

TP 2 - Architecture et networking

Enseignant : Adrien Humiliere

01/04/2015

A la fin de ce TP :

- *Faire une archive contenant les projets Xcode des exercices.*
- *Envoyer l'archive à adrien.humiliere@djit.fr.*
- *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 : Client REST

L'objectif de l'exercice (et du TP) est de réaliser un client simple pour afficher des données fournies par une API REST distante.

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.

- Lancer Xcode et créer un nouveau projet de type Master-Detail Application (iOS > Application > Master-Detail Application). Prendre le temps de se familiariser avec le code généré par Xcode.
- A l'aide de la librairie AFNetworking, récupérer les données de la requête `GET /beers` sur `openbeerdatabase.com`. Afficher le résultat de la requête dans la console.
- Créer un Model pour stocker les données récupérées de l'API. Ce Model pourrait prendre la forme d'un singleton, contenant un tableau d'objets `Beer`, avec les propriétés fournies dans l'API.
- Dans chaque cellule de la TableView, afficher les données associées à une bière (nom et brasserie). Utiliser les propriétés et styles fournis nativement par la classe `UITableViewCell`.

- (e) Au clic sur une cellule, afficher un autre Controller, contenant le nom, la description et le taux de chaque bière.
- (f) [bonus] Ajouter un mécanisme permettant d'afficher plus de données si l'utilisateur le souhaite. Au clic sur un bouton ou en arrivant en bas de liste, une nouvelle requête sera faite à l'API et les nouvelles données seront ajoutées à celles déjà présentes dans la liste.