



INF8405 – Informatique mobile

Travail pratique 1 - Application de jeu Android - Tic-tac-toe!

1. Informations générales

Session	Hiver 2020
Chargé de laboratoire	Groupe 1 - Mehdi Kadi (mehdi.kadi@polymtl.ca)
Date de début	8 janvier 2020
Date de remise	11 février 2020
Laboratoire	L-4708
Pondération	10%
Directives particulières	<ol style="list-style-type: none">1. Tout projet rendu sera pénalisé de 4 points s'il est soumis par une équipe dont la taille est différente de (3) étudiants, sans l'approbation préalable du chargé de laboratoire.2. Tout travail sera pénalisé de 4 points si la composition des équipes de travail n'est pas communiquée aux chargés de laboratoires avant le 17 janvier..3. Rapport à rendre en format PDF.4. Le code source + le rapport sont à rendre dans une archive ZIP, avec le fichier .APK de l'application5. Soumission du rapport et des travaux réalisés par Moodle uniquement (http://moodle.polymtl.ca).6. Tout retard dans la soumission du travail réalisé mènera à une pénalité de 4 points par jour de retard
Auteur original	Mehdi Kadi.
Révision par	-

2. Connaissances préalables

- Programmation orientée objet, développement d'interfaces graphiques
- Langage XML
- Utilisation de l'API Android

3. Environnement et outils nécessaires

- Environnement de développement intégré (IDE) : Android Studio
- Tablette Android

4. Objectifs du laboratoire

L'objectif principal de ce laboratoire est d'initier l'étudiant au développement de services pour des terminaux mobiles. De manière spécifique, au terme de ce laboratoire, il s'agira pour l'étudiant de :

- Développer une application pour terminaux mobiles;
- Se familiariser avec les caractéristiques des applications mobiles;
- Se familiariser avec un système d'exploitation pour terminaux mobiles.

Ce travail pratique consiste, par la même occasion, à évaluer deux des 12 qualités de l'ingénieur définies par le BCAPG (Bureau canadien d'agrément des programmes de génie). Le Bureau d'agrément a pour mandat d'attester que les futurs ingénieurs ont atteint ces 12 qualités à un niveau acceptable. Les deux qualités en question sont:

Qualité 4 (Conception) : capacité de concevoir des solutions à des problèmes d'ingénierie complexes et évolutifs et de concevoir des systèmes, des composants ou des processus qui répondent aux besoins spécifiés, tout en tenant compte des risques pour la santé et la sécurité publiques, des aspects législatifs et réglementaires, ainsi que des incidences économiques, environnementales, culturelle et sociales.

Qualité 7 (Communication) : habileté à communiquer efficacement des concepts d'ingénierie complexes, au sein de la profession et au public en général, notamment lire, rédiger, parler et écouter, comprendre et rédiger de façon efficace des rapports et de la documentation pour la conception, ainsi qu'à énoncer des directives claires et y donner suite.

5. Présentation de l'application

Ce travail pratique consiste à réaliser une application de jeu pour deux joueurs. Il s'agira de développer une version du jeu "Tic-tac-toe", aussi appelé « morpion ». Le "Tic-tac-toe" est un jeu de réflexion qui se joue à deux joueurs. Le support du jeu est une grille dans le cadre de ce laboratoire. Les joueurs inscrivent tour à tour leur symbole sur une grille. Le premier joueur qui réussit à faire un alignement de ses symboles horizontalement, verticalement ou en diagonale gagne la partie.

5.1 Explication du jeu sur une grille de 3×3

Deux joueurs s'affrontent. Ils doivent remplir chacun à leur tour une case de la grille avec le symbole qui leur est attribué : **O** ou **X**. Le gagnant est celui qui arrive à aligner trois symboles identiques, horizontalement, verticalement ou en diagonale.

Une partie gagnée par le joueur **X** :



Une partie nulle :



Source : "Tic-tac-toe," Wikipédia. 15-Dec-2019.

