

Projet : Aide à la mémorisation
Version du 28 novembre

1 Modalités

Le projet est à réaliser en binôme, chaque membre du binôme doit faire une partie conséquente du projet et connaître sa totalité. Projets en solo sont tolérés, mais fortement déconseillés

Les soutenances auront lieu le 11 janvier pour les binômes avec étudiants Genial et le 16 janvier pour les autres. Le développement du projet sur git doit terminer avant 23h59 le 9 janvier. La soutenance se fait sur machine, inutile de faire les transparents. Lors de la soutenance vous démontrerez les fonctionnalités principales de votre projet, l'architecture de l'application, et certains détails du code. Les deux membres de binôme doivent participer à la présentation.

Nous pourrions utiliser un ou des détecteurs de plagiat pour repérer les copies et autres « emprunts ».

Dépôt git du projet (obligatoire)

Vous devez créer pour votre projet un dépôt git sur gaufre

gaufre.informatique.univ-paris-diderot.fr

avant le 20 novembre.

Les deux enseignants de l'UE doivent être désignés comme **Reporter** de votre projet sur git. Pour stocker votre projet Android dans le dépôt git on procède de façon suivante :

- (1) Créer un projet sur AndroidStudio.
- (2) Sur AndroidStudio aller vers VCS -> **Enable Version Control Integration**.
- (3) **Select a version control system to associate with the project** : choisir Git.
Les noms de fichiers à sauvegarder dans git sont désormais coloriés en rouge.
- (4) Dans le menu de AndroidStudio apparaît un onglet **Git** à la place de VCS. Allez vers **Git -> Manage Remotes**
Dans la fenêtre qui s'ouvre faire un clic sur +. Une autre fenêtre s'ouvre où vous copiez le lien https vers votre dépôt git.
- (5) L'étape suivante est peut-être inutile si votre nom et mail Git sont déjà renseignés globalement sur votre machine. Aller vers le répertoire de votre projet et exécuter sur le terminal deux commandes :

```
git config user.name your_user_name
git config user.email your_email
```

`your_user_name` c'est évidemment votre nom sur Gitlab écrit entre les guillemets doubles. Vous pouvez voir ce nom en vous connectant sur le Gitlab gaufre et en allant vers l'onglet le plus à droite et en sélectionnant **Settings** (juste avant **Sign out**). D'habitude c'est votre nom et prénom avec l'espace entre les deux. `your_mail` c'est votre mail sur Gitlab gaufre, visible également dans **Settings**.

- (6) Dans AndroidStudio aller vers **Git -> Commit**. Une sous-fenêtre s'ouvre à gauche avec tous les fichiers que AndroidStudio propose de sauvegarder dans le dépôt. Sélectionner tout. Écrire un **Commit message** et faire **Commit** ou directement **Commit and push**. Notez que commit est impossible si vous n'avez pas écrit quelque chose dans la sous-fenêtre **Commit message**.

Votre binôme devrait commencer par cloner le projet.

Dans le dépôt git ajouter **README.md** avec les noms des membres de votre équipe de projet.

2 Présentation du projet

Le but du projet est de concevoir et implémenter une application Android qui aide l'utilisateur à apprendre un sujet (vocabulaire d'une langue étrangère, formules mathématiques, dates historiques, etc.). Pour chaque sujet on aura un jeu de questions avec leurs réponses. Chaque jeu de questions est séparé des autres. On ne mélange pas l'apprentissage des dates avec l'anglais.

Le principe de l'apprentissage est le suivant : Au début toutes les questions ont le même statut (niveau 1) et on prend une question qui est demandée à l'utilisateur. Si l'utilisateur répond correctement, le statut de la question est incrémenté, sinon le statut est remis à 1. Les questions avec statut 1 seront vu tous les jours, celles avec statut 2 tous les deux jours, celles avec statut 3 tous les quatre jours, et ainsi de suite.

L'application doit être capable de gérer un nombre quelconque de jeux de questions chacun composé d'un nombre quelconque de questions. Les jeux de questions sont stockés dans une base de données. Pour l'instant, une question et sa réponse sont du texte. Pour répondre correctement à une question l'utilisateur doit donner le texte exacte (ou presque) de la réponse.

Alternativement, vous pouvez implémenter des questions à choix multiples.

Les fonctionnalités suivantes devraient être implémentées, les deux premières sont indispensables.

- (1) **Création/Suppression** : L'utilisateur peut
 - créer/supprimer un nouveau jeu de questions
 - ajouter/enlever une question à un jeu de questions
- (2) **Aide à la mémorisation** : L'utilisateur peut
 - choisir un jeu de questions et commencer/continuer l'apprentissage (répondre à une série de questions) ;
 - consulter la réponse à une question trop difficile ;
 - choisir le statut d'une question manuellement, ou même supprimer une question ;
 - se faire afficher des statistiques pour un jeu de questions et pour tous les jeux.
- (3) **Rappel** : A minima, l'utilisateur lance l'application tous les jours. Il serait beaucoup mieux de le lui rappeler via une notification.
- (4) **Chargement de questions** : Il n'est pas très pratique d'entrer les questions une par une à la main. Pour cette raison, l'utilisateur peut choisir de télécharger un jeu de questions

dans un fichier à partir d'une adresse **http**. Le format de ce fichier (texte, format CSV, XML, JSON, ...) n'est pas imposé.

- (5) **Paramétrage** : Le comportement de l'application devrait de être paramétrable. Par exemple, le temps de réponse avant de considérer que la réponse à une question est fausse. Ou bien la couleur du fond ou la taille des polices.
- (6) **Extensions** : Vous pouvez proposer et implémenter des extensions. Par exemple, on peut imaginer que les questions ne sont pas uniquement du texte mais contiennent des images ou des sons.

3 Aspects techniques et autres critères d'évaluation

Impérativement le projet doit être réalisé en Kotlin sous AndroidStudio (version Giraffe).

Le projet doit se compiler et s'exécuter sans problèmes majeurs (sur votre machine et sur celles des enseignants).

On préconise l'utilisation de la technologie Jetpack Compose (avec ViewModel) pour les interfaces graphiques et Room pour la base de données. L'absence de ces technologies sera pénalisée.

On appréciera l'utilisation des technologies vues en cours (coroutines, datastore, mise en forme material 3, adaptation à l'écran, menus, navigation, services, work manager, notifications, download manager, broadcast receivers etc.), et d'autres technologies pertinentes.

La qualité de programmation sera un critère important : pensez à l'architecture de votre application, à la modularité, à la clarté du code et à la bonne conception de la base de données.

Le projet doit être ergonomique et facile à utiliser. Par ailleurs le comportement doit être correct quand vous tournez l'écran. La qualité graphique sera un critère mineur (ne passez pas trop de temps avec cela).