

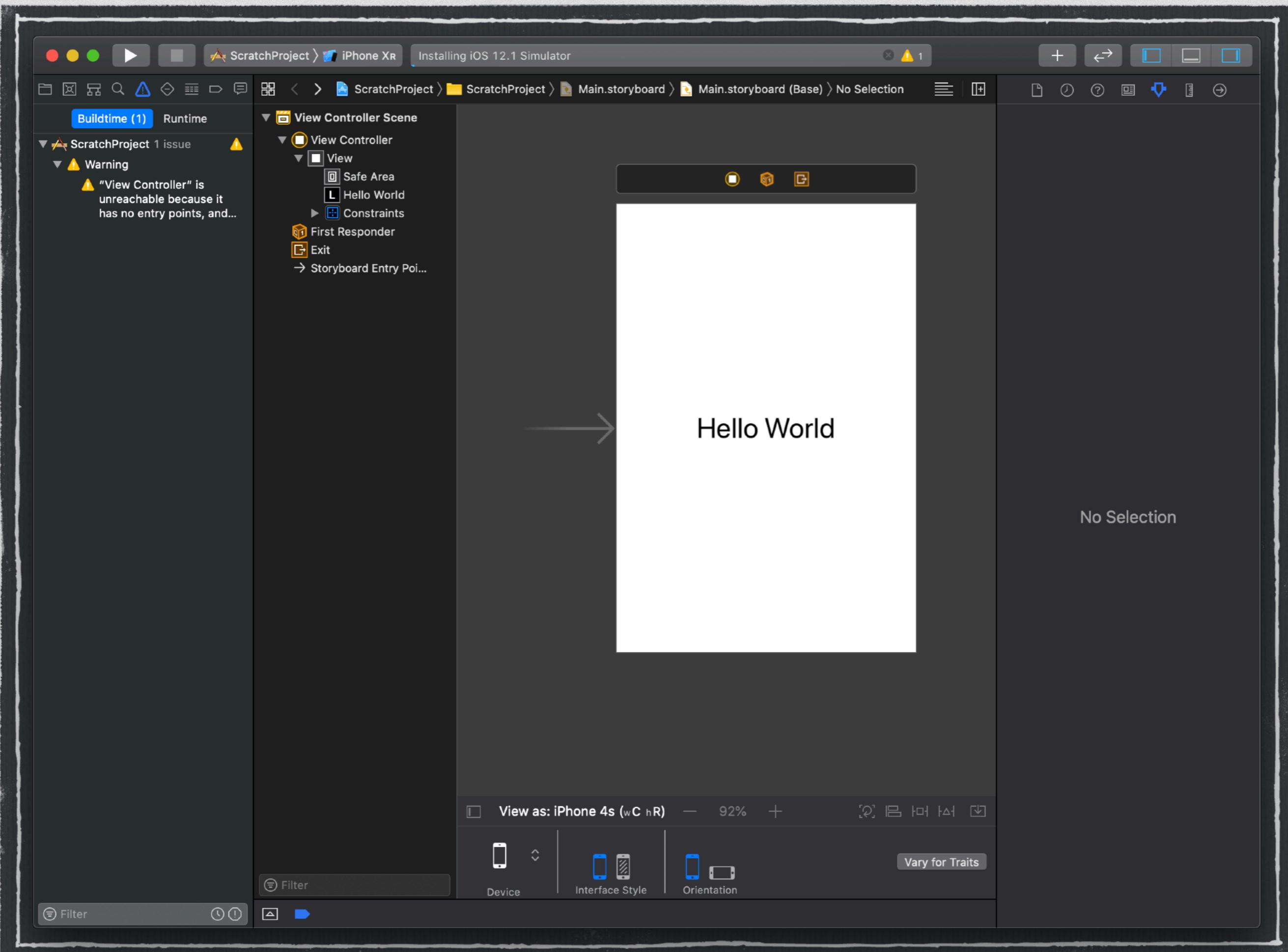
Splitting Views

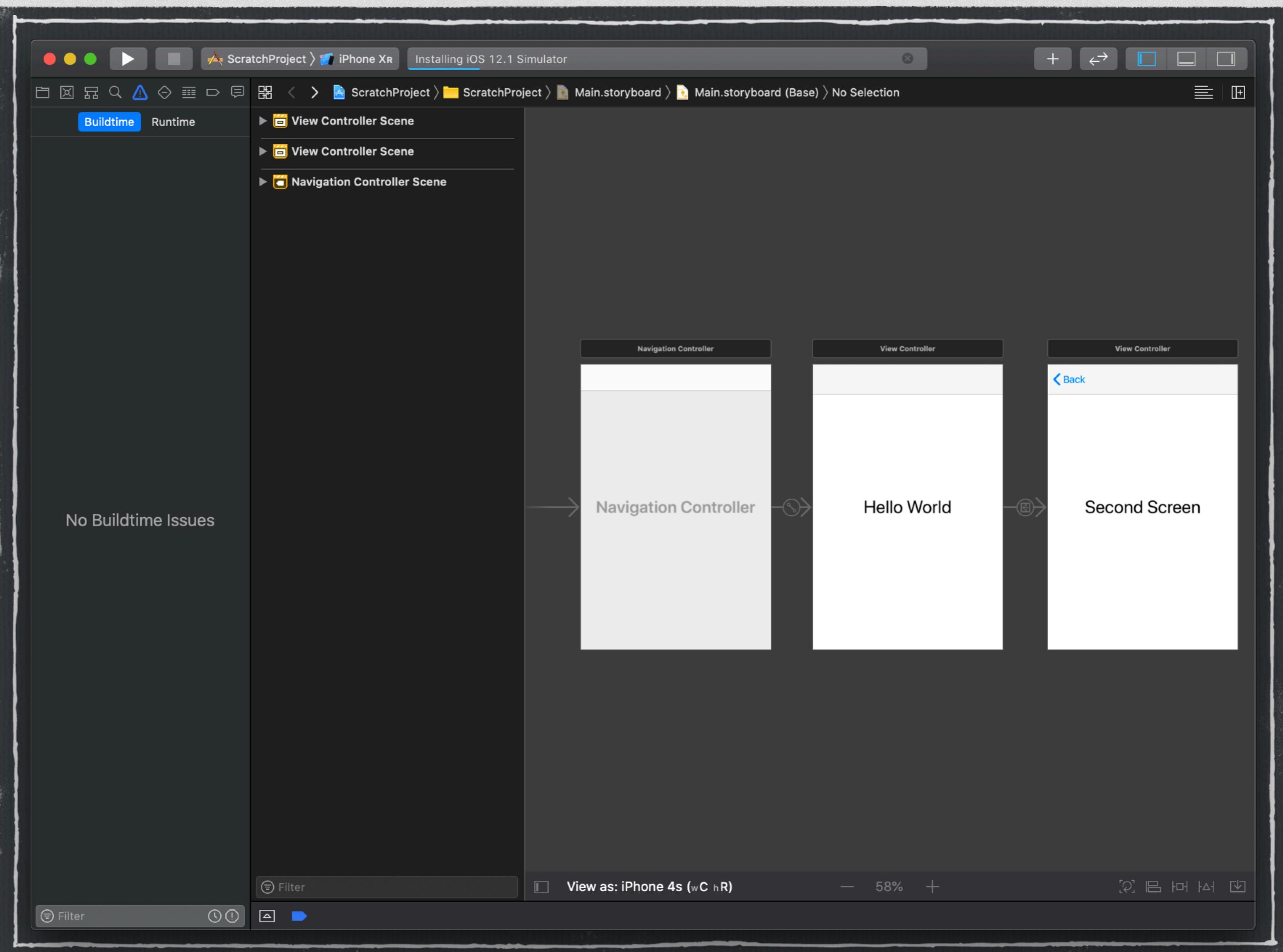
The Nitty Gritty of Container Views

Calvin Chestnut









View vs ViewController

UIViewController

`viewDidLoad()`

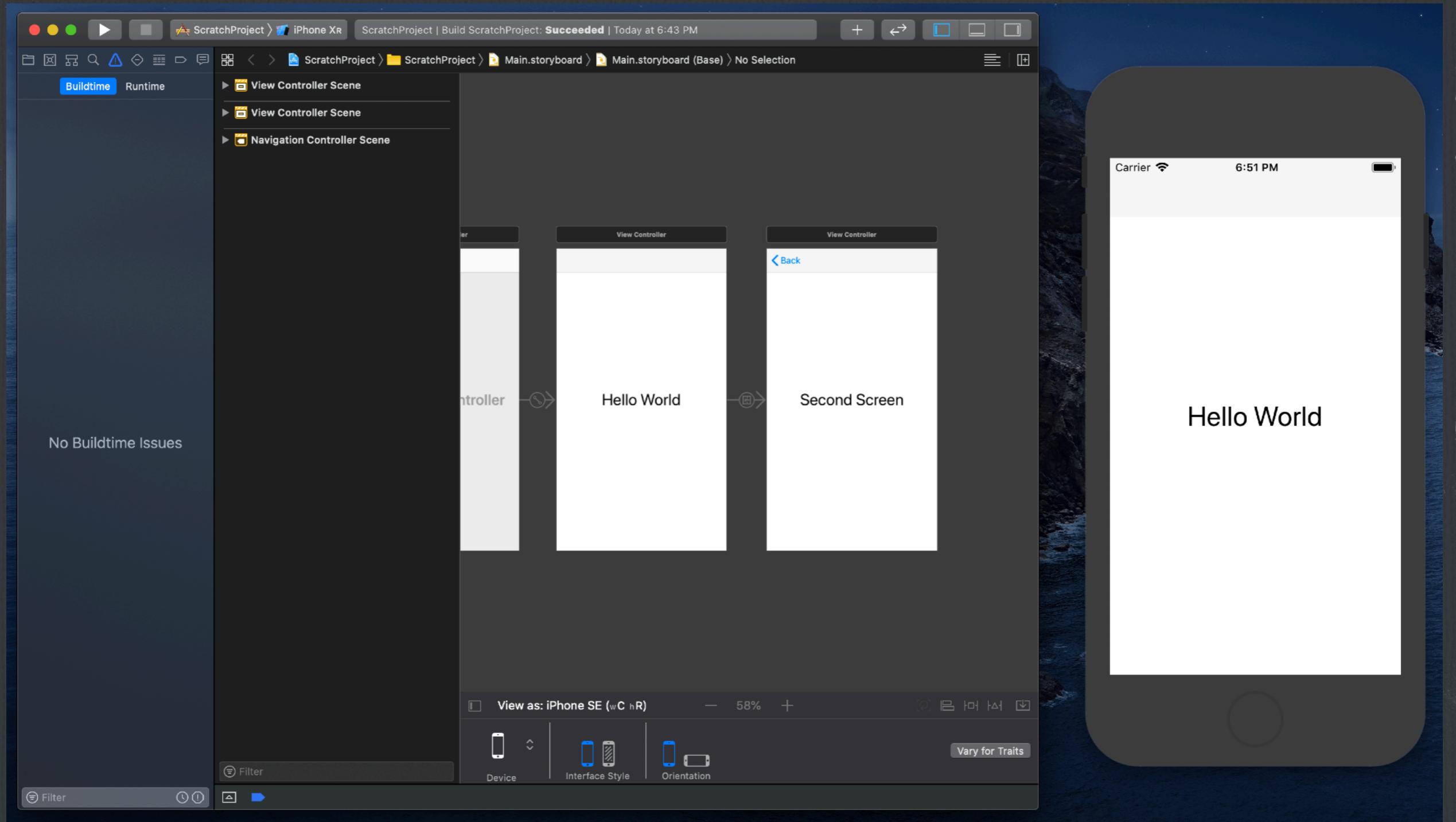
`viewWillAppear(_:)`

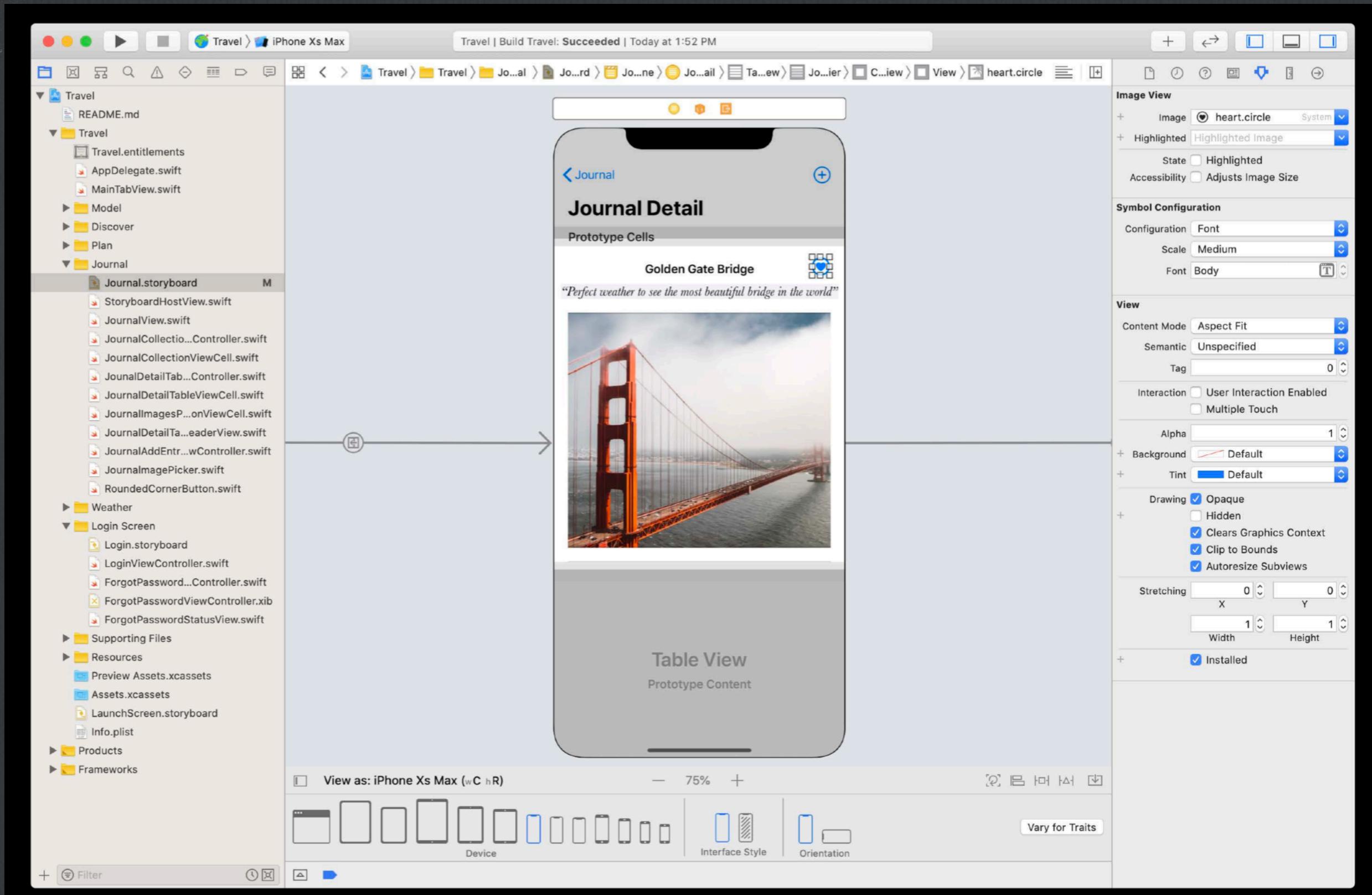
`viewWillDisappear(_:)`

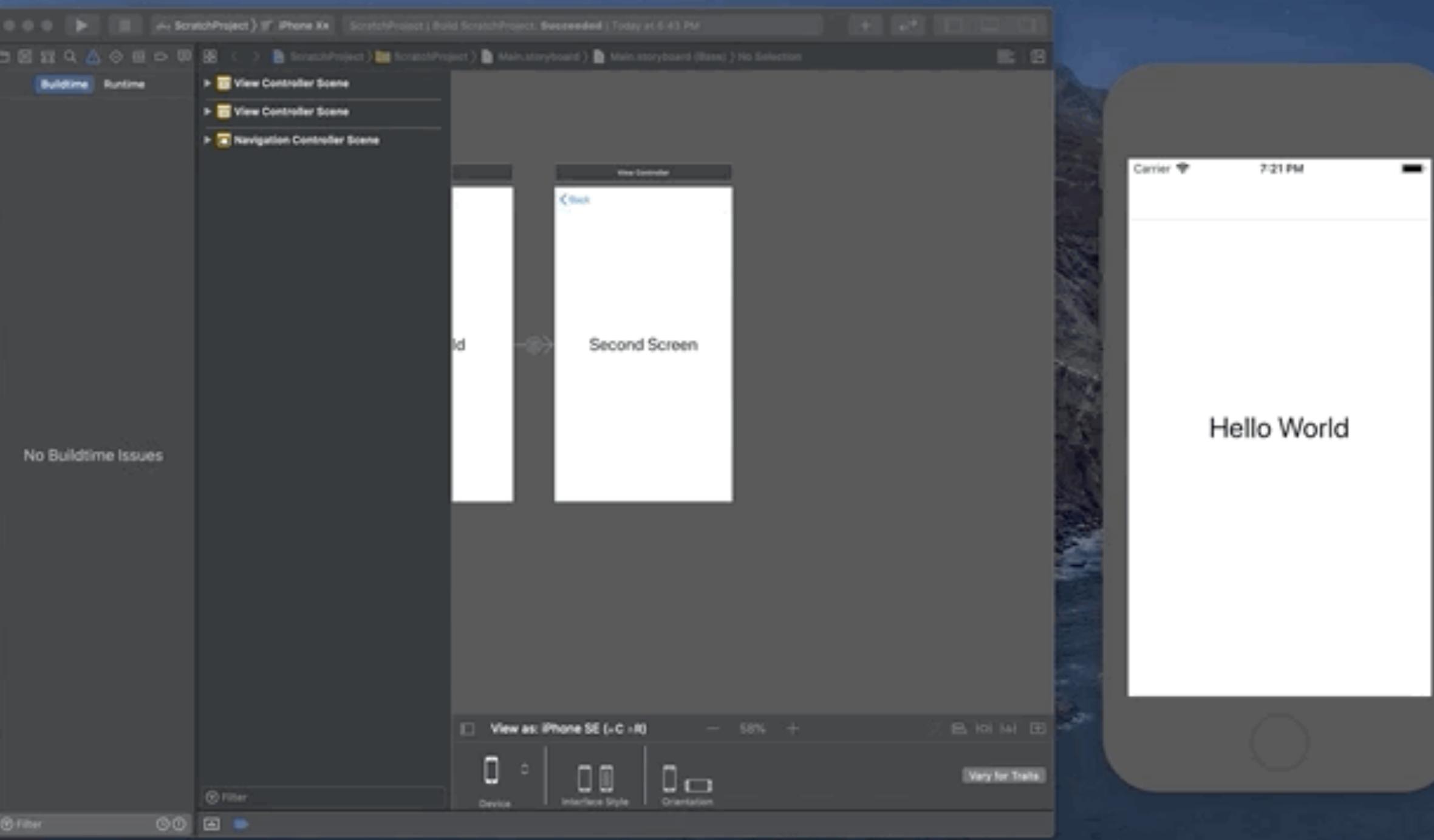
`didReceiveMemoryWarning()`

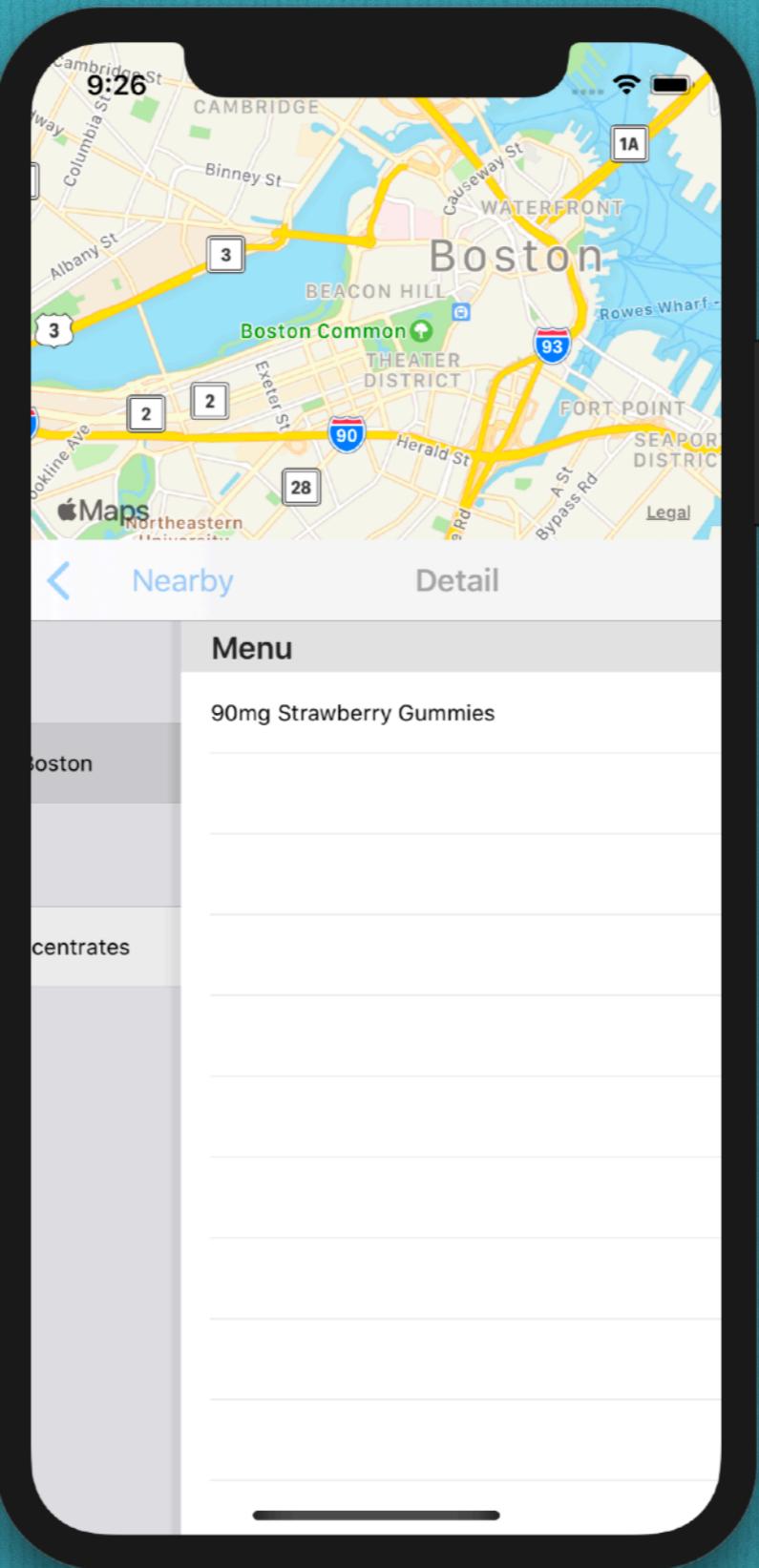
UIView

`draw(_:)`

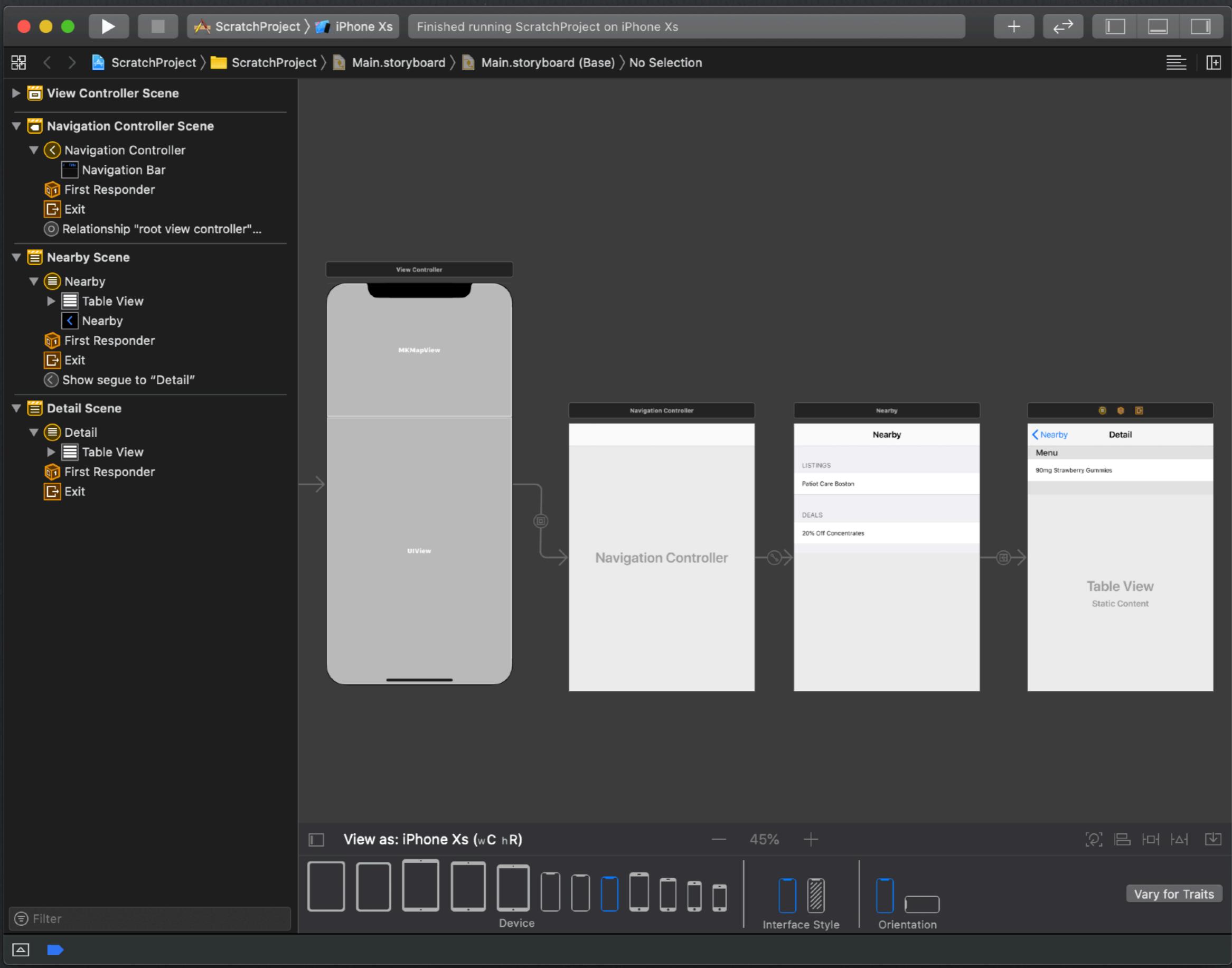


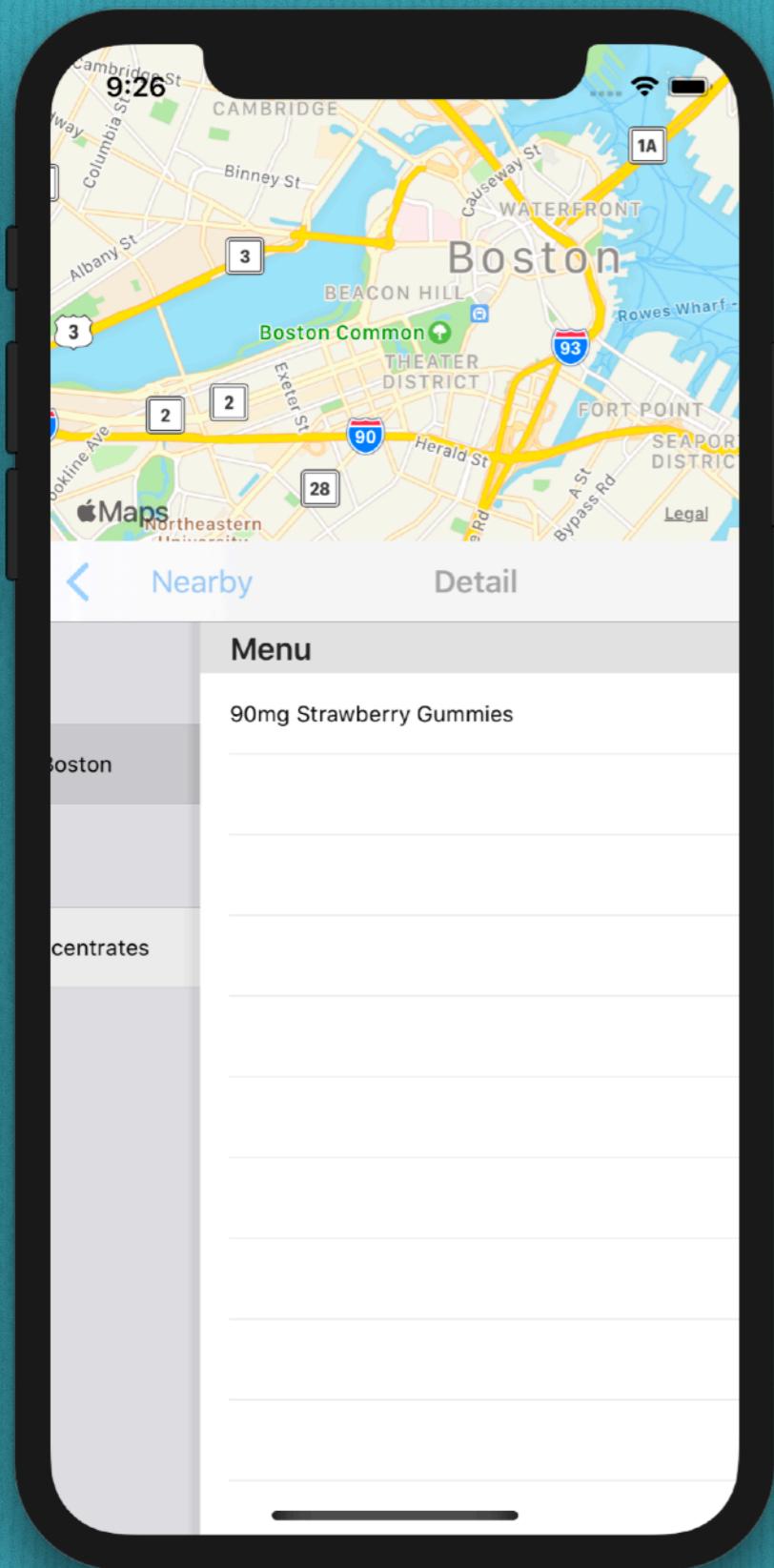






iPhone Xs — 13.0

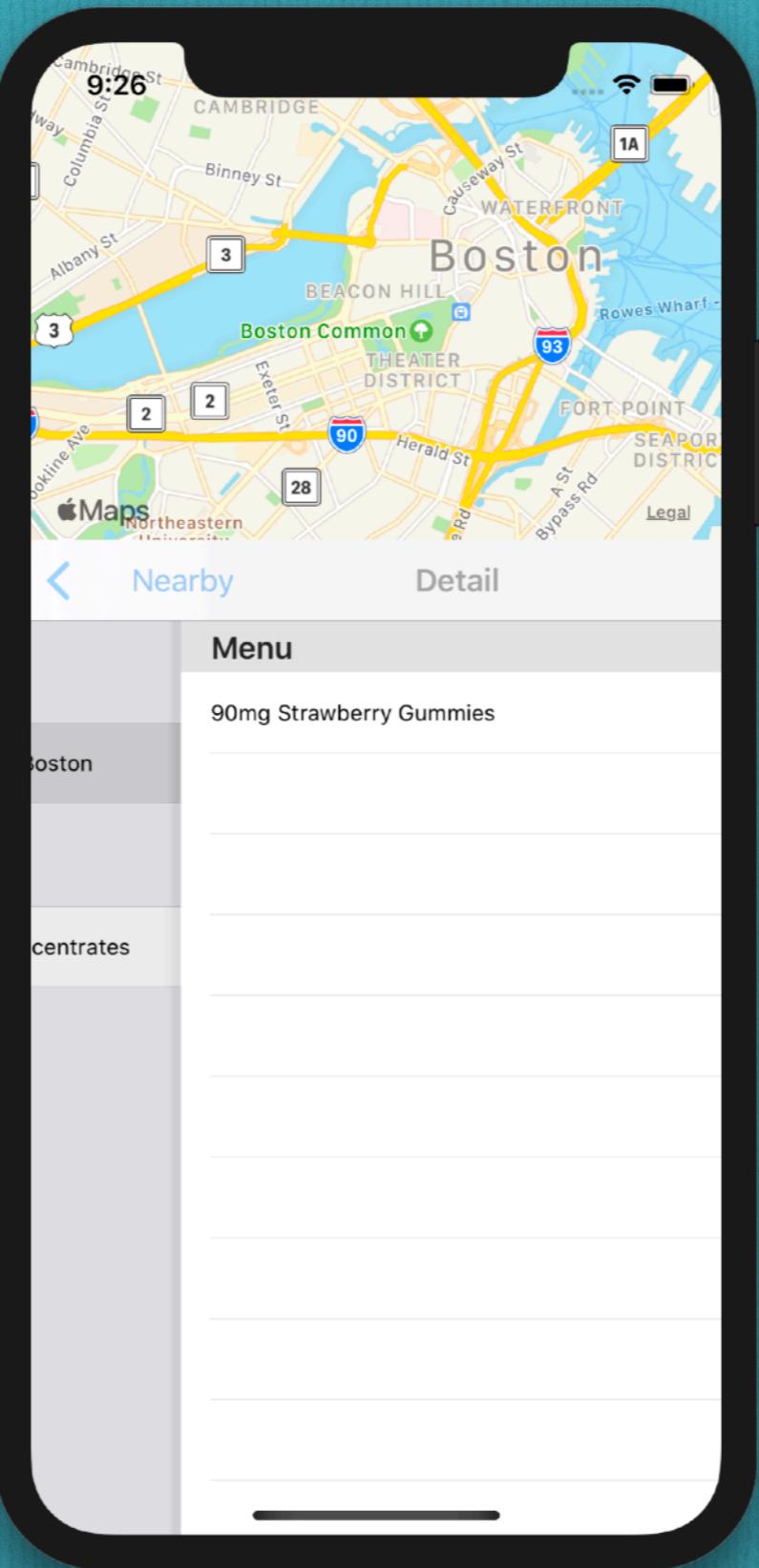




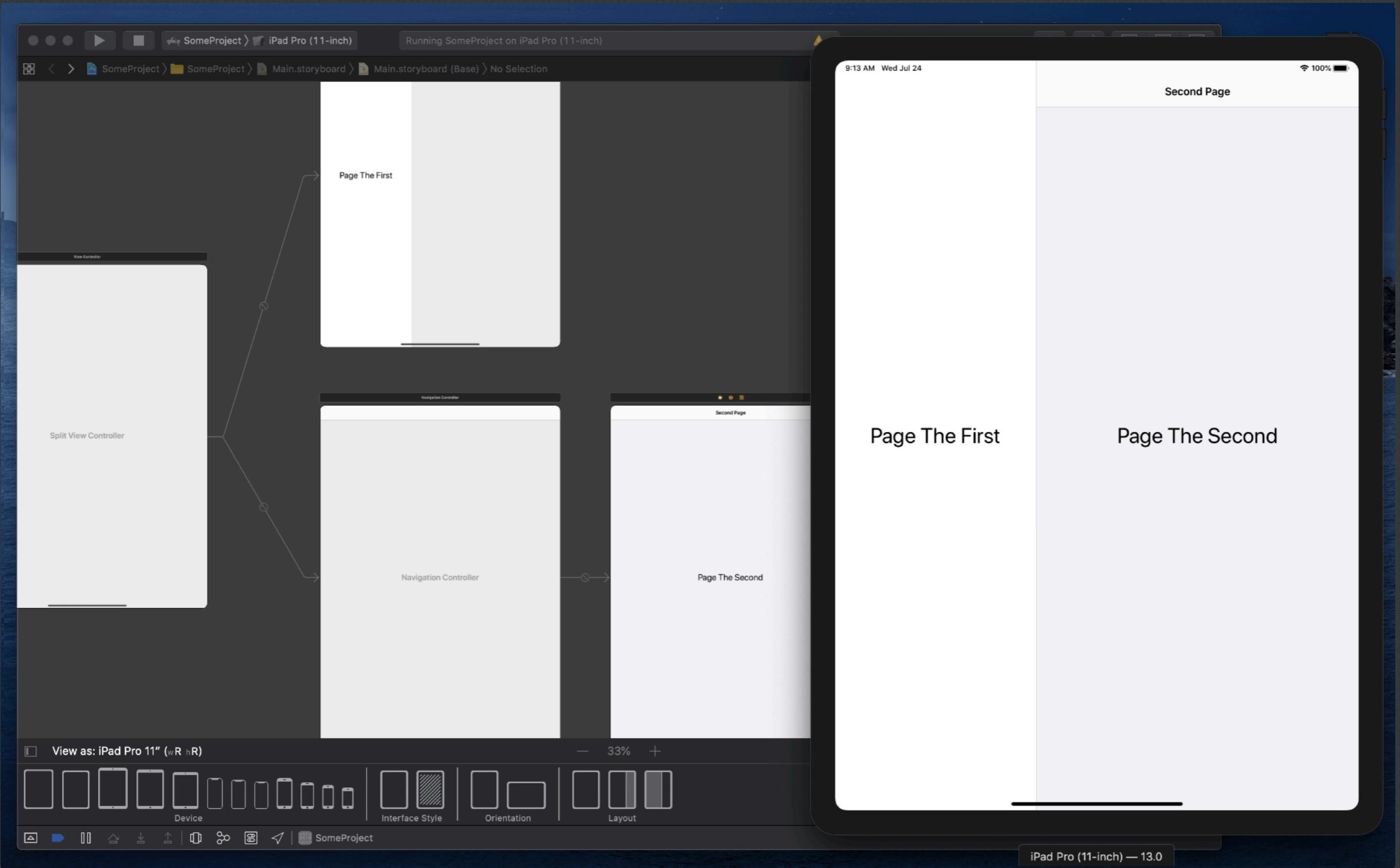
Storyboard Container Views Embed Segues

UIViewController

`addChild(_:)`
`willMove(toParent:)`
`didMove(toParent:)`



iPhone Xs — 13.0

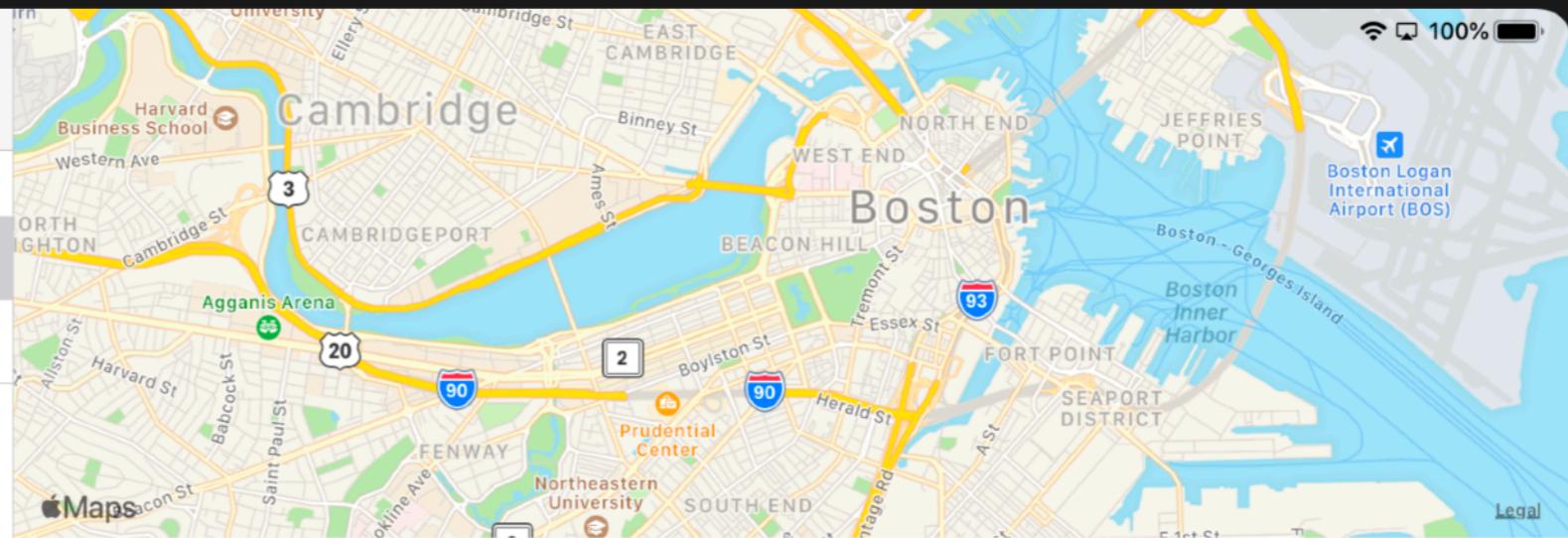


7:44 AM Fri Jul 12

Green

Nearby

Search



Nearby

LISTINGS

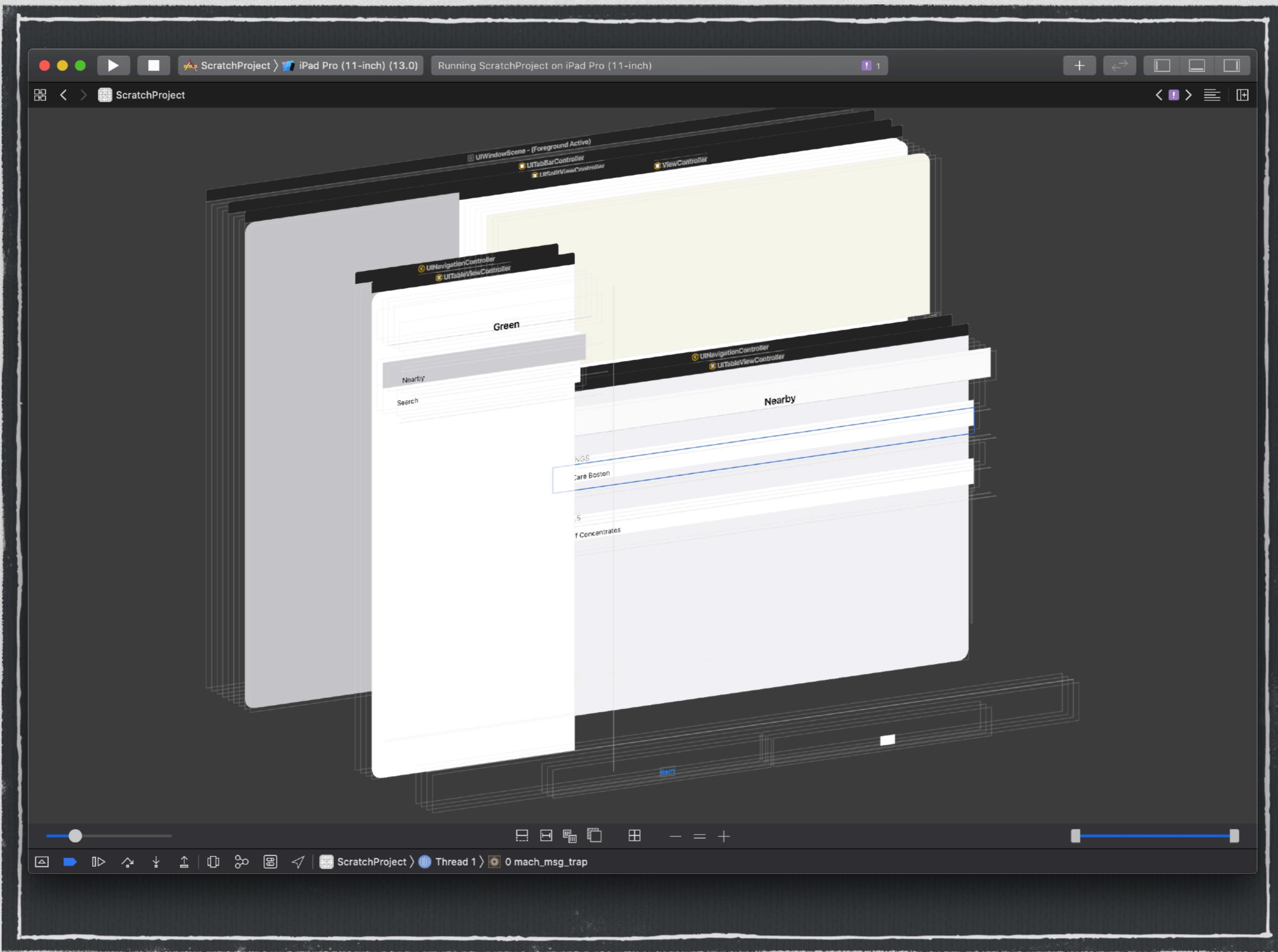
Patiot Care Boston

