

## TP 3 - Interfaces personnalisées

*Enseignant : Adrien Humiliere*

*08/04/2015*

*A la fin de ce TP :*

- *Faire une archive contenant les projets Xcode des exercices.*
- *Envoyer l'archive à [adrien.humiliere@djit.fr](mailto:adrien.humiliere@djit.fr).*
- *Le rendu ne sera pas noté mais fera l'objet d'une correction.*
- *Si le TP est fait par plusieurs étudiants, merci de préciser les noms et adresses mail de chacun.*

### **Pour se connecter aux machines : m2sar/sarM2.**

Ces sessions sont partagées et régulièrement effacées. Ne stockez rien d'important dessus, récupérez vos fichiers et déconnectez vos sessions après usage.

Des terminaux (iPod et iPad) sont disponibles pour effectuer vos tests.

### **Note importante pour l'utilisation d'Xcode avec les terminaux :**

Faire particulièrement attention aux messages d'erreurs concernant les certificats.

**Ne jamais cliquer sur le bouton "Revoke and request".**

En cas de doute, adressez-vous à l'enseignant.

## **Exercice 1**

L'objectif de l'exercice est de personnaliser des éléments d'interface à partir d'une application où l'ensemble du modèle est déjà prêt..

Les différentes UI que vous serez amenés à réaliser pendant ce TP devront être harmonieuses et adaptées à toutes les tailles d'écrans grâce à AutoLayout. Aidez-vous des appareils disponibles et du simulateur, qui permet d'émuler toutes ces tailles.

- Récupérer le projet initial sur [adhumi.fr/teaching](http://adhumi.fr/teaching). L'ouvrir et se familiariser avec les différentes classes et le storyboard.  
L'application va simplement chercher une liste de musiques sur le profil d'un utilisateur SoundCloud et affiche ces titres dans une table view.
- Dans chaque cellule de la table view, essayer d'afficher, en plus du titre, le nom de l'artiste et la durée de la musique. Pour cela, utiliser la propriété `detailTextLabel` et les différents styles de Table View Cell dans le storyboard.
- On se rend rapidement compte des difficultés pour afficher les informations différemment de ce qui est prévu par défaut. Dans ce cas, nous devons créer une subclass de Table View Cell.

Créer une nouvelle classe héritant de `UITableViewCell`. Dans le storyboard, sélectionner le style `custom` pour la Table View Cell et indiquer comme Custom class le nom de la classe que vous venez de créer.

Dans cette Cell, ajouter les différents éléments UI que vous souhaitez voir apparaître et référencez les dans la classe associée. Afficher les données dans ces champs.

- (d) Tester le rendu de la question précédente sur différents appareils avec le simulateur. Améliorer la vue jusqu'à ce que le rendu soit bon sur chaque appareil (utiliser les contraintes AutoLayout).
- (e) Pour chaque musique, ajouter l'image d'illustration dans la Cell. Ces images sont hébergées sur un serveur distant, utiliser `UIImageView+AFNetworking` pour les récupérer.
- (f) [bonus] Au clic sur une cellule, afficher une interface permettant de lire la musique associée. Pour cela :
  - Créer un nouveau View Controller dans le Storyboard. Déclencher une action **show** sur ce View Controller au moment où la cellule est sélectionnée. Compiler : un View Controller vide est affiché au clic sur une ligne.
  - Trouver un moyen de récupérer l'objet que l'on souhaite afficher dans le View Controller affiché.
  - Réaliser dans le View Controller une interface, s'adaptant à tous types d'appareils, permettant d'utiliser un lecteur de musique (bouton Play/Pause, progression de la musique, cover).
- (g) [bonus] A l'aide de la classe `AVPlayer`, lire la musique dont l'URL est fournie dans le modèle.