



# SOMMAIRE

Introduction .....	2
Méthodologie .....	3
1) Création de l'écran de démarrage (Splash Screen) .....	3
2) Création du projet .....	4
3) Structure du projet .....	4
4) Mise en place de SQLITE Room .....	5
5) Implémentation du RecyclerView .....	7
6) Gestion des fonctionnalités de l'application .....	7
7) Tests et débogage.....	8
8) Documentation et livrables .....	8
Conclusion .....	9

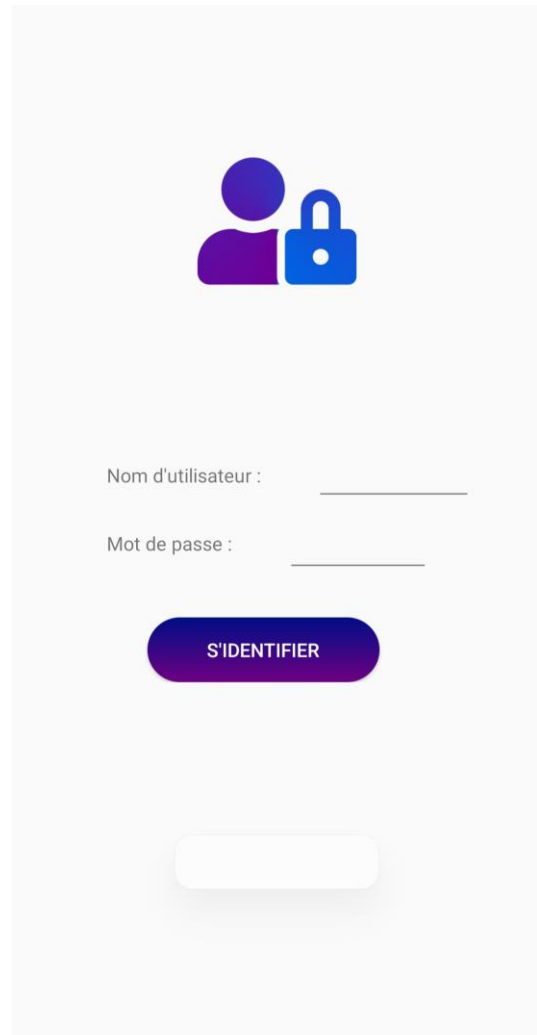


Dans cette étude, nous aborderons le développement d'une application de stockage et d'affichage des données relatives à un pays en utilisant SQLite Room et RecyclerView. L'objectif principal de ce projet est de créer une application fonctionnelle qui permettra à l'utilisateur de gérer et d'afficher les informations essentielles d'un pays, telles que le nom du pays, le nombre d'habitants et le nom de la capitale.

Pour réaliser cela, nous suivrons les consignes fournies dans le cadre de cet examen.

