**Atelier :développement mobile**

TP1

**Étudiantes: Farhaoui Eya , Guesmi Ikram**

**Enseignant: Maddouri Faouzi**

Année universitaire:2024/2025

Table des matières

**[Les etapes de la création d’un avd: 2](#_Toc24485)**

**[Hiérarchie du projet: 3](#_Toc21081)**

**[L’activité de l’application réalisée: 5](#_Toc16906)**

# Les étapes de la création d’un avd:

Dans Android Studio :

**Accéder au gestionnaire d’AVD** :

* Cliquez sur l'onglet "Tools" dans la barre de menu.
* Sélectionnez "AVD Manager" (ou "Android Virtual Device Manager").

**Créer un nouvel AVD** :

· Dans l’AVD Manager, cliquez sur le bouton "Create Virtual Device".

· cliquez sur le bouton "New harware profile".

· **Configurer les paramètres AVD** :

· Donnez un nom à votre AVD.

* Vous pouvez personnaliser les options comme la taille de la mémoire, l'orientation (portrait ou paysage), etc.

· **Ajouter un skin personnalisé** :

Aprés avoir télecharger le skin de votre appareil ;

* Recherchez la section **"Skin"**. Vous pouvez choisir un skin prédéfini ou personnaliser le skin en utilisant un fichier image. Pour un skin personnalisé, vous devez fournir les fichiers de skin sous forme de dossier contenant layout, drawable, etc.
* Assurez-vous que votre skin est compatible avec les dimensions de votre appareil virtuel.

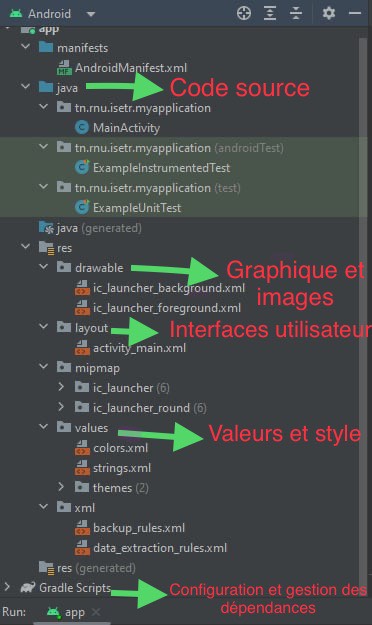
· **Finaliser la création de l’AVD** :

* Une fois que vous avez configuré les paramètres et ajouté le skin, cliquez sur **"Finish"** pour créer l’AVD.

· **Démarrer l'AVD** :

* L’AVD que vous venez de créer apparaîtra dans la liste de l’AVD Manager.
* Cliquez sur le bouton **"Play"** (icône de lecture) pour lancer l'AVD.

# Hiérarchie du projet:



### ****Dossier** java**

* **Rôle** : Contient le code source de l'application.
* **Sous-dossiers** :
  + **Package** : Les fichiers Java sont organisés en packages, qui représentent des espaces de noms pour les classes. Chaque activité, service, ou classe de modèle sera généralement dans son propre fichier .java.

### ****Dossier** res**

* **Rôle** : Contient toutes les ressources non-code (images, layouts, chaînes de caractères, etc.) utilisées par l'application.
* **Sous-dossiers** :
  + drawable : Contient des images et des graphiques (JPEG, PNG, XML pour les formes, etc.) utilisés dans l'interface utilisateur.
  + layout : Contient des fichiers XML qui définissent la structure et l'apparence des interfaces utilisateur. Chaque fichier représente un écran ou une partie de l'interface.
  + values : Contient des fichiers XML pour les valeurs simples telles que les chaînes de caractères (strings.xml), les dimensions (dimens.xml), et les styles (styles.xml). Cela permet de centraliser et de gérer facilement les ressources utilisées dans l'application.

**Dossier** gradle

* **Rôle** : Contient les fichiers et configurations liés à Gradle, l'outil de construction utilisé par Android pour gérer les dépendances et les builds.
* **Fichiers principaux** :
  + build.gradle **(niveau projet)** : Déclare les dépendances du projet, les configurations communes, et les plugins utilisés par l'application.
  + build.gradle **(niveau module)** : Contient des configurations spécifiques au module de l'application, comme les dépendances spécifiques, les configurations de compilation, et les configurations d'application (comme le nom de l'application, la version, etc.).

# L’activité de l’application réalisée:

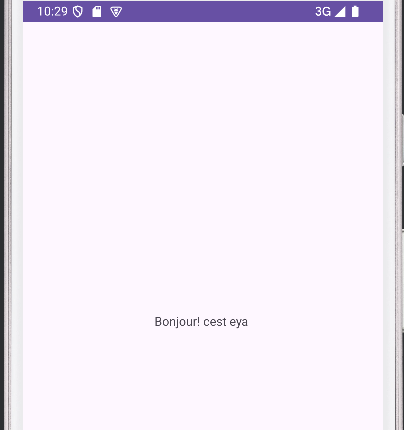


Figure 1: activité de l'application