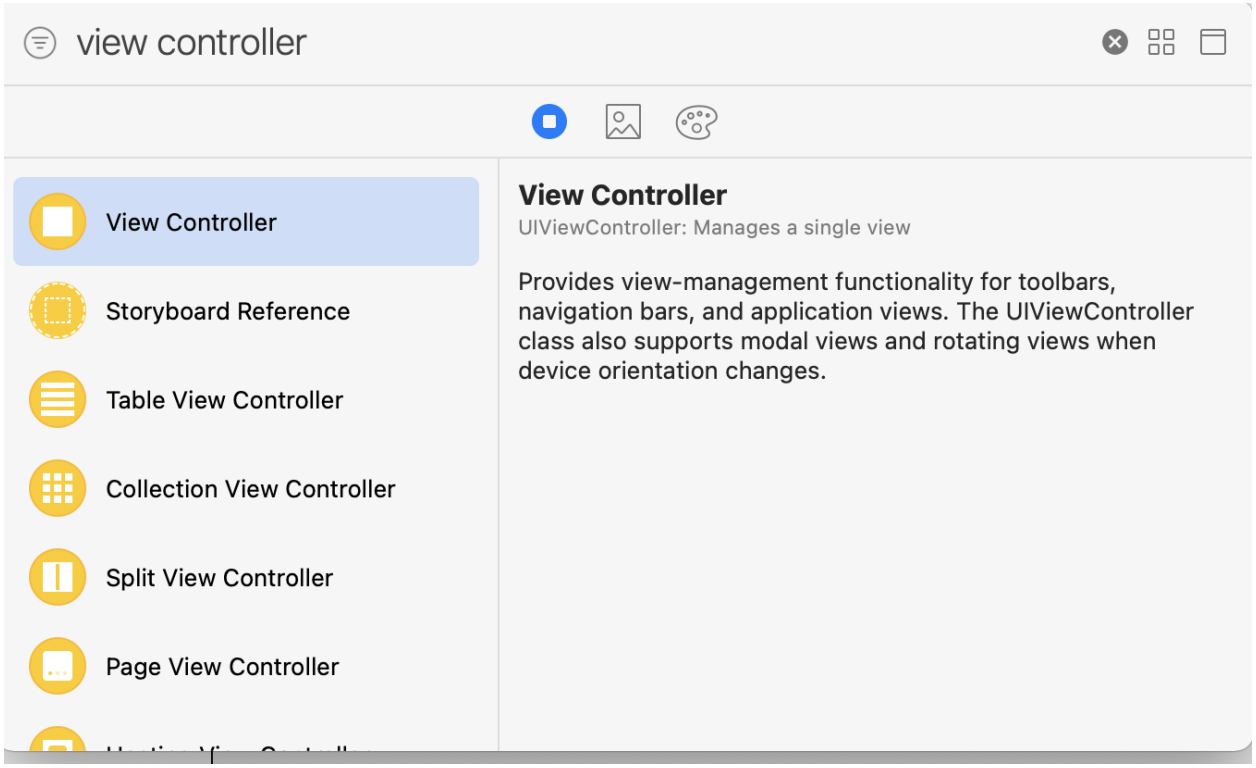
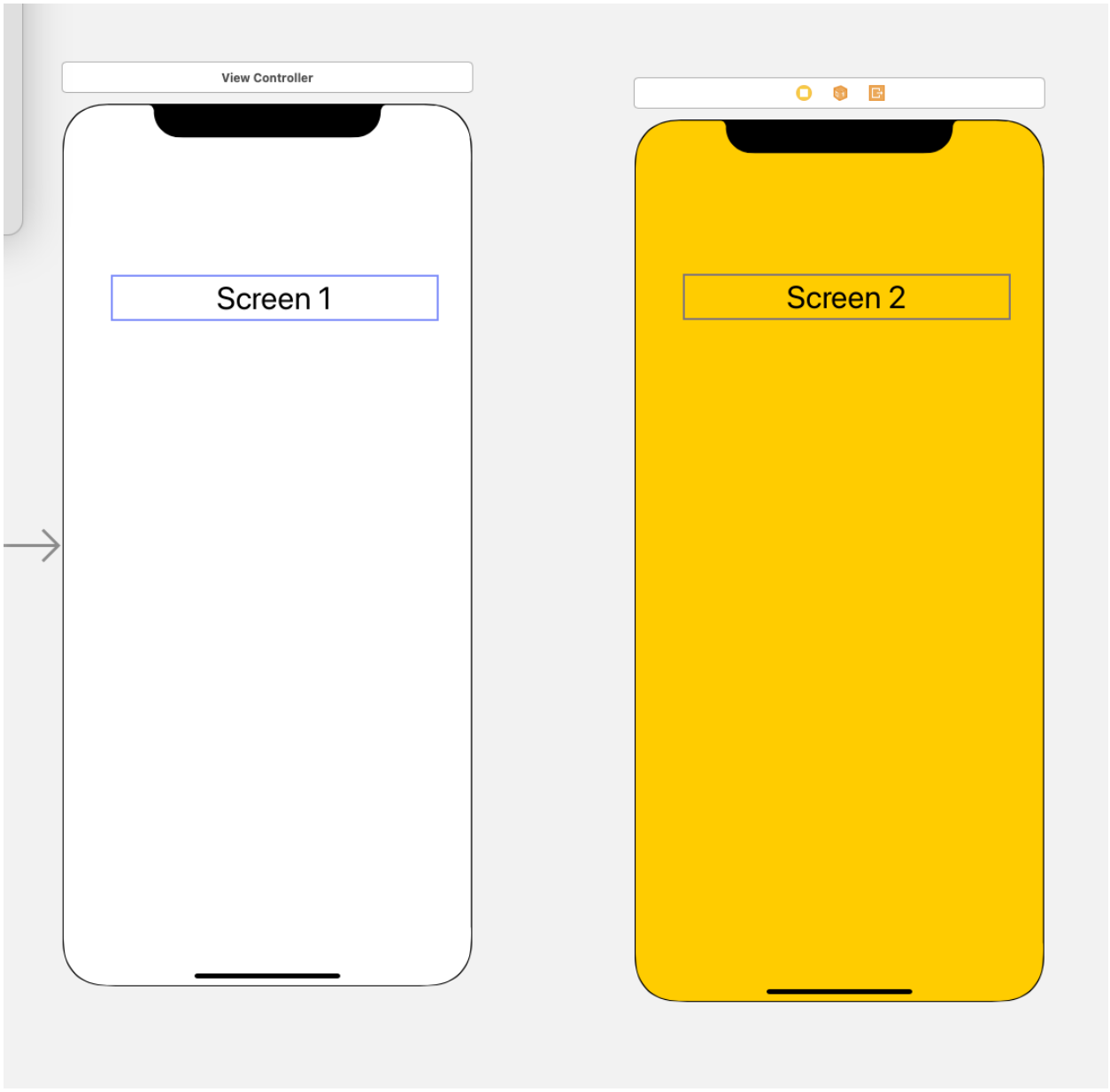
**Multiscreen application**