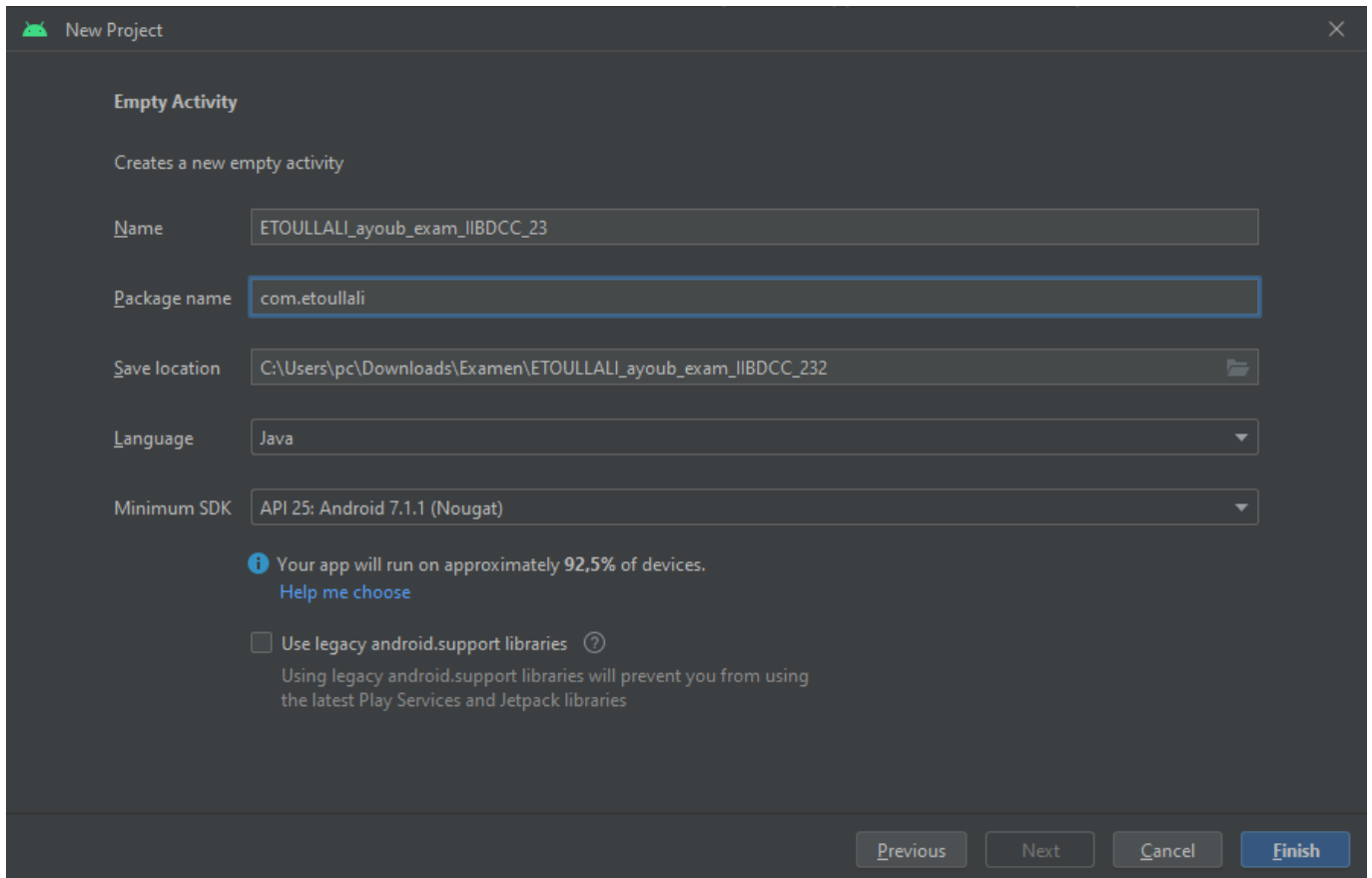
### 1) Création de l'écran de démarrage (Splash Screen)

Nous commencerons par mettre en place l'écran de démarrage de l'application, qui s'affiche brièvement avant d'accéder à l'interface principale.



## 2) Création du projet

Nous suivrons les consignes et créerons le projet en utilisant un nom approprié, conformément aux directives données.



New Project

Empty Activity

Creates a new empty activity

Name: ETOULLALI\_ayoub\_exam\_IIBDCC\_23

Package name: com.etoullali

Save location: C:\Users\pc\Downloads\Examen\ETOULLALI\_ayoub\_exam\_IIBDCC\_232

Language: Java

Minimum SDK: API 25: Android 7.1.1 (Nougat)

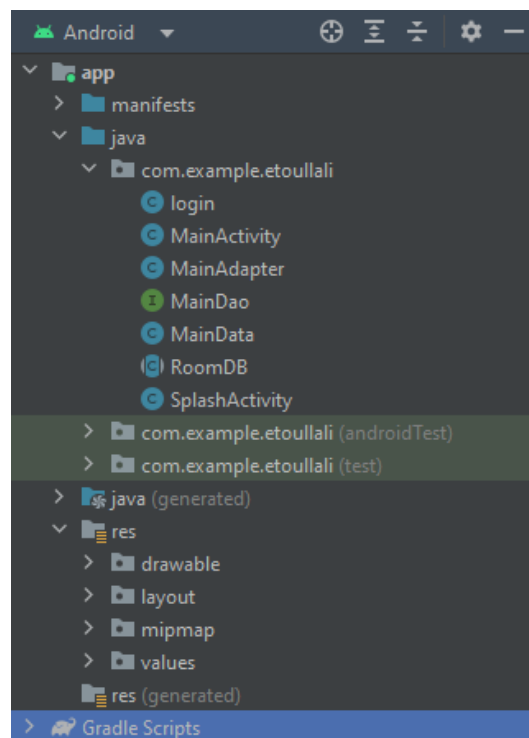
☒ Your app will run on approximately 92,5% of devices.  
[Help me choose](#)

