

INF8405 - Informatique Mobile

Travail Pratique N°1 : Application de jeu pour Android

1. Informations générales

Session	Hiver 2014
Date de début	Mercredi 29 Janvier 2014
Date de remise	Mercredi 26 Février 2014 à 13h45 au plus tard + Présentation de l'application au cours de la séance de TP.
Laboratoire	L4708
Pondération	15%
Nombre d'étudiants	2
Directives particulières	<ol style="list-style-type: none"> 1. Tout rapport sera pénalisé de 3 points s'il est soumis par une équipe dont la taille est différente de celle définie sans l'approbation préalable du chargé de laboratoire. 2. Rapport à rendre en format PDF ou Word. 3. Soumission du rapport et des travaux réalisés par moodle uniquement (http://moodle.polymtl.ca). 4. Tout retard de soumission du rapport et des travaux réalisés sera pénalisé de 3 points par jour de retard.
Proposition	Eric-Mayeul O. Fafolahan
Chargé de laboratoire	Eric-Mayeul O. Fafolahan (eric.fafolahan@polymtl.ca)

2. Connaissances préalables.

- Langage Java (programmation orientée objet, développement d'interface graphique, programmation événementielle).
- Langage XML.

3. Environnement et outils nécessaires.

- Environnement de développement Intégré (IDE) : **Eclipse + SDK Android + Plug-in Eclipse d'Android (ADT)**
- 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 :

- concevoir et réaliser une application pour terminaux mobiles ;
- se familiariser avec les caractéristiques des applications mobiles ;
- se familiariser avec un système d'exploitation pour terminaux mobiles.

5. Présentation de l'application

Au cours de ce travail pratique, vous aurez à concevoir et réaliser en mode mono-joueur une version simplifiée du jeu de pierres précieuses (*gems*) appelé **bejeweled**¹. Le support de bejeweled est une grille de 8 x 8 cellules dont chacune contient un type de pierre précieuse. A l'initialisation d'une partie, toutes les cellules de la grille sont remplies de manière aléatoire des différents types de gems (**figure 1**).



Figure 1: Grille de jeu à l'initialisation.

Le but du jeu est de permuter à l'horizontale ou à la verticale une pierre précieuse d'une cellule avec celle d'une cellule adjacente pour former une chaîne horizontale ou verticale de pierres identiques de longueur n (n supérieur ou égal à 3). Une permutation est dite valide lorsqu'elle mène à la formation d'une chaîne de plus de 3 pierres identiques. Les pierres disparaissent lorsque les chaînes sont formées. Les positions vides ainsi générées sur la grille sont comblées par les pierres situées sur la même colonne à un niveau supérieur. Celles-ci sont déplacées vers le bas suivant la colonne correspondante. Lorsqu'aucune pierre ne se trouve en position supérieure pour compenser une cellule vide, alors

¹ <http://en.wikipedia.org/wiki/Bejeweled>

l'application génère de manière aléatoire une nouvelle pierre dans la cellule vide. Les permutations invalides ne sont pas autorisées. Le but du jeu est donc de réussir à former un maximum de chaînes en exploitant toutes les permutations possibles sur la grille de jeu jusqu'à la condition de fin (consulter la rubrique suivante). A la fin d'une partie, le joueur peut accéder à une nouvelle partie ou quitter l'application.

6. Requis

6.1 Requis fonctionnels

Dans le cadre du TP, le jeu à programmer fonctionne globalement comme suit : deux modes de jeu sont proposés au démarrage de l'application. Ceci permet de déterminer la condition de fin de la partie. Le premier mode (jeu chronométré) spécifie une durée de jeu de 60 secondes. Le deuxième mode de jeu spécifie un nombre maximal de 10 permutations valides à faire. Pour chaque mode il n'existe qu'un seul niveau de jeu. A la fin, le joueur a la possibilité de changer de mode de jeu, de rejouer selon le même mode ou quitter l'application. De manière spécifique, les fonctionnalités attendues de l'application sont :

- La sélection d'options au démarrage de l'application. Le joueur peut à cette phase :
 - ✓ Consulter la *liste des meilleurs scores obtenus par joueur*. La liste doit être présentée suivant un classement décroissant en fonction des scores obtenus et contenir au moins (5) joueurs.
 - ✓ *Sélection du mode de jeu* au lancement de l'application
 - ✓ Quitter le jeu.
- L'identification du joueur au début de la partie et mise à jour de son score (conserver le meilleur score) ou l'insérer à la liste des meilleurs scores au terme de la partie de jeu.
- L'initialisation d'une partie de jeu selon la **Figure 1** quel que soit le mode de jeu choisi.
- Règle de calcul du score, suite à une permutation valide :
 - ✓ Une chaîne de 3 pierres identiques octroie **100 points** au joueur.
 - ✓ Attribuer **50 points** par pierres supplémentaire pour une chaîne de plus de 3 gems identiques. *Exemple* : une chaîne de 5 gems octroie donc **200** points au joueur.
 - ✓ Les chaînes formées en cascade (chaînes générées pendant la descente des pierres supérieures) et les gems spéciaux de bonification ne sont pas demandés.
- Jeu en mode tactile.
- Affichage à l'écran du score courant du joueur. Ce score est initialisé à zéro et incrémenté en fonction du gain obtenu au cours du jeu.

