particular city. The front page displays the Max and Min temperature of the particular city. To check he forecast for next 5 days click on the ñForecast for next 5 daysò button.

It will take you to another page where the date as well as the time and temperatures for those particular days are displayed. It also tells the GMT time.

This app can be used to check and plan your schedule according to the weather at the time.

METHODOLOGY

First of all create a Weather API from **https://developer.accuweather.com/**.

Create a new project in Android Studio in a normal way.

Open your **AndroidManifest.xml** file and add **internet connect permission**.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.pallavgupta.weatherapptesting">

    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".StartPage">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".MainActivity"></activity>
    </application>

</manifest>
```

Open your **activity_main.xml** file.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="@drawable/goodweather"
    android:padding="16dp"
    tools:context="com.example.pallavgupta.weatherapptesting.MainActivity">

    <TextView
        android:layout_width="354dp"
        android:layout_height="wrap_content"
        android:text="Daily forecast "
```

```
        android:textAlignment="center"
        android:textColor="#ff0000"
        android:textSize="20sp"
        android:textStyle="bold" />

    <ListView
        android:id="@+id/idListView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />


</LinearLayout>
```

Now we are going to create a Java class file named **Weather.java.**

```
package com.example.pallavgupta.weatherapptesting;

/**
 * Created by Pallav Gupta on 18-03-2018.
 */

public class Weather {

    String date;
    String minTemp;
    String maxTemp;
    String link;

    public String getDate() {
        return date;
    }

    public void setDate(String date) {
        this.date = date;
    }

    public String getMinTemp() {
        return minTemp;
    }

    public void setMinTemp(String minTemp) {
        this.minTemp = minTemp;
    }

    public String getMaxTemp() {
        return maxTemp;
    }

    public void setMaxTemp(String maxTemp) {
        this.maxTemp = maxTemp;
    }

    public String getLink() {
        return link;
    }

    public void setLink(String link) {
        this.link = link;
    }
}
```

We use the **HttpURLConnection** class to make the remote request.

```
package com.example.pallavgupta.weatherapptesting;

import android.net.Uri;
import android.util.Log;

import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Scanner;

import javax.net.ssl.HttpsURLConnection;

/**
 * Created by Pallav Gupta on 18-03-2018.
 */

public class NetworkUtils {
    private static final String TAG="NetworkUtils";
    private final static String WEATHERDB_BASE_URL=
            "http://dataservice.accuweather.com/forecasts/v1/daily/5day/190795";

    private final static String API_KEY="YwAjkF079AOJkshFNPxGX6dTAQWVlxtY";


    private final static String METRIC_VALUE="true";

    private final static String PARAM_API_KEY="apikey";

    private final static String PARAM_METRIC="metric";

    public static URL buildUrlForWeather() {

        Uri buildUri=Uri.parse(WEATHERDB_BASE_URL).buildUpon()
                .appendQueryParameter(PARAM_API_KEY,API_KEY)
                .appendQueryParameter(PARAM_METRIC,METRIC_VALUE)
                .build();

        URL url = null;
        try {
            url = new URL(buildUri.toString());
        } catch (MalformedURLException e){
            e.printStackTrace();
        }

        Log.i(TAG, "buildUrlForWeather: url: "+url);
        return url;
    }

    public static String getResponseFromHTTPUrl(URL url) throws IOException{

        HttpURLConnection urlConnection=(HttpURLConnection) url.openConnection();
        try {
            InputStream in = urlConnection.getInputStream();

            Scanner scanner= new Scanner(in);
            scanner.useDelimiter("\\A");
```

```
            boolean hasInput = scanner.hasNext();
            if (hasInput) {
                return scanner.next();
            }else {
                return null;
            }
        } finally {
            urlConnection.disconnect();
        }
    }
}
```

```
package com.example.pallavgupta.weatherapptesting;

import android.content.Context;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.ArrayList;

/**
 * Created by Pallav Gupta on 18-03-2018.
 */

public class WeatherAdapter extends ArrayAdapter<Weather>{

    public WeatherAdapter(@NonNull Context context, ArrayList<Weather>
weatherArrayList) {
        super(context, 0,weatherArrayList);
    }

    @NonNull
    @Override
    public View getView(int position, @Nullable View convertView, @NonNull
ViewGroup parent) {
        Weather weather=getItem(position);


        if (convertView==null){

            convertView=
LayoutInflater.from(getContext()).inflate(R.layout.list_item ,parent,false);
        }


        TextView dateTextView= convertView.findViewById(R.id.tvDate);
        TextView minTextView= convertView.findViewById(R.id.tvLowTemperature);
        TextView maxTextView= convertView.findViewById(R.id.tvHighTemperature);


        dateTextView.setText(weather.getDate());
        minTextView.setText(weather.getMinTemp());
        maxTextView.setText(weather.getMaxTemp());

        return convertView;



    }
}
```