☐ Use legacy android.support libraries ?  
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Previous Next Cancel Finish

## 3) Structure du projet

Nous veillerons à ce que tous les packages de l'application contiennent votre nom et prénom, comme spécifié dans les consignes.



## 4) Mise en place de SQLite Room

Nous configurerons la bibliothèque SQLite Room, qui facilitera le stockage et la gestion des données relatives aux pays.

### RoomDB

```
package com.example.etoullali;

import android.content.Context;

import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;

//Add database entities
@Database(entities={MainData.class},version=1,exportSchema = false)

public abstract class RoomDB extends RoomDatabase {

    // create database instance
    private static RoomDB database;

    // Define database name
    private static String DATABASE_NAME="database";

    public synchronized static RoomDB getInstance(Context context)
    {
        // check condition
        if(database==null)
        {
            // when database is null
            // Initialize database
            database=
Room.databaseBuilder(context.getApplicationContext(),RoomDB.class,DATABASE_NAME)
                .allowMainThreadQueries().fallbackToDestructiveMigration().build();
        }
        // Return database
        return database;
    }

    //Create DAO
    public abstract MainDao mainDao();
}
```

### MainData

```
package com.example.etoullali;

import androidx.room.ColumnInfo;
import androidx.room.Entity;
import androidx.room.PrimaryKey;

import java.io.Serializable;

@Entity(tableName ="table_name")

public class MainData implements Serializable {

    //Create id column
    @PrimaryKey(autoGenerate = true)
    private int ID;

    // Create text column
    @ColumnInfo(name="text")
    private String text;
}
```

```

//Generate getters and setters

public int getID() {
    return ID;
}

public String getText() {
    return text;
}

public void setID(int ID) {
    this.ID = ID;
}

public void setText(String text) {
    this.text = text;
}
}

```

## MainDao

```

package com.example.etoullali;
import androidx.room.Dao;
import androidx.room.Insert;
import androidx.room.Query;

import java.util.List;

import static androidx.room.OnConflictStrategy.REPLACE;

@Dao
public interface MainDao {

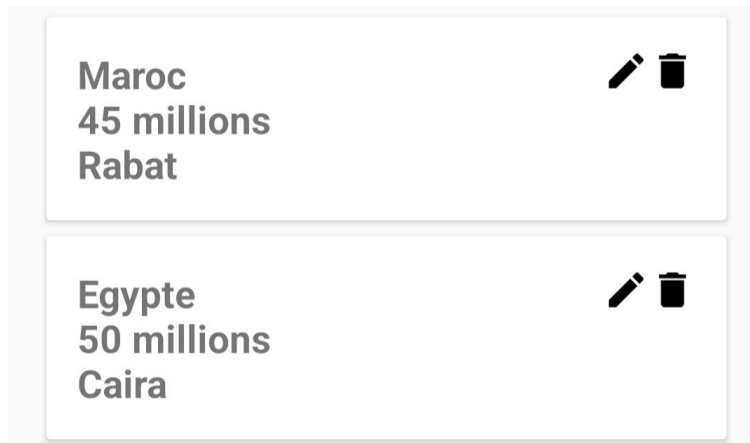
    // Insert query
    @Insert(onConflict = REPLACE)
    void insert(MainData mainData);
    // Delete query
    /*
    @Delete
    void delete(List<MainData> mainData);
    //to do
    */
    // delete Query
    @Query("Delete from table_name where ID=:sID")
    void delete(int sID);
    // Update query
    @Query("UPDATE table_name SET text= :sText where ID=:sID")
    void update(int sID, String sText);

    // Get all data query
    @Query("SELECT * FROM table_name")
    List<MainData> getAll();
}

