**Miniproiect PPA**

**Weather app**

***- Faza Finala -***

Moise Andrei

Mihailescu Bogdan

**422C/421B**

**1.Descrierea temei**

Aceasta aplicatie are rolul de a oferi informatii referitoare la vremea actuala. Odata cu deschiderea aplicatiei, se va introduce numele orasului dorit si se vor furniza informatiile meteorologice.

Pentru a dezvolta aplicatia, s-a impartit proiectarea sa in 5 pasi:

* Crearea interfetei grafice
* Declararea widgets-urilor
* Cautarea informatiilor meteorologice
* Instantierea unui Task pentru primirea informatiilor in timp real
* Crearea obiectelor si simplificarea informatiilor obtinute pentru a fi usor de citit
* **Modificarea plasarii textview-urilor**
* **Adaugarea a 3 noi textview-uri pentru generarea de noi informatii suplimentare**
* **Modificarea unor butoane**
* **Tratarea unor erori ce cauzau blocarea aplicatiei**

**Prin modificarile aduse, utilizatorul beneficiaza de o interfata mai usor de inteles si mai sugestiva, precum si de o analiza mai elaborata a vremii. Un alt avantaj al modificarilor aduse este ca aplicatia nu se va bloca in cazul in care este apasat butonul de start fara completarea orasului.**

**2.Descrierea punctului de start**

Tutorialul folosit se gaseste la link-ul:

<https://www.youtube.com/watch?v=xri8nlp68oQ>

Acest tutorial proiecteaza aplicatia de la 0, pas cu pas. Se invata lucruri precum adaugarea de text sau de imagini de background la deschiderea aplicatiei si metode de a-l lasa pe utilizator sa introduca date de intrare, de exemplu numele orasului, si a i se comunica date de iesire pe baza datelor de intrare, in acest caz stirile meteo. Pentru obtinerea informatiilor de pe internet, in timp real, se foloseste o cheie API.

Screenshot 1 aplicatie in faza initiala:

Se pot remarca: fundalul, mesajul de intampinare, precum si spatiul unde se cere introducerea numelui orasului, care este completat cu „london”.

Se observa aparitia mesajului “Clouds:broken clouds” in stanga sus a ecranului.



Screenshot 2 aplicatie faza initiala:

Se remarca aparitia mesajului „Clouds:few clouds” in stanga sus a ecranului la introducerea textului „bucharest” .



**3. Descrierea proiectarii elementelor de noutate**

initial dupa adaugarea elementelor de noutate

O prima modificare consta in plasarea mai corecta pe ecran a campurilor cu informatii si mesajului de intampinare, permitand utilizatorului o navigare mult mai usoara prin meniu.

De asemenea se observa schimbarea grafica a butonului din dreapta jos ce trebuie apasat dupa completarea orasului pentru a furniza starea meteo. Acestuia i s-a atribuit o imagine mai sugestiva decat cea initiala (care ducea cu gandul la o aplicatie de email).

In faza initiala daca se apasa butonul pentru generarea de informatii, aplicatia se bloca. Acest lucru reprezenta o problema destul de serioasa pentru utilizator, avand chiar potentialul sa il faca sa doreasca inlocuirea acestei aplicatii. Problema a fost insa remediata.



O alta modificare importanta este adaugarea de casute suplimentare pentru o descriere mult mai detaliata si mai folositoare a vremii. O astfel de casuta ofera informatii referitoare la temperatura orasului selectat de catre utilizator, temperatura a carei valoare este generata in grade celsius. O alta casuta distribuie valoarea exacta a umiditatii in procente iar ultima casuta arata presiunea atmosferica, presiune ce este indicata in hPa(1hPa este echivalent cu 100 Pascal(Pa)).

De asemenea, fiecare camp a fost evidentiat prin plasarea acestuia in interiorul unui contur pentru a se observa cu usurinta informatia furnizata de aplicatie, oferind astfel o imagine mai placuta si mai usor de descifrat.