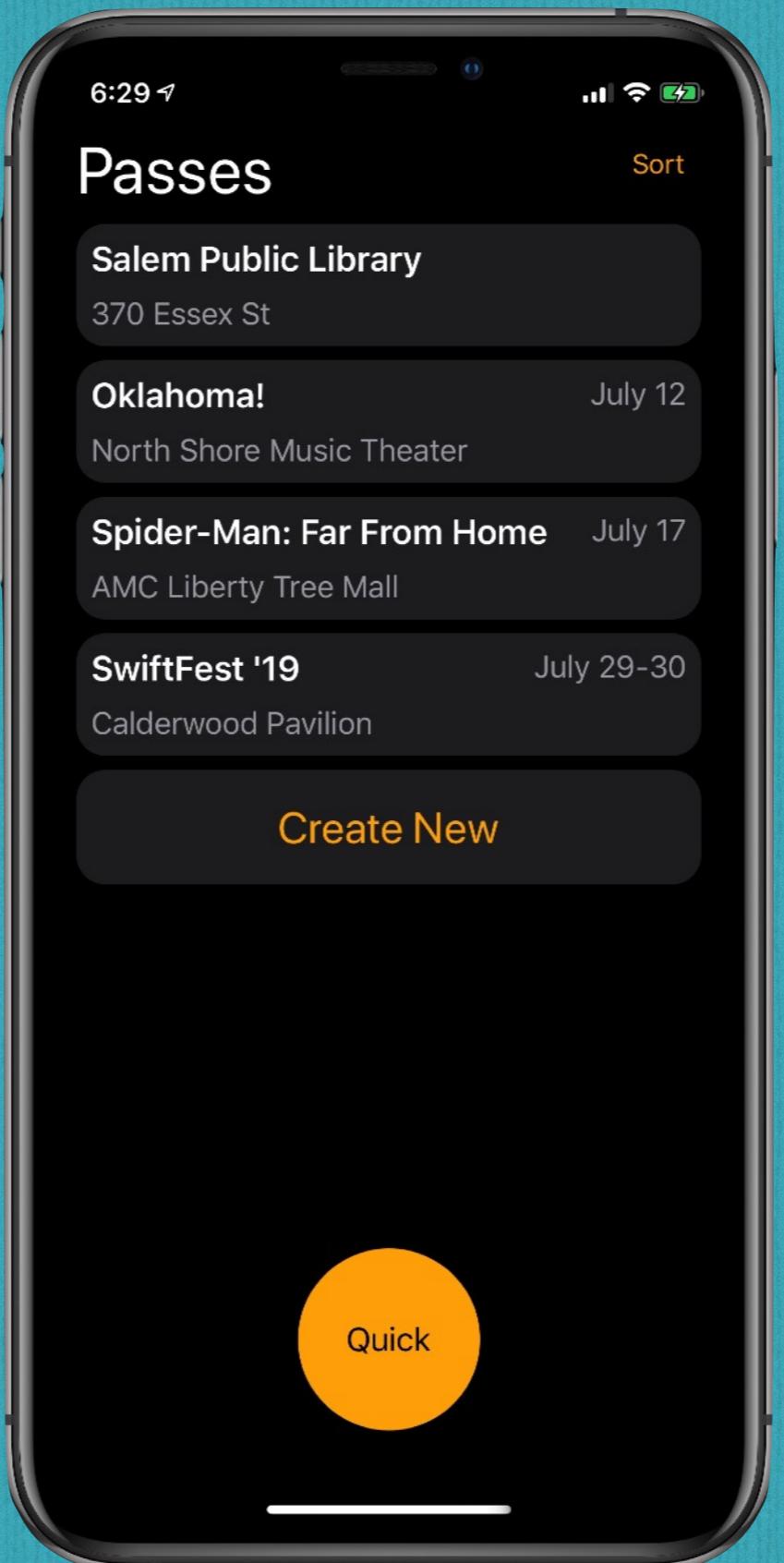
DEALS

20% Off Concentrates

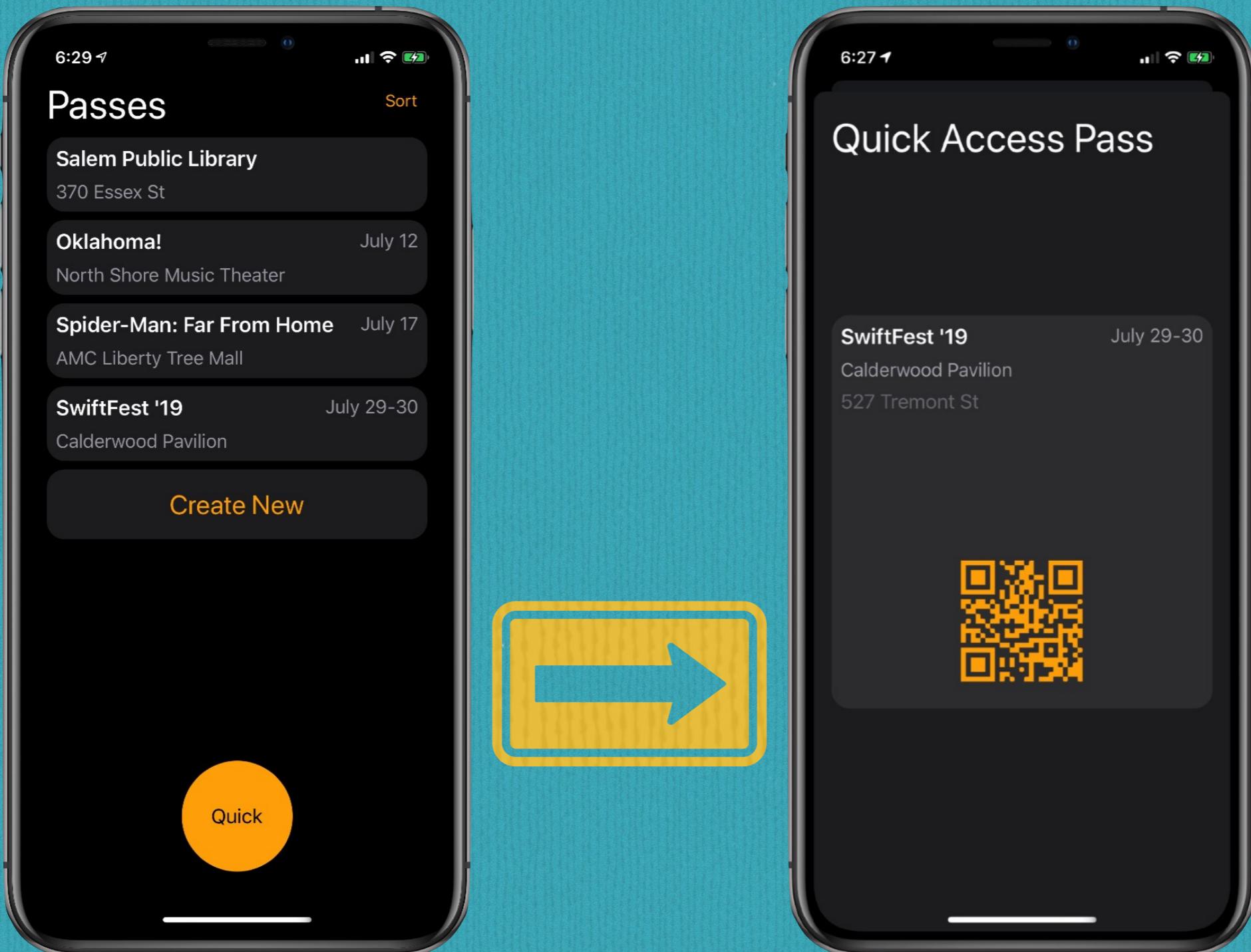
Item

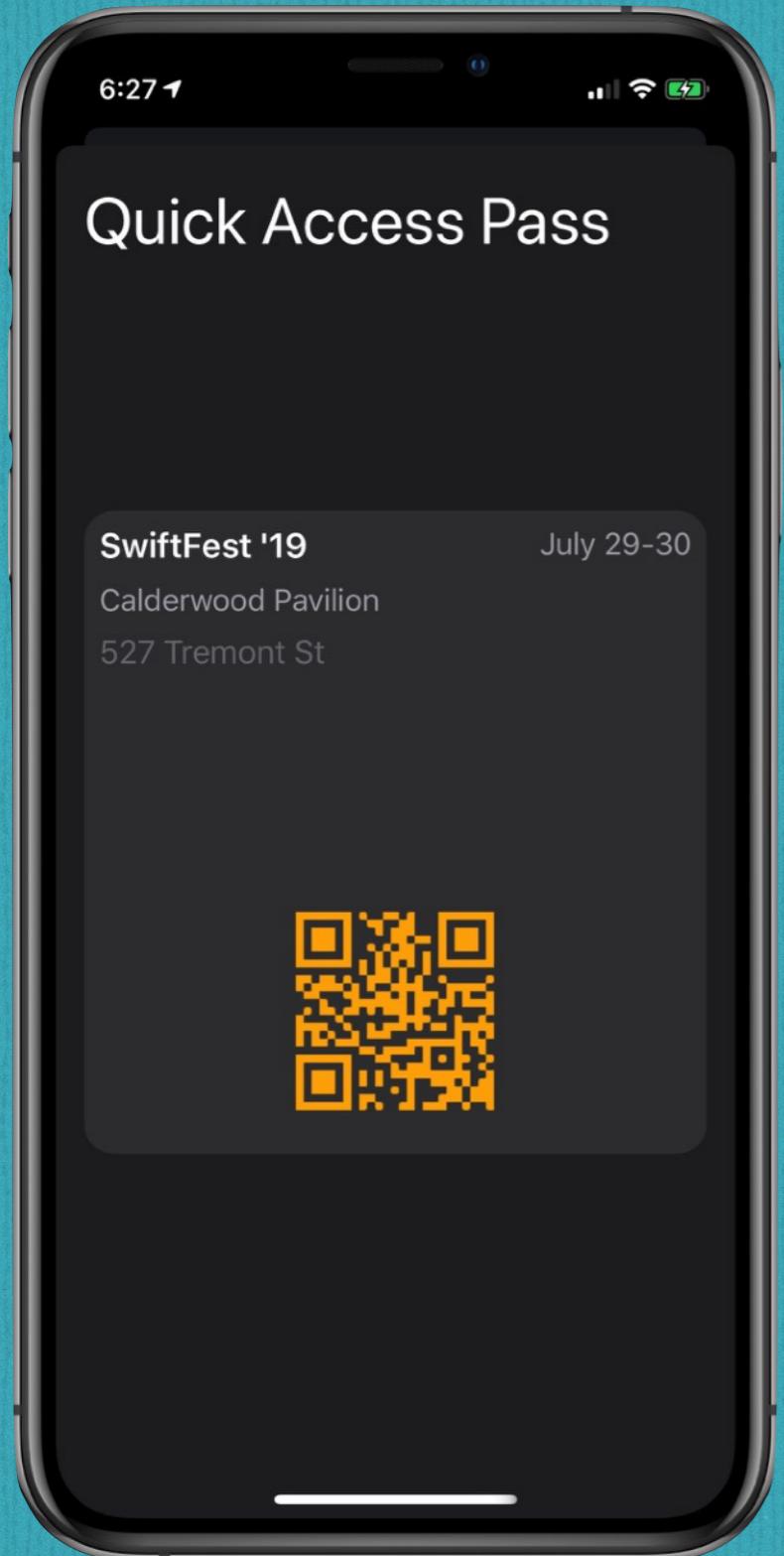
Item





QuickAccessViewController



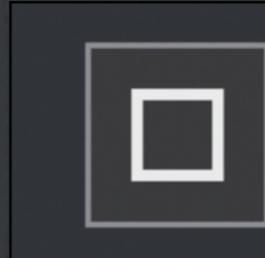


QuickAccessViewController

PassPreviewViewController

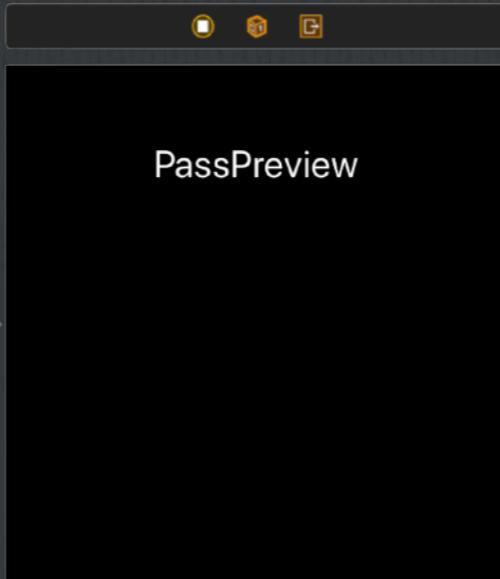
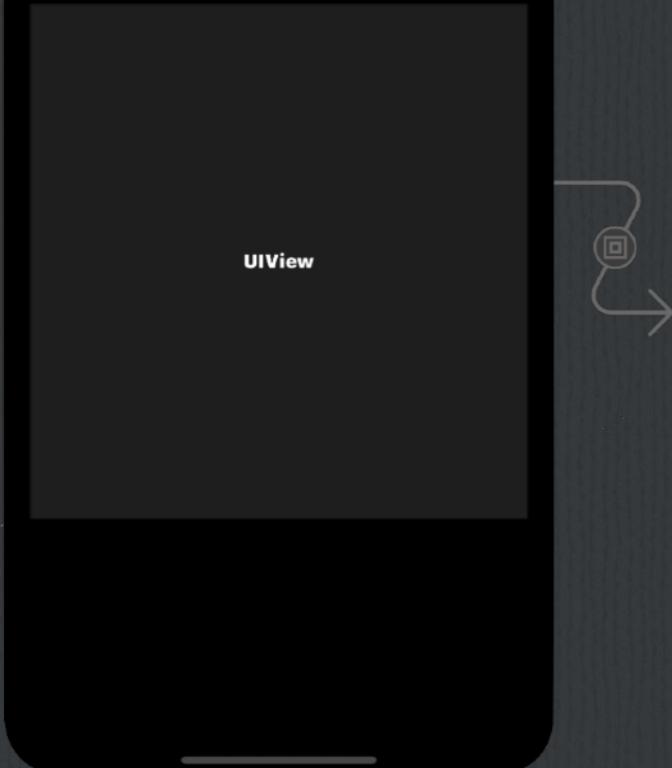
Setting Up Nested Controllers





Container View

Quick Access Pass



- UIView w/
Triggered Scene**
- Embed Segue**
- Child Added
During Segue**



The Debate

- Code is ‘Smaller’
- IBSegueAction
- World Beyond Xcode
- No visual representation

8:28 AM Fri Jul 26

Main.storyboard

Plain text

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <document type="com.apple.InterfaceBuilder3.CocoaTouch.Storyboard.XIB" version="3.0" toolsVersion="14460.31"
targetRuntime="iOS.CocoaTouch" propertyAccessControl="none" useAutolayout="YES" useTraitCollections="YES"
useSafeAreas="YES" colorMatched="YES" initialViewController="49e-Tb-3d3">