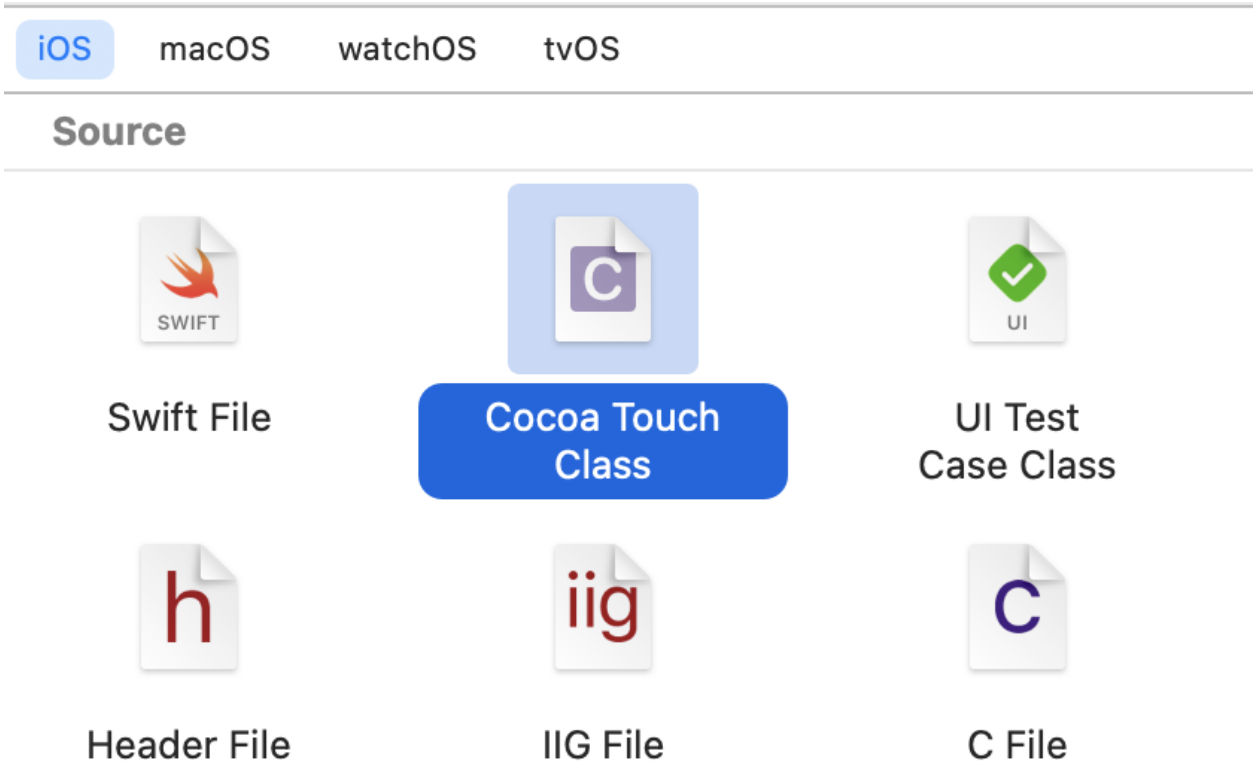
Adding another screen

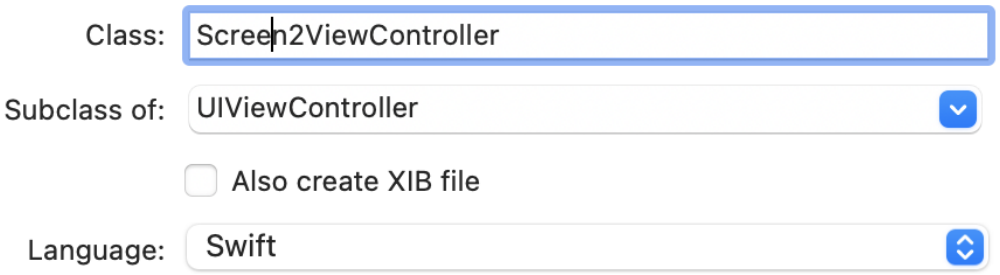
1/ Add a new screen (View Controller) to the storyboard