```

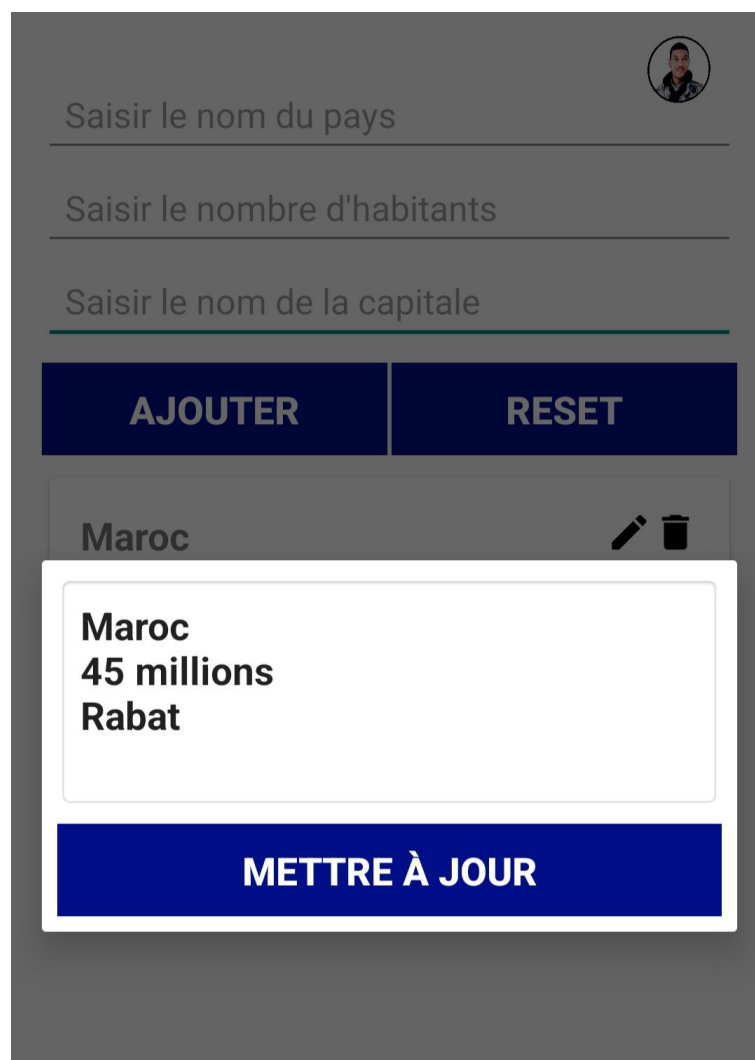
## 5) Implémentation du RecyclerView

Nous utiliserons le composant RecyclerView pour afficher les données des pays de manière efficace et interactive.



## 6) Gestion des fonctionnalités de l'application

Nous ajouterons les fonctionnalités nécessaires pour permettre à l'utilisateur d'ajouter, modifier et supprimer des pays, ainsi que d'afficher les détails de chaque pays.





