A Master/Detail App is great for apps that need a menu/display functionality. In addition, apps where menu's grow and shrink.

1) Create a new project
   a. File > New Project
   b. Select "Master / Detail Application" > Next
   c. Fill in options screen > Next
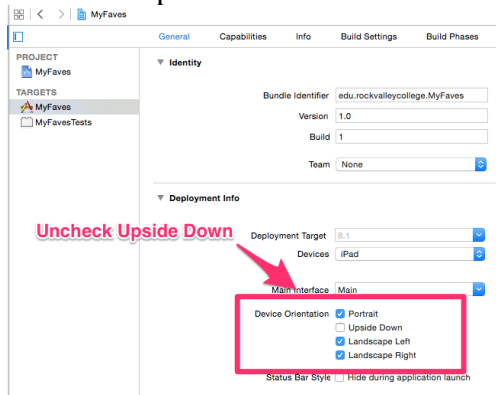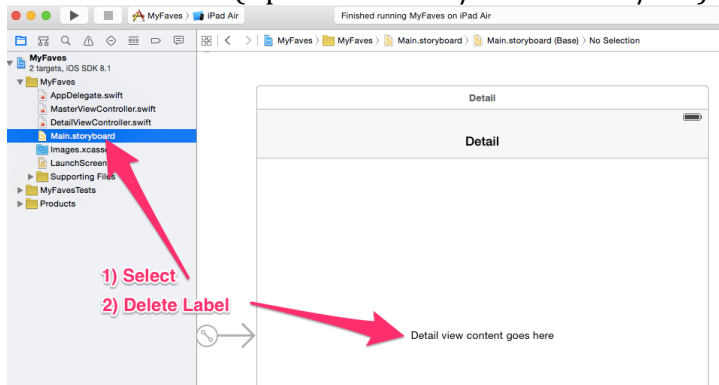

   d.
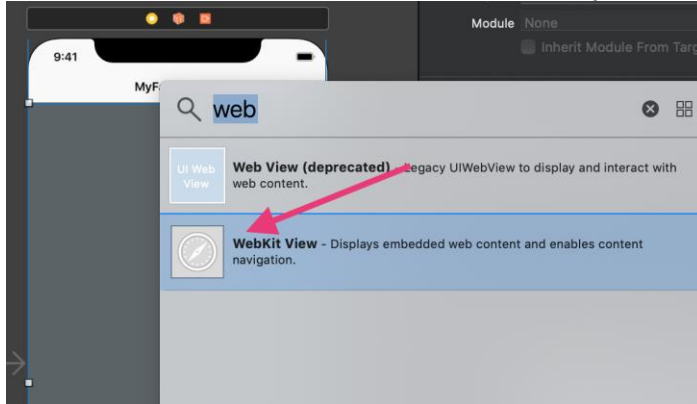   e. Select "Desktop" (You can move later) > Create
2) UnCheck Upside Down

3) Edit Detail View (tip: command +/- to zoom in/out)

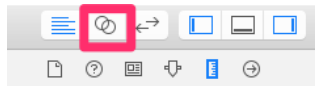4) Add WkWebView to DetailViewController (fill entire viewcontroller)
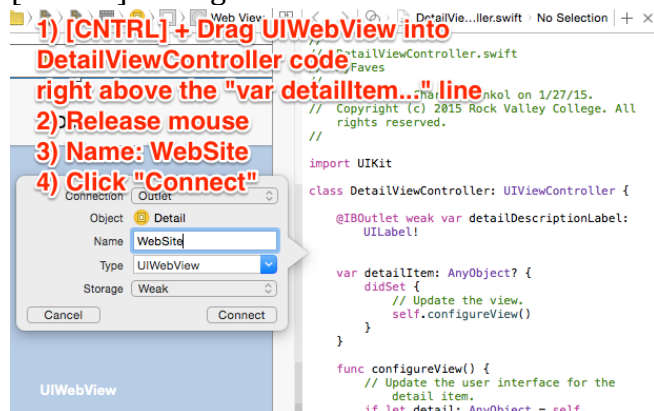

5) Add Constraints
   a. Select DetailViewController
   b. Editor > Resolve Auto Layout Issues > Reset to Suggested Constraints
6) Add Outlet for WkWebView
   a. Select WkWebView by clicking ONCE with mouse
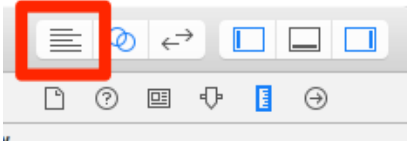   b. Select "Assistant Editor"

   c. [CNTRL] + Drag UIWEBVIEW to DetailViewController

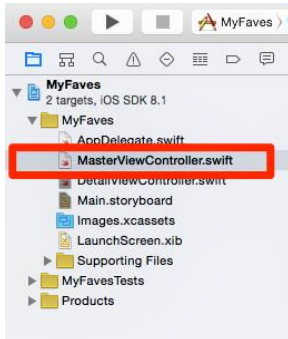d. Add import WebKit to top of DetailViewController code

```
 1  //
 2  //  DetailViewController.swift
 3  //  mf
 4  //
 5  //  Created by Charles Konkol on 9/8/19.
 6  //  Copyright © 2019 Charles Konkol. All rights reserved.
 7  //
 8
 9  import UIKit
10  //1 Add import
11  import WebKit
12
```

