Helpful Information…

\* Images for iOS should always be PNG. Xcode does not accept any other file type.

\* Helful when creating Icons for both iOS and Android apps. Upload Image and this site will take your image and email you a zipped file with all the icons you need for all devices for free.

<https://makeappicon.com/>

Launch Screen sizes

\* iPhone Portrait iOS 8,9

Retina HD 5.5 (iPhone 6 Plus) - 1242px x 2208px - Default-736h@3x.png

Retina HD 4.7 (iPhone 6) - 750px x 1334px - Default-667h@2x.png

\* iPhone Landscape iOS 8,9

Retina HD 5.5 (iPhone 6 Plus) - 2208px x 1242px - Default-Landscape-736h@3x.png

\* iPhone Portrait iOS 7-9

2x (iPhone 4) - 640px x 960px - Default@2x.png

Retina 4 (iPhone 5) - 640px x 1136px - Default-568h@2x.png

\* iPhone Portrait iOS 5,6

1x (iPhone 3) - 320px x 480px - Default.png

2x (iPhone 4) - 640px x 960px - Default@2x.png

Retina 4 (iPhone 5) - 640px x 1136px - Default-568h@2x.png

Navbar Icons

\* Icons for navbars are two sizes:

60x60px for Retina

30x30px standard

\* This site lets you search and download Icons for free with the pixel size you need

https://icons8.com/web-app/new-icons/all

Let's begin...

-Open Xcode

-Create a new Xcode Project

-Select Single View Application > Next

Product Name: "Name of your project"

Team: Is your development team (You might need to create an appleID for this)

Language: Swift

Devices: Universal > Next

-Place where you want to save it > Create

Most of your layout will happen in the Main.storyboard

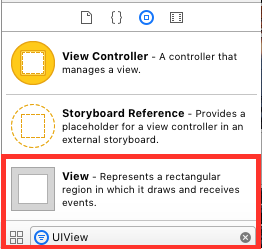
You will find a view controller and yout main story board when you select it.

Creating a Navbar with a Scrollview:

Top of toolbar:

Editor > Embed In > Navigation Controller

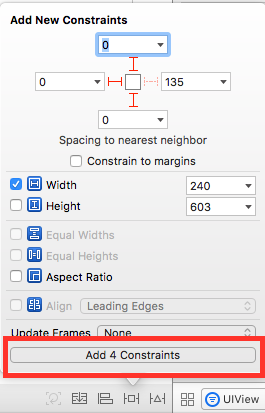
Drag UIView to View Controller



add constraints

you want it to stick to the side, top and bottom of your screen

and you'll want to add the width of your view (240px exp.)



add Scroll View (just like you did with the UIView)

Macintosh HD:Users:mined_minds:Desktop:Screen Shot 2017-03-29 at 10.25.20 AM.png if you select the circle icon at the top right it will open up the viewController.swift file along side your main.storyboard.

control drag scrollview to viewController.swift file

-place under your class viewController

-connection: Outlet

-name: "name of view" (use camel casing for titles)

-Type: UIScrollView

-Connect

Now you are able to manipulate the UIScrollView

Add under viewDidLoad :

name of view.contentSize.height = 1000

(this sets the height of the scroll)

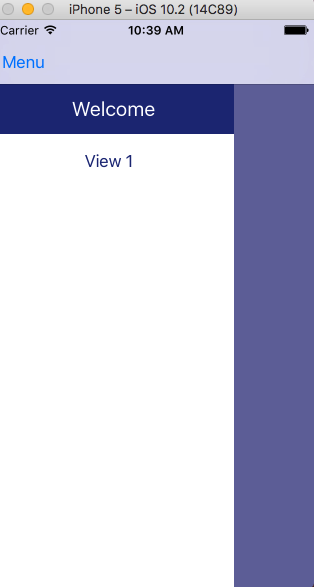
Let's add Labels and buttons:

add Bar Button Item for "Menu" action

a suggestion if you add color to the main view it’s easier to see the scrollview.

drag label

drag buttons

Macintosh HD:Users:mined_minds:Desktop:Screen Shot 2017-03-29 at 10.38.49 AM.png Select what phone you want to see in on and press play!! 

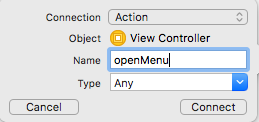
Open and close scrollView/navbar:

control drag view.leading = leading (constraint) to viewController.

This will become outlet

name it leadingConstraint

link bar button item "Menu" by control drag into viewController and make it an action.



add the following code:

set var to...

var menuShowing = false

if (menuShowing) {

leadingConstraint.constant = -240

UIView.animate(withDuration: 0.3, animations: {

self.view.layoutIfNeeded()

})

}

else {

leadingConstraint.constant = 0

UIView.animate(withDuration: 0.3, animations: {

self.view.layoutIfNeeded()

})

}

menuShowing = !menuShowing

}

Macintosh HD:Users:mined_minds:Desktop:Screen Shot 2017-03-29 at 10.38.49 AM.png check it out!! You should have a menu that opens and closes now when you press the “Menu Button” you have created.

Shadow effect to the navbar for demention:

control drag view that scrollView is on to viewController

make it a new outlet and add code to viewController

shadowView.layer.shadowOpacity = 1

shadowView.layer.shadowRadius = 6

Creating a Webview:

On your main view…

drag webView to main view

add constraints to top, leading, trailing and bottom

Control drag "webView" to ViewController and make new outlet

to display a url use the following code:

let url = URL(string: "https:// ")

nameOfWebOutlet.loadRequest(URLRequest(url: url!))

This will be placed below the viewDidLoad function

note to self: add to p.list

App Transport Security Settings

Allow Arbitrary Loads >Boolean > Yes

(this allows you to display http and https urls)

Linking buttons to URLs:

control drag button to viewController and add the url code as above

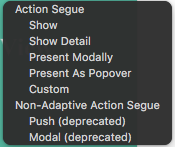
let url = URL(string: "https:// ")

webView.loadRequest(URLRequest(url: url!))

Lets link the button to a new viewController.

Drag ViewController to Main StoryBoard

Control drag button to new view and select SHOW

 Now if you select that button it will go to the new View

Creating LaunchScreen:

Select Assets.xcassets folder + sign adds a launchImage option

add pictures you have sized. On the right you will see settings

you can select the phones you want to target.

info plist:

add Launch Image > "name of your launch screen"

LaunchStoryboard:

set to "Initial view controller"

General

App Icon Source > AppIcon

Launch Image Source > LaunchImage

Icon set up:

SELECT Assets.xcassets folder and drag the pixel icons you have created and place them in the correct size