

# PROJET ANDROID (15%)

Dans le cadre du cours conception et développement d'un projet informatique, 420-6S5-BA.

## MODALITÉS

En équipe de 2 ou 3. Le projet doit être hébergé sur GitHub avec l'utilisation du bon gitignore. Le projet doit également être fait dans Android studio. Vous ne pouvez pas être en équipe avec les mêmes personnes que dans votre projet d'ESP. Vous devrez également avoir une version fonctionnelle de votre application pour la présentation. Cette date vous sera communiquée officiellement ultérieurement, mais sur le calendrier actuel le vendredi 8 mai est la date prescrite. La remise officielle est toutefois après cette date.

Pour chacun des sprints, vous devrez prendre un rendez-vous pour venir présenter l'avancement du projet.

## ÉNONCÉ

### PRÉSENTATION GÉNÉRALE

Vous devez concevoir une application de votre choix qui respecte les restrictions de la section suivante. Toutefois, la conception de votre application doit contenir certains éléments et vous devez avoir suffisamment d'éléments optionnels pour obtenir un total de points prescrit. Attention d'autres éléments sont présent dans la section livrable. Le projet sera corrigé en utilisant une tablette du CÉGEP à moins d'avis contraire de votre part. Vous devez donc tester éventuellement votre projet sur une des tablettes ou prévoir un arrangement.

Vous êtes invité à essayer de faire une application cohérente, mais éventuellement vous serez probablement forcé d'insérer une ou plusieurs fonctionnalités incohérentes ou peu pertinentes dans votre projet dans le but d'accuser les points optionnels nécessaires. Cela est tout à fait acceptable.

---

### ÉLÉMENTS OBLIGATOIRES

- Votre application doit avoir une interface graphique fonctionnelle.
- Votre application doit contenir des contrôles de base tels que les boutons, les boîtes de texte...
- Votre application doit utiliser au minimum une boîte de dialogues.
- Votre application doit utiliser au moins 1 intent pour changer d'activité.
- Votre application doit utiliser des couleurs personnalisées.(À faire dans le sprint1)
- Votre application doit contenir au moins une animation.
- Votre application doit utiliser les listeners comme dans la méthode vue en classe.

---

### ÉLÉMENTS OPTIONNELS

Vous devez avoir au moins 10 points(équipe de 2) ou 12 points (équipe de 3) dans les éléments optionnels. Des éléments peuvent être ajoutés à cette liste. Si vous souhaitez ajouter un élément particulier à votre application, et celui n'est pas dans la liste, vous pouvez venir me consulter pour ajouter votre élément à cette liste et j'évaluerai la valeur en nombre de points de votre élément. Vous n'avez pas à décider immédiatement, mais à la fin du projet 10 points doivent être présents.

- Utilisation de la caméra et des photos. Les photos doivent être sauvegardées dans l'application (4 points)
- Utilisation du GPS (2 points)
- Utilisation de l'accéléromètre (2 points)
- Utilisation de Google Map (3 points)
- Utilisation de la boussole (2 points)
- Utilisation d'une authentification avec Firebase(2 points)<sup>1</sup>
- Utilisation du stockage Firebase(2 points)<sup>1</sup>
- Utilisation d'une base de données Firestore NOSQL avec Firebase(4 points)<sup>1</sup>
- Utilisation d'une base de données avec SQLite (3 points) \*déconseillé\*
- Utilisation de RecyclerView (3 points)
- Création de notification (3 points)
- Utilisation de thread (2point)
- Utilisation d'une API qui n'a pas été vue en classe (2 points)
- Utilisation de son (1 point)
- Gestion du changement d'orientation avec du code (1 point)
- Création d'une « customView » en implémentant la fonction « onDraw » (3 points)
- Intégration de « gesture » personnalisé (3 points)
- Intégration de réalité augmentée avec gesture(6 points)
  - Ne fonctionnera pas sur les tablettes du CÉGEP. Vous devez avoir un téléphone qui supporte ARCore : API Level 27.2.9 et OpenGL ES 3.0
- Éléments de votre crue et préapprouvés par moi (Nombre de points à déterminer).

#### RESTRICTION

- Le projet ne peut pas être un jeu de style platformer. Par contre, un jeu basé sur une interface (exemple : Trivia crack) est acceptable.
- Le projet doit être réalisé dans Android Studio en utilisant Java. Si vous le souhaitez vraiment nous pouvons prendre entente pour Kotlin. Toutefois, le support enseignant sera grandement affecté par le choix de cette technologie.
- Le projet doit être réalisé en utilisant GitHub.
- Le projet doit être destinée pour des téléphones ou tablette. À moins d'entente préalable.
  - La version d'Android minimal autorisé est : 5.1 Lollipop API LEVEL 21
  - La version d'Android maximal autorisé est : 7.1 Nougat API LEVEL 25

#### DÉROULEMENT

##### CHOIX DU PROJET, 21 MARS (15 POINTS)

Vous pouvez choisir un projet de votre choix. Ce projet ne peut toutefois pas être en lien avec votre ESP.

#### LIVRABLES

- Un document Word d'un quart à une demi-page qui énonce le but de votre application

<sup>1</sup> Il n'est pas possible de cumuler plus de 6 points avec Firebase.

