Université de Paris

Weather APP

Programmation Avancée

M1 MIAGE FIC/FA
2022-2023

Réaliser par:
BETTAHAR Mohamed El Amine
MISTRY Manishkumar
SLIMENE Randa

Encadré par:
POIROT Gilles

Sommaire

01 Présentation – Technologie utilisée

02 Diagramme UML

03 Les points d'attentions

04 Démonstration

01 Présentation

« WeatherAPP » une application de “météo” orientée événements sur Android codé en Java

Donne la situation de la météo :

- Température *(en °C)*
- Température ressentie *(en °C)*
- Humidité *(en %)*
- Description de la météorologiques
- Vitesse de vent *(en m/s)*
- Nuages - Pression *(en hPa)*

Technologies

Diagramme UML V1_En_Classe

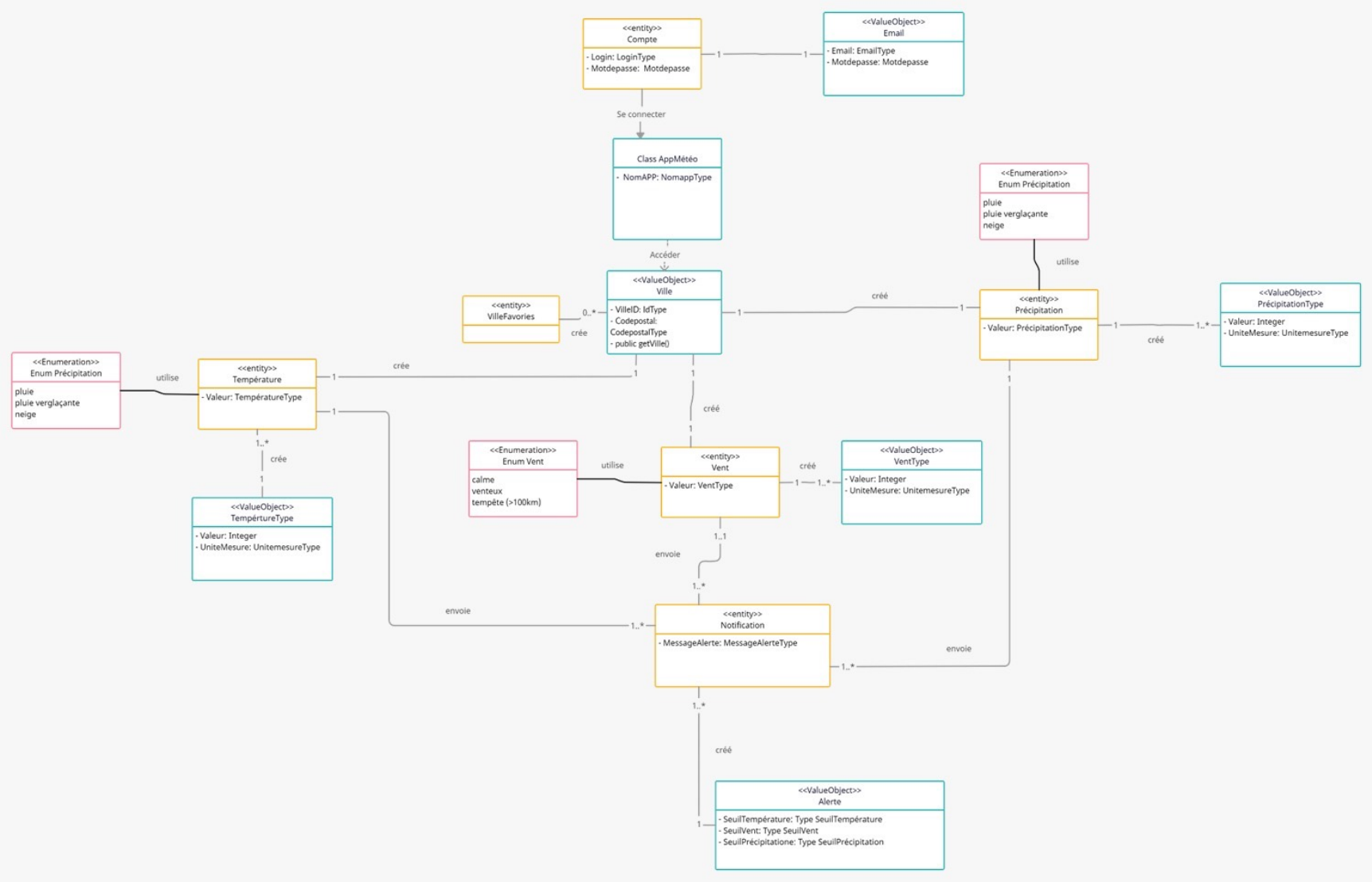
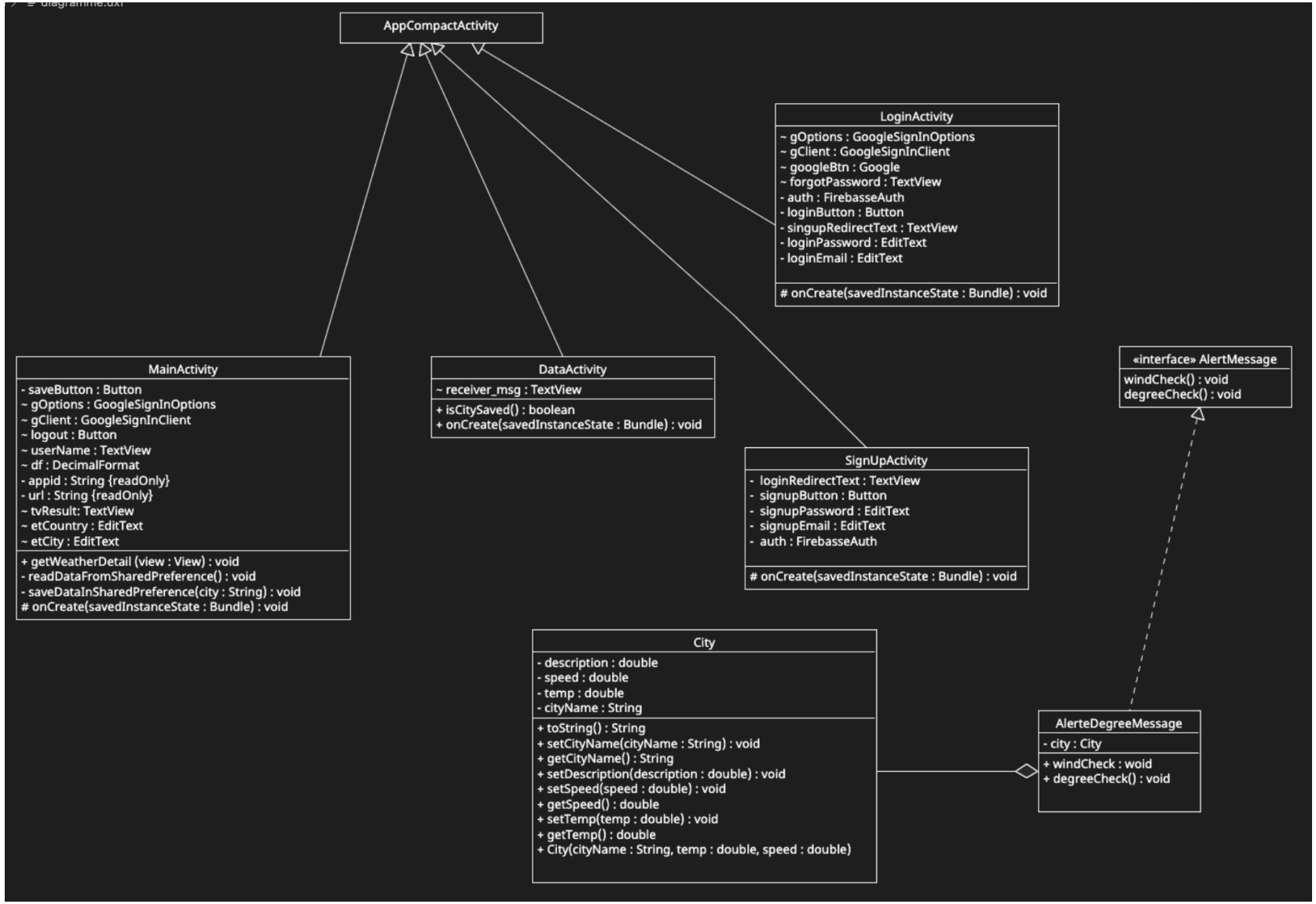