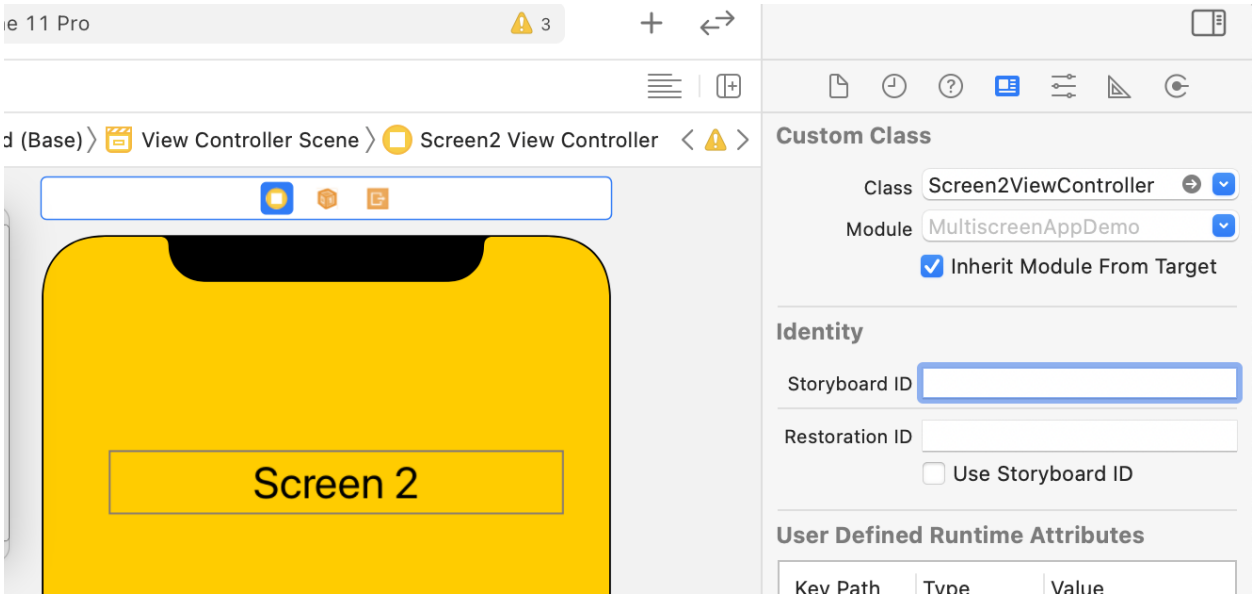



Create a Swift file for the second screen

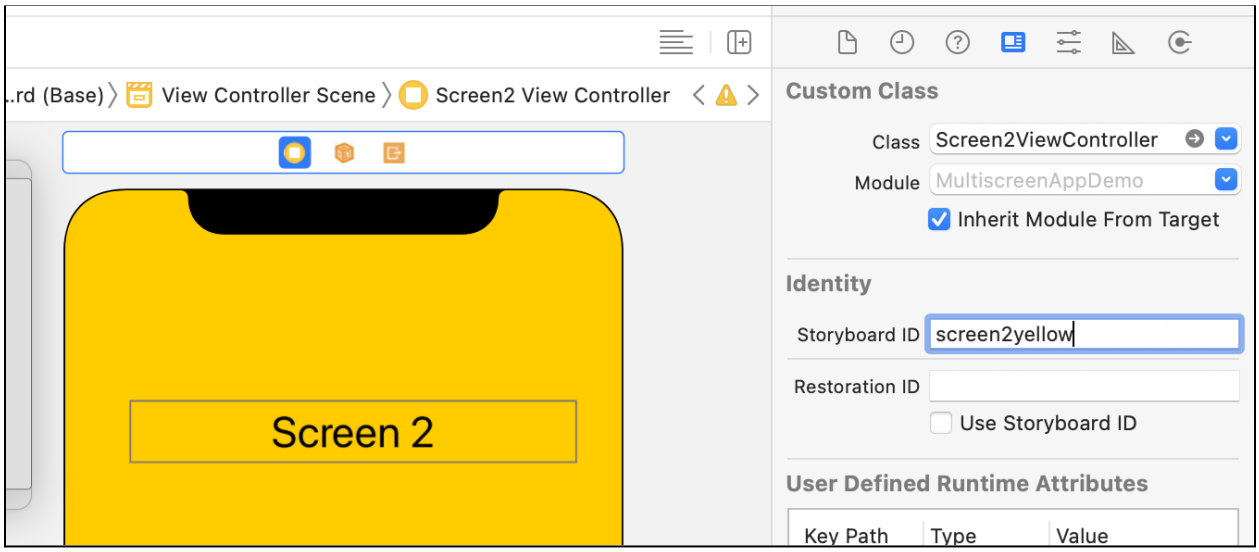



Associate screen 2 view controller with new storyboard screen




2/ Give the ViewController an id




3/ In the ViewController.swift file, retrieve an instance of the storyboard ViewController


```
import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }

    // action
    @IBAction func btnPressed(_ sender: Any) {
        // 1. get a reference to the second screen
        guard let screen2 = storyboard?.instantiateViewController(identifier:
"screen2yellow") as? Screen2ViewController else {
            print("Cannot find a screen with an id of screen2yellow")
            return
        }

    }

}
```

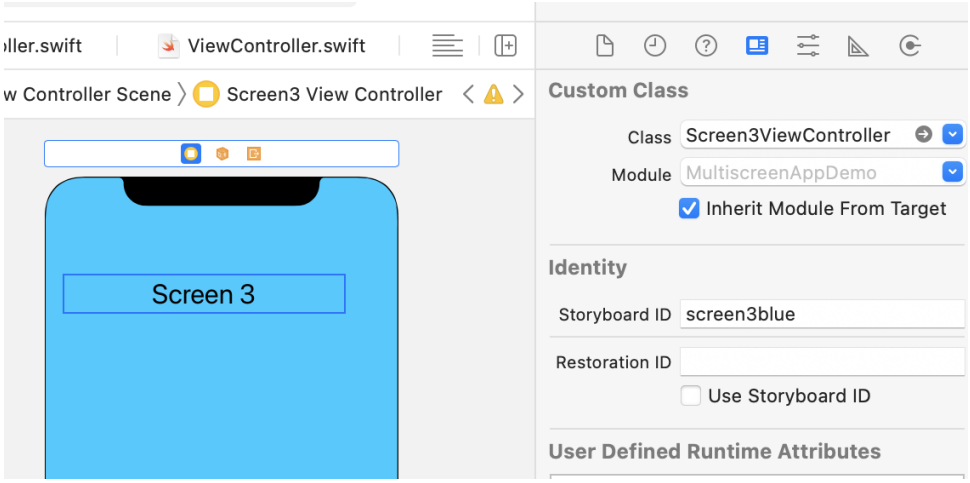
4/ Pass the instance to the show() function

```
import UIKit

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }

    // action
    @IBAction func btnPressed(_ sender: Any) {
        // 1. get a reference to the second screen
        guard let screen2 = storyboard?.instantiateViewController(identifier:
"screen2yellow") as? Screen2ViewController else {
            print("Cannot find a screen with an id of screen2yellow")
            return
        }

        // 2. show the screen
        show(screen2, sender:self)
    }

}
```

5/ Done!