3   <device id="retina5_9" orientation="portrait">
4     <adaptation id="fullscreen"/>
5   </device>
6   <dependencies>
7     <plugin identifier="com.apple.InterfaceBuilder.IBCocoaTouchPlugin" version="14460.20"/>
8     <capability name="documents saved in the Xcode 8 format" minToolsVersion="8.0"/>
9   </dependencies>
10  <scenes>
11    <!--Tab Bar Controller-->
12    <scene sceneID="y12-sM-oP">
13      <objects>
14        <tabBarController id="49e-Tb-3d3" sceneMemberID="viewController">
15          <tabBar key="tabbar" contentMode="scaleToFill" id="WZ-zg-YXA">
16            <rect key="frame" x="0.0" y="975" width="768" height="49"/>
17            <autoresizingMask key="autoresizingMask" widthSizable="YES" flexibleMinY="YES"/>
18            <color key="backgroundColor" red="0.0" green="0.0" blue="0.0" alpha="0.0" colorSpace="custom"
customColorSpace="sRGB"/>
19            <color key="barTintColor" red="1" green="0.7032746640000005" blue="0.105873724" alpha="1"
colorSpace="custom" customColorSpace="displayP3"/>
20          </tabBar>
21          <connections>
22            <segue destination="9N5-1f-GDL" kind="relationship" relationship="viewControllers" id="euS-
cT-3pN"/>
23            <segue destination="nBi-rQ-2SG" kind="relationship" relationship="viewControllers" id="yKW-oP-
Dcp"/>
24            <segue destination="0rj-CB-SQK" kind="relationship" relationship="viewControllers" id="ZFR-gr-
jkh"/>
25          </connections>
26        </tabBarController>
27        <placeholder placeholderIdentifier="IBFirstResponder" id="Hu-YB-40B" sceneMemberID="firstResponder"/>
28      </objects>
29      <point key="canvasLocation" x="0.0" y="0.0"/>
30    </scene>
31    <!--Home-->
32    <scene sceneID="sfl-oO-jEj">
33      <objects>
34        <viewControllerPlaceholder storyboardName="Home" id="9N5-1f-GDL" sceneMemberID="viewController">
35          <tabBarItem key="tabBarItem" title="Home" id="R6l-bR-JE3"/>
36          <viewControllerPlaceholder>
37            <placeholder placeholderIdentifier="IBFirstResponder" id="z65-wp-OKZ" userLabel="First Responder"
sceneMemberID="firstResponder"/>
38            <objects>
39              <point key="canvasLocation" x="-416" y="456"/>
40            </objects>
41            <!--Favorites-->
42            <scene sceneID="Q1I-Ds-10F">
43              <objects>
44                <viewControllerPlaceholder storyboardName="Favorites" id="nBi-rQ-2SG" sceneMemberID="viewController">
45                  <tabBarItem key="tabBarItem" systemItem="favorites" id="onN-rm-aY1"/>
46                  <viewControllerPlaceholder>
47                    <placeholder placeholderIdentifier="IBFirstResponder" id="t8g-Ur-1Bu" userLabel="First Responder"
sceneMemberID="firstResponder"/>
48                    <objects>
49                      <point key="canvasLocation" x="-3" y="456"/>
50                    </objects>
51            </scene>
52            <!--Me-->
53            <scene sceneID="Wd0-01-hyF">
54              <objects>
55                <viewControllerPlaceholder storyboardName="Me" id="0rj-CB-SQK" sceneMemberID="viewController">
56                  <tabBarItem key="tabBarItem" title="Me" id="vbW-zr-KnJ"/>
57                  <viewControllerPlaceholder>
58                    <placeholder placeholderIdentifier="IBFirstResponder" id="vyL-nb-Q8h" userLabel="First Responder"
sceneMemberID="firstResponder"/>
59                    <objects>
60                      <point key="canvasLocation" x="366" y="456"/>
61                    </objects>
62            </scene>
63        </objects>
64      </scene>
65    </scenes>
66  </document>
```

Content

Changes

Status



Three Step Process

`parentVC.addChild(childVC)`

`parentVC.view.addSubview(childVC.view)`

`childVC.didMove(toParent: parentVC)`





Removal

`childController.removeFromParent()`

`childView.removeFromSuperview()`

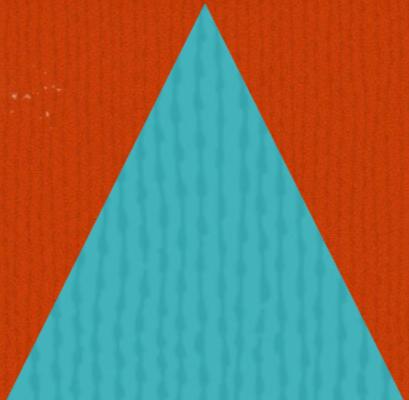


```
extension UIViewController {  
    func removeViewFromParent() {  
        view.removeFromSuperview()  
        self.removeFromParent()  
    }  
}
```

When Should A View Controller Have Children?

Never/Rarely

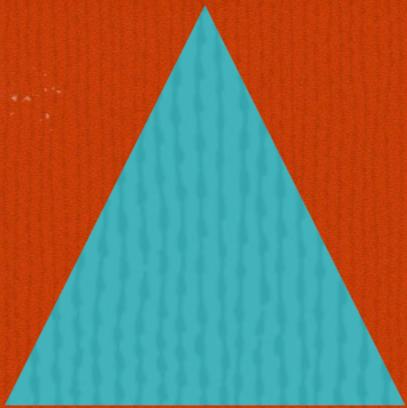
Every Subview



When Should A View Controller Have Children?

Never/Rarely

Every Subview

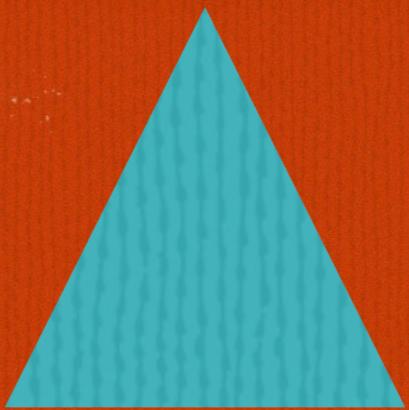


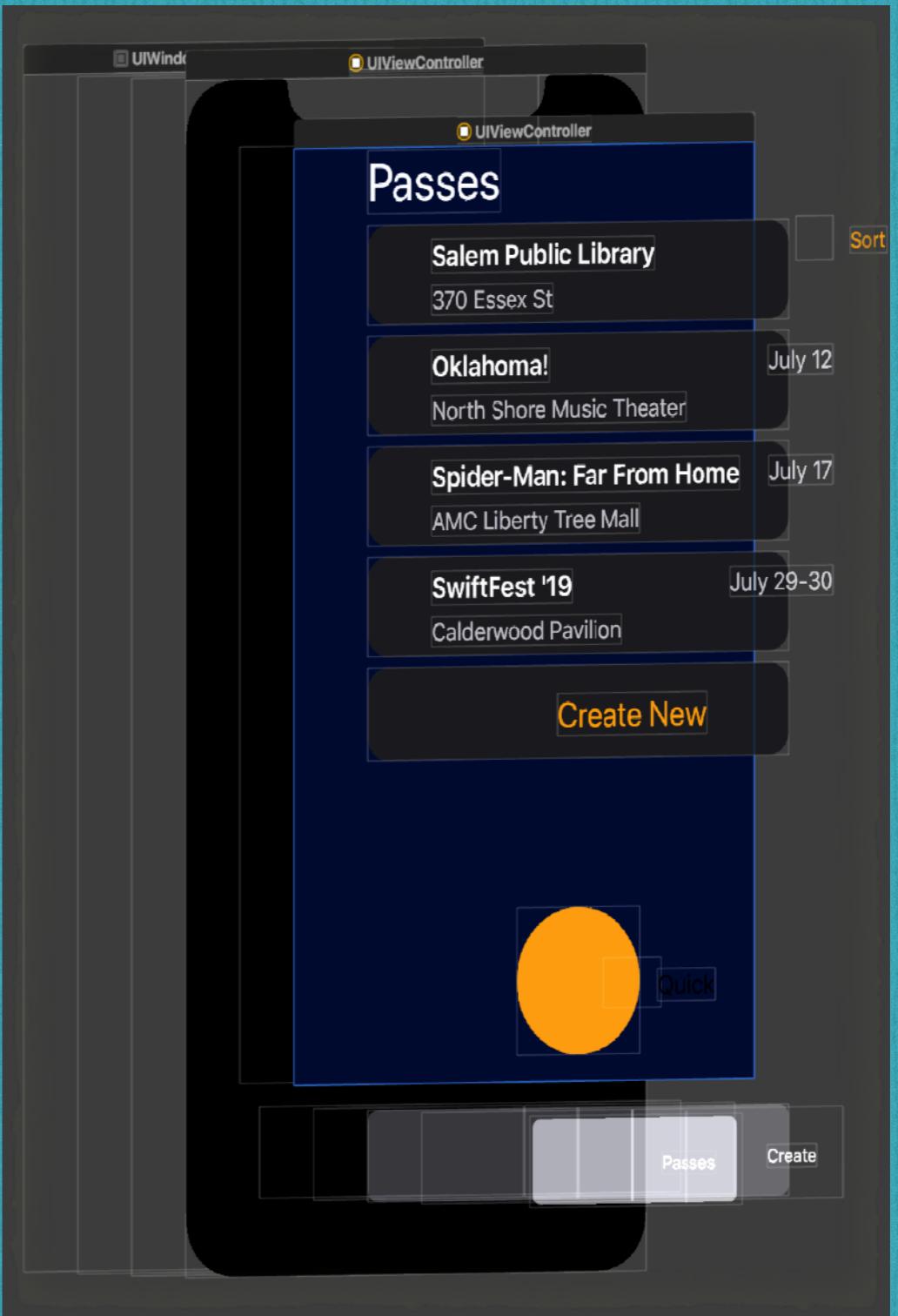
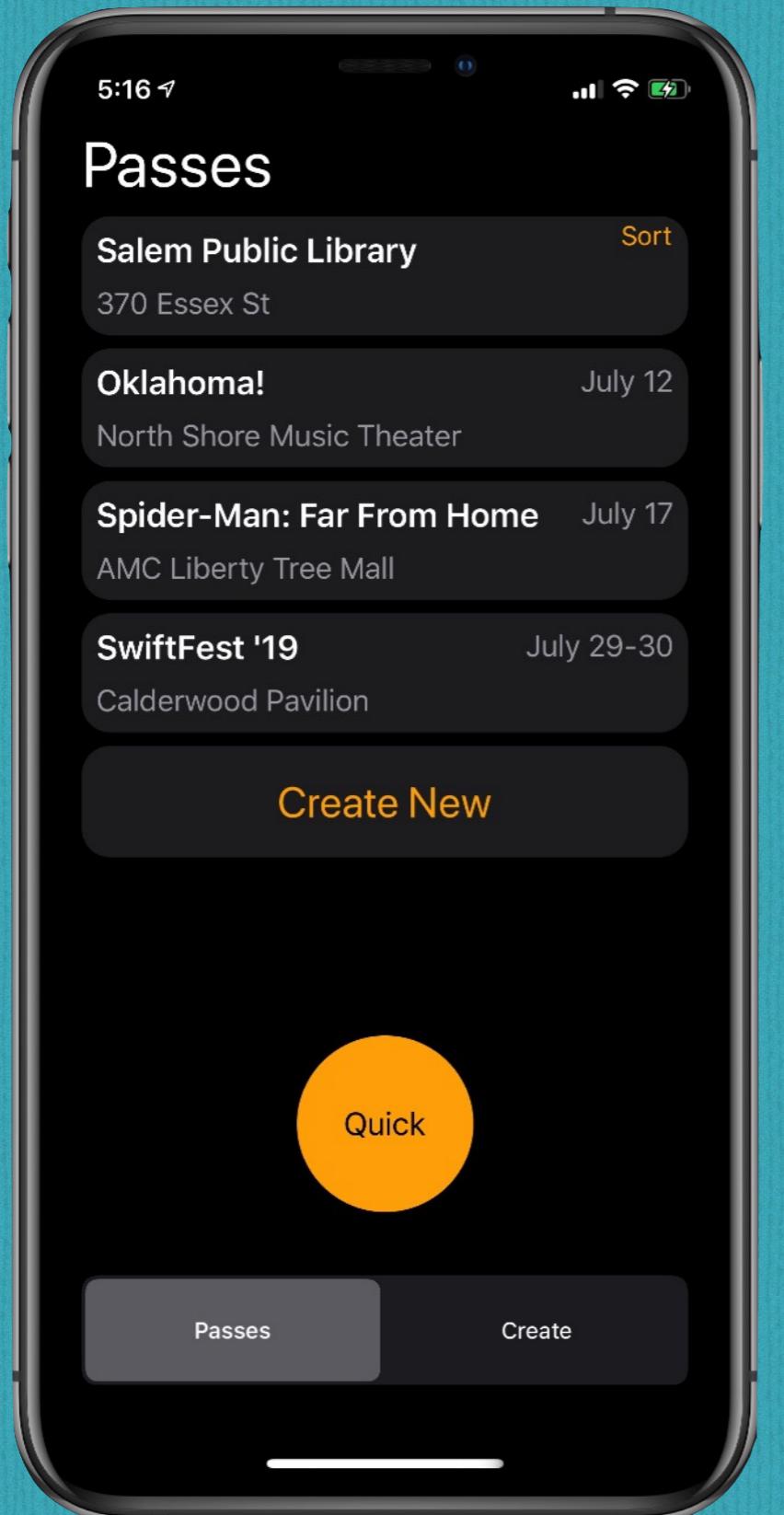
Find Your Balance

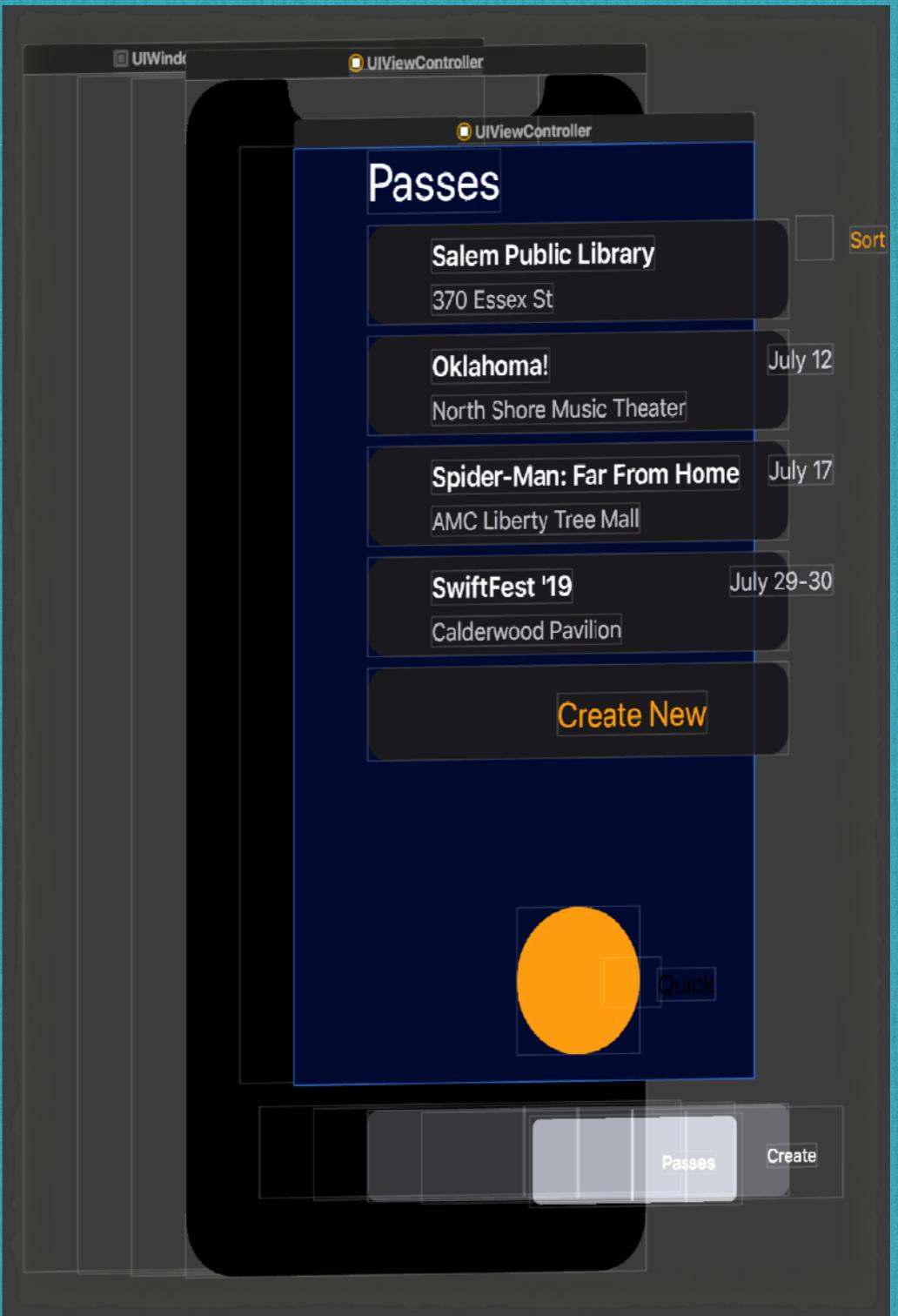
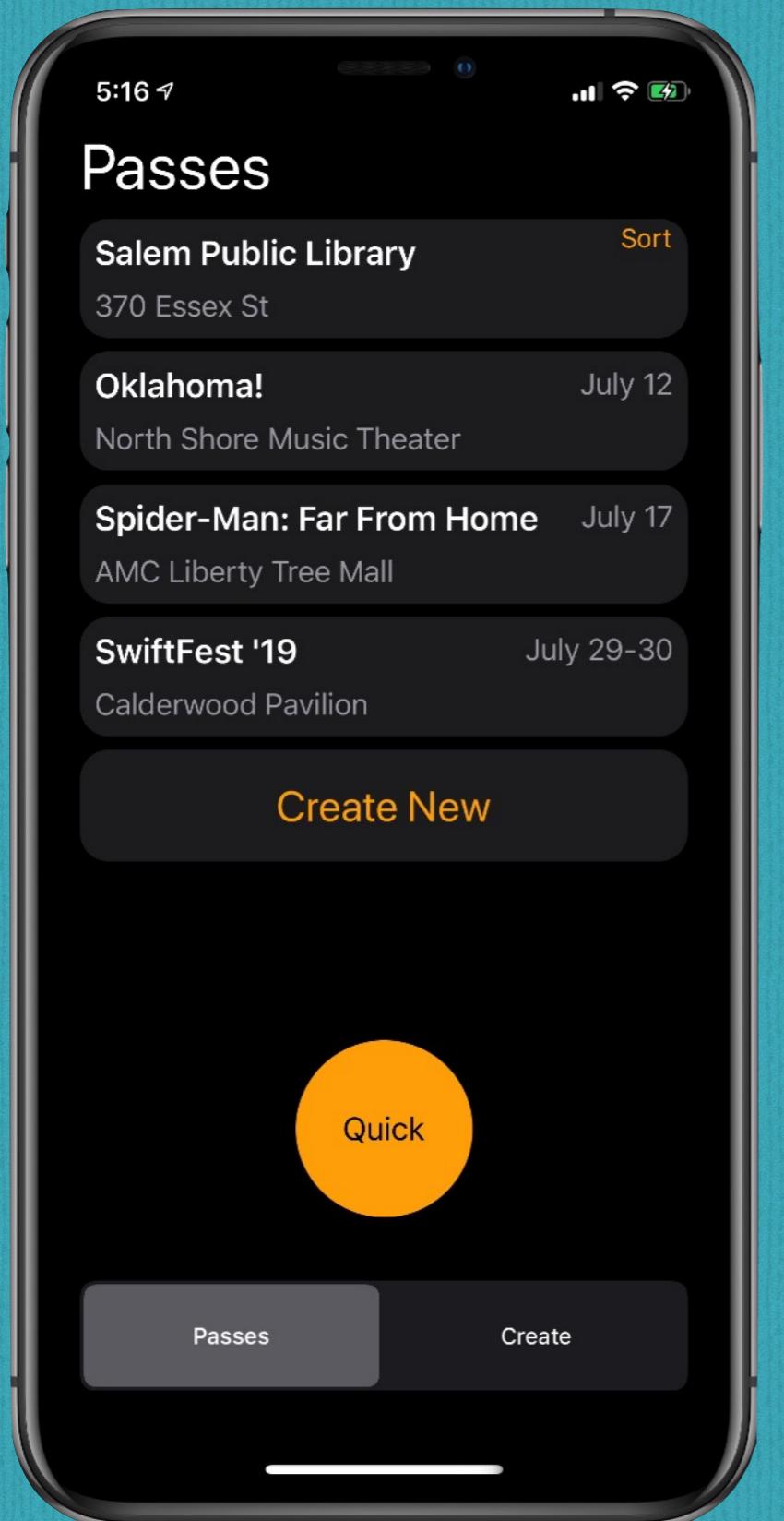
Never/Rarely



Every Subview







ONLINE

North Shore Music Theater

Spider-Man: Far From Home July 17

AMC Liberty Tree Mall

SwiftFest '19 July 29-30

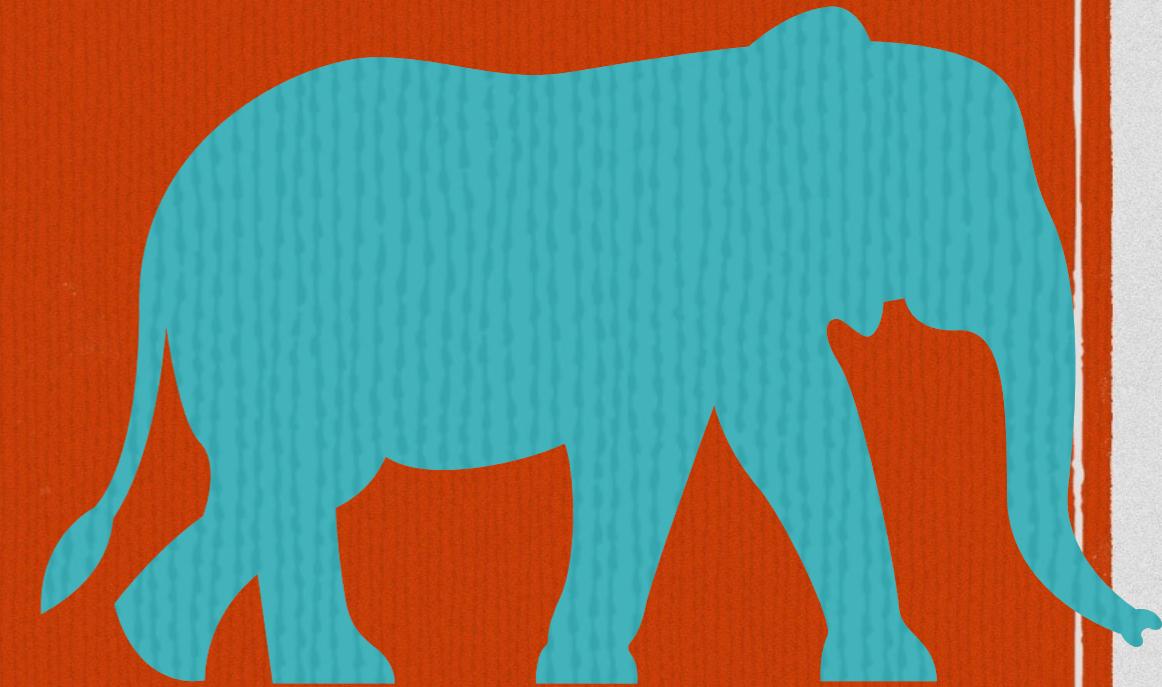
Calderwood Pavilion

!

Create New



SwiftUI





