# Gestion du changement d’orientation du téléphone

Un téléphone Android est équipé d’un capteur permettant de définir si celui-ci est tenu en mode portrait ou paysage. L’application lancée au moment du changement d’orientation pivote en conséquent. Notre application doit donc tenir compte de ce changement et avoir une interface ergonomique pour le mode paysage également.

Le répertoire “res/layout“ contient les différents fichiers XML de notre interface. Cependant, il est possible de créer des interfaces spécifiques pour chaque type de configuration (dans notre cas, le changement d’orientation). Un répertoire “res/layout-land“ peut être créé. Il contiendra l’interface en mode paysage. On peut également créer un répertoire “res/layout-port“ pour le mode portrait. Le projet doit néanmoins contenir obligatoirement un répertoire “res/layout“ qui sera le répertoire par défaut. Dans notre projet nous avons choisi de mettre dans ce répertoire l’interface portrait et dans un répertoire “res/layout-land“, l’interface paysage.

**(layout\_paysage\_1.PNG : interface en mode paysage)**

Quelques légers changements ont été faits dans le mode paysage par rapport au mode portrait :

* La fenêtre des fonctionnalités : en portrait, cette fenêtre est placée entre la barre de titre de la chanson et la barre de progression. En paysage l’espace entre les deux est trop petit pour que tous les icones figurant sur la fenêtre des fonctionnalités s’affichent. Nous avons donc placé cette fenêtre en haut et ajouté un bouton pour fermer les fonctionnalités

**(layout\_paysage\_2.PNG : aspect de la fenêtre des fonctionnalités en mode paysage)**

* La fenêtre des paroles : nous avons modifiés cette fenêtre pour les mêmes raisons que pour les fonctionnalités. En portrait, celle-ci se ferme en appuyant sur le bouton fonctionnalités ou sur le bouton de la liste des musiques. Dans ce mode là, ces deux boutons étant cachés, nous avons rajouté un bouton “fermer“ en bas de la fenêtre.

**(paroles\_paysage.PNG : fenêtre d’affichage des paroles en mode paysage)**

### Action lors du changement de configuration du téléphone :

Certaines configurations du téléphone peuvent être modifiées en cours d’exécution (dans notre cas, le changement d’orientation). Lorsqu’un tel changement survient, Android redémarre l’activité en cours (par un appelle de la méthode onDestroy() puis de onCreate()). Il nous est donc primordial d’avoir un système de sauvegarde et de restauration de l’état de l’application avant et après le changement d’orientation.

Android fournir des méthodes permettant ces actions :

* La méthode @Override **public** Object onRetainNonConfigurationInstance() est appelé juste avant le onDestroy(). Elle permet de stocker dans un object de type Object toutes les données que l’on souhaite récupérer lorsque que le changement de configuration est effectué. Nous avons donc crée un tableau d’objects où l’on a stocké notamment le MediaPlayer contenant la musique en cours, les valeurs de début et de fin de la boucle, l’état de la lecture (play/pause et mode boucle sélectionnée), quelle fenêtre est ouverte, …
* Dans le onCreate() nous avons récupéré cet objet grâce à getLastNonConfigurationInstance()

Si celle-ci n’est pas nulle, alors on modifie les éléments de l’activité en fonction des données que l’on a stockées.

Ainsi, lorsqu’on change l’orientation du téléphone on ne remarque pas que l’activité a redémarré.