Exercise: Practice adding another screen to your application

- Add the screen to your storyboard

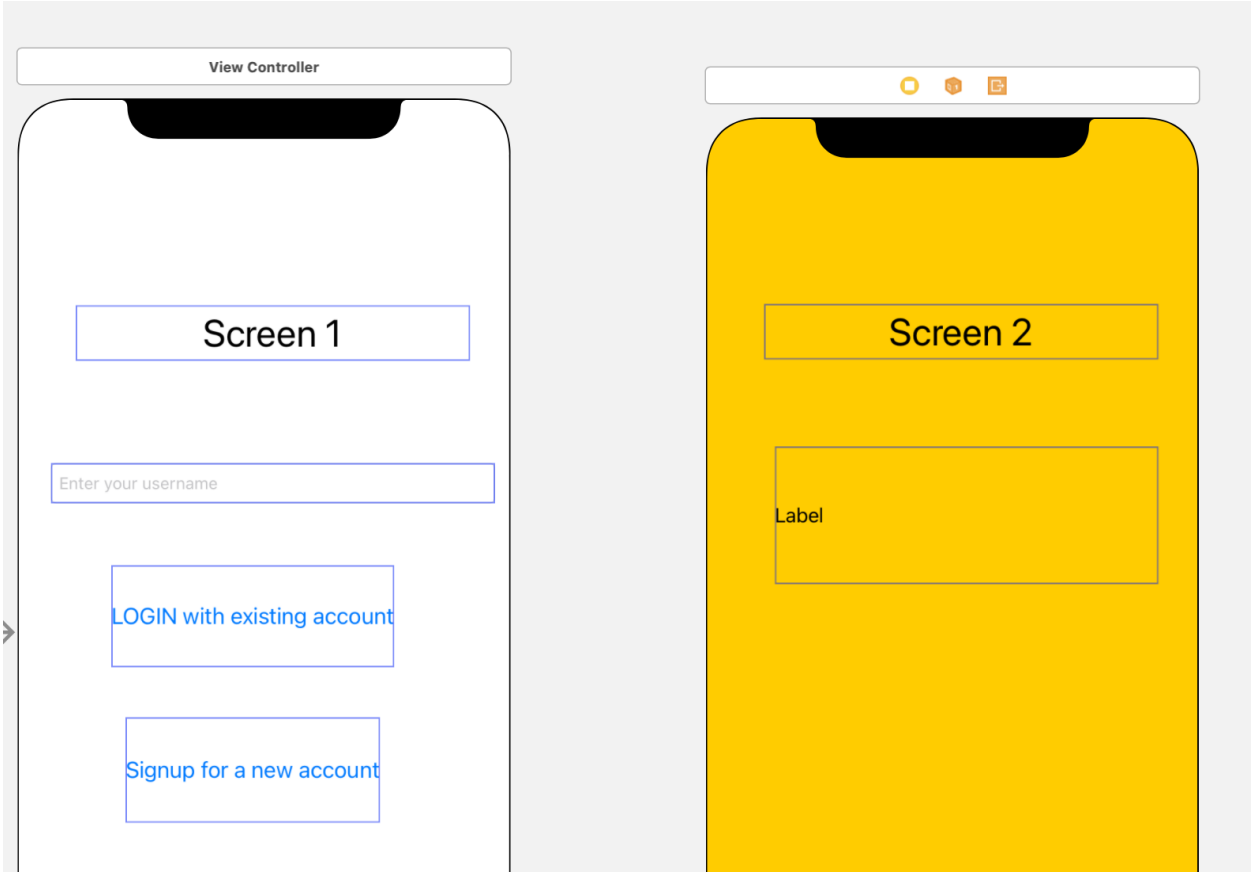



- On screen 1 - add a button to navigate to screen 3

```
    @IBAction func btnGoToScreen3Pressed(_ sender: Any) {
        guard let screen3 = storyboard?.instantiateViewController(identifier:
"screen3blue") as? Screen3ViewController else {
            print("Cannot find a screen with an id of screen3blue")
            return
        }

        // 2. show the screen
        show(screen3, sender:self)

    }
```

## Passing Data to Another Screen

1/ Add a second screen to storyboard



```
import UIKit

class ViewController: UIViewController {

    // outlets
    @IBOutlet weak var txtUsername: UITextField!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }

    // action
    @IBAction func btnPressed(_ sender: Any) {

        // 1. get a reference to the second screen
        guard let screen2 = storyboard?.instantiateViewController(identifier:
"screen2yellow") as? Screen2ViewController else {
            print("Cannot find a screen with an id of screen2yellow")
            return
        }

        // 2. Get the username from the text box
        let username = txtUsername.text!
        print("The user name  from the text box is: \(username)")

        // 2. show the screen
       // show(screen2, sender:self)
    }
}
```

2/ In the second screen's ViewController.swift file, create a **variable** that has the same **data type** as the data you want to send. This variable is called a "receiving variable"

3/ Write code to do something with that data

```
class Screen2ViewController: UIViewController {

    // stored property
    var age:Int = 5

    // receving variable: It will store any data that was received from
Screen #1
    // data type = match the data type of the information that is being sent
    var usernameFromScreen1:String = ""

    override func viewDidLoad() {
        super.viewDidLoad()

        // do something with the data you got from screen 1
        print("Screen 1 sent me: \(self.usernameFromScreen1)")
    }
}
```

4/ In the first screen, use the *instance* of the second screen to set the value of the **receiving** variable

```
@IBAction func btnPressed(_ sender: Any) {

        // 1. get a reference to the second screen
        guard let screen2 =
storyboard?.instantiateViewController(identifier: "screen2yellow") as?
Screen2ViewController else {
            print("Cannot find a screen with an id of screen2yellow")
            return
        }

        // 2. Get the username from the text box
        let username = txtUsername.text!
        print("The user name  from the text box is: \(username)")

        // 3. "Send" the username to screen 2
        screen2.usernameFromScreen1 = username

        // 4. show the screen
        show(screen2, sender:self)
    }
```

## Exercise:  Send an object to screen 2

## 1/ Model an object

```
class Dog {
    var name:String
    var breed:String
    var color:String

    init(name:String, breed:String, color:String) {
        self.name = name
        self.breed = breed
        self.color = color
    }

    func bark() {
        print("\(self.name) says WOOF WOOF!")
    }
}
```

## 2/ Screen 2 View Controller:

- Add the dog object to your list of receiving variables

```
import UIKit

class Screen2ViewController: UIViewController {

    // receiving variable: It will store any data that was received from Screen #1
    // data type = match the data type of the information that is being sent
    var usernameFromScreen1:String = ""
    var password:String = ""
    var age:Int = 0
    var isSleeping:Bool = false

    var dogFromScreen1:Dog?


    override func viewDidLoad() {
        super.viewDidLoad()

        // do something with the data you got from screen 1
```

```
        print("Screen 1 sent me: \(self.usernameFromScreen1)")
        print("Screen 1 sent me: \(self.password)")
        print("Screen 1 sent me: \(self.age)")
        // using self is optional
        print("Screen 1 sent me: \(isSleeping)")

        guard let dog = dogFromScreen1 else {
            return
        }
        print("Dog from screen 1: \(dog.name)")
        dog.bark()

    }

}
```

### 3/ Screen 1 View Controller

- Set the value of the dog

```
   // action
    @IBAction func btnPressed(_ sender: Any) {

        // 1. get a reference to the second screen
        guard let screen2 = storyboard?.instantiateViewController(identifier:
"screen2yellow") as? Screen2ViewController else {
            print("Cannot find a screen with an id of screen2yellow")
            return
        }

        // 2. Get the username from the text box
        let username = txtUsername.text!
        print("The user name  from the text box is: \(username)")

        // 3. "Send" the username to screen 2
        screen2.usernameFromScreen1 = username

        // 3b. hardcode some data to send to screen 2
        screen2.age = 55
        screen2.isSleeping = true
        screen2.password = "thisismySuperSecretP@55word!"

        // dog
        screen2.dogFromScreen1 = Dog(name:"Polo", breed:"Poodle", color:"pink")
```

```
        // 4. show the screen
        show(screen2, sender:self)
    }
```