**4. Descrierea implementarii**

* MainActivity.java:
* package com.weatherapp.moise.weatherapp;  
    
  import android.graphics.Color;  
  import android.os.AsyncTask;  
  import android.os.Bundle;  
  import android.support.design.widget.FloatingActionButton;  
  import android.support.design.widget.Snackbar;  
  import android.support.v7.app.AppCompatActivity;  
  import android.support.v7.widget.Toolbar;  
  import android.view.Gravity;  
  import android.view.View;  
  import android.view.Menu;  
  import android.view.MenuItem;  
  import android.widget.EditText;  
  import android.widget.TextView;  
  import android.widget.Toast;  
    
  import org.json.JSONArray;  
  import org.json.JSONException;  
  import org.json.JSONObject;  
    
  import java.io.IOException;  
  import java.io.InputStream;  
  import java.io.InputStreamReader;  
  import java.net.HttpURLConnection;  
  import java.net.MalformedURLException;  
  import java.net.URL;  
    
  public class MainActivity extends AppCompatActivity {  
    
   **private** EditText cityField;  
   **private** TextView resultWeather;  
   **private** String cityToFind;  
   **private** **TextView temperatura,umiditate,presiune;** @Override  
   protected void onCreate(Bundle savedInstanceState) {  
   super.onCreate(savedInstanceState);  
   setContentView(R.layout.*activity\_main*);  
   Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
   setSupportActionBar(toolbar);  
    
   cityField = (EditText) findViewById(R.id.*editText*);  
   resultWeather = (TextView) findViewById(R.id.*textView*);  
   **temperatura = (TextView) findViewById(R.id.*temperatura\_field*);  
   umiditate = (TextView) findViewById(R.id.*umiditate\_field*);  
   presiune = (TextView) findViewById(R.id.*presiune\_field*);**  
    
   FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.*fab*);  
   fab.setOnClickListener(new View.OnClickListener() {  
   @Override  
   public void onClick(View view) {  
   Snackbar.*make*(view, "Loading Weather", Snackbar.*LENGTH\_LONG*)  
   .setAction("Action", null).show();  
    
   FindWeather(view);  
    
    
   }  
   });  
   }  
    
   public void FindWeather(View v) {  
   cityToFind = cityField.getText().toString();  
   try {  
   ExecuteTask tasky = new ExecuteTask();  
   tasky.execute(("http://api.openweathermap.org/data/2.5/weather?q=" + cityToFind + "&APPID=951782e766706ff422a872d5adf31efd"));  
   } catch (Exception e) {  
   e.printStackTrace();  
    
   }  
    
   }  
    
    
   public class ExecuteTask extends AsyncTask<String, Void, String> {  
   @Override  
   protected String doInBackground(String... strings) {  
    
   String result = "";  
   URL url;  
   HttpURLConnection urlConnection = null;  
    
   try {  
   url = new URL(strings[0]);  
   urlConnection = (HttpURLConnection) url.openConnection();  
   InputStream is = urlConnection.getInputStream();  
   InputStreamReader reader = new InputStreamReader(is);  
   int data = reader.read();  
   while (data != -1) {  
   char current = (char) data;  
   result += current;  
   data = reader.read();  
   }  
    
   return result;  
    
   } catch (MalformedURLException e) {  
   e.printStackTrace();  
   } catch (IOException e) {  
   e.printStackTrace();  
   }  
    
    
   return null;  
   }  
    
   @Override  
   protected void onPostExecute(String s) {  
   try {  
   super.onPostExecute(s);  
    
   try {  
   **String Temperatura1 = "";  
   String Umiditate1 = "";  
   String Presiune1 = "";**  
   String message = "";  
   JSONObject jsonObject = new JSONObject(s);  
    
   String infoWeatherToday = jsonObject.getString("weather");  
   JSONArray array = new JSONArray(infoWeatherToday);  
   for (int i = 0; i < array.length(); i++) {  
   JSONObject jsonSecondary = array.getJSONObject(i);  
   String main = "";  
   String description = "";  
   main = jsonSecondary.getString("main");  
   description = jsonSecondary.getString("description");  
   if (main != "" && description != "") {  
   message += main + ":" + description + "\r\n";  
   }  
   }  
    
