

Dans ce TD nous continuerons à développer notre application donnant la météo des stations de ski. Ce TD est dédié à l'amélioration de quelques aspects de l'application.

→ Si vous avez accumulé du retard lors des dernières séances, demandez à un autre binôme de vous donner son application en l'état, que vous ferez re-évoluer indépendamment. Il est important, dans ce cas, de l'indiquer *dès le début de la séance* à qui vous l'avez emprunté, afin que ni eux ni vous ne soyez pénalisés. (Si vous attendez la fin de la séance pour le dire, les deux groupes seront pénalisés.)

1 Evitez les chaînes de caractères codées en dur

Toutes les chaînes de caractères statiques de votre application doivent être listés dans le fichier `res/values/strings.xml`. Y a-t-il des chaînes de caractères codées en dur dans votre code ou votre layout XML ? Si c'est le cas, un warning devrait s'afficher. Faites en sorte que toutes les chaînes de caractères statiques de votre application se trouvent dans le fichiers `res/values/strings.xml`.

2 Internationalisation

Puisque toutes les chaînes de caractères statiques de votre application sont répertoriées dans un seul fichier `res/values/strings.xml`) il est très facile de traduire l'application dans une autre langue. Faites une version en français et une autre en anglais à l'aide de ces fichiers (voir la page *Supporting Different Languages* sur developer.android.com). Changez ensuite la langue du téléphone via ses paramètres afin de tester le résultat.

3 Gérer les différentes densités d'écrans

Android tourne sur de multiples appareils qui peuvent posséder des densités d'écrans différentes. Par exemple, le Samsung Galaxy S3 a une résolution de 1280x720 pixels pour un écran de 4.8 pouces, il possède donc une très haute densité de pixels. A l'inverse, le Samsung Galaxy S2 n'a qu'une résolution de 480x800 pixels pour un écran de 4.2 pouces. Une image de 100x100 pixels paraîtra donc plus petite sur le Galaxy S3 que sur le Galaxy S2. Android offre un moyen de gérer cette variété de densité d'écrans. Android définit quatre catégories de densité d'écrans : basse, moyenne, haute et très haute densité (ldpi, mdpi, hdpi, xhdpi). Fournissez des images de résolutions différentes pour que votre UI s'adapte à n'importe quel téléphone.

Vous pouvez tester votre application dans le mode `design` du fichier `.xml` en choisissant le modèle de téléphone à utiliser (en haut, en remplaçant "Nexus 4").

4 Localisation de la station de ski sur Google Maps

On désire localiser la station de ski qui nous intéresse. Modifiez votre UI pour intégrer un bouton *Locate*. Ce bouton doit permettre de lancer Google Maps en recherchant la station de ski rentrée par l'utilisateur. Implémentez cette fonctionnalité (indice : il faut utiliser une Intent).

5 Persistance légère des données

Faites en sorte que votre application mémorise la station, de sorte à la recharger automatiquement lorsque l'application démarre la fois suivante. Le plus facile pour cela est d'utiliser des *Shared Preferences*.

6 Seconde activité de sélection de station (optionnel)

Les Intents permettent à une activité d'en lancer d'autres, soit extérieures à l'application (comme ci-dessus avec google map), soit intérieure à l'application. Créez une nouvelle activité que vous appellerez *StationListActivity* qui hérite de la classe *ListActivity*. Le but de cette activité consiste à lister un ensemble de stations de ski pour permettre à l'utilisateur d'en sélectionner une. Faites un bouton "sélectionner" depuis l'activité principale qui lancera cette activité (en utilisant `startActivityForResult()`). Dans un premier temps cette activité listera quelques noms de station en dur. (Vous aurez probablement besoin d'un *ArrayAdapter* pour l'alimenter à partir d'un tableau.) Lorsque l'utilisateur choisit une de ces stations, la seconde activité termine et la première récupère le résultat.

Une fois que ça fonctionne, faites en sorte que cette liste soit alimentée par les choix passés de l'utilisateur (d'abord limités aux choix de l'exécution courante, puis pour les plus aguerris, en utilisant une base de donnée qui survivra au redémarrage de l'application).

7 À rendre par courriel :

Une archive zip nommée `td5-nom.zip` si vous êtes seul ou `td5-nom1-nom2.zip` si vous êtes deux, où `nom1` est le nom de famille alphabétiquement le plus petit de votre binôme, et `nom2` le suivant. L'archive doit contenir :

- 1) Fichier de layout `.xml`
- 2) Fichiers strings `.xml`
- 3) Manifeste `.xml`
- 4) Fichiers sources `.java`