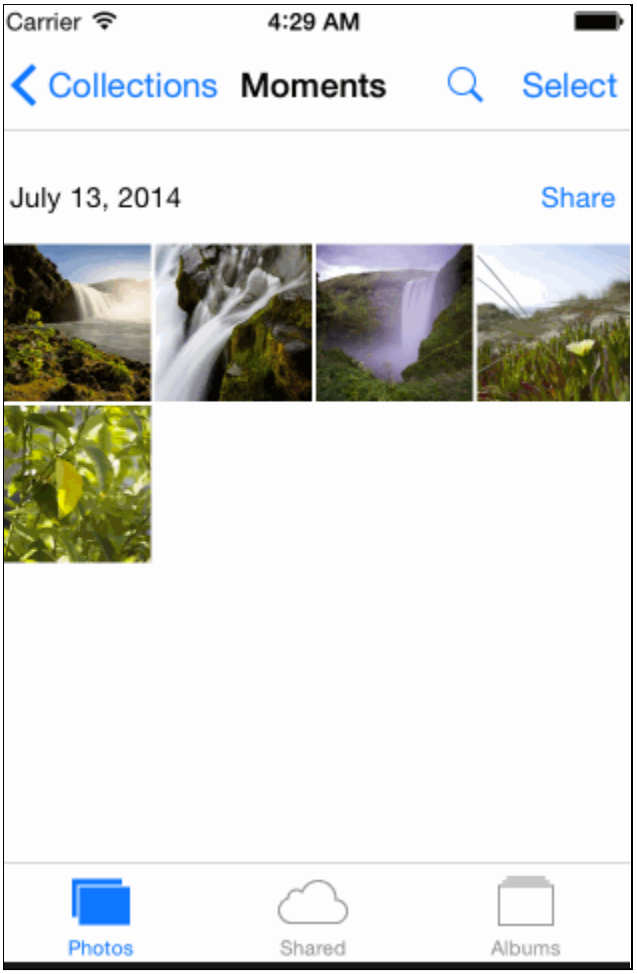
Expected output:

```
The user name  from the text box is: Peter
Screen 1 sent me: Peter
Screen 1 sent me: thisismySuperSecretP@55word!
Screen 1 sent me: 55
Screen 1 sent me: true
Dog from screen 1: Polo
Polo says WOOF WOOF!
```

## Working With Built in Controllers for Multiscreen Navigation

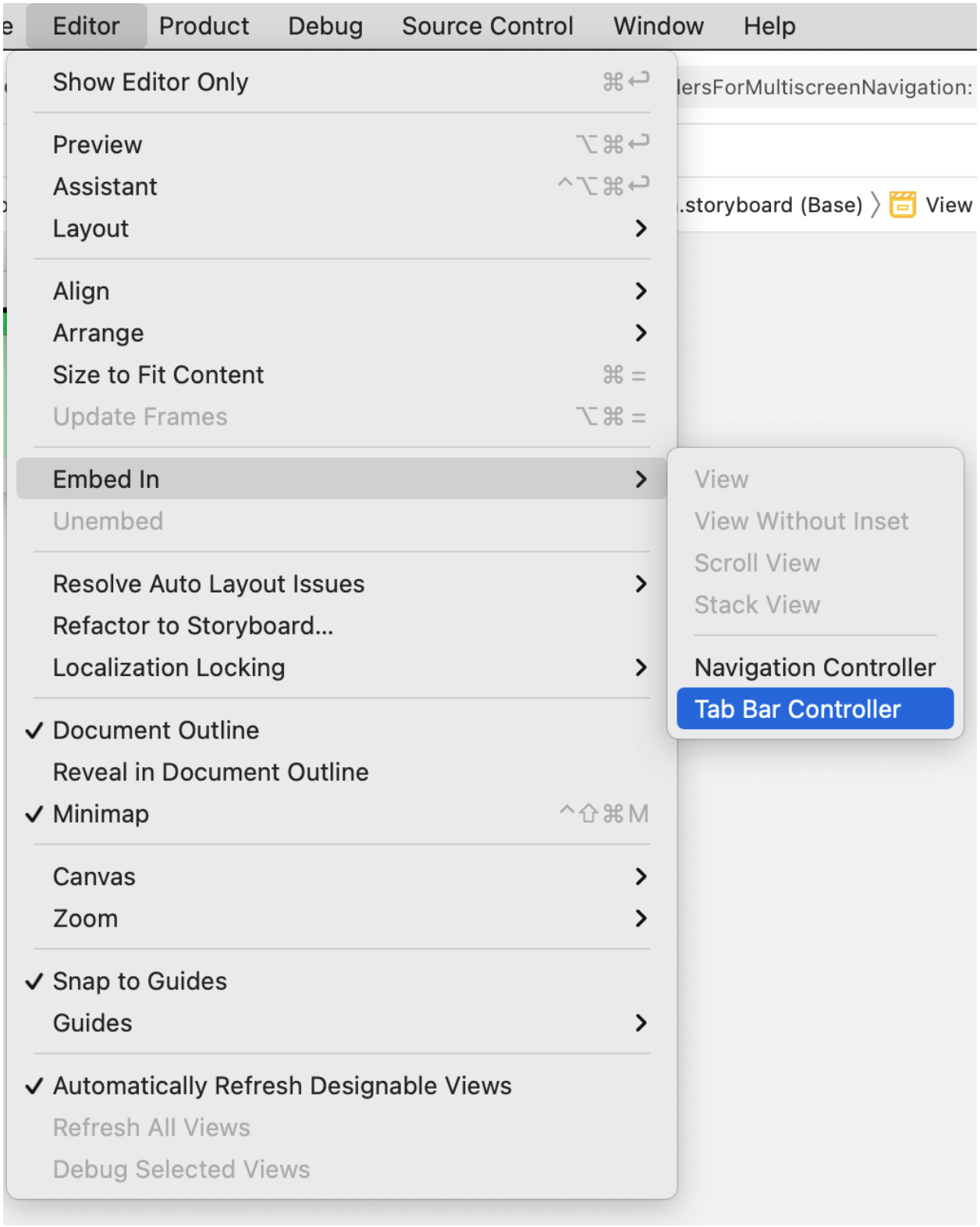
## 1/ Navigation Controller[1]