Diagramme UML_VFinal



02 Les points d’attentions

Data Type

```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/etCity"
    android:digits="abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
    android:layout_marginBottom="10dp"
    android:hint="Enter City Name"
    android:inputType="textPersonName"/>

<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/etCountry"
    android:layout_marginBottom="10dp"
    android:ems="10"
    android:hint="Enter Country code (Op)"
    android:inputType="textPersonName" />
```

Appels Asynchrones

```java
public void getWeatherDetails(View view) {
    String tempUrl = "";
    String city = etCity.getText().toString().trim();
    String country = etCountry.getText().toString().trim();
    if(city.equals("")){
        tvResult.setText("City field can not be empty!");
    }else{
        if(!country.equals("")){
            tempUrl = url + "?q=" + city + "," + country + "&appid=" + appid;
        }else{
            tempUrl = url + "?q=" + city + "&appid=" + appid;
        }
        StringRequest stringRequest = new StringRequest(Request.Method.POST, tempUrl, new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                String output = "";
                try {
                    JSONObject jsonResponse = new JSONObject(response);
                    JSONArray jsonArray = jsonResponse.getJSONArray( name: "weather");
                    JSONObject jsonObjectWeather = jsonArray.getJSONObject( index: 0);
                    String description = jsonObjectWeather.getString( name: "description");
                    JSONObject jsonObjectMain = jsonResponse.getJSONObject("main");
                    double temp = jsonObjectMain.getDouble( name: "temp") - 273.15;
                    double feelsLike = jsonObjectMain.getDouble( name: "feels_like") - 273.15;
                    float pressure = jsonObjectMain.getInt( name: "pressure");
                    int humidity = jsonObjectMain.getInt( name: "humidity");
                    JSONObject jsonObjectWind = jsonResponse.getJSONObject("wind");
                    String wind = jsonObjectWind.getString( name: "speed");
                    JSONObject jsonObjectClouds = jsonResponse.getJSONObject("clouds");
                    String clouds = jsonObjectClouds.getString( name: "all");
                    JSONObject jsonObjectSys = jsonResponse.getJSONObject("sys");
                    String countryName = jsonObjectSys.getString( name: "country");
                    String cityName = jsonResponse.getString( name: "name");
                    tvResult.setTextColor(Color.rgb( red: 68, green: 134, blue: 199));
                    output += "Current weather of " + cityName + " (" + countryName + ")"
                            + "\n Temp: " + df.format(temp) + " °C"
                            + "\n Feels Like: " + df.format(feelsLike) + " °C"
                            + "\n Humidity: " + humidity + "%"
                            + "\n Description: " + description
                            + "\n Wind Speed: " + wind + "m/s (meters per second)"
                            + "\n Cloudiness: " + clouds + "%"
                            + "\n Pressure: " + pressure + " hPa";
                    tvResult.setText(output);} catch (JSONException e) {e.printStackTrace(); }}}, new Response.ErrorListener(){
            @Override
            public void onErrorResponse(VolleyError error) { Toast.makeText(getApplicationContext(), error.toString().trim(), Toast.LENGTH_SHORT).show();
            }});RequestQueue requestQueue = Volley.newRequestQueue(getApplicationContext()); requestQueue.add(stringRequest);
```

```java
public class MainActivity extends AppCompatActivity {
    EditText etCity, etCountry;
    TextView tvResult;
    private final String url = "https://api.openweathermap.org/data/2.5/weather";
    private final String appid = "e53301e27efa0b66d05045d91b2742d3";
```