- Un backlog avec les stories que vous souhaitez faire pour votre projet dans Excel
  - Assurez-vous d'avoir tous les éléments obligatoires et suffisamment d'éléments optionnels pour obtenir le nombre de points prescrit par le nombre de membres de votre équipe.
- Dans le même Excel que le backlog, une feuille qui se nomme sprint#1. Une liste des stories sélectionnées pour le sprint1.
- Un repository sur « GitHub » et mettre agagnon01 comme collaborateur
  - Si vous êtes une équipe de 3, le projet devra être public

---

## CRITÈRE

- Énoncé de projet
  - Le projet est décrit adéquatement
- Backlog
  - Les « stories » sont suffisamment découpées
  - Le format des « stories » respecte le format « En tant que... je... afin de... »
  - Les critères d'acceptation présentent les « stories » adéquatement et couvrent les cas d'exception.
  - L'estimation du nombre d'heures requis pour les « stories » semble appropriée.
  - Les priorités sont appropriées.
- GitHub
  - Le projet est créé avec un nom significatif
  - Agagnon01 est un collaborateur

## SPRINT 1, 16 AVRIL AVANT 23H59 (ÉVALUÉ EN ÉQUIPE) 10 POINTS

Selon la planification actuelle qui peut changer, vous aurez 2 cours pour un total d'environ 6h en plus du temps que vous aurez lorsque vous aurez complété les exercices du cours. Ne soyez pas trop ambitieux. Vous devrez faire une courte présentation de vos stories fonctionnelles à la fin du sprint pour démontrer leur fonctionnement. Les stories lors du sprint 1 sont à faire en équipe.

---

## LIVRABLES

- Créer un « Release » sur GitHub avec le nom « Sprint 1 »
- Remettre le choix des stories pour le sprint 2
  - Les stories du sprint 2 sont à faire individuellement. Chaque membre de l'équipe doit donc se voir assigner environ 1 ou 2 stories à faire.
  - Mettre à jour le backlog en retirant les stories complétées et en ajoutant les nouvelles.
  - Créer une feuille pour chaque membre et nommer celle-ci « sprint#2PRENOM\_DU\_MEMBRE ». Mettre la ou les stories assignées à ce membre dans cet onglet.

---

## CRITÈRE SANS S'Y LIMITER

- Les stories sélectionnées fonctionnent
- L'application est exempte de problème
- Les interfaces sont esthétiques et peuvent bien s'adapter à différents affichages.
- Lisibilité des informations (Taille de police, choix de couleur)
- Les couleurs de l'application sont personnalisées et utilisées

**SPRINT 2, 17 MAI AVANT 23H59 (ÉVALUÉ DE INDIVIDUELLEMENT) 10 POINTS  
FONCTIONNEMENT ET 53,5 POINTS POUR LA REMISE FINALE**

Selon la planification actuelle qui peut changer, vous aurez 2 cours pour un total d'environ 6h en plus du temps que vous aurez lorsque vous aurez complété les exercices du cours. Ne soyez pas trop ambitieux. Vous devrez faire une courte présentation de vos stories fonctionnelles à la fin du sprint pour démontrer leur fonctionnement. Les stories lors du sprint 2 sont à faire individuellement.

---

**LIVRABLES**

- Créer un « Release » sur GitHub avec le nom « Remise finale »
- Un APK fonctionnelle qui sera utilisé pour tester votre application
- Document explicatif contenant les sections suivantes
  - Page titre
  - Table des matières
  - Pour chaque élément obligatoire
    - Indiquer dans quels fichiers le code pour cet élément est présent
    - Indiquer les instructions pour visualiser dans l'app(si applicable)
    - Exemple : Boite de dialogue  
La boite de dialogue est utilisée pour afficher un message lorsque l'adresse courriel de l'utilisateur est erronée. Le code est présent dans le fichier « login.java » dans la racine du répertoire contenant les fichiers de code Java. Il est possible de visualiser la boite de dialogue en écrivant une adresse courriel invalide « aa.com » et en appuyant ensuite sur le bouton « Connexion ».
  - Pour chaque élément optionnel
    - Indiquer dans quels fichiers le code pour cet élément est présent
    - Indiquer dans quel but cet élément est utilisé
    - Exemple : GPS  
Le capteur GPS est utilisé dans le fichier « localisationUtilisateur.java » dans le répertoire « GPS ». Le code permet de connaître la localisation de l'utilisateur pour ensuite afficher Restaurant le plus près.

---

**CRITÈRE SANS S'Y LIMITER**

La correction sera faite sur une tablette du CÉGEP à moins d'entente préalable différente.

- Les stories sélectionnées fonctionnent. (10 points)
- L'application est exempte de problème (10 points)
- Les interfaces sont esthétiques et peuvent bien s'adapter à différents affichages. (5 points)
- Le document explicatif contient une page titre (1 point)
- Le document explicatif contient une table des matières(0,5 point)
- Le document explicatif contient pour chaque élément obligatoire de manière bien identifiée les indications nécessaires pour retrouver le code impliqué ainsi que les instructions pour appeler ledit code.(3 points)
- Le document explicatif contient pour chaque élément optionnel de manière bien identifiée les indications nécessaires pour retrouver le code impliqué et ce à quoi sert ledit code. (20 points)
- Les éléments obligatoires sont présents et fonctionnels(3 points)

- Les éléments optionnels sélectionnés sont présents et fonctionnels (10 points)
- Il y a au moins 10(2 membres) ou 12(3 membres) points d'éléments optionnels. (-2 points par point non fonctionnel des éléments optionnels).
- Le document possède une présentation propre
- La couleur est modifiée (2 points)
- Lisibilité (2 points)
- Propreté de la présentation (2 points)
- Une APK est remise (5 points)