

# [Innovations technologiques]

*Projet : Site de vente d'objets d'occasion*

## 1 Énoncé

Il est demandé de développer une application Android en Java permettant d'acheter des objets en seconde main.

Un utilisateur inscrit et connecté pourra :

- ajouter un objet à vendre
- obtenir la liste des tous les objets et filtrer cette liste en n'affichant que les objets proches de lui
- visualiser les informations détaillées d'un objet
- placer un objet dans sa liste des objets à suivre et afficher sa liste d'objets à suivre

Les différentes informations seront stockées dans une base de données. Cette base de données sera manipulée par des services distants (RPC PHP ou services web RESTfull en Java) qui renverront des informations au format texte pur ou au format JSON.

## 2 La base de données

La base de données devra stocker des informations sur les utilisateurs et sur les objets. Voici quelques exemples d'informations qu'il faudra manipuler.

### 2.1 Les utilisateurs

- Un ID
- Un identifiant (pseudo, adresse email, numéro de téléphone, etc. à vous de choisir)
- Mot de passe
- Coordonnées de l'endroit où se trouveront les objets
- etc.

### 2.2 Les objets

- Un ID
- Nom de l'objet
- Type de l'objet (bricolage, jeu vidéo, musique, etc.)
- Prix souhaité
- Descriptif de l'objet
- Photo(s)
- Etc.

## **2.3 La liste des objets à suivre**

- Un ID
- L'ID de l'utilisateur voulant suivre l'objet
- L'ID de l'objet à suivre

## **3 Cahier des charges**

### **3.1 Activité « Écran d'accueil »**

L'écran d'accueil possédera :

- un formulaire de connexion
- un bouton pour accéder à l'écran d'inscription

#### **Le formulaire de connexion**

Le formulaire de connexion contiendra :

- 2 zones de saisie : l'une pour l'identifiant<sup>1</sup> et l'autre pour le mot de passe
- un bouton de connexion

Pour se connecter, l'utilisateur doit saisir son identifiant et son mot de passe correspondant, puis cliquer sur le bouton de connexion. Si les informations saisies sont correctes, il accédera à l'écran de listing des objets.

#### **Le bouton d'inscription**

Cliquer sur le bouton d'inscription permettra d'afficher l'écran d'inscription d'un nouvel utilisateur.

### **3.2 Activité « Inscription »**

#### **Informations à saisir**

Le formulaire de saisie contiendra les éléments suivant :

- une zone de saisie pour l'identifiant
- 2 zones de saisie pour le mot de passe

Une fois les informations encodées, l'utilisateur devra cliquer sur un bouton. Si les infos saisies par l'utilisateur ne sont pas correctes, il faut afficher un message d'erreur, sinon on ajoute l'utilisateur dans la DB et on revient à l'écran d'accueil.

---

<sup>1</sup> Comme username, vous pouvez choisir un pseudo, une adresse email, un numéro de téléphone, etc.

### **3.3 Activité « Listing des objets en vente »**

Cet écran contiendra différentes parties:

- une zone dans laquelle seront affichés les objets en vente
- une zone permettant d'affiner les objets affichés en fonction de la distance
- un bouton permettant d'accéder à l'écran d'ajout d'un objet à vendre
- un bouton permettant d'afficher la liste des objets à suivre
- un zone contenant un bouton de déconnexion

#### **Les objets en vente affichés**

La liste des objets les plus proches<sup>2</sup> seront affichés. En cliquant sur un objet de la liste, une nouvelle activité sera affichée, avec les informations détaillées de l'objet sélectionné.

#### **Mise en veille ou changement d'orientation**

La liste des objets doit être rechargée (ou complétée) automatiquement chaque fois qu'on réactive l'application après une mise en veille du smartphone ou si on change l'orientation du smartphone.

#### **Recherche des objets en fonction de la distance**

Il doit y avoir moyen de modifier la distance max de recherche des objets.

#### **Bouton « Ajouter un objet à vendre »**

Si l'utilisateur clique sur ce bouton, il accédera à une nouvelle activité lui permettant d'encoder les informations d'un objet à vendre.

#### **Bouton « Liste des objets à suivre »**

Si l'utilisateur clique sur ce bouton, il accédera à une nouvelle activité dans laquelle seront affichés tous les objets suivis par l'utilisateur.