e. Select the "Standard Editor" again

7) Edit the "MasterViewController.swift" file
   a. Select "MasterController.swift file under Project Navigator

   b. See MasterViewController.swift code: MasterViewController.swift
   c. Read and Complete 10 Steps in Comments
8) Edit "DetailViewController.swift" file
   a. Select DetailViewController.swift file under Project Navigator
   b. See DetailViewController.swift code: DetailViewController.swift
   c. Edit Code (Read and Complete 5 Steps in Comments)
9) Create group folder named images
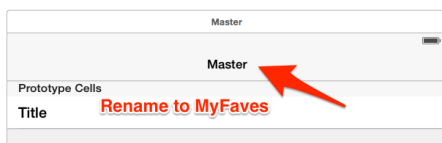10) Add images to that group (check copy items)
11) Rename Titles for Master/Detail Views
    a. Select Main.StoryBoard
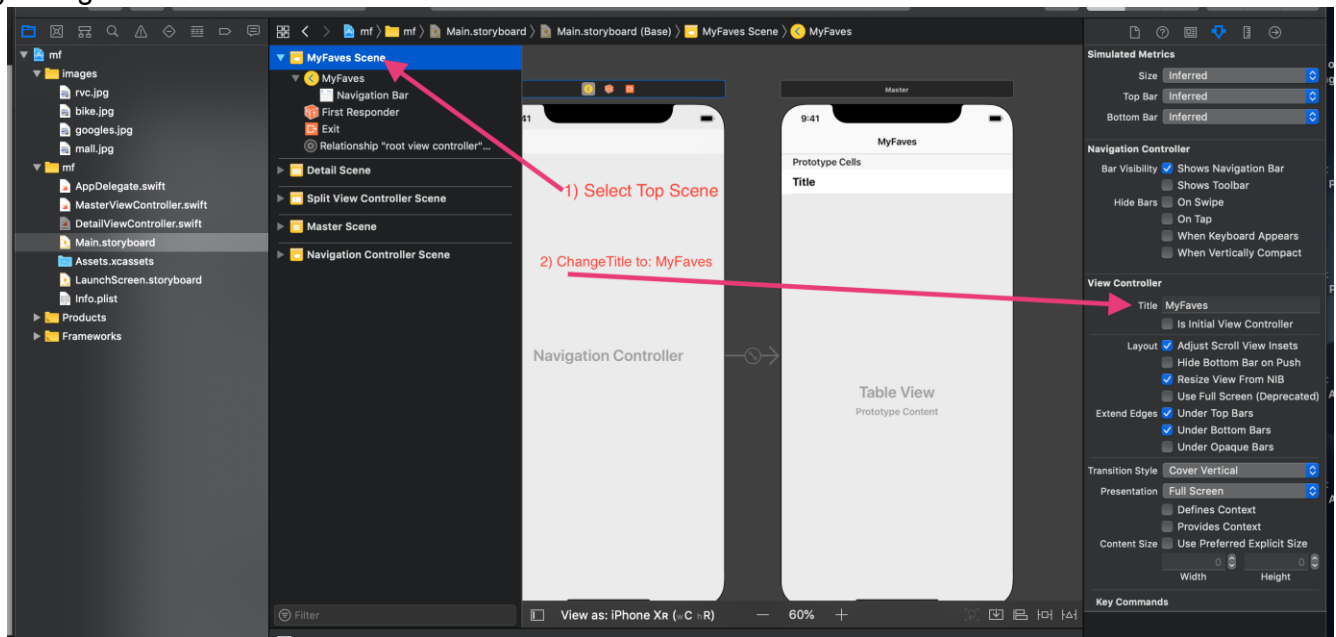   b. Select Detail View & Rename "Detail" to "MyFaves Detail"

   Detail
   Rename to MyFaves Detail

    c. Select Master Detail View & Rename "Master" to "MyFaves"

   Master
   Master
   Prototype Cells
   Title   Rename to MyFaves

12)Change title of Master Nav

13) Add below code to info.plist

Transport Security

Available from iOS 9.0

As of iOS 9.0, you can't work with HTTP web data by default, because it's blocked by something called App Transport Security that effectively requires data to be transmitted securely. If possible, you should switch to HTTPS and use that instead, but if that's not possible for some reason – e.g. if you're working with a third-party website – then you need to tell iOS to make exceptions for you.

Note: the very fact that iOS calls these "exceptions" does imply the exception option may go away in the future.

Exceptions be defined per-site or globally, although if you're going to make exceptions obviously it's preferable to do it for individual sites. This is all set inside your application's Info.plist file, and this is one of the very few times when editing your plist as source code is faster than trying to use the GUI editor in Xcode.

So, right-click on your **Info.plist** file and choose **Open As** > **Source Code**.

Your plist should end like this:

```
</dict>
</plist>
```

Right before the </dict> line, I'd like you to paste the below code and save!

```xml
<key>NSAppTransportSecurity</key>
  <dict>
      <!--Include to allow all connections (DANGER)-->
      <key>NSAllowsArbitraryLoads</key>
      <true/>
</dict>
```