Dans le cadre de ce laboratoire, vous allez devoir créer 3 modes ayant une taille de grille différente (**3x3**, **4x4**, **5x5**).

L'application ne permet pas à des joueurs de jouer en réseau. Les deux joueurs utilisent le même appareil Android pour jouer. Quand le joueur 1 appuie sur une case, ce sera le tour au joueur 2 de sélectionner une case sur la grille.

6. Requis

6.1.Requis fonctionnels

Qualité évaluée :

4.3 Procéder à la conception

Critère d'évaluation : Intégrer les concepts de programmation en réseautique retenus au premier laboratoire en répondant aux besoins et en respectant les requis fonctionnels du projet courant.

Au lancement de l'application, on affiche à l'utilisateur un *splash screen* qui dure 4 secondes. À la fin de ce délai l'utilisateur est dirigé vers le menu principal. Trois boutons sont affichés, positionnés verticalement, l'un par-dessus l'autre. Les boutons sont "Play", "About" et "Exit".

De manière spécifique, les fonctionnalités attendues de l'application sont :

6.1.1 Splash Screen

- ~~Un animation doit être jouée au lancement de l'application. Un logo animé peut être affiché au démarrage de l'application, une écriture peut être déplacée pour créer un effet artistique, etc. Le *splash screen* dure 4 secondes. Vous pouvez faire preuve d'imagination pour cette partie.~~
- ~~Durant l'animation, un son doit être joué et doit s'arrêter lorsque l'utilisateur est redirigé vers le menu principal.~~

6.1.2 Menu principal

- ~~Le bouton "Play" permet de lancer une fenêtre permettant à l'utilisateur de choisir la taille de la grille sur laquelle il souhaite jouer (3x3, 4x4, 5x5). Un bouton sur cette fenêtre permet de lancer la partie. La grille choisie s'affiche et il est possible aux joueurs de jouer chacun leur tour.~~
- ~~Le bouton "About" permet d'afficher une fenêtre contenant les noms et prénoms des membres de l'équipe ainsi que leur numéro de groupe (présent dans la boîte de remise sur Moodle.);~~
- ~~Le bouton "Exit" permet de fermer l'application de jeu.~~

6.1.3 Vue du jeu

- ~~Un bouton permet de revenir au menu principal et de relancer une nouvelle partie. L'état d'avancement de l'ancienne partie ne doit pas être sauvegardé; le score de chaque joueur doit être sauvegardé par contre.~~

- ~~Quand le joueur 1 appuie sur une cellule, un symbole (X ou O) qui lui est propre est affiché dans la cellule. Le même comportement s'applique pour le joueur 2;~~
- ~~Quand un joueur appuie sur une cellule, un son propre à chaque joueur doit être joué. L'application doit toujours être utilisable quand le son est joué; La durée du son doit être brève.~~
- ~~Une fois une cellule occupée, il n'est pas possible de remplacer le symbole déjà présent ou bien de le retirer; C'est le tour au prochain joueur de jouer.~~
- ~~En tout temps, il est possible de lire le score (nombre de partie gagnées par chaque joueur). Si l'application est relancée, le score de chacun des joueurs doit être maintenu.~~
- ~~Un bouton "Reset" permet de réinitialiser le score des joueurs à 0.~~

6.1.4 Fenêtre de succès

- ~~Une fenêtre ou bien un *toast* doit apparaître lorsque la partie est terminée, on affiche quel joueur a gagné;~~
- ~~La fenêtre de succès doit disparaître à l'aide d'une animation *fade-out* après 3 secondes;~~
- ~~À la fin de chaque partie vous devez donner la possibilité à l'utilisateur de relancer la même partie ou bien de choisir un nouveau format de grille et de lancer la partie une fois la sélection faite par l'utilisateur.~~
- ~~Le score de chacun des joueurs est mis à jour. Chaque joueur qui gagne une partie obtient 1 point.~~



Fig. 1: Rendu possible de la fenêtre de vue pour une grille 3x3 (Note: certains requis/fonctionnalités sont manquants dans cette illustration).

