

M4104 – Projet : l'école des loustics

- Training : <http://developer.android.com/training/>
- Guide : <http://developer.android.com/guide/>
- Documentation : <http://developer.android.com/reference/>
- Documentation java : <http://docs.oracle.com/javase/7/docs/api>
- Tutoriel en français : <http://fr.openclassrooms.com/informatique/cours/creez-des-applications-pour-android>

Présentation

L'éducation nationale propose de vous engager pour réaliser une application mobile destinée aux écoles primaires. Cette application permettra aux élèves de s'auto-évaluer sur des petits exercices mathématiques et des exercices à base de connaissances.

Les fonctionnalités **minimales** de l'application seront :

- Proposer des exercices mathématiques : dans un premier temps, table de multiplication et exercice proposant une série de 10 additions simples à réaliser ;
- Créer des comptes pour chaque élève (prénom et nom) ;
- Proposer des exercices de culture générale : dans un premier temps, un exercice de type question/réponse en français ;

Exemples d'applications réalisées par des étudiants (le projet proposé à ces étudiants était légèrement différent mais l'idée était la même) :

- <https://www.youtube.com/watch?v=ow--EYXRK9g>

EVALUATION :

Le projet se base sur la notion de « développer plus pour gagner plus » ! Une application minimale vous est demandée (ci-dessus et développée dans les étapes ci-après). Des fonctionnalités supplémentaires vous sont proposées à chaque étape pour avoir des points supplémentaires. Vous pourrez choisir vos fonctionnalités en fonction de vos envies.

Vous ferez attention à ce que votre code soit **correctement structuré**, en distinguant les parties modèle (classes java), vue (XML) et contrôleur (Activités). De plus, votre **code devra être clair et lisible** : correctement indenté, respecter les conventions de nommage et utiliser des noms explicites. Vous **commenterez les parties de code** un peu longue afin de détailler les étapes de vos algorithmes.

AUCUN BUG ne devra être présent dans votre application sous peine d'un fort malus !

RENDU du projet :

Vous devrez rendre avec votre projet un rapport. Ce document, **SOUS FORME PDF**, expliquera les fonctionnalités présentes (minimales et supplémentaires) dans votre projet et le workflow de votre application.

Enfin, le **nom de votre projet devra commencer par votre login**, de même que l'archive que vous rendrez.

Le non respect de ces consignes impactera fortement l'évaluation de votre projet.

Etape 1 : Mise en place et réalisation du workflow

Objectif : concevoir l'enchaînement des différentes parties de l'application sur papier et réaliser ce workflow en utilisant les intentions explicites vu dans le TD2/TP2.

Mettez en place le workflow de votre application. Pour ce faire, réalisez un storyboard sur **papier** représentant les différentes fenêtres composant votre application. Par exemple, au lancement de votre application une fenêtre d'accueil contiendra la liste des comptes des élèves (pour le moment ad hoc, simulant des comptes – réalisation à l'étape 3 BD). L'application proposera aussi de « jouer » en anonyme. Lors du choix d'un compte élève ou anonyme, une autre fenêtre vous proposera un choix d'exercices. Lors du choix de l'élève, l'exercice se lancera et évaluera les compétences de l'enfant. Prévoir une aide expliquant chaque exercice à l'élève.

Création de votre projet : sur android studio > new project (Empty activity) > Minimum API 15 (pour couvrir le maximum de devices).

Etape 2 : Créer des exercices mathématiques

Objectif : mettre en place une interaction utilisateur et un traitement des données (réf. cours 1 / TP1). Transfert de données entre activités (réf. cours 2 / TP 2).

L'étape 2 consiste à proposer des exercices mathématiques. Dans un premier temps, intégrez l'exercice table de multiplication réalisé dans le TP2. Ensuite, réalisez un exercice proposant une série de 10 additions simples à compléter en conservant le nombre de bonnes réponses pour le donner à l'élève à la fin de l'exercice. Pour ce faire, créez un scénario de l'exercice sur papier et réalisez ensuite ce scénario à l'aide d'une ou plusieurs activités.

Exemple de réalisation (correction fonctionnalité supplémentaire) : <https://youtu.be/18kosN4J9LE>

IMPORTANT : vous **DEVEZ** séparer vos données, vos vues et votre traitement (MVC : modèle, vue, contrôleur). Par exemple, dans l'exercice pierre/feuille/ciseaux, les données étaient représentées par les classes Jeu et Resultat, la vue par la représentation XML de l'activité et le traitement par l'activité même.