    
   **JSONObject jsonTemp = jsonObject.getJSONObject("main");  
   String str\_umiditate = "";  
   Double str\_temperatura = null;  
   String str\_presiune = "";  
   str\_presiune = jsonTemp.getString("pressure");  
   str\_temperatura = jsonTemp.getDouble("temp");  
   Double str\_temperatura\_C = (double) str\_temperatura;  
   str\_temperatura\_C = str\_temperatura\_C - 275.5;  
   str\_umiditate = jsonTemp.getString("humidity");  
   if (str\_presiune != "" && str\_temperatura != null && str\_umiditate != "") {  
   Presiune1 += "Presiune: " + str\_presiune + "hPa";  
   Temperatura1 += "Temperatura: " + str\_temperatura\_C + "°C";  
   Umiditate1 += "Umiditate: " + str\_umiditate + "%";  
   }  
    
    
   if (message != "") {  
   resultWeather.setText(message);  
   temperatura.setText(Temperatura1);  
   umiditate.setText(Umiditate1);  
   presiune.setText(Presiune1);  
   } else {  
   Toast.*makeText*(MainActivity.this, "An Error Occurred", Toast.*LENGTH\_SHORT*).show();  
    
   }  
   } catch (JSONException e) {  
   e.printStackTrace();  
   }  
   }  
   catch(Exception e) {  
   Toast tst = Toast.*makeText*(MainActivity.this, "Invalid City, change it!", Toast.*LENGTH\_LONG*);  
   tst.setGravity(Gravity.*CENTER*, 0, 0);  
   tst.show();  
   e.printStackTrace();  
   }  
    
   }  
    
   }**  
    
   @Override  
   public boolean onCreateOptionsMenu(Menu menu) {  
   // Inflate the menu; this adds items to the action bar if it is present.  
   getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
   return true;  
   }  
    
   @Override  
   public boolean onOptionsItemSelected(MenuItem item) {  
   // Handle action bar item clicks here. The action bar will  
   // automatically handle clicks on the Home/Up button, so long  
   // as you specify a parent activity in AndroidManifest.xml.  
   int id = item.getItemId();  
    
   //noinspection SimplifiableIfStatement  
   if (id == R.id.*action\_settings*) {  
   return true;  
   }  
    
   return super.onOptionsItemSelected(item);  
   }  
  }
* AndroidManifest.xml:
* <?xml version="1.0" encoding="utf-8"?>  
  <manifest xmlns:android="http://schemas.android.com/apk/res/android"  
   package="com.weatherapp.moise.weatherapp">  
    
   <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=".MainActivity"  
   android:label="@string/app\_name"  
   android:theme="@style/AppTheme.NoActionBar">  
   <intent-filter>  
   <action android:name="android.intent.action.MAIN" />  
    
   <category android:name="android.intent.category.LAUNCHER" />  
   </intent-filter>  
   </activity>  
   </application>  
    