6.2 Requis non fonctionnels

- ~~L'application sera développée pour une plate-forme Android. À des fins de compatibilité, la version de la plate-forme à considérer est **Android 4.3.1 (API 18)** et versions supérieures. Le minSdkVersion est donc **Android 4.3.1 (API 18)**;~~
- ~~Le jeu doit être accessible aux terminaux mobiles notamment les tablettes. Dans le cadre de l'évaluation, la présentation de l'application se fera uniquement sur des tablettes répondant aux requis de plate-forme indiqués ci-haut;~~
- ~~L'application doit être facile d'utilisation. Un soin doit être accordé au système de navigation et à l'interface en général;~~
- ~~La production d'un code lisible et suffisamment documenté (commentaires du code) est vivement attendue. Le développement doit se faire avec l'IDE Android Studio.~~

7. Livrables

7.1 Langages et bibliothèques autorisés

- L'application doit être développée uniquement en Java ou bien en Kotlin.
- Usage permis de bibliothèques externe. Si vous utilisez ces bibliothèques, veuillez justifier dans la section « Présentation » de votre rapport la raison de leur utilisation.

Tout non-respect de ces consignes entraînera la note de 0.

7.2 Soumission

Outre la présentation de l'application (démonstration), les livrables à fournir à terme seront regroupés dans une archive (ZIP) ayant le format suivant:

INF8405_TP1_matriculeX_matriculeY_matriculeZ où $\text{matriculeX} > \text{matriculeY} > \text{matriculeZ}$

Votre archive contiendra les fichiers suivants :

- Le projet Android Studio de l'application incluant les fichiers sources (.java), autrement dit le dossier en entier contenant votre projet;
- Le rapport au format PDF;
- **Le fichier contenant votre application (.apk).**

En cas d'absence d'un élément parmi les livrables, le travail ne sera pas corrigé et vaudra à toute l'équipe la note de 0. Le non-respect des autres consignes est sujet à pénalité.

****Assurez-vous que les livrables compilent et s'exécutent adéquatement sur les ordinateurs du laboratoire! ****

7.3 Rapport

Qualité évaluée :

7.1 Lire et rédiger de la documentation

Critère d'évaluation : Rédiger un rapport technique documentant efficacement le travail d'ingénierie réalisé dans ce projet en utilisant différentes formes de langage (naturel, informatique, etc.)

Le rapport, d'une longueur maximale de 3 pages (excluant la page de présentation), doit comporter les éléments suivants :

- **Page présentation** qui doit contenir le nom ou le logo de l'école, le libellé et l'identifiant du cours, la session, le numéro et l'identification du projet, la date de remise, les matricules et noms des membres de l'équipe, la mention « Soumis à : **nom et prénom du chargé de laboratoire** ».
- **Introduction** en vos propres mots pour mettre en évidence le contexte et les objectifs du TP.
- **Présentation** de vos travaux. Une explication de votre solution mettant en lumière la prise en compte des principaux requis du système. Si vous utilisez des configurations particulières des bibliothèques ou des projets, précisez-les également.
- **Difficultés rencontrées** lors de l'élaboration du TP et les éventuelles solutions apportées.
- **Critiques et Améliorations** : Il serait intéressant d'inclure vos suggestions pour améliorer le laboratoire.
- **Conclusion** : Expliquez en quoi ce laboratoire vous a été utile, ce que vous avez appris, si vos attentes ont été comblées, etc.

8. Évaluation

La grille d'évaluation est la suivante :

Rubriques	Points
Évaluation de l'exécutable : respect des requis, aucun arrêt abrupte, convivialité des interfaces, etc.	12
Évaluation de l'implémentation : gestion adéquate de toutes ressources (création, utilisation, libération), gestion des erreurs, logique de développement, documentation du code , etc.	3
Rapport	4
Conformité de la remise (remise du fichier .apk)	1
Total des points	20