

---

# SMB116-Projet d'application Android

Date de rendu : 30 Avril 2022

## VUE D'ENSEMBLE

Conception et développement d'une application Android utilisant une API (The Movie Database). L'application permettra à un utilisateur de s'inscrire et de se connecter à l'application pour accéder à des listes de films et à leurs détails. Un utilisateur pourra rechercher des films, les ajouter ou les supprimer de sa liste de visionnage. Il aura également accès à un écran pour personnaliser son profil.

## OBJECTIFS

1. Le premier objectif est de restituer les différentes notions de la conception et du développement d'une application Android.
2. Le second objectif est de sensibiliser à la production d'une application "professionnelle" que ça soit : par l'architecture logiciel établie ; la qualité du code ; la gestion du code source.

## CARACTÉRISTIQUES

L'application doit disposer des fonctionnalités suivantes :

- **Inscription** : L'utilisateur doit pouvoir renseigner un email, un mot de passe ainsi que sa date de naissance (DatePickerDialog). Pour des soucis de gain de temps et de faciliter, l'inscription sera traitée localement et ne nécessitera pas de serveur. Une fois inscrit, l'utilisateur est redirigé vers la page de connexion avec les informations pré-remplies.
- **Connexion** : L'utilisateur doit pouvoir se connecter à l'application en saisissant son email et son mot de passe. Entre deux sessions, ses informations doivent rester enregistrées pour faciliter la reconnexion.
- **Navigation** : Un ou des composants de navigation permettent à l'utilisateur de naviguer dans les différents écrans de l'application.
- **Consultation de la liste des tendances** : Une fois connecté, l'écran principal de l'application doit afficher dans une liste scrollable, les tendances actuelles des films.

---

Chaque entrée de la liste affiche l'image, le titre et la date de sortie du film.

- **Détail d'un film** : Un clic sur un film dirige l'utilisateur vers une vue de détails (Master/Detail flow). Libre à vous de choisir les informations à afficher ici. Cette page proposera également deux actions :
  - Partager un film au format textuel (titre et description)
  - Ajouter un film à sa liste de visionnage
- **Liste de visionnage** : L'utilisateur peut ajouter et retirer des films de sa liste de visionnage, par souci de simplicité et de temps, la liste sera gérée localement mais sauvegardée entre deux sessions.
- **Recherche** : L'application doit permettre la recherche d'un film par son titre, les résultats de recherche seront affichés de la même manière que la liste des tendances. Un clic sur film permet d'accéder à ses détails.
- **Profil** : Un écran permettra à l'utilisateur de mettre à jour son mot de passe et de sélectionner ou de prendre une photo comme image de profil. Cette photo est utilisée uniquement dans le contexte de cet écran et ne sera pas sauvegardée.

## Critères de notations

Seront évaluées uniquement les consignes données, qui sont basées sur les TDs / TP's effectués dans le cadre des cours. Inutile donc de faire plus que demandé.

Une note de groupe sur 20 sera attribuée en fonction de la qualité du travail rendu. Cette note tiendra compte des aspects suivants par ordre d'importance :

- L'application répond aux besoins énoncés ci-dessus.
- L'utilisation des composants graphiques adéquats.
- L'architecture et la qualité du code.
- La propreté de votre gestion de sources.

## Ressources

L'application utilisera l'API public de The Movie Database (TMDB), il sera nécessaire de créer un compte pour obtenir une clé API. Attention, il est préférable de ne pas mettre cette clé API sur votre système de gestion de source. Vous trouverez ci-dessous les liens utiles :

- 
- [Inscription TMDB](#)
  - [Enregistrement clé API](#)
  - [Documentation développeur TMDB](#)

Je vous conseille de vous pencher sur des librairies existantes pour gérer certains aspects de votre application, inutile de réinventer la roue. Libre à vous cependant d'utiliser celles de votre choix.

Requêtes API :

- <https://square.github.io/retrofit/>
- <https://square.github.io/okhttp/>

Affichage des images depuis une URL public :

- <https://github.com/bumptech/glide>
- <https://square.github.io/picasso/>
- <https://coil-kt.github.io/coil/>

Stockage des données :

- <https://developer.android.com/training/data-storage/shared-preferences>
- <https://developer.android.com/training/data-storage/room>