AIDE : Nous vous conseillons de traiter l'affichage de la série d'opérations dans la même activité. L'affichage d'une nouvelle opération s'obtiendra par la mise à jour graphique de l'activité.

Fonctionnalités supplémentaires possibles :

- Proposer à l'élève de se corriger (comme sur la vidéo ci-dessus)
- Choisir le format des opérands (unité, dizaine et centaine). Exemple de réalisation : https://youtu.be/_ssg_PRXej8
- Choisir entre addition, multiplication, soustraction et division ! Pour ce faire, améliorer votre modèle pour qu'il devienne générique
- Introduire un timer. L'élève devra faire le plus d'opération en une minute. Exemple de réalisation : <https://youtu.be/NKWtUSiiKGY>

AIDE : une possibilité (mais pas la seule) est d'utiliser la classe `CountDownTimer` présente dans la librairie d'Android (<https://developer.android.com/reference/android/os/CountDownTimer>).

Etape 3 : Gérer les comptes élèves

Objectif : manipulation de la base de données interne (réf. cours 3 / TP3), utilisation d'intention implicite (cours 2 / TP1).

L'étape 3 consiste à gérer des comptes d'élèves sur l'appareil mobile. Dans un premier temps, l'application proposera à l'élève de se créer un compte : un prénom et un nom (login et mdp ne sont pas nécessaire, l'application est destinée à des écoles primaires). Ces informations seront sauvegardées en base de données (la librairie Android Room doit être utilisée). Les comptes créés seront ensuite proposés à l'accueil de l'application.

Exemple de réalisation: <https://youtu.be/fGFDZ3Pjpmk> (login et avatar pas nécessaire dans la version de base).

AIDE : Nous vous conseillons pour afficher une liste d'utiliser la classe `ListView` avec son adapter. Pour plus d'informations, reportez vous au document M4104C-2.TP2.exercice5 sur Chamilo et au lien suivant : <https://developer.android.com/reference/android/widget/ListView>

Vous pouvez également utiliser la classe `RecyclerView` plus avancée et plus flexible que `ListView` mais plus difficile à utiliser : <https://developer.android.com/guide/topics/ui/layout/recyclerview>

Fonctionnalités supplémentaires possibles :

- Proposer d'associer un avatar au compte (une image présente dans les ressources de votre projet) comme dans la vidéo ci-dessus

AIDE : Pour réaliser une boîte de dialogue plusieurs solutions sont possibles : <https://developer.android.com/guide/topics/ui/dialogs>

- Prendre une photo ou récupérer une photo dans la galerie pour l'avatar
- Enregistrer les meilleurs résultats de chaque élève sur les différents exercices (reportez vous au TP3 section « Comment faire pour conserver l'utilisateur courant ? »)

Etape 4 : Créer des exercices de culture générale

Objectif : manipulation de la base de données interne (réf. cours 3 / TP3).

L'étape 4 consiste à proposer des exercices évaluant les connaissances des élèves. Dans un premier temps, réalisez un exercice de type question/réponse constitué d'une série de 10 questions avec, pour chaque question, 3 réponses proposées dont une bonne (comme vu dans le TP1 exercice 2, sauf que la bonne réponse n'est pas toujours à la même place !). On vous laisse libre de trouver une liste de questions/réponses pour le français (par exemple conjugaison, trouver le bon mot, etc.). Comme dans les exercices mathématiques, conservez le nombre de bonnes réponses pour le donner à l'élève à la fin de l'exercice. Pour ce faire, créez un scénario de l'exercice sur papier, concevez ensuite la base de données nécessaire et réalisez le scénario.

Fonctionnalités supplémentaires possibles :

- Choisir entre des exercices de type question/réponse spécifiques : français, histoire ou géographie ! Pour ce faire, améliorer votre modèle pour qu'il devienne générique.

AIDE : la base de données contiendra des questions/réponses associées à un tag (français, histoire ou géographie). L'activité proposant l'exercice recevra le tag en paramètre (extra) et proposera les bonnes questions.

Etape 5 : Développer plus pour gagner plus

Ajouter des fonctionnalités ou des exercices supplémentaires. Quelques exemples possibles :
Mettre en place des niveaux de difficulté à débloquent, créer des exercices de logique, etc.