---

<sup>2</sup> A vous de fixer une distance max par défaut

### **3.4 Activité « Création d'un nouvel objet à vendre »**

Dans cette activité, un formulaire permettra d'encoder les informations de l'objet à vendre. Dans les informations à remplir, il faudra choisir le type de l'objet.

#### **Le type de l'objet**

Le choix du type de l'objet s'effectuera grâce à un wizard accessible en cliquant sur un bouton. Ce wizard sera géré par une activité développée "à la main" par le programmeur. Dans cette activité, la liste des types sera affichée. L'utilisateur devra alors choisir un type dans la liste.

L'utilisateur aura aussi la possibilité de taper le type au clavier s'il ne veut pas utiliser le wizard.

### **3.5 Activité « Informations détaillées d'un objet »**

Cette activité permettra d'afficher :

- les informations détaillées de l'objet
- le pseudo de l'utilisateur qui vend l'objet
- la distance où se situe l'utilisateur
- un bouton permettant de placer l'objet dans sa liste des objets à suivre

### **3.6 Activité « Liste des objets à suivre »**

Cette activité affiche la liste des objets que l'utilisateur est en train de suivre. En cliquant sur un objet de la liste, une nouvelle activité sera affichée, avec les informations détaillées de l'objet sélectionné.

## 4 Contraintes

### 4.1 Gestion des erreurs

Gérez un maximum d'erreurs possibles en affichant des messages d'erreur personnalisés partout où c'est possible.

Il faut vérifier les saisies de l'utilisateur et fixer quelques contraintes de saisie. Par exemple :

- les 2 mots de passe soient les mêmes
- pseudo pas trop court
- mot de passe d'un certain nombre de lettres
- nom de l'objet pas trop court
- prix positif
- aucun champ vide
- etc.

### 4.2 Ergonomie et présentation des écrans

- Soignez votre programme afin que les informations soient présentées clairement à l'écran. Il faut aussi que l'utilisation du programme soit aisée.
- La présentation des écrans de l'application doivent garder une même structure et cohérence (couleurs, polices)
- Le programme ne doit pas planter si vous changez l'orientation du device
- Le programme doit exister en version française et en version anglaise<sup>3</sup>. La langue d'affichage changera automatiquement selon la langue de l'OS.

---

<sup>3</sup> Excepté le contenu de la base de données qui, lui, restera stocké dans la langue d'encodage

## 5 Remarques

- Le projet doit être réalisé avec Android Studio 2020.3.1
- C'est un projet personnel. Cela signifie que :
  - tout bout de code qui se retrouve chez plusieurs étudiants sera sanctionné par un 0/20 chez les étudiants ayant le code commun
  - vos RPC ou services web doivent également être les vôtres
- Les fonctionnalités citées dans l'énoncé doivent toutes être réalisées
- Des points seront attribués à la personnalisation du projet. Il est donc demandé d'ajouter quelques fonctionnalités supplémentaires. Exemple :
  - connexion via Facebook
  - gestion des photos pour les objets
  - une miniature est affichée dans la liste des objets
  - possibilité d'enlever un objet de sa liste des objets à suivre
  - possibilité de voir les autres objets d'un utilisateur
  - communiquer avec un utilisateur
  - modification des informations d'un objet
  - suppression d'un objet mis en vente
  - etc.

## 6 Rapport explicatif

Un rapport (au format PDF) devra être remis. Celui-ci contiendra :

- la liste des modifications à apporter au code<sup>4</sup>, afin de faire tourner le projet sur un autre ordinateur (remarque : ces modifications doivent être minimales, l'application étant un maximum portable).
- les informations (username/password) d'un user déjà existant prêt à être utilisé
- la liste des fonctionnalités supplémentaires qui ont été développées

## 7 Remise du travail

- Le projet doit être remis dans un dépôt Moodle pour le 26 août, au plus tard à 18h00.
- Le projet (frontend et backend) et le rapport devront être remis dans un fichier compressé (format ".zip"), ce fichier étant nommé de la manière suivante: **NomPrenom.zip**.

---

<sup>4</sup> Est-il nécessaire d'installer les services distants en local ? Si oui, faut-il modifier des adresses IP pour les requêtes ? Où doit-on faire cela ?