

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

Choose options for your new project:

Product Name:

Organization Name:

Organization Identifier:

Bundle Identifier:

Language:

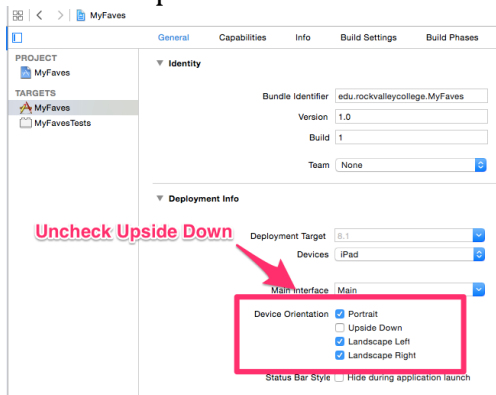
Devices:

☐ Use Core Data

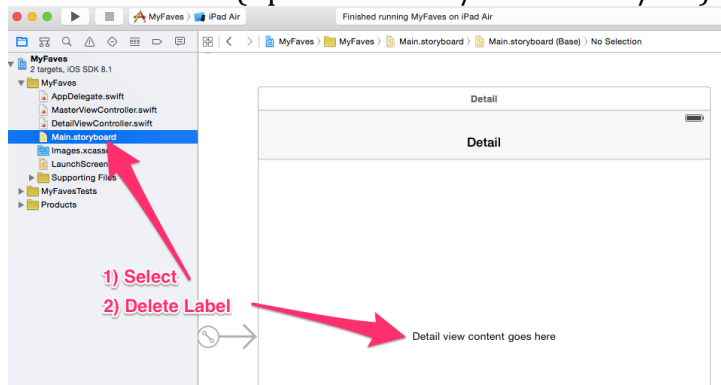
Cancel Previous Next

- d.
- e. Select "Desktop" (You can move later) > Create

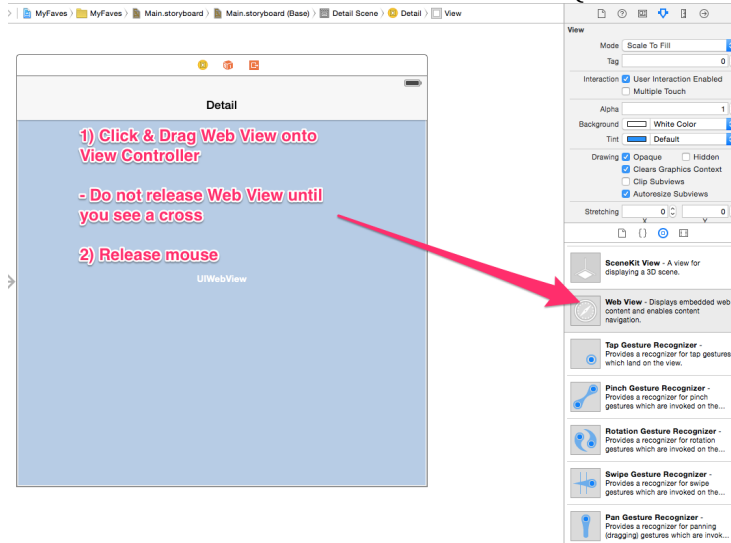
2) Uncheck Upside Down



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

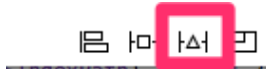


4) Add UIWebView to DetailViewController (fill entire viewcontroller)



5) Add Constraints to UIWebView

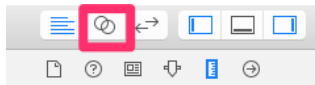
- Select UIWebView by clicking ONCE with mouse
- Select "Resolve Auto Layout Issues"



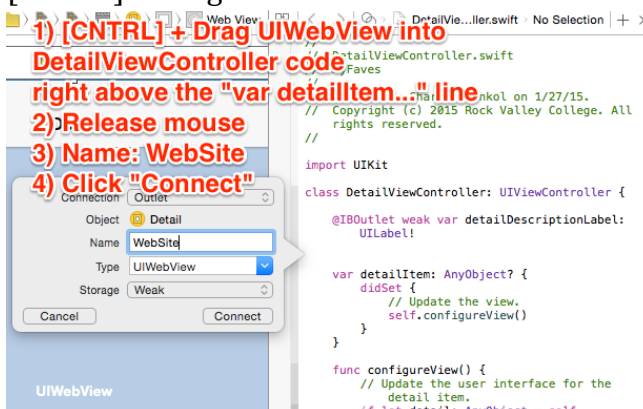
- Under "Selected View" Choose & Select "Reset to Suggested Constraints"

6) Add Outlet for UIWebView

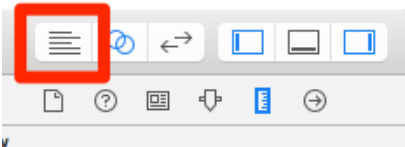
- Select UIWebView by clicking ONCE with mouse
- Select "Assistant Editor"



- [CNTRL] + Drag UIWEBVIEW to DetailViewController

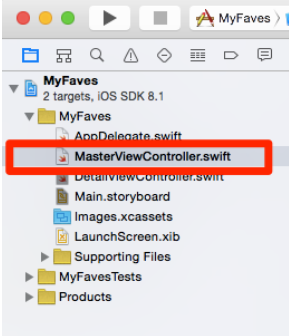


- d. Select the “Standard Editor” again



7) Edit the “MasterViewController.swift” file

- a. Select “MasterController.swift file under Project Navigator



- b. See MasterViewController.swift code: [MasterViewController.swift](#)

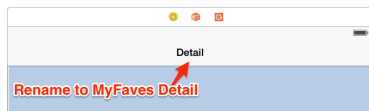
- c. Edit Code (Read and Complete 10 Steps in Comments)

8) Edit “DetailViewController.swift” file

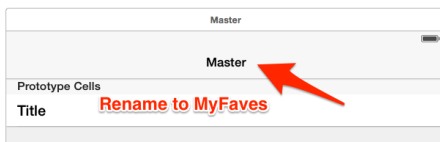
- a. Select DetailViewController.swift file under Project Navigator
- b. See DetailViewController.swift code: [DetailViewController.swift](#)
- c. Edit Code (Read and Complete 2 Steps in Comments)

9) Rename Titles for Master/Detail Views

- a. Select Main.StoryBoard
- b. Select Detail View & Rename “Detail” to “MyFaves Detail”



- c. Select Master Detail View & Rename “Master” to “MyFaves”



10) Add 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 this and save!

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