  </manifest>
* Content\_main.xml:
* <?xml version="1.0" encoding="utf-8"?>
* <android.support.constraint.ConstraintLayout 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:background="@drawable/background"
* app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"
* tools:context=".MainActivity"
* tools:showIn="@layout/activity\_main">
* <TextView
* android:id="@+id/textView2"
* **android:layout\_width="wrap\_content"**
* **android:layout\_height="wrap\_content"**
* **android:layout\_marginLeft="8dp"**
* **android:layout\_marginRight="8dp"**
* **android:layout\_marginTop="56dp"**
* **android:background="@drawable/camp\_data"**
* **android:gravity="center"**
* **android:text="@string/header\_text"**
* android:textSize="24sp"
* **android:textStyle="bold|italic"**
* **app:layout\_constraintHorizontal\_bias="0.611"**
* **app:layout\_constraintLeft\_toLeftOf="parent"**
* **app:layout\_constraintRight\_toRightOf="parent"**
* **app:layout\_constraintTop\_toTopOf="parent"** />
* <EditText
* android:id="@+id/editText"
* **android:layout\_width="wrap\_content"**
* **android:layout\_height="wrap\_content"**
* **android:layout\_marginEnd="8dp"**
* **android:layout\_marginLeft="8dp"**
* **android:layout\_marginRight="8dp"**
* **android:layout\_marginStart="8dp"**
* **android:layout\_marginTop="52dp"**
* **android:background="@drawable/camp\_data"**
* android:ems="10"
* **android:hint="@string/city"**
* android:inputType="textPersonName"
* **app:layout\_constraintEnd\_toEndOf="parent"**
* **app:layout\_constraintHorizontal\_bias="0.503"**
* **app:layout\_constraintStart\_toStartOf="parent"**
* **app:layout\_constraintTop\_toBottomOf="@+id/textView2"**
* **tools:ignore="MissingConstraints"** />
* **<TextView**
* **android:id="@+id/temperatura\_field"**
* **android:layout\_width="wrap\_content"**
* **android:layout\_height="35dp"**
* **android:layout\_marginLeft="32dp"**
* **android:layout\_marginStart="32dp"**
* **android:layout\_marginTop="64dp"**
* **android:background="@drawable/camp\_data"**
* **android:text="@string/temp"**
* **app:layout\_constraintStart\_toStartOf="parent"**
* **app:layout\_constraintTop\_toBottomOf="@+id/editText"**
* **tools:ignore="MissingConstraints"**
* **tools:textSize="24sp" />**
* **<TextView**
* **android:id="@+id/umiditate\_field"**
* **android:layout\_width="wrap\_content"**
* **android:layout\_height="35dp"**
* **android:layout\_marginLeft="32dp"**
* **android:layout\_marginStart="32dp"**
* **android:layout\_marginTop="108dp"**
* **android:background="@drawable/camp\_data"**
* **android:text="@string/umiditate"**
* **app:layout\_constraintStart\_toStartOf="parent"**
* **app:layout\_constraintTop\_toBottomOf="@+id/editText"**
* **tools:ignore="MissingConstraints"**
* **tools:textSize="24sp" />**
* <TextView
* android:id="@+id/textView"
* android:layout\_width="wrap\_content"
* **android:layout\_height="35dp"**
* **android:layout\_marginLeft="32dp"**
* **android:layout\_marginStart="32dp"**
* **android:layout\_marginTop="20dp"**
* **android:background="@drawable/camp\_data"**
* **android:text="@string/vremea"**
* **app:layout\_constraintStart\_toStartOf="parent"**
* **app:layout\_constraintTop\_toBottomOf="@+id/editText"**
* **tools:ignore="MissingConstraints"**
* tools:textSize="24sp" />
* **<TextView**
* **android:id="@+id/presiune\_field"**
* **android:layout\_width="wrap\_content"**
* **android:layout\_height="35dp"**
* **android:layout\_marginLeft="32dp"**
* **android:layout\_marginStart="32dp"**
* **android:layout\_marginTop="152dp"**
* **android:background="@drawable/camp\_data"**
* **android:text="@string/presiunea"**
* **app:layout\_constraintStart\_toStartOf="parent"**
* **app:layout\_constraintTop\_toBottomOf="@+id/editText"**
* **tools:ignore="MissingConstraints"**
* **tools:textSize="24sp" />**
* </android.support.constraint.ConstraintLayout>
* Strings.xml:

<resources>

<string name="app\_name">Weather App</string>

<string name="action\_settings">Settings</string>

<string name="header\_text">Welcome To Weather App\n Please Enter A City</string>

<string name="city">City</string>

**<string name="presiunea">Presiunea</string>**

**<string name="umiditate">Humidity</string>**

**<string name="temp">Temperature</string>**

<string name="vremea">Weather Condition</string>

</resources>

**5. Prezentarea executiei aplicatiei**

Screenshot1: la deschiderea aplicatiei.

