



Unité d'enseignement : [Catégorie]

Titre : Android

Objectif : Prise en main des outils de développement sous Android. Développement de quelques application de complexité croissante

Pour tous les exercices présentés, je vous conseille la consultation des deux documents pdf fournis ou bien encore la recherche de codes exemples sur internet...

Les exercices proposés sont de complexité croissante. Essayez de suivre la progression proposée. Je vous laisse le choix dans la conception et la créativité de vos IHM, aussi je ne fournis pas de capture d'écran en exemple.

1. Layouts :

Afin de prendre en main l'outil de développement et le concept des IHM sous Android, réaliser 2 ou 3 projets en concevant pour chaque projet une IHM composés des widgets couramment utilisés. Vos applications ne feront rien de particulier pour le moment. Vérifiez comment sont alignés les widgets et modifier les mises en pages en changeant ces alignements.

2. Evènement sur un bouton :

L'application que vous allez créer doit permettre la saisie du prénom de l'utilisateur dans une zone d'édition ; l'appui sur un bouton réaffiche le prénom qui a été saisi dans un widget de type « textview ».

3. Authentification :

L'application doit permettre la saisie d'un login et d'un mot de passe. L'appui sur un bouton permettra de vérifier l'authentification (login et mot de passé placés en dur dans le code). La réussite ou non de cette authentification sera affichée à l'aide d'un message de type « Toast »

4. Orientation :

Créer une IHM qui contient 1 bouton et une zone de saisie qui occupent la largeur de l'écran.

Réaliser une autre IHM (nouveau fichier XML), changer l'orientation en mode paysage et replacer les mêmes widgets qui occupent la largeur de la page.

Tester que vous avez bien le passage en mode paysage quand vous retournez le mobile (CTRL+F12 sur l'émulateur) et que les données que l'utilisateur commence à saisir se retrouvent bien dans la zone de saisie lors d'un changement d'orientation.

5. Calculette :

Réaliser une calculette simple : deux zones de saisie pour les deux termes de l'opération. Une zone d'affichage pour le résultat. Le choix de l'opération se fera à l'aide de boutons radios (prendre un « RadioGroup »).

6. Listes :

Très utilisés sur les mobiles. Vous allez créer une page qui affiche des données sous forme de liste. La liste sera composée d'une icône et d'un texte à côté (la jaquette d'un dvd et le titre du film à côté par exemple).

Les éléments de la liste seront sélectionnables et un message vérifiera que la gestion de l'évènement sur la sélection d'un élément est bien gérée.

7. **Fichiers :**

Créer une application qui permet la saisie d'un contact (nom, prénom, mail etc...) et qui enregistre ce contact dans un fichier (ou qui l'ajoute au fichier si celui-ci existe). Créer également la fonction de recherche d'un contact et qui affichera alors les détails du contact recherché.

8. **Timer :**

Cela existe déjà sur tous les mobiles mais on vous demande de concevoir une application qui permet de créer une alerte par un signal visuel et/ou sonore. Cette alerte est programmable sur 24h.

9. **Gestion du GPS :**

Votre mobile est équipé d'un GPS. On vous demande de réaliser une application qui donne les coordonnées GPS où vous vous trouvez.

10. **Accéléromètre :**

Vu à la TV : une application tournant sur smartphone utilise l'accéléromètre afin de réveiller un individu sur une plage horaire donnée mais uniquement lorsqu'il se trouve dans sa phase de sommeil léger. Cela évite la fatigue de la journée et les réveils difficiles.

Le smartphone est situé sur le lit non loin de la personne et détecte si la personne bouge (sommeil léger). Si la personne ne bouge pas c'est qu'elle est dans une phase de sommeil profond et il faudrait éviter le réveil à ce moment-là.

Commencer par réaliser une application qui donne les positions relatives en X, Y et Z du mobile. Une fois cette phase vérifiée, passer aux réglages du réveil et à la gestion de celui-ci.

11.

12. Carnet

13. Reveil avec capteur accéléromètre en sommeil léger plutôt que profond

14. Client-Serveur :

Ecrire un script php d'un