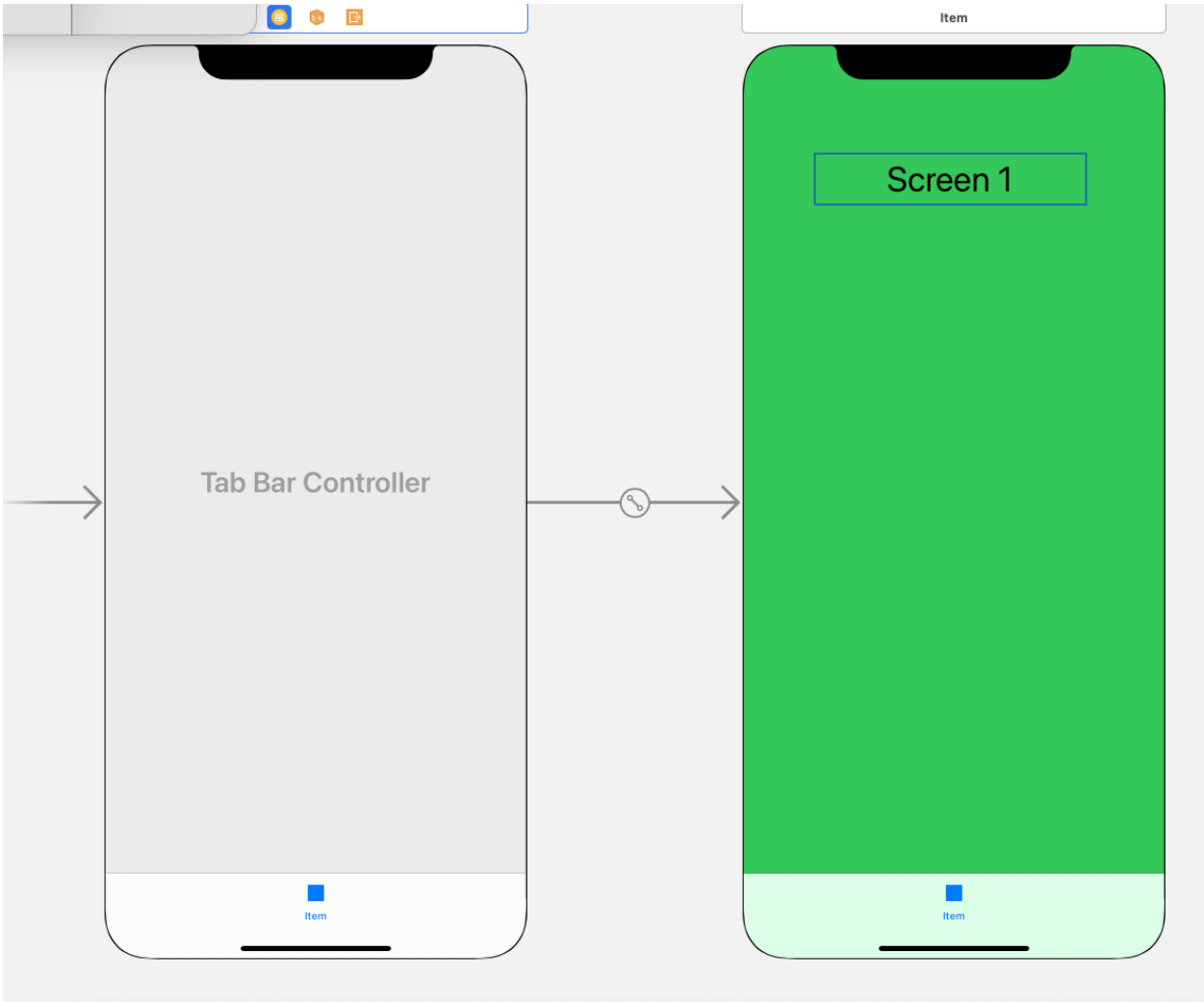
## When you want to link several screens together in sequential order



