TD 03 : TableView / Segues / Adress book

Ce TD portera sur l'utilisation des objets graphiques avancés et des connexions à la disposition des développeurs plus expérimentés sous IOS.

1. AniBook, le livre des animaux

Dans cette application l'utilisateur choisit un animal dans la liste proposée et la photo de celui-ci apparaît en bas de l'écran avec son nom.

Pour réaliser cette application à l'aide d'un storyboard vous utiliserez :

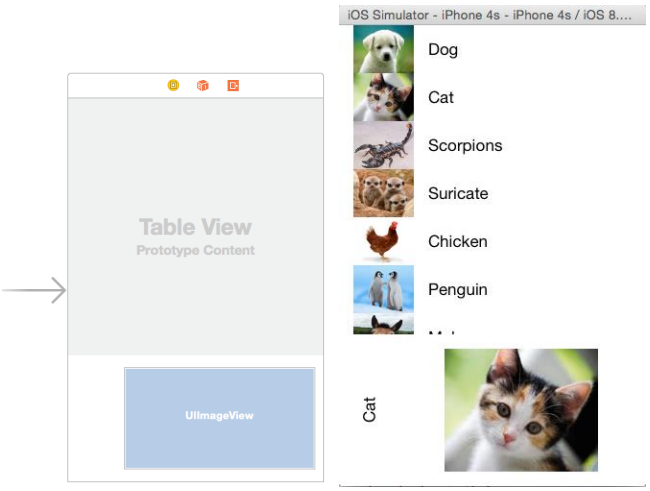
- une tableView
- une imageview
- un label orienté à 90°
- des IBOutlet et des IBAction

En Objective C vous utiliserez :

- NSDictionnay pour associer le nom et l'image de l'animal
- (label).transfom pour la rotation du label
- les méthodes issues du protocole UITableViewDelegate et UITableViewDataSource pour renseigner le nombre de lignes de la tableview (numberofrowsinsection), le contenu de la cellule courante (cellforrowatindexpath) et pour définir le comportement lors de la sélection (didselectrowatindexpath)
- les IBAction et les IBOutlet résultant du lien mis en place à partir de l'IB

Vous penserez lier le datasource et le delegate de la tableview au viewcontroller comme vu en cours.

Résultat attendu sous storyboard et sous émulateur:



2. lightMyNights, lampe torche connectée … à un autre viewcontroller

A l'aide d'un pickerview l'utilisateur choisi une couleur et après sélection, la première vue disparaît au profit d'une vue ayant le fond de la couleur et un bouton au centre pour revenir à la première vue et modifier son choix.

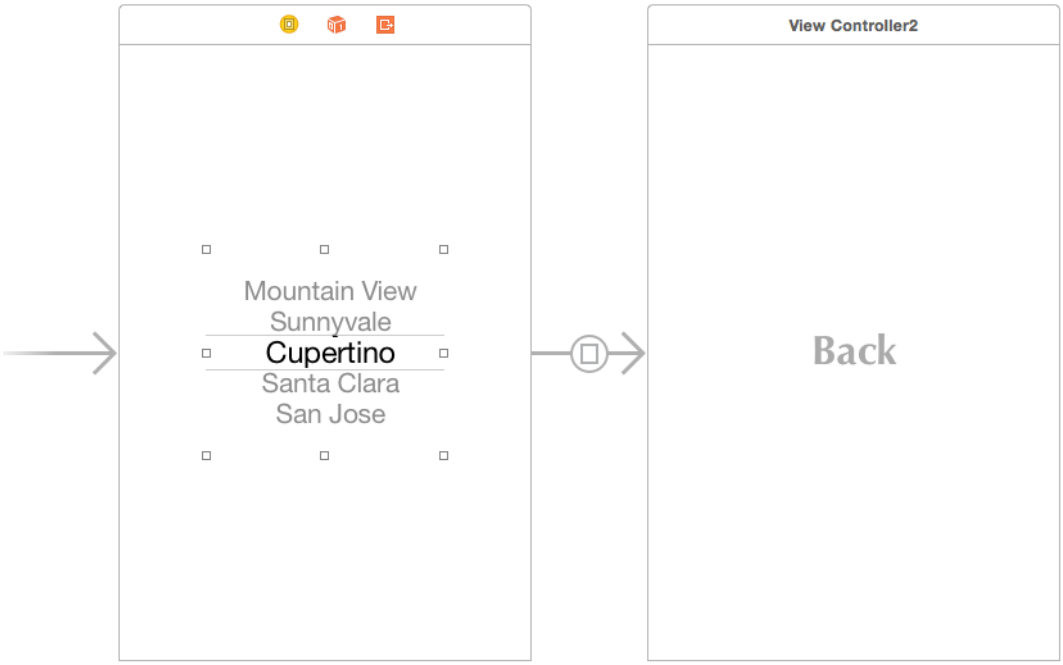
Pour réaliser cette application à l'aide d'un storyboard vous utiliserez :

