IUT - Département Informatique

Android - 2013-2014

Android TD4

4ème séance - Base de données (suite) et communication en activités

Dans ce TD, nous allons continuer le tutoriel de création d'un gestionnaire de notes textuelles, en faisant l'exercice 2. Cet exercice couvre en particulier :

- l'utilisation de plusieurs activités dans une application, ces dernière pouvant communiquer pour échanger des paramètres ou des résultats par `Intent`[1] ;
- la création d'un menu contextuel ;
- quelques manipulations de plus concernant les bases de données ;
- quelques éléments d'interface graphique.

0a Téléchargez l'archive `td4.zip` décompressez-là dans votre répertoire de travail Android. Cette archive contient le squelette de l'exercice 2 du tutoriel en ligne, auquel nous avons apporté quelques modifications pour le rendre compatible avec le dernier SDK. Cette archive comporte aussi une version électronique de cette feuille de TD.

0b Rendez-vous sur la page du tutoriel : `http://developer.android.com/training/notepad/notepad-ex2.html` et commencez à suivre le tutoriel. Ce dernier se décompose en plusieurs étapes (Step 1, Step 2, etc.). Pensez à revenir à cette feuille de TD après chaque étape, car nous ferons souvent le point sur les notions parcourues.

Step 1 La liste des méthodes indiquées en fin de paragraphe illustre l'importance de la redéfinition de méthodes de type « callback » dans Android. Celles mentionnées ici nous permettront d'intervenir, respectivement, lors de la création du menu contextuel (on y ajoutera une commande de suppression), lors d'une sélection d'item dans ce menu (on y déclenchera l'opération en question), lors d'un clique sur une note de la liste (où on lancera l'activité secondaire pour éditer la note) et enfin lors de la fin d'exécution d'une activité qu'on a lancé (où l'on récupèrera le résultat correspondant).

Step 2 Il y a en fait un menu contextuel pour chaque item de la liste ! Grâce à l'appel à `registerForContextMenu()`, la méthode `onCreateContextMenu()` sera invoquée pour chaque item de la liste (passé sous forme de `View`).

Step 3 Le framework Android est sympa. Lorsque la liste est remplie, l'identifiant de chaque item est sauvegardé (de manière transparente), ce qui nous permet ensuite d'agir directement sur l'enregistrement correspondant en base de données lorsqu'un clique est effectué. (Note : ce type de choses ne fonctionne que si les tables de la base respectent certaines conventions de nommage, comme ici le fait d'avoir un champs '`_id`'.)

Step 4 N'oubliez pas le petit encart à droite. On peut tout aussi bien utiliser une `intent` pour lancer une activité donnée, que pour demander le lancement d'une activité quelconque sachant remplir une tâche désirée (p.ex. prendre une photo, voir une page web).

Step 5 Bien comprendre la différence entre `position` et `id`. Noter la manière dont les arguments sont passés. Noter l'ajout d'un entier (ici `ACTIVITY_EDIT`) qui identifie l'action demandée à l'activité appelée (cette dernière sachant potentiellement faire plusieurs choses).

Step 6 Visualiser le principe général d'un traitement asynchrone, c'est à dire un traitement dont le résultat nous reviendra par l'invocation d'une méthode locale. Dans le cas présent (lancement d'une autre activité), cela a pour effet d'arrêter l'exécution de l'activité appelante (car il ne peut y avoir qu'une activité s'éxecutant à un instant donné). Mais nous verrons plus tard qu'une activité peut aussi déclencher des traitements en tâche de fond tout en continuant à s'exécuter (`AsyncTask`).

Step 8 Demandez de l'aide si vous ne trouvez pas les menus dans votre version d'Eclipse/ADT.

Step 9 Point 6 : On peut créer des classes à la volée pour « implémenter » une interface donnée, sans même lui donner un nom. Cela évite d'avoir à définir une classe exprès pour cela, ou bien d'implémenter l'interface voulue directement dans votre classe principale (ici, `Notepadv2`).

Step 12 Testez votre application. Si tout fonctionne, vous pouvez passer à l'exercice 3 du tutoriel. Cet exercice consolide les connaissances vues au TD 2 (cycle de vie), en les combinant avec les fonctionalités de cette application. Vous verrez en particulier comment ne pas perdre d'information ou re-mettre l'affichage à jour au moment opportun.

1. http ://developer.android.com/reference/android/app/Intent.html