Now time to move towards the **MainActivity.java** class.

When we have the complete response, we convert it to a **JSONObject** object.

```
package com.example.pallavgupta.weatherapptesting;

import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.ListView;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Iterator;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = MainActivity.class.getSimpleName() ;
    private ArrayList<Weather>weatherArrayList= new ArrayList<>();
    private ListView listview;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        listview=findViewById(R.id.idListView);

        URL weatherUrl= NetworkUtils.buildUrlForWeather();
        new FetchWeatherDetails().execute(weatherUrl);
        Log.i(TAG , "onCreate: weatherUrl:"+weatherUrl);
    }



    private class FetchWeatherDetails extends AsyncTask<URL,Void,String>{


        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }

        @Override
        protected String doInBackground(URL... urls) {
            URL weatherUrl= urls[0];
            String weatherSearchResults= null;

            try {
```

```
                weatherSearchResults=
NetworkUtils.getResponseFromHTTPUrl(weatherUrl);
            } catch (IOException e){
                e.printStackTrace();
            }
            Log.i(TAG, "doInBackground:
weatherSearchResults:"+weatherSearchResults);
            return weatherSearchResults;
        }

        @Override
        protected void onPostExecute(String weatherSearchResults) {
            if (weatherSearchResults != null && !weatherSearchResults.equals("")) {
                weatherArrayList=parseJSON(weatherSearchResults);

                Iterator itr = weatherArrayList.iterator();
                while (itr.hasNext()) {
                    Weather weatherInIterator = (Weather) itr.next();
                    Log.i(TAG, "onPostExecution: Date: " +
weatherInIterator.getDate() +
                            "Min: " + weatherInIterator.getMinTemp() +
                            "Max: " + weatherInIterator.getMaxTemp() +
                            "Link: " + weatherInIterator.getLink());
                }
            }

            super.onPostExecute(weatherSearchResults);
        }
    }

    private ArrayList<Weather> parseJSON(String weatherSearchResults) {

        if (weatherArrayList != null){

            weatherArrayList.clear();
        }

        if (weatherSearchResults != null){
            try {
                JSONObject rootobject = new JSONObject(weatherSearchResults);
                JSONArray results= rootobject.getJSONArray("DailyForecasts");

                for (int i = 0; i <results.length() ; i++) {
                    Weather weather= new Weather();


                    JSONObject resultsObj =results.getJSONObject(i);
                    String date= resultsObj.getString("Date");
                    weather.setDate(date);



                    JSONObject temperatureObj=
resultsObj.getJSONObject("Temperature");
                    String minTemperature =
temperatureObj.getJSONObject("Minimum").getString("Value");
                    weather.setMinTemp(minTemperature);


                    String maxTemperature =
temperatureObj.getJSONObject("Maximum").getString("Value");
                    weather.setMaxTemp(maxTemperature);



                    String link = resultsObj.getString("Link");
                    weather.setLink(link);
```

```
                    /* Log.i(TAG,"parseJSON: date:"+date +" "+
                            "Min: "+minTemperature+" "+
                     "Max: "+maxTemperature+" "+
                     "Link: "+link);*/

                     weatherArrayList.add(weather);

                     }


                     if (weatherArrayList != null){

                     WeatherAdapter weatherAdapter = new
WeatherAdapter(this,weatherArrayList);
                     listview.setAdapter(weatherAdapter);

                     }


                     return weatherArrayList;

            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        return null;
    }
}
```