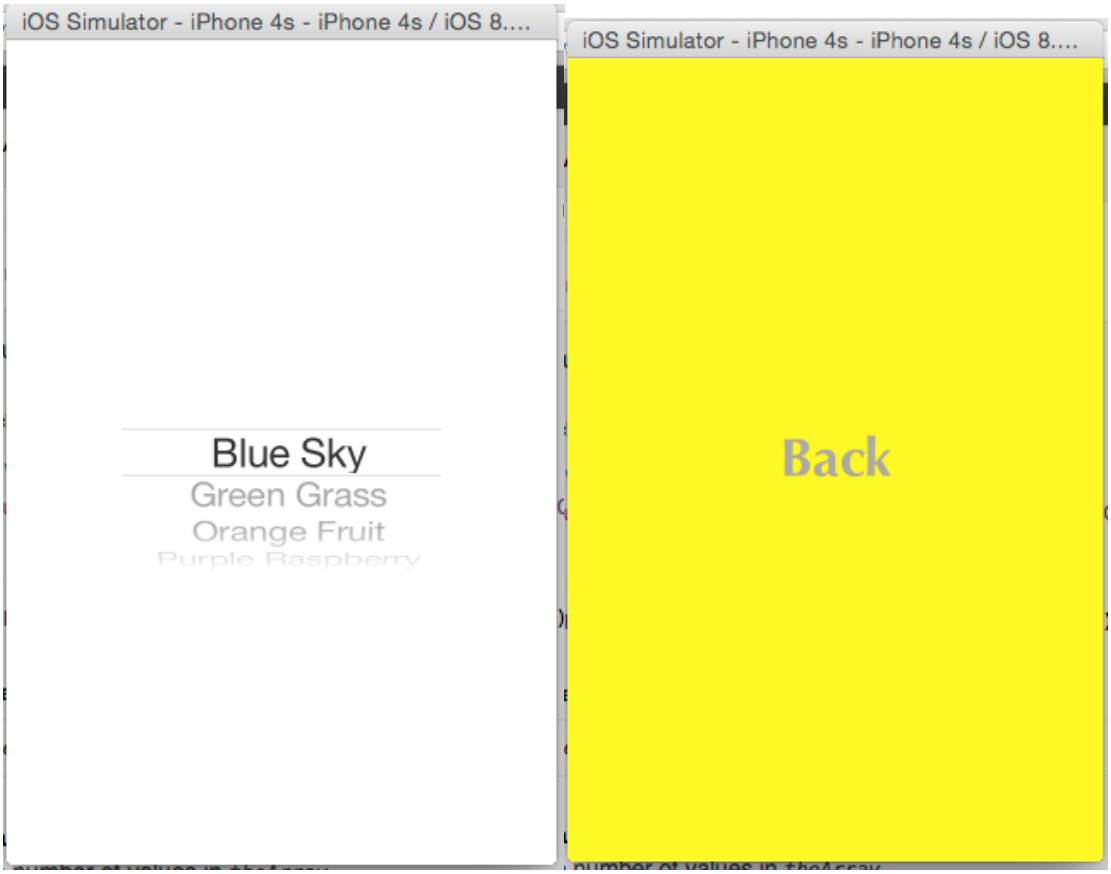
- 2 viewcontrollers
- un segue (present modally) liant la vue 2 à la vue 1
- une pickerview pour permettre le choix des couleurs
- un bouton

En Objective C vous utiliserez :

- NSArray (tableau pour stocker les couleurs)
- les méthodes issues du protocole UITableViewDelegate et UITableViewDataSource dont celle permettant de définir le nombre de ligne de colonne du pickerview, et son comportement lors d'un choix de l'utilisateur
- la méthode prepareForSegue et performSegueWithidentifier pour utiliser la connexion d'un segue
- les UI couleurs

Résultat attendu sous storyboard et sous émulateur:





3. myAddressBook, gestion du répertoire de contacts

L'utilisateur peut voir, ajouter, effacer, rechercher dans son répertoire de contact.

Pour réaliser cette application à l'aide d'un storyboard vous utiliserez :

- un label avec un nombre infini de ligne
- 4 boutons, pour voir, ajouter, effacer, rechercher
- 3 textfields pour interargir avec le clavier du iPhone

En Objective C vous utiliserez :

- des callbacks sur les boutons et les textsfields
- le framework addressBook et addressBookUI
- le protocole <ABPeoplePickerNavigationControllerDelegate>
- des types ABAddressBookRef pour définir des address books
- des types AbRecordRef pour créer de nouveaux contacts ou lire des contacts existants
- des ABRecordSetValue
- des ABAddressBookSave
- des CFArrayRef et autres types du CoreFundation, vous casterez en conséquence en bridgeant vos types (__bridge CFStringRef) pour une NSString vers une CSStringRef et (__bridge NSString) pour l'inverse

Cet exercice, s'il est construit méthodiquement, est tout à fait à la portée de tous.

Faites les choses dans l'ordre :

- dans un premier temps apprenez à afficher la liste de contact du iPhone dans un label
- puis ajouter des contacts
- puis effacer les
- puis réaliser la recherche en dernier lieu.

Dans cet ordre les étapes s'enchainent logiquement.

La liste de contact est accessible dans l'émulateur, vous pouvez la reseter dans le menu IOS Simulator/Reset content and settings. Pensez à cette étape de temps en temps pour voir vos avancées dans ce projet.

Résultat attendu sous storyboard, sous émulateur, la liste des contacts est disponible dans le simulateur: