

ANDROID - TP2

Source et pdf du cours et de ce TP :
<https://bitbucket.org/olevitt/technologies-mobiles/src>

L'objectif de ce TP est de réaliser une todo list (liste de “choses à faire”).
Ce TP fera appel aux notions suivantes :

- ListView
- RelativeLayout
- Intents
- Stockage de données

Il est fortement conseillé d'avoir le cours à portée de main et à ne pas hésiter à lire la documentation officielle <http://d.android.com> ainsi que la multitude de tutoriaux disponibles.

1 L'écran principal : affichage des tâches

- Créer un nouveau projet android avec une activity
 - Modifier le layout de cette activity pour y ajouter une ListView qui prendra toute la largeur et toute la hauteur
1. Quel composant est chargé de faire le lien entre les données et la ListView ? (on l'implémentera plus tard)
 2. On souhaite ajouter un bouton “ajouter une tâche” en dessous de la ListView. Quel layout proposez vous ?
 3. Pourquoi un LinearLayout avec orientation “vertical” ne convient-il pas ?
 - Ajouter le bouton en lui faisant prendre la place nécessaire en hauteur et toute la largeur.

2 Un écran secondaire : ajout d'une tâche

- Créer une deuxième activity (rappel : ne pas oublier de l'ajouter au manifest !)
 - Ajouter deux champs de texte (nom et commentaires) ainsi qu'un bouton “ajouter” à cette activity (libre à vous de définir le layout)
4. Cette activity n'étant pour l'instant appellable que depuis l'écran principal, quels intent-filters proposez vous ?

3 Enchaînement des écrans

3.1 Dans un sens ...

- Sur l’activity principale, récupérer les clicks sur le bouton “ajouter une tâche”
- 5. On veut lancer l’activity d’ajout d’une tâche lors d’un click sur ce bouton. Quel concept android va t’on utiliser ? (indice : il peut être soit implicite soit explicite)
- 6. Dans ce cas, est-il implicite ou explicite ?
- 7. Pensez vous qu’il faille passer des données lors de l’appel à l’activity “ajouter une tâche” ? Si oui, lesquelles ?
- Lancer l’activity d’ajout de tâche lors d’un click sur le bouton.

3.2 ...et dans l’autre

- Lors d’un click sur le bouton “ajouter” de l’activity d’ajout, on souhaite retourner à l’écran principal.
- 8. Est-il judicieux de lancer un intent vers l’activity principale ?
- En réalité, android fait une pile des activités lancées (stack). Documentation officielle sur les stacks
- En faire l’expérience en lançant un intent vers l’activity principale depuis cette activity et en appuyant ensuite sur la touche retour de l’appareil.
- Utiliser la méthode finish() de Activity pour fermer l’activity et ainsi revenir à l’écran principal lors d’un click sur le bouton ajouter

4 Place aux données !

- Créer une classe destinée à contenir les données : Item. Chaque élément de la todo list contiendra au moins un nom et une description.

4.1 Afficher les données dans la ListView

- 9. Au vu de la simplicité des données à afficher pour chaque élément (2 champs de texte), quel type d’adapter vous paraît le plus intéressant ? (Rappel : on a le choix entre ArrayAdapter, CursorAdapter et faire sa propre implémentation à partir de BaseAdapter)
- Implémenter la solution choisie en utilisant un jeu de données de test (initialiser les données à la volée)

4.2 Stocker les données

10. Quelle solution de stockage des données vous paraît le plus adaptée ? (Rappel : on a le choix entre Preferences, fichier plat, base de données SQLite et stockage distant)
11. Pour des raisons de simplicité, on va opter dans la suite pour un stockage en fichier plat (libre à vous de le remplacer par une base SQLite)
 - Créer une méthode qui écrit une liste d'éléments Item dans un fichier.
 - Créer une méthode qui instancie une liste d'éléments Item à partir d'un fichier.
 - Remplir la ListView à partir des données contenues dans le fichier.