- Affichage à l'écran du nombre de chaînes formées (ne pas préciser la longueur). Ce nombre, initialisé à zéro (0) doit être régulièrement incrémenté au cours du jeu.
- Affichage à l'écran :
 - ✓ du temps de jeu restant (en secondes) pour le mode de jeu chronométré. Le temps est initialisé à 60 secondes au début de la partie.
 - ✓ du nombre de permutations valides restantes avant la fin de la partie dans le deuxième mode de jeu.
- Réinitialisation à tout moment d'une partie de jeu en cours (recommencer la partie).
- Possibilité de mettre fin au jeu à n'importe quel moment, avec émission de message de confirmation de fin de partie pour prévenir les sorties en cas de mauvaises manipulations. Le joueur doit confirmer la fermeture de l'application.

6.2 Requis non fonctionnels

Les exigences non fonctionnelles attendues sont les suivantes :

- Le jeu sera développé pour une plate-forme Android. A des fins de compatibilité, la version de la plate-forme à considérer est Android 3.0 (API 11). Toutefois, il sera accepté toute autre version entre Android 3.0 (API 11) et Android 4.0.3 (API15) inclusivement. Les versions Android 4.1.2 (API16), Android 4.2 (API 17) et les versions antérieures à Android 3.0 (API 11) ne seront donc pas tolérées.
- 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 à l'interface en général.
- La production d'un code lisible et suffisamment documenté (commentaires du code) est souhaité. Le développement doit se faire avec l'IDE Eclipse.

7. Livrables

Outre la présentation de l'application (démonstration), les livrables à fournir à la fin du TP seront regroupés dans une archive (ZIP ou RAR) dont le nom est formé des matricules des membres de l'équipe, séparés par le caractère « trait de soulignement » c'est-à-dire ('_'). L'archive contiendra un rapport de TP au format PDF ou Word et le code source de l'application. Le non-respect des consignes est sujet à pénalité.

7.1 Rapport de TP

Votre rapport doit contenir :

- ✓ **Une page de présentation** (indispensable) faisant mention des éléments suivants : Nom et logo de l'école ; le sigle et le titre du cours ; la session ; le libellé du TP ; les noms, prénoms et matricules des membres de l'équipe; la mention « Soumis à : *nom et prénoms du chargé de laboratoire* » ; la date de soumission.
- ✓ **Un résumé** de moins de 200 mots des objectifs et des réalisations de votre travail.
- ✓ **Une introduction** qui mentionne le contexte dans lequel le travail a été réalisé ainsi qu'une brève description de la méthodologie (aspects techniques)
- ✓ Une section dédiée à la **méthodologie** (aspects techniques) de réalisation du TP. Cette section doit mentionner :
 - Une indication de la répartition des travaux au sein de l'équipe.
 - Une indication des différentes phases de réalisation du TP.
 - La structure de l'application (classes, packages, méthodes/variables publiques privées, composantes utilisées). Expliquer autant que faire se peut, les choix effectués.
 - La séquence d'exécution. (Il est suggéré d'inclure un diagramme à cet effet).
 - Les difficultés rencontrées ainsi que les décisions qui ont été prises pour les résoudre.
- ✓ Une section dédiée aux **tests d'exécution** (inclure des captures d'écran) pour valider les fonctions réalisées.
- ✓ Une **conclusion et les travaux futurs**. Cette section doit mentionner :
 - Un résumé de l'expérience acquise après la réalisation du projet.
 - Des améliorations susceptibles d'être apportées à votre projet si un délai supplémentaire devait être accordé.
 - Vos critiques et suggestions pour l'amélioration du contenu.

7.2 Code source

Afin de faciliter la correction, vous devrez remettre un fichier .zip généré à l'aide de l'outil d'exportation d'Eclipse. Les étapes sont les suivantes : Clic droit sur le projet > Export... > General > Archive File > Next > S'assurer que le projet est bien coché ainsi que tous les dossiers et fichiers> Donner un nom au

fichier archive avec précision du chemin complet d'accès en remplissant le champ « *To archive file* » >
 Cocher l'option « *Save in Zip format* » > Finish.

8. Grille d'évaluation

Rubriques	Points
Présentation (Démonstration + réponses aux questions)	4
Évaluation complémentaire de l'exécutable : (fonctionnalité, ergonomie, portabilité, performance).	4
Évaluation du code source (implémentation de l'architecture, logique de programmation, documentation du code).	6
Rapport de TP (fond, forme).	6
Total	20