**Application Mobile – Examen Florian Rouget, 23/11/2016**

# Présentation d’Android

1. **Qu’est-ce qu’Android ? Quel intérêt qu’il soit basé sur un Kernel Linux ?**

Android est un système d’exploitation utilisé sur beaucoup de machines portables. L’intérêt d’avoir un système d’exploitation Android basé sur un Kernel Linux est qu’il offre plus de liberté pour les développeurs au niveau du développement pur et des droits : Android est open-source.

1. **Quelle est la version actuelle d’Android ?**

La version d’Android actuelle est Android 7.0 « nougat ».

1. **A quoi sert un émulateur ?**

Un émulateur sert à tester son application directement sur son PC, sans avoir besoin d’avoir un appareil Android à portée de main. Un émulateur « simule » le comportement qu’aurait la machine.

1. **Pourquoi ne se sert-on pas d’Eclipse ?**

Eclipse est un IDE qui permet de faire de l’Android, mais il n’est pas dédié à Android, alors que Android Studio fait du Android nativement.

# APK

1. **Qu’est-ce qu’un APK ? de quels types peut-il être ? Que contient-il ? Quel est l’inconvénient de l’un des 2 types d’APK ?**

Un APK (Android Package) est un export d’un projet Android. C’est un équivalant d’un « setup.exe », dans le sens où on charge un APK sur sa machine Android, puis on l’exécute pour vraiment installer l’application. Un APK contient donc tous les fichiers nécessaires pour que l’application fonctionne (ressources compressés, fichiers de données…). Un APK peut être signé ou non-signé, l’avantage étant qu’un APK signé peut être uploadé sur le google play, alors qu’un APK non-signé ne peut pas l’être.

1. **Le Keystore, à quoi sert-il ?**

Le Keystore contient les clés de signatures qui permettent de signer un APK et de le distribuer.

1. **A quoi sert l’alias ?**

L’alias correspond au nom et au mot de passe d’une clé de signature. L’alias sert à assurer la sécurité des données de la clé.

# Composantes d’Android

1. **Qu’est-ce qu’une activité ?**

Une activité correspond à un écran de l’application. Il s’agit d’un fichier XML contenant tous les objets visuels présents et utilisables par l’utilisateur (listes, boutons, images, textes…).

1. **Qu’est-ce qu’un service ?**

Un service est un processus qui s’exécute en tâche de fond de l’application. Il ne s’arrête que lorsqu’il a fini sa tâche ou sera interrompu. Un service peut servir à échanger des paquets, par exemple.

1. **Qu’est-ce qu’un content provider ?**

Un content provider permet de partager les données d’une application, qui peuvent être choisies en base de données SQLite, dans des fichiers ou sur le web. Le but est de permettre à d’autres applications de requêter ces données.

# Cycle de vie d’une activité

1. **4 états à expliquer succinctement :**
   1. En cours d’exécution : l’état de l’application lors d’une utilisation normale.
   2. En pause : l’application n’est pas utilisée, mais partiellement visible.
   3. Stopper : l’application n’est pas utilisée, mais encore lancée en fond.
   4. Tuer : l’application est fermée.
2. **Quelle est la méthode du cycle d’une activité que vous avez employé jusqu’à maintenant ?**

Jusqu’à maintenant nous avons principalement utilisé une application en cours d’exécution.

# Contexte d’une application

1. **Que représente le contexte ?**

Le contexte d’une application représente l’état courant d’une application et les informations sur son environnement. Il sert à récupérer des objets transmis par d’autres parties de notre application.

1. **Que font les méthodes suivantes :**
   1. getApplicationContext() : permet de récupérer le contexte de notre application.
   2. getContext() : permet de récupérer le contexte dans lequel la vue courante est exécutée.
   3. getBaseContext() : permet de récupérer le contexte défini grâce à la méthode setBaseContext.
   4. This :

# Le Manifeste

1. **A quoi sert ce fichier XML ?**

Le fichier manifeste est une description complète de l’application, il répertorie les composantes, les activités, les services, les permissions, les fournisseurs de contenu, etc. Il est vital pour le projet, car sans lui toutes les données structurantes sont perdues.

1. **Où le trouve-t-on ?**

On trouve le fichier manifeste à la racine de tout projet Android. Souvent, on trouve AndroidManifest.xml dans le dossier « manifests », lui-même dans le dossier « app ».

1. **Que contient-il ?**

Il contient les composantes, les activités, les services, les permissions, les fournisseurs de contenu. Il détient une liste de tous les éléments existants dans l’application, en somme.

1. **Décrivez le contenu de la partie application.**

Entre les balises <application>…</application>, il y a : Un définition de l’icône de l’application, le nom de l’application, le thème de l’application, l’autorisation de sauvegarde, une définition de « MainActivity » entre balises <activity></activity>.

# Architecture d’un Projet

1. **Que contient le répertoire « Java » ?**

Le répertoire « Java » contient tous les fichiers .java qui contiennent le code de l’application.

1. **Que contient le répertoire « RES » ? Décrivez le contenu de chaque sous-répertoire.**

Le répertoire « RES » contient toutes les ressources de l’application (images, sons, layout…). Dans « drawable », il y a toutes les images et les icônes utilisé dans les interfaces, dans « Layout » il y aura toutes nos interfaces (dispositions des vues sur les écrans), dans « mipmap » sont placés les icônes systèmes de l’application, et dans « values », on trouve toutes les valeurs de configuration, les textes.

1. **Dans le répertoire « Values » figure le Strings.xml. Comment-vous en servir pour l’internationalisation de votre application ? Comment changez-vous la langue au final ?**

Dans le fichier Strings.xml se trouvent des chaines de caractères. Il suffit de créer plusieurs types de ressources dédiés aux différentes langues contenant chacune les textes que l’on veut utiliser dans notre application. Pour ce qui est du changement de langue,