Lire et enregistrer avec SharedPreference

```
private void saveDataInSharedPreferences (String city) throws JSONException {

    SharedPreferences sharedPreferences = getSharedPreferences( name: "MySharePref", MODE_PRIVATE);
    SharedPreferences.Editor myEdit = sharedPreferences.edit();

    myEdit.putString("city", city);
    myEdit.commit();
    Toast.makeText(getApplicationContext(), "saved", Toast.LENGTH_SHORT).show();
    tvResult.setText(city);
}

private void readDataFromSharePreference() {
    SharedPreferences sh = getSharedPreferences( name: "MySharePref", MODE_PRIVATE);
    String city = sh.getString( key: "CITY",  defValue: "");
    if(city.isEmpty()){
        saveButton.setText("Save Data");
    }
    else {
        saveButton.setText("Update data ");
    }
}
```

Transfert des données dans Data Activity

```java
// Sauvegarder le nom de la ville
saveButton.setOnClickListener(v -> {
    String s = tvResult.getText().toString();
    Intent intent = new Intent(getApplicationContext(), DataActivity.class);
    intent.putExtra( name: "message_key", s);
    startActivity(intent);
    try {
        saveDataInSharedPreferences(tvResult.getText().toString().trim());
    } catch (JSONException e) {
        throw new RuntimeException(e);
    }
});
```

```java
public class DataActivity extends AppCompatActivity {

    TextView receiver_msg;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.data_activity);

        receiver_msg = findViewById(R.id.received_value_id);

        Intent intent = getIntent();

        String str = intent.getStringExtra( name: "message_key");

        receiver_msg.setText(str);

    }
```

Interface Alerte

```java
package com.example.wheathera;

public interface AlertMessages {
    void degreeCheck();
    void windCheck();
    //void descriptionCheck();
}
```

```java
public class AlertDegreeMessage implements  AlertMessages{
    private City city;

    public AlertDegreeMessage(City city) { this.city = city; }


    @Override
    public void degreeCheck() {
        final double minTemp = -5; //par défault on a mis des valeurs
        final double maxTemp = 45;
        try{
            if(city.getTemp() < minTemp || city.getTemp() > maxTemp) throw new Exception("");

        }catch (Exception e){
            e.printStackTrace();
        }

    }

    @Override
    public void windCheck() {
        try{
            if(city.getSpeed() < 100) throw new Exception("Rester vigilant");
        }catch (Exception e){
            e.printStackTrace();
        }

    }


    /*@Override
    public void descriptionCheck() {
        try{
            if (city.getCityName()) throw new Exception("Alerte Sounami"));
            else{
                throw new Exception("Alerte Volcan");
            }

        } catch (Exception e) {
            e.printStackTrace();
        }

    }*/
```