This is the list_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:background="#afeeee"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/tvDate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="2018-03-18"
        android:textStyle="bold"
        android:textSize="20sp"
        android:textColor="@color/colorPrimary" />

    <TextView
        android:id="@+id/tvTemperature"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Temperature"
        android:textStyle="italic"
        android:textSize="18sp"
        android:textColor="@color/colorPrimaryDark" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_weight="0.5"
```

```
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="High"
            android:textStyle="normal"
            android:textSize="18sp"
            android:textColor="@color/colorAccent" />

        <TextView
            android:id="@+id/tvHighTemperature"
            android:layout_weight="1"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="33"
            android:textStyle="normal"
            android:textSize="18sp"
            android:textColor="@color/colorAccent" />

    </LinearLayout>


    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_weight="0.5"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="Low"
            android:textStyle="normal"
            android:textSize="18sp"
            android:textColor="#32cd32" />

        <TextView
            android:id="@+id/tvLowTemperature"
            android:layout_weight="1"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="23"
            android:textStyle="normal"
            android:textSize="18sp"
            android:textColor="#32cd32" />

    </LinearLayout>



</LinearLayout>
```

We make the front page

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:background="@drawable/wow"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.pallavgupta.weatherapptesting.StartPage">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
```

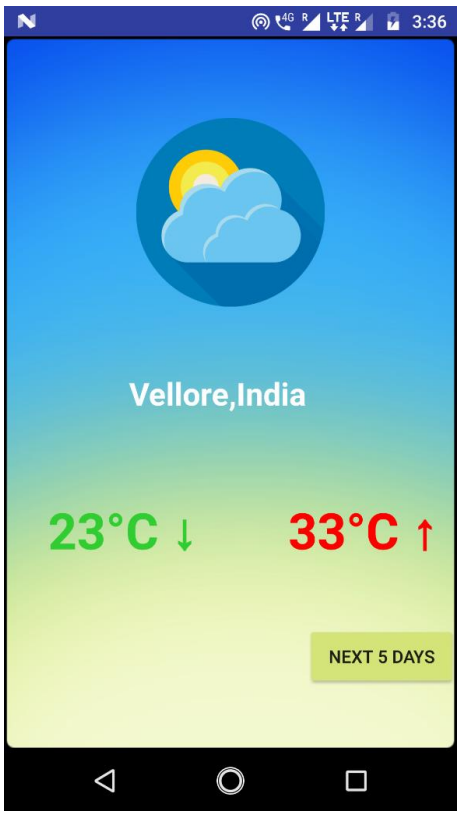
```
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_toLeftOf="@+id/Forecast"
        android:layout_toStartOf="@+id/Forecast"
        android:text="Vellore,India"
        android:textColor="#fff"
        android:textSize="25sp"
        android:textStyle="bold" />

    <ImageView
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:layout_above="@+id/textView"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="52dp"
        android:adjustViewBounds="false"
        android:cropToPadding="false"
        app:srcCompat="@drawable/logo" />
    <Button
        android:id="@+id/Forecast"
        android:layout_width="120dp"
        android:layout_height="50dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_marginBottom="50dp"
        android:backgroundTint="@color/colorGreen"
        android:text="Next 5 days" />


    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/Forecast"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_marginBottom="49dp"
        android:layout_marginLeft="35dp"
        android:layout_marginStart="35dp"
        android:text="23°C ↓ "
        android:textColor="#32cd32"
        android:textSize="40sp"
        android:textStyle="bold"
        android:id="@+id/minTemp" />

    <TextView
        android:id="@+id/maxTemp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/minTemp"
        android:layout_alignBottom="@+id/minTemp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_marginEnd="16dp"
        android:layout_marginRight="16dp"
        android:text="33°C ↑"
        android:textColor="#ff0000"
        android:textSize="40sp"
        android:textStyle="bold" />



</RelativeLayout>
```

