

# Android Reactive Programming project

Vous devez construire en équipe une application mobile Android avec le thème de votre choix. J'attends une démonstration de votre maîtrise de l'ensemble des concepts vus en cours.

La note maximale pour le socle commun sera de 16 points. Pour obtenir jusqu'à 4 points supplémentaires, vous devrez implémenter des fonctionnalités plus avancées et/ou plus complexes que celles présentées en cours.

Les critères d'évaluation ont un impact plus ou moins élevé en fonction de s'il est rempli ou non, pensez donc à au moins respecter tous les critères pour vous assurer une note maximale.

## Contraintes:

1. Un groupe de 3 personnes.
2. **Les conventions de nommage:** Suivez les conventions de nommage vues en cours pour garantir un code clair et cohérent, et faciliter la maintenance.
3. **La qualité des interfaces graphiques:** Les interfaces doivent être soignées pour offrir une expérience utilisateur agréable et intuitive (principes d'interface Android vus en cours).
4. **La deadline:** Le premier jour de retard entraîne 1 point en moins sur la note et doublera chaque jour de plus:
  - 1 jour → -1 point
  - 2 jours → -2 points
  - 3 jours → -4 points
  - 4 jours → -8 points
  - ...

## **Livrables attendus:**

1. L'URL de dépôt Git.
2. La contribution des 3 étudiants au code.
3. Le code source complet du projet, accompagné de commentaires appropriés.
4. Des captures d'écran ou des maquettes des interfaces utilisateur.
5. La documentation technique expliquant l'architecture et les choix de conception (1 ou 2 pages suffisent).

## Grille de notation:

1. La qualité de l'arbre Git, des branches, des commits et du workflow Git. (0/20 pour l'élève qui n'a aucun commit). *1 point*
2. La gestion du dark mode partout dans l'application. *0,5 point*
3. Une architecture propre MVVM (Modèle-Vue-ViewModèle) avec une séparation claire des responsabilités et une gestion efficace des données et de l'interface utilisateur. *2 points*
4. L'implémentation d'un CRUD sur une ou plusieurs API de votre choix. *1 point*
5. L'implémentation de plusieurs écrans. *0,5 point*
6. Le respect des conventions de nommage. *0,5 point*
7. La qualité et la cohérence du code. *0,5 point*
8. L'utilisation des concepts de Programmation Orientée Objet vus en cours. *1 point*
9. La pertinence des fonctionnalités implémentées. *0,5 point*
10. La qualité des interfaces graphiques. *1 point*
11. La gestion des erreurs et des exceptions. *0,5 point*
12. La gestion de la nullabilité. *0,5 point*
13. La gestion des cas d'usage spéciaux. L'application doit gérer des comportements inattendus de l'utilisateur. *0,5 point*
14. La soutenance. *6 points*
15. Les bonus pour tout développement approfondi pas vu en cours. *4 points*