Création de compte SignUpActivity

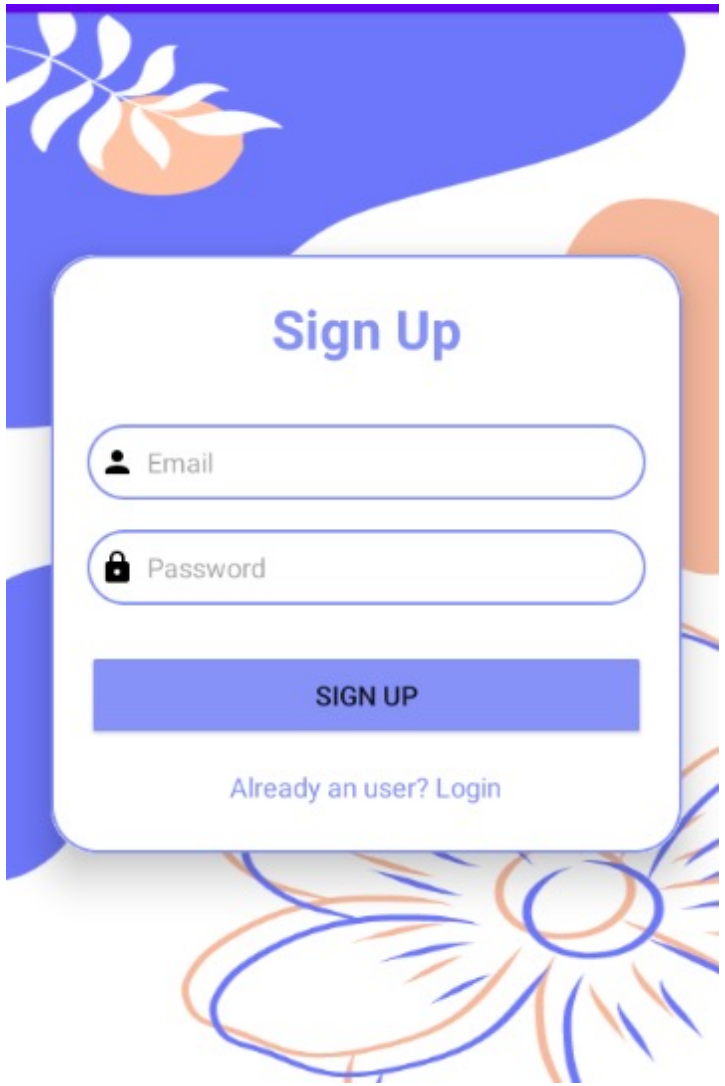
```java
public class SignUpActivity extends AppCompatActivity {

    private FirebaseAuth auth;
    private EditText signupEmail, signupPassword;
    private Button signupButton;
    private TextView loginRedirectText;

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

        auth = FirebaseAuth.getInstance();
        signupEmail = findViewById(R.id.signup_email);
        signupPassword = findViewById(R.id.signup_password);
        signupButton = findViewById(R.id.signup_button);
        loginRedirectText = findViewById(R.id.loginRedirectText);

        signupButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String user = signupEmail.getText().toString().trim();
                String pass = signupPassword.getText().toString().trim();

                if (user.isEmpty()){
                    signupEmail.setError("Email cannot be empty");
                }
                if (pass.isEmpty()){
                    signupPassword.setError("Password cannot be empty");
                } else{
                    auth.createUserWithEmailAndPassword(user, pass).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete(@NonNull Task<AuthResult> task) {
                            if (task.isSuccessful()) {
                                Toast.makeText(SignUpActivity.this, "SignUp Successful", Toast.LENGTH_SHORT).show();
                                startActivity(new Intent( packageContext: SignUpActivity.this, LoginActivity.class));
                            } else {
                                Toast.makeText(SignUpActivity.this, "SignUp Failed" + task.getException().getMessage(), Toast.LENGTH_SHORT).show();
                            }
                        }
                    });
                }
            }
```

