

## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Création d'une vue**

#### **Description de l'acquis**

Apprentissage de la création d'une vue (composée de boutons et de parties textuelles) pour l'utilisateur à l'aide de Android Studio.

#### **Démontré par**

Utilisation du fichier XML activity\_nomActivité.xml, modification du code afin de l'adapter au besoin de l'application, ajout d'éléments, personnalisation des éléments.

### **Acquis : Rendre l'application « Responsive »**

#### **Description de l'acquis**

Permettre à l'utilisateur d'utiliser notre application et faire en sorte que l'application évolue au fil du temps ou fil des actions de l'utilisateur.

#### **Démontré par**

Lien entre le Java et le XML permettant lors d'un clic sur notre bouton d'effectuer un traitement en Java. (Augmentation du compteur + Modification du contexte en fonction de la valeur du compteur).

### **Acquis : Menu Contextuel**

#### **Description de l'acquis**

Apprentissage de la création d'un menu contextuel pouvant contenir certaines options.

#### **Démontré par**

Création d'un menu contextuel permettant de quitter l'application.

### **Acquis : Shared Preferences**

#### **Description de l'acquis**

Utilisation des Shared Preferences afin de sauvegarder les préférences de l'utilisateur et permettent de récupérer ces informations lors du redémarrage de notre application.

#### **Démontré par**

Utilisation des Shared Preferences afin de sauvegarder notre highscore permettant ainsi à chaque utilisation de notre application de savoir quel était notre highscore précédent.

## **Acquis : Démarrer une nouvelle activité**

### **Description de l'acquis**

Utilisation de diverses activités permettant de rendre l'application plus complète et de mieux découper le code.

### **Démontré par**

Démarrage d'une application à l'aide de `startActivityForResult` (ou `startActivity`) en passant un intent explicite à cette méthode (permet de spécifier quelle Activité va être appelée).

## **Acquis : Passage de données entre activités**

### **Description de l'acquis**

Passage d'informations au sein des activités afin de rendre l'application cohérente.

### **Démontré par**

Utilisation du `intent.putExtra(« Type », « valeur »)` afin de sauvegarder l'information au sein de l'intent. Récupération au sein de la classe appelée à l'aide de `intent.getStringExtra(« Type », « valeurDéfaut »)` ;

## **Acquis : Appel Intent implicite**

### **Description de l'acquis**

Utilisation d'intent implicite afin d'appeler des API externe à notre application.

### **Démontré par**

Utilisation de l'API permettant les appels à l'aide de : `new Intent(Intent.ACTION_CALL)`; Attribution de permissions au sein du manifeste afin de permettre à l'application appelée de réaliser sa fonction.

## **Acquis : Sauvegarde de données lors de la rotation de l'écran**

### **Description de l'acquis**

Sauvegarde de l'état de l'application lors de la rotation de l'écran afin de permettre à l'utilisateur de continuer à utiliser l'application comme si de rien n'était.

### **Démontré par**

Création à l'aide de la méthode `onSaveInstanceState` d'un objet `SavedInstanceState` qui sera récupéré après la rotation de notre écran. On pourra remettre l'application dans l'état où elle était avant la rotation de l'écran à l'aide de la méthode `onRestoreInstanceState()`.

## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Apprentissage du modèle MVC**

#### **Description de l'acquis**

Apprentissage de l'architecture MVC.

#### **Démontré par**

Division en trois couches afin de faciliter la programmation et de coder de manière saine :

Modèle : Responsable de l'état et des données du processus.

Contrôleur : Responsable de capturer les entrées de l'utilisateur. Permet de modifier le modèle.

Vue : Responsable d'afficher les données à l'utilisateur.

### **Acquis : Utilisation de l'attribut « tag »**

#### **Description de l'acquis**

Utilisation de tag afin de faciliter la programmation

#### **Démontré par**

Utilisation de l'attribut tag afin d'automatiser la gestion des chiffres tout comme des opérateurs de notre calculatrice.

### **Acquis : Utilisation d' « Observer »**

#### **Description de l'acquis**

Utilisation d'observer afin de prévenir les vues d'un changement au sein de la couche modèle.

#### **Démontré par**

Enregistrement des vues au sein du modèle en temps qu'observers et avertissement de ces derniers lors d'un changement au sein de la couche modèle à l'aide la méthode notifyAll() appelant la méthode notifyChange() de chacun des observers.

## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Utilisation d'API externes**

#### **Description de l'acquis**

Utilisation d'API externes en autorisant notre application à les utiliser et utilisation de ces derniers afin d' étoffer notre propre application.

#### **Démontré par**

Autorisation et utilisation de l'API permettant la réception de SMS

### **Acquis : Utilisation d'un Broadcast Receiver**

#### **Description de l'acquis**

Utilisation d'un Broadcast Receiver afin de récupérer les SMS reçu par notre appareil et traitement de ceux-ci.

#### **Démontré par**

Récupération du contenu d'un SMS ainsi que le numéro de téléphone de l'expéditeur afin de les utiliser au sein de notre application.

### **Acquis : Amélioration de la persistance des données avec SQLite**

#### **Description de l'acquis**

Utilisation d'une base de données au format SQLite afin d'améliorer la persistance de nos données et permettre à l'utilisateur de les réutiliser par la suite. (Il s'agit d'une base de données présente directement sur l'appareil).

#### **Démontré par**

Création de la base de données, création d'une table, insertion de lignes au sein de celles-ci, sélection de données présentes dans notre base de données.

### **Acquis : Utilisation de ListView et Adapter**

#### **Description de l'acquis**

Utilisation de listView dans le but de permettre l'affichage des données présentes au sein de la base de données correctement. Utilisation de l'Adapter afin de permettre cette fonctionnalité.

#### **Démontré par**

Récupération du cursor reçu grâce à notre requête et utilisation de celui-ci afin de paramétrer notre ListView (A l'aide de l'ArrayAdapter afin d'afficher tel quel le contenu de notre table, A l'aide d'un

Nom : Sacré Christopher  
Date : 30/03/2017

Fiche Numéro : 4

cursor Adapter afin de paramétrer l’affichage et de choisir les colonnes à afficher et dans quel ordre les afficher).

## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Utilisation d'API externes**

#### **Description de l'acquis**

Utilisation d'API externes en utilisant des liens URL afin d'obtenir des données, des informations depuis des sites externes.

#### **Démontré par**

Utilisation de l'API de LastFM dans le cadre d'une application visant à afficher les artistes ayant le plus été écoutés.

### **Acquis : Utilisation de JSON**

#### **Description de l'acquis**

Désérialisation d'un objet JSON et obtention d'informations sur base de ce dernier.

#### **Démontré par**

Utilisation de diverses opérations sur un objet JSON récupéré depuis LastFM.

### **Acquis : Approfondissement des compétences sur l'ArrayAdapter**

#### **Description de l'acquis**

Apprentissage de nouveaux éléments au sujet des ArrayAdapter.

#### **Démontré par**

Utilisation d'un ArrayAdapter afin d'afficher le contenu d'un JSON au sein d'une ListView.

### **Acquis : Apprentissage des fragments**

#### **Description de l'acquis**

Utilisation de fragments afin de rendre notre application dynamique et plus cohérente.

#### **Démontré par**

Génération d'un projet Android Studio : Master / Details Flow et apprentissage du fonctionnement de ce dernier. Apprentissage du passage entre différents fragments.

## **Acquis : Découverte de la WebView**

### **Description de l'acquis**

Découverte d'un nouveau format d'affichage de données : la WebView. Permet d'afficher directement du contenu récupéré sur le Web au sein de notre application.

### **Démontré par**

Recherche de la page web de LastFM correspondant à l'artiste sélectionné.

## **Acquis : Découverte du LongClickListener**

### **Description de l'acquis**

Découverte d'un nouveau type de listener : le LongClickListener (Permet de ne pas lancer un événement au moindre clic mais d'attendre un clic plus long).

### **Démontré par**

Utiliser pour la sélection d'un des artistes et l'affichage de ces détails.

## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Utilisation de sa propre API**

#### **Description de l'acquis**

Utilisation de l'API de PAE via le serveur de l'IPL.

#### **Démontré par**

Connexion à l'API de PAE et utilisation des méthodes précédemment codées dans le cadre du cours de PAE. (Envoi d'une requête contenant les bons arguments (Passage d'arguments à l'aide du JSON) et réception de la réponse (La réponse sera également au format JSON). Traitement des différentes réponses (redirection en cas de login, affichage des données demandées, ....).

### **Acquis : Création d'une LoginActivity**

#### **Description de l'acquis**

Utilisation du template Login Activity proposé par Android Studio.

