**MobileSpace Tutorial Series**

Introduction to iOS Apps

**Project 2: MovieGuide**

**Objective:** Learning how to make a request to a web API and process the response JSON into model objects. Also work with data persistence.

**Milestones:**

1. Building the Movies Table view with custom UITableViewCell and detail view controller
2. Making the API call and getting a response object; storing it in relevant data models
3. Implementing local persistence of movies with Realm database.

**API:**

**The Movie Database** - https://www.themoviedb.org/

**The Movie Database API documentation** - docs.themoviedb.apiary.io

**Endpoint used -** http://api.themoviedb.org/3/movie/now\_playing

**Pods for MovieGuide:**

pod 'Alamofire', '~> 3.5.0'

pod 'AlamofireImage', '~> 2.5'

pod 'RealmSwift', '0.98.8'

**URL:**

https://api.themoviedb.org/3/movie/now\_playing?api\_key=a07e22bc18f5cb106bfe4cc1f83ad8ed

**Helper Code:**

pod 'Alamofire', '~> 3.5.0'

pod 'AlamofireImage', '~> 2.5'

**//MovieController**

var movie: Movie?

self.titleLabel.text = movie?.movieTitle

self.overviewLabel.text = movie?.movieOverview

if(movie?.movieBackdropPathUrl != nil) {

backdropImageView.af\_setImageWithURL(movie!.movieBackdropPathUrl!)

}

**//MovieCell**

import AlamofireImage

var movie : Movie! {

didSet {

movieTitleLabel.text = movie.movieTitle

if(movie.moviePosterUrl != nil) {

moviePosterImageView.af\_setImageWithURL(movie.moviePosterUrl!)

}

}

}

**//Movie**

let baseImageURL = <http://image.tmdb.org/t/p/w500>

var moviePosterUrl: NSURL?

var movieTitle: String?

var movieOverview: String?

var movieBackdropPathUrl: NSURL?

init(dictionary: NSDictionary) {

if let moviePosterUrlString = dictionary["poster\_path"] as? String {

moviePosterUrl = NSURL(string: baseImageURL + moviePosterUrlString)!

} else {

moviePosterUrl = nil

}

if let movieBackdropPathString = dictionary["backdrop\_path"] as? String {

movieBackdropPathUrl = NSURL(string: baseImageURL + movieBackdropPathString)!

} else {

movieBackdropPathUrl = nil

}

self.movieTitle = dictionary["title"] as? String

self.movieOverview = dictionary["overview"] as? String

}

class func movies(array: [NSDictionary]) -> [Movie] {

var movies = [Movie]()

for dictionary in array {

let movie = Movie(dictionary: dictionary)

movies.append(movie)

}

return movies

}

**//ViewController**

import Alamofire

let apiKey = "a07e22bc18f5cb106bfe4cc1f83ad8ed"

var movies: [Movie]? = []

let cell = tableView.dequeueReusableCellWithIdentifier("MovieCell", forIndexPath: indexPath) as! MovieCell

cell.movie = movies![indexPath.row]

cell.selectionStyle = UITableViewCellSelectionStyle.None

func makeAPICall() {

Alamofire.request(.GET, "https://api.themoviedb.org/3/movie/now\_playing?api\_key=\(apiKey)").responseJSON { response in

if let json = response.result.value {

if let status\_code = json["status\_code"] as? Int {

print("ERROR: Unable to hit the API with status code: \(status\_code)")

print("Got status message: \(json["status\_message"] as! String)")

}

else {

print("Connection to API successful!")

self.movies = Movie.movies((json["results"] as? [NSDictionary])!)

self.tableView.reloadData()

}

}

}

}

override func prepareForSegue(segue: UIStoryboardSegue, sender: AnyObject?) {

let cell = sender as! UITableViewCell

let indexPath = tableView.indexPathForCell(cell)

let movie = movies![indexPath!.row]

let movieController = segue.destinationViewController as! MovieController

movieController.movie = movie

}