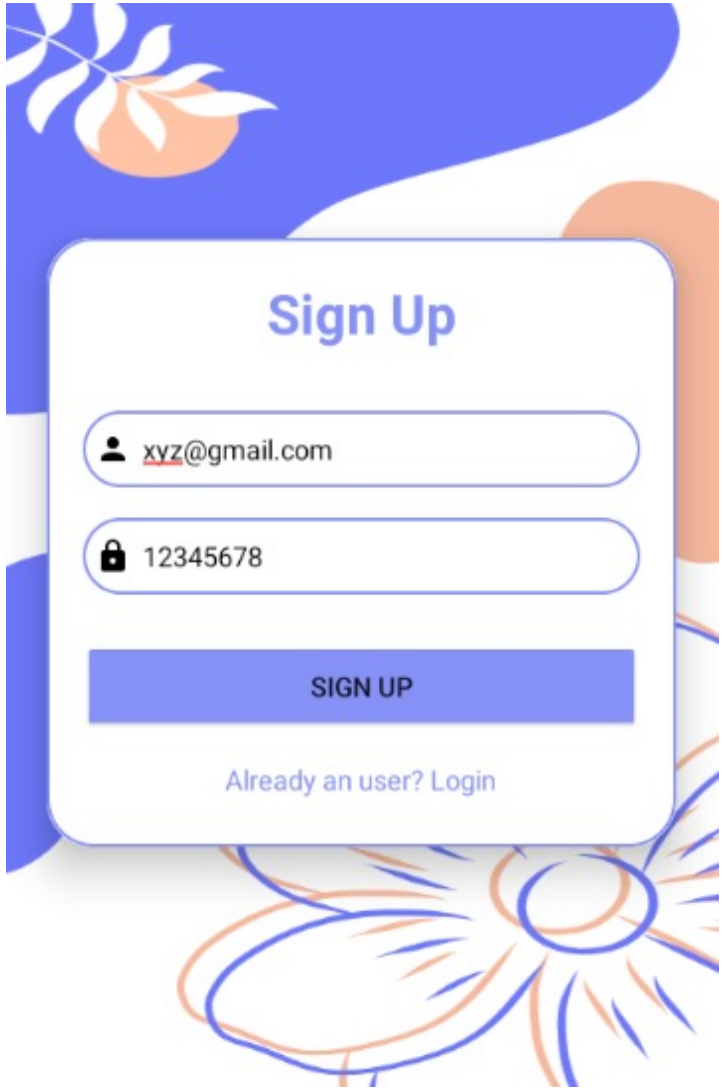
Login Activity

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

    loginEmail = findViewById(R.id.login_email);
    loginPassword = findViewById(R.id.login_password);
    loginButton = findViewById(R.id.login_button);
    signupRedirectText = findViewById(R.id.signUpRedirectText);
    forgotPassword = findViewById(R.id.forgot_password);
    googleBtn = findViewById(R.id.googleBtn);

    auth = FirebaseAuth.getInstance();

    loginButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {

            String email = loginEmail.getText().toString();
            String pass = loginPassword.getText().toString();

            if (!email.isEmpty() && Patterns.EMAIL_ADDRESS.matcher(email).matches()) {
                if (!pass.isEmpty()) {
                    auth.signInWithEmailAndPassword(email, pass)
                            .addOnSuccessListener(new OnSuccessListener<AuthResult>() {
                                @Override
                                public void onSuccess(AuthResult authResult) {
                                    Toast.makeText(LoginActivity.this, "Login Successful", Toast.LENGTH_SHORT).show();
                                    startActivity(new Intent( packageContext: LoginActivity.this, MainActivity.class));
                                    finish();
                                }
                            }).addOnFailureListener(new OnFailureListener() {
                                @Override
                                public void onFailure(@NonNull Exception e) {
                                    Toast.makeText(LoginActivity.this, "Login Failed", Toast.LENGTH_SHORT).show();
                                }
                            });
                } else {
                    loginPassword.setError("Empty fields are not allowed");
                }
            } else if (email.isEmpty()) {
                loginEmail.setError("Empty fields are not allowed");
            } else {
                loginEmail.setError("Please enter correct email");
            }
        }
```

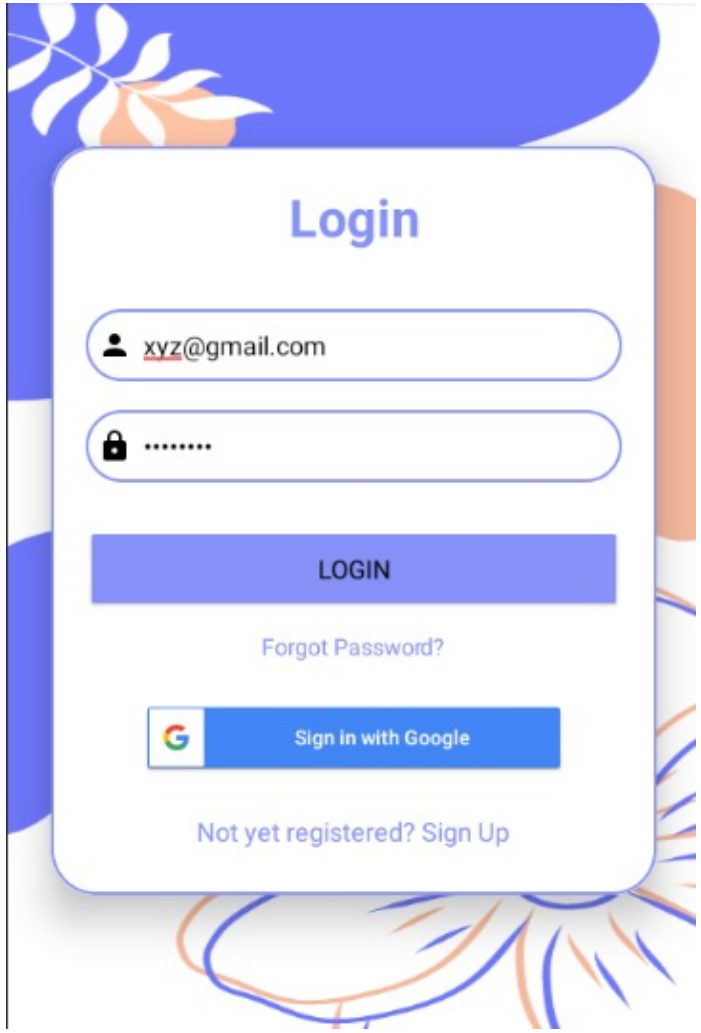
Démonstration

Créer un compte

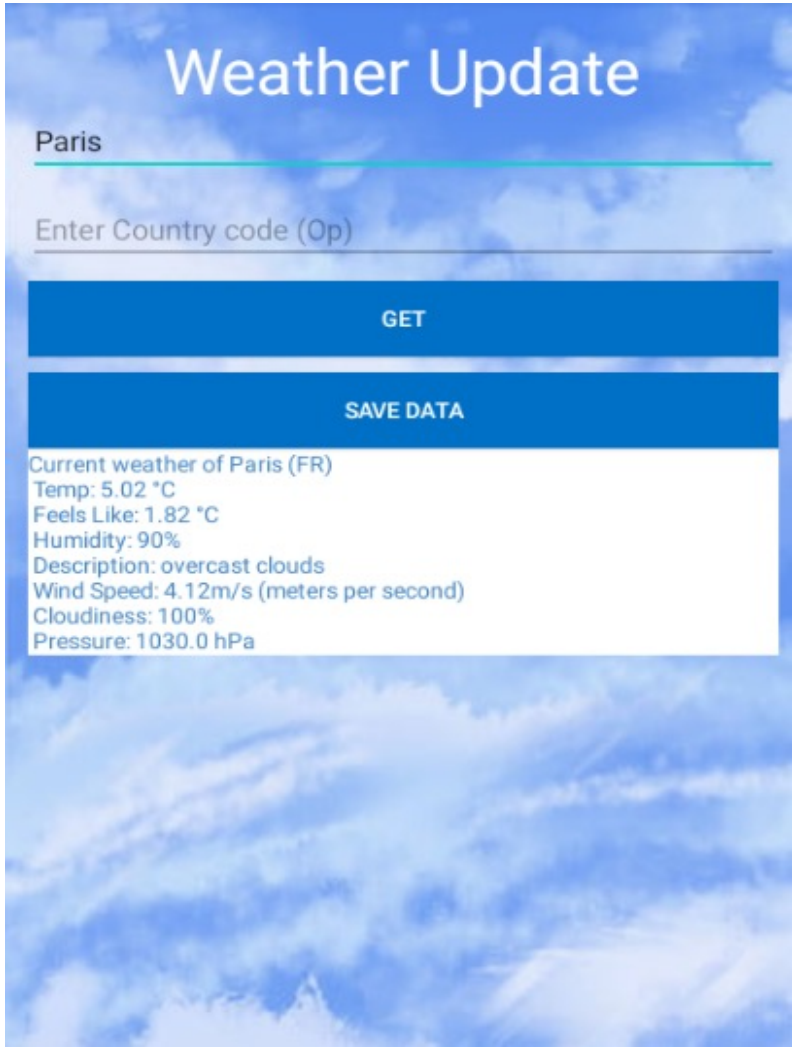


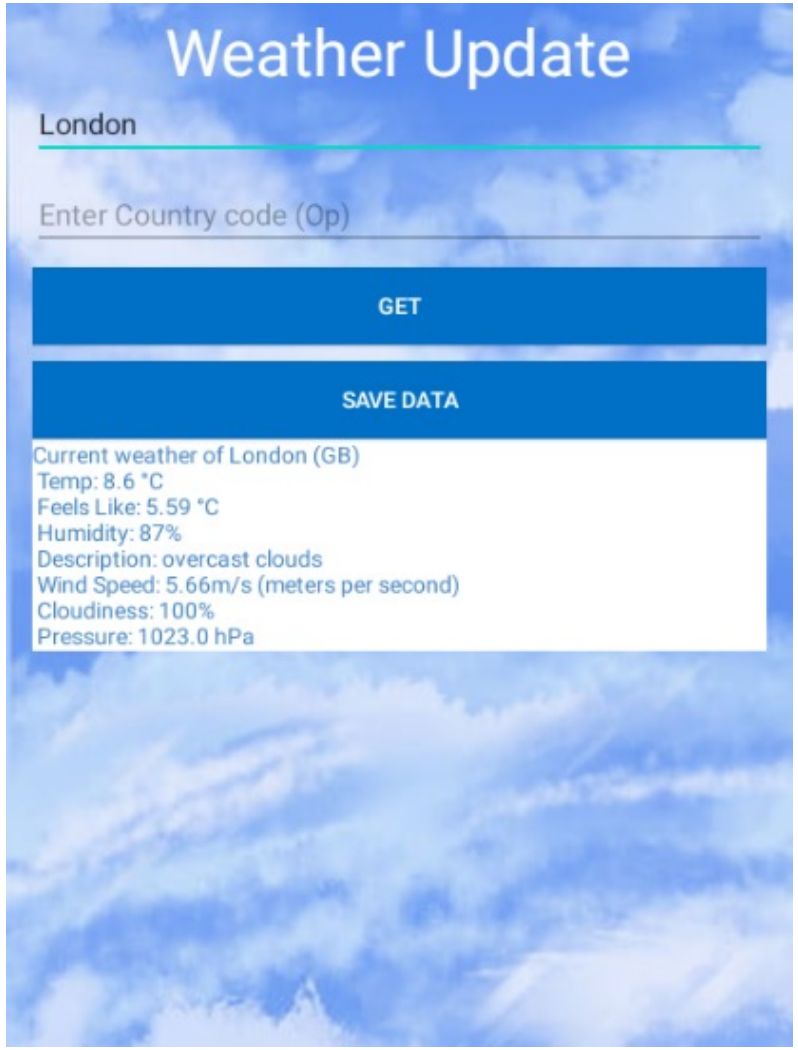


Login


Login
xyz@gmail.com
LOGIN
Forgot Password?
Sign in with Google
Not yet registered? Sign Up

Affichage des détails météo Paris et London





Sauvegarde des données

Conclusion

Merci