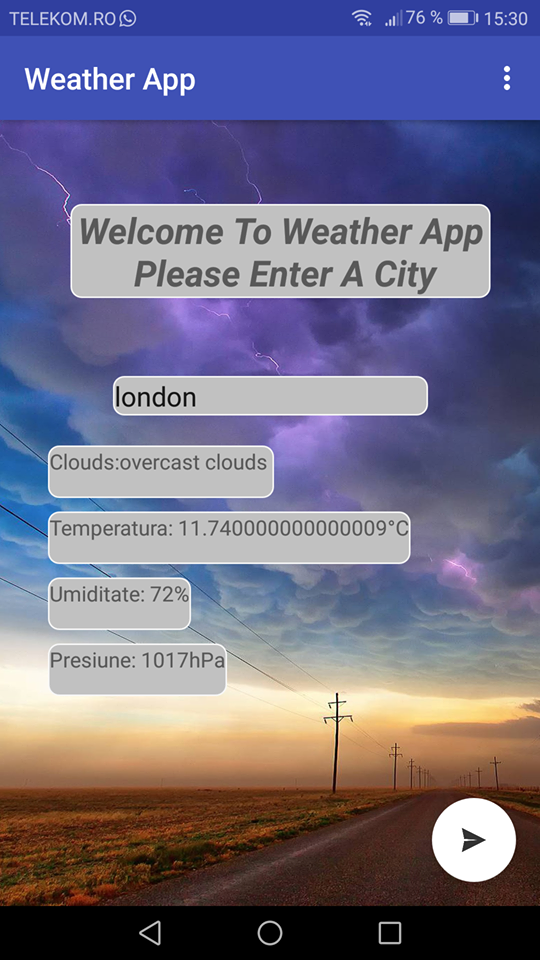
Se pot remarca: fundalul, mesajul de intampinare, precum si spatiul unde se cere intoducerea numelui orasului.

Dupa completarea campului orasului, aplicatia va furniza datele meteo referitoare la umiditate, temperatura, precum si starea cerului.



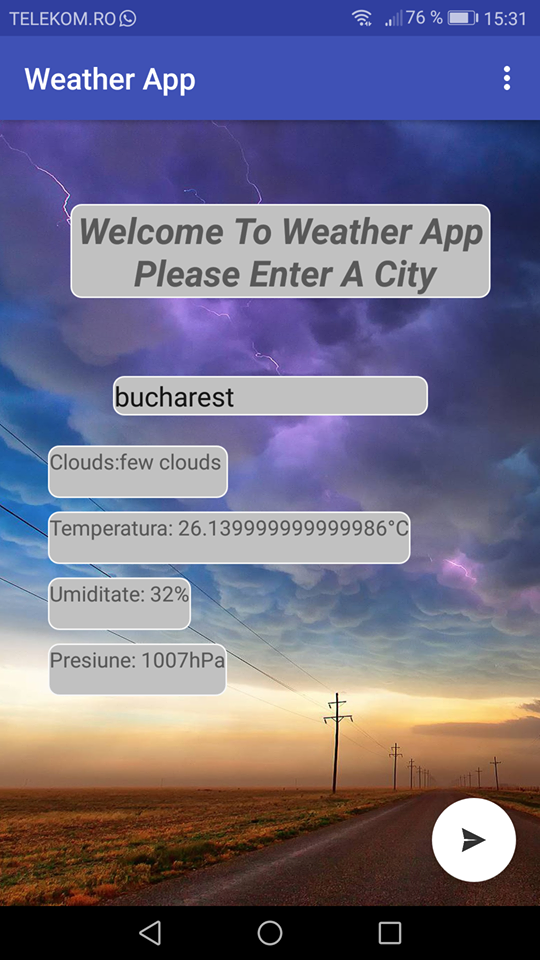
Screenshot2: dupa ce se completeaza campul orasului cu “london”

se remarca aparitia mesajului “mist:mist,haze:haze” in dreptul conditiilor meteo. De asemenea apare si valoarea in grade Celsius 11.74 a temperaturii, umiditatea 72%, precum si presiunea 1017 in hPa (1hPa=100Pa(pascal)).



Screenshot3: dupa ce se completeaza campul orasului cu “bucharest”

se remarca aparitia mesajului “clear:clear sky” in dreptul condtiilor meteo. De asemenea apare si valoarea in grade Celsius 26.13 a temperaturii, umiditatea 32%, precum si presiunea 1007 in hPa (1hPa=100Pa(pascal)).



Screenshot4: dupa ce se completeaza campul orasului cu “paris”

Se remarca aparitia mesajului “clear:clear sky” in dreptul condtiilor meteo. De asemenea apare si valoarea in grade Celsius 15.93 a temperaturii, umiditatea 88%, precum si presiunea 1012 in hPa (1hPa=100Pa(pascal)).