#### **Démontré par**

Apprentissage des fonctionnalités relatives à ce template (affichage de messages d'erreurs, Spinner lors du chargement de la requête, utilisation de la AsyncTask).

### **Acquis : Récapitulatif des précédents acquis**

#### **Description de l'acquis**

Récapitulatif de différents éléments vus aux précédents cours de mobile et familiarisation avec ceux-ci.

#### **Démontré par**

Création d'un Intent comme vu précédemment afin de changer d'activité, réutilisation du template Master-Details proposé par Android Studio. Utilisation d'un ArrayAdapter afin d'afficher une liste d'éléments. Utilisation du JSON, création de onClickListener, révision du modèle MVC.



## **Fiche réflexive d'acquis d'apprentissage**

### **Acquis : Savoir utiliser les bases d'Android Studio**

#### **Description de l'acquis**

Se familiariser avec Android Studio, nous permettant par la suite de faciliter notre résolution des exercices. Comprendre les mécanismes de ce dernier tout comme les outils qu'il nous met à disposition.

#### **Démontré par**

Paramétrage de l'environnement de développement, Création d'un projet ainsi que création d'un appareil virtuel et exécution correcte de ce dernier, voyage entre les différents onglets proposés par l'IDE, utilisation du SDK manager.

### **Acquis : Observation de la structure**

#### **Description de l'acquis**

Exploration de la structure de fichiers et essai de compréhension de différents fichiers, apprentissage de l'utilisation de ces derniers.

#### **Démontré par**

Android Studio nous permet de facilement nous déplacer au sein de notre projet (à l'aide de différentes vues (Android, Project, ...) ainsi qu'une structure de fichiers facile à utiliser), les fichiers se trouvent pour une application MyApp au sein de : MyApp/app

À partir de là on distingue trois sous dossiers :

App > contient la description de notre application, les permissions accordées, ainsi que les Activity qu'elle contient

Java> contient le code java (au format .java)

Res > Contient les images, les icones, la description des écrans, les fichiers XML servant à l'application, c'est notamment dans cette partie du projet que l'on utilise l'outil de conception graphique.

## Acquis : Utilisation de l'outil de conception graphique

### Description de l'acquis

Android Studio nous permet via un outil de conception graphique de modifier facilement notre application.

### Démontré par

Accès au fichier `res/layout/activity_main.xml` et `res/layout/content_main.xml` à modifier, Passage en mode design et utilisation la palette pour placer les objets souhaités (Texte, icônes, ...). Modification des différentes données en double cliquant dessus ou via leur zone propriétés associée.

## Acquis : Affichage de texte au sein du Log

### Description de l'acquis

Modifier le code java pour inclure des messages dans la console (logcat) lors de l'exécution du programme

### Démontré par

Ajout de la ligne `Log.priorite("MonTag" + id);` dans la méthode `onOptionsItemSelected` pour afficher le texte lorsqu'un `OptionsItem` (Clic sur les trois points en haut à gauche, puis settings) est sélectionné dans l'application.

## Acquis : Utilisation du debugger

### Description de l'acquis

Apprentissage de la bonne utilisation du debugger propose par Android Studio

### Démontré par

Ajout d'un breakpoint au sein du code source java. Aperçu de la fonction debugger proposé par Android Studio, observation et utilisation de ce dernier au sein de Android Studio.

## Acquis : Découverte du cycle de vie d'une application

### Description de l'acquis

Apprentissage des différentes étapes dans lesquelles un processus peut passer lors de son cycle de vie.

### Démontré par

Affichage au sein de la console de message nous permettant de savoir le changement entre les différentes étapes. Visionnage du changement d'étape à l'aide du debugger du changement d'état.

## **Acquis : Utilisation d'ADB**

### **Description de l'acquis**

Apprentissage et familiarisation de/avec l'outil ADB. Apprentissage des bonnes pratiques au sein de cet outil.

### **Démontré par**

Recherche de commandes à l'aide de commande « adb logcat help », essai de diverses commandes, analyse des résultats. Utilisation de la commande « logcat » et utilisation de divers filtres.

## **Réponse aux questions posées sur la fiche :**

### **Question 1**

Ces descriptions se trouvent dans le fichier res/layout/activity\_main.xml

### **Question 2**

On y fait appel lors de l'appel de la méthode setContentView dans la méthode onCreate.