DEVOIR APPLICATION MOBILE

# Présentation d’Android

1. Qu’est-ce qu’Android ? quel est l’intérêt qu’il soit basé sur le système Kernel Linux ?

C’est un système complet pour smartphones et tablettes, qui comporte une gestion matérielle, un API de programmation…

Android est aussi un framework permettant le développement d’application mobile.

L’intérêt qu’il soit basé sur le système Kernel Linux, c’est qu’il sépare la couche matérielle de la couche logicielle.

1. Quelle est la version actuelle d’Android ?

La version actuelle d’Android est Marshmallow.

1. A quoi sert un émulateur ?

Un émulateur sert à tester les applications développer sous Android avant de de sortir l’APK de celle-ci. Il simule à l’écran un smartphone pour voir si l’affichage et le code fonctionnent.

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

# APK

1. Qu’est-ce qu’un APK ? Que contient-il ? de quels types ? Inconvénient de l’un des deux types d’APK ?

C’est un binaire représentant une application, pour le créer l’application est compilée et packagée dans un fichier contenant :

* Le code de l’application compilé
* Les ressources
* Les assets
* Les certificats
* Le fichier de manifest

Il existe 2 types d’APK, le signé et le non-signé, ce dernier peut être utilisé pour tester l’application mais pas pour être publié sur le googleplay.

1. Le Keystore, à quoi sert-il ?

Le Keystore sert à stocker les clefs créées pour les différentes applications, les clefs seront demandées à chaque création d’un apk. Cette clef servira à distribuer l’application.

1. A quoi sert l’alias ?

L’alias est le nom de la clef, mais aussi son mot de passe, il sécurise la clef.

# Composantes d’Android

1. Qu’est-ce qu’une activité ?

Une activité se trouve dans l’interface utilisateur d’une application Android, une activité correspond donc à un écran. C’est la composante principale, elle représente l’implémentation et les interactions de nos interfaces.

1. Qu’est-ce qu’un service ?

Le service ne possède pas d’interface, mais il permet l’exécution d’un traitement en tâche de fond, il ne s’arrête que quand il est fini.

1. Qu’est-ce qu’un Content Provider ?

Un Content Provider permet de partager les données d’une application, Ces données peuvent être stockées en base SQLite, dans des fichiers ou sur le web. Le but étant 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’activité se trouve au premier plan et reçoit les interactions utilisateur.
   2. En pause, l’activité est visible mais l’utilisateur ne peut pas interagir avec.
   3. Stoppée, l’activité n’est plus visible mais toujours en cours d’exécution.
   4. Tuée, l’activité est comme son nom l’indique tuée, elle n’est plus en cours d’exécution et disparaît dans la BackStack.
2. Quelle est la méthode du cycle d’une activité que vous avez employé jusqu’à présent ?

La méthode du cycle d’une activité utilisé jusqu’à présent est onCreate() ;

# Contexte d’une application

1. Que représente le contexte ?

Le contexte d’une application représente son état courant et les informations sur son environnement. Il sert à récupérer des objets transmis par d’autres parties de votre application. On peut récupérer le contexte de 4 manières différentes, voir ci-dessous.

1. Que font les méthodes suivantes ?
   1. GetApplicationContext() ;

Cette méthode récupère le contexte de notre application.

* 1. getContext() ;

Cette méthode récupère la vue courante.

* 1. getBaseContext() ;

récupèrer le contexte défini grâce à cette méthode.

* 1. This

S’utilise uniquement si on est dans la classe héritante.

# Le Manifeste

1. A quoi sert se fichier XML ?

Le Manifeste est la description complète de tous les composants Android, c’est à cet endroit qu’est fait l’inventaire du contenu de l’application.

1. Où le trouve-t-on ?

Le fichier Manifeste se trouve à la racine de tout fichier Android.

1. Que contient-il ?
   1. Dans la 1ère partie

La déclaration xmlns, elle est obligatoire.

Le package(identifiant) de l’application.

1. Décrivez le contenu de sa partie application.
   1. Selon le contenu de l’application, le Manifeste varie

Les permissions.

Uses-configurations

Uses-features

Supports-screen

Contient toutes les spécificités et les composantes de l’application :

* + - Le nom
    - L’icône
    - Le thème
    - …

# Architecture du projet

1. Que contient le répertoire Java ?

Le répertoire java contient tous les fichiers sources, le code source et le 2ème code source des tests.

1. Que contient le répertoire RES ? Décrivez le contenu de chaque sous répertoire.

Le répertoire RES est un ensemble de fichiers xml et d’images appartenant à l’interface, dans les sous-dossiers on retrouve :

* + Le drawable, images, icônes utilisées dans l’interface.
  + Les layouts, ce sont les interfaces, la disposition des vues et des écrans.
  + Le menu, ce sont les menus contextuels ou d’application.
  + Le mipmap, images, icônes utilisées dans l’interface.
  + Le values, valeur de configuration…

1. Dans le répertoire Values, figure le String.xml, comment vous en servir pour l’internationalisation de votre application ? Comment changer vous de langue au final ?

Dans le String.xml en allant dans l’éditeur, nous pouvons créer une nouvelle clef, nous permettant de changer la langue, et donc les valeurs des chaines de notre application. Cette action va créer de nouveaux fichiers Strings.xml, mais dans différentes langues.

Pour changer de langue rien de plus simple, toujours dans l’éditeur il nous suffit de selectionner la bonne clef et de changer la langue.