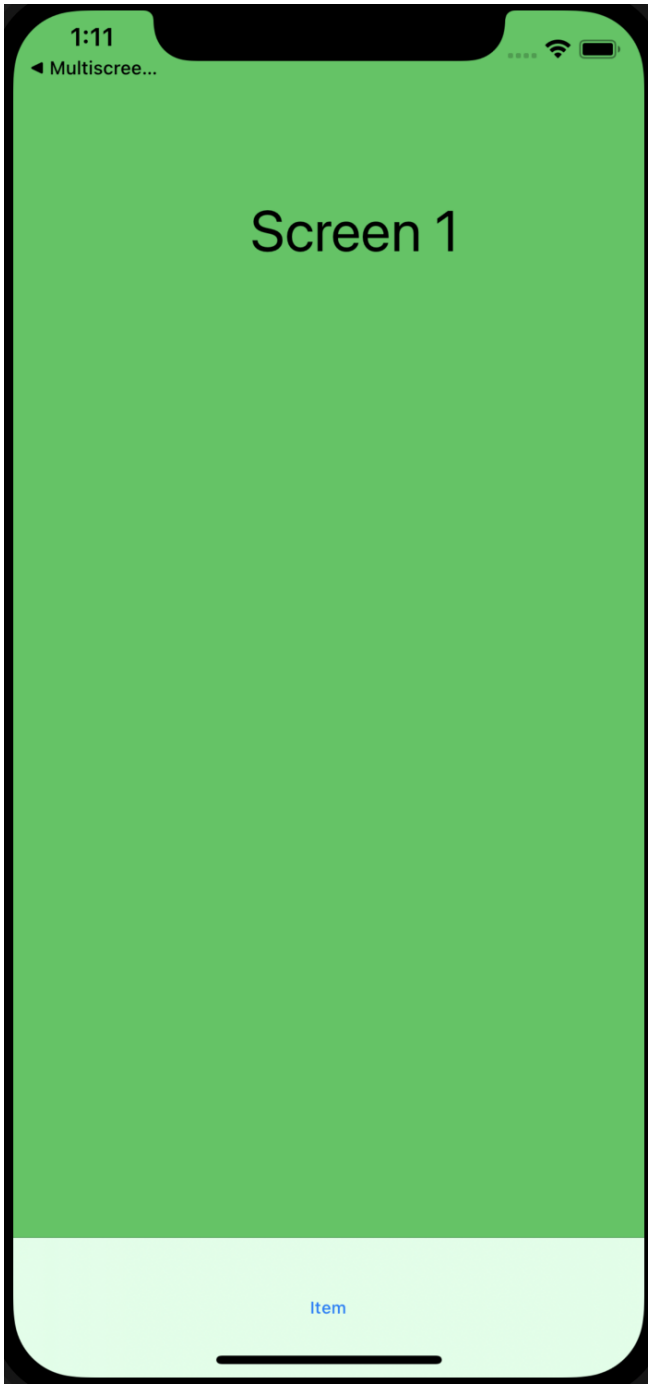

[1] Image from: https://guides.codepath.com/ios/Navigation-Controller-Quickstart

# Tab Bar Controller


Editor
Product
Debug
Source Control
Window
Help
Show Editor Only
Preview
Assistant
Layout
Align
Arrange
Size to Fit Content
Update Frames
Embed In
Unembed
Resolve Auto Layout Issues
Refactor to Storyboard...
Localization Locking
Document Outline
Reveal in Document Outline
Minimap
Canvas
Zoom
Snap to Guides
Guides
Automatically Refresh Designable Views
Refresh All Views
Debug Selected Views
View
View Without Inset
Scroll View
Stack View
Navigation Controller
Tab Bar Controller
lersForMultiscreenNavigation:
.storyboard (Base)
View

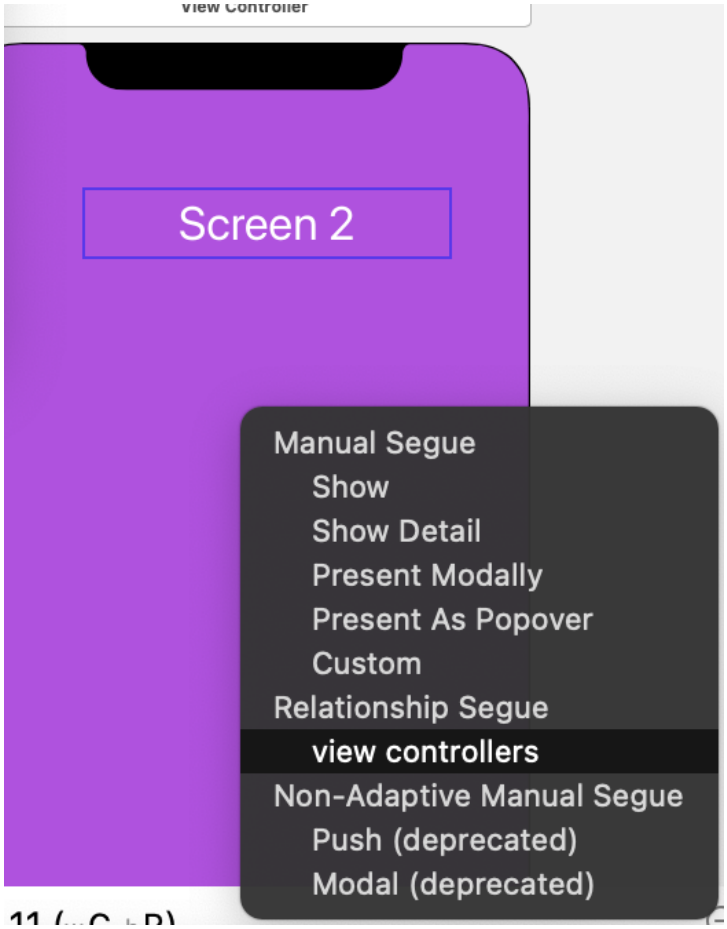