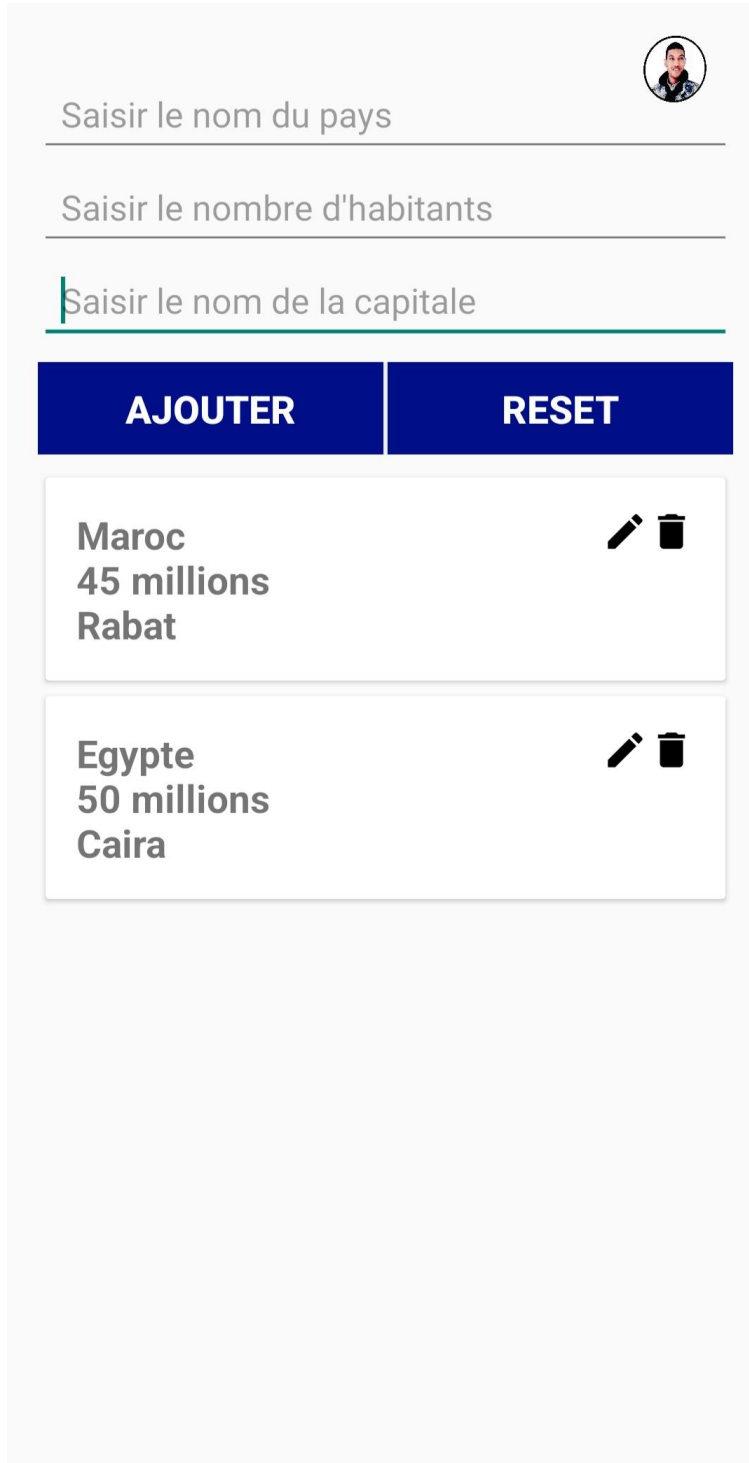
## 7) Tests et débogage

Nous effectuerons des tests approfondis pour vérifier le bon fonctionnement de l'application et corriger les erreurs éventuelles.

## 8) Documentation et livrables

Nous préparerons une courte vidéo démo de l'application en action, ainsi que des captures d'écran pour accompagner le code source.

**Lien de démo :** [XRecorder Edited 26052023 121256.mp4 - Google Drive](#)



The screenshot shows a web application interface with a light gray background. At the top right, there is a circular profile picture of a man. Below it, there are three input fields with placeholder text: "Saisir le nom du pays", "Saisir le nombre d'habitants", and "Saisir le nom de la capitale". Below the input fields are two dark blue buttons with white text: "AJOUTER" and "RESET". Below the buttons are two white cards with rounded corners. The first card contains the text "Maroc", "45 millions", and "Rabat". The second card contains the text "Egypte", "50 millions", and "Caira". To the right of each card are two icons: a pencil and a trash can.

Country	Population	Capital	Actions
Maroc	45 millions	Rabat	Edit, Delete
Egypte	50 millions	Caira	Edit, Delete

En conclusion, ce projet nous a permis de développer une application de stockage et d'affichage des données relatives à un pays en utilisant SQLite Room et RecyclerView. Nous avons réussi à respecter les consignes spécifiées dans le cadre de cet examen, en créant une application fonctionnelle et conviviale. Grâce à l'utilisation de SQLite Room, nous avons pu gérer facilement le stockage des données, tandis que RecyclerView nous a permis d'afficher les informations des pays de manière efficace. Ce projet nous a également permis d'approfondir nos connaissances en développement mobile et de mettre en pratique les concepts clés de cette discipline.