```
class QuickAccessViewController: UIViewController {
    override func viewWillAppears(_ animated: Bool) {
        super.viewWillAppears(animated)
        // Prepare to appear
    }
    override func viewDidAppears(_ animated: Bool) {
        // React to appearance
        super.viewDidAppears(animated)
    }
    override func viewWillDisappear(_ animated: Bool) {
        super.viewWillDisappear(animated)
        // Prepare to disappear
    }
    override func viewDidDisappear(_ animated: Bool) {
        // React to disappearance
        super.viewDidDisappear(animated)
    }
}
```



```
class QuickAccessViewController: UIViewController {  
    override func viewWillAppears(animated: Bool) {  
        super.viewWillAppears(animated)  
        // Prepare to appear  
    }  
    override func viewDidAppears(animated: Bool) {  
        // React to appearance  
        super.viewDidAppears(animated)  
    }  
    override func viewWillDisappear(animated: Bool) {  
        super.viewWillDisappear(animated)  
        // Prepare to disappear  
    }  
    override func viewDidDisappear(animated: Bool) {  
        // React to disappearance  
        super.viewDidDisappear(animated)  
    }  
}
```



```
struct QuickAccessView: View {  
    var body: some View {  
        return PassPreview()  
            .onAppear {  
                // React to appearance  
            }.onDisappear {  
                // React to disappearance  
            }  
    }  
}
```



Calvin Chotard

iCalvin.dev

@iCalvin

CastKit.fm

@CastKitFM

Questions?