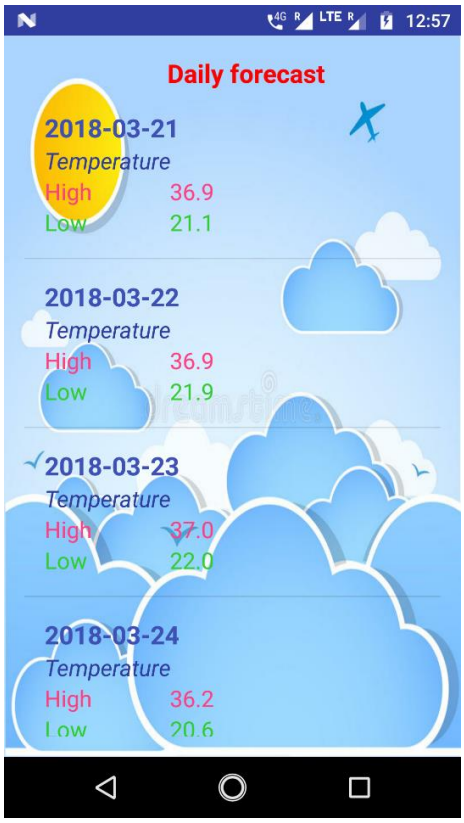
```
package com.example.pallavgupta.weatherapptesting;
```

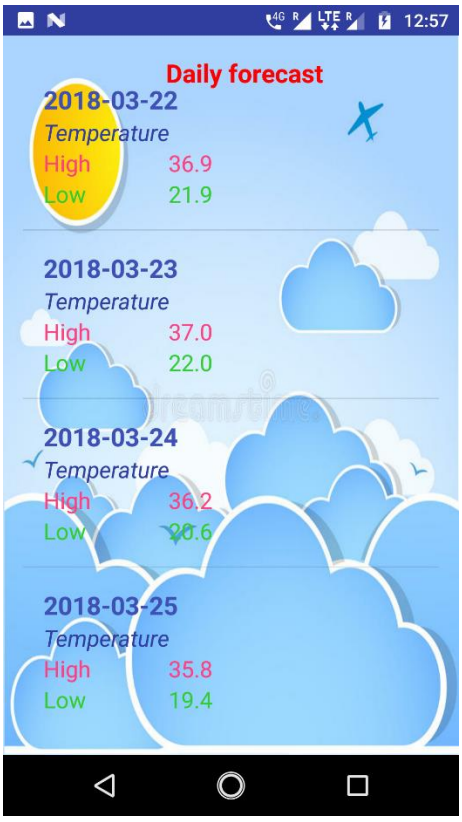
```java
import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

public class StartPage extends AppCompatActivity {

    public Button btn;


    public void init() {

        btn = findViewById(R.id.Forecast);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent toy = new Intent(StartPage.this, MainActivity.class);
                startActivity(toy);
            }
        });

    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_start_page);

        init();


    }


}
```

The API used is API 1: https://developer.accuweather.com/

RESULT OF THE EXECTUION







Since I had no basic knowledge of Java and JavaScript, creating an App was certainly a difficult task. I tried using the google volley library from the git but due to some error I was not able to apply the function from the library. So I had to make different java files for the connection. I had to take reference from many sources.

CONCLUSION

So from developing this App I draw this conclusion that any data from the web can be connected to an app using an API.

Also I can conclude that, if you search on Google Play Store, you'll find most weather apps either full of ads or require too many permissions, or includes unwanted features that most of us never use, we can develop an app without any ads or requiring too many permissions.

REFERENCES

www.google.com
www.youtube.com
https://developer.android.com/studio/index.html
https://www.androidauthority.com/android-studio-tutorial-beginners-637572/
https://www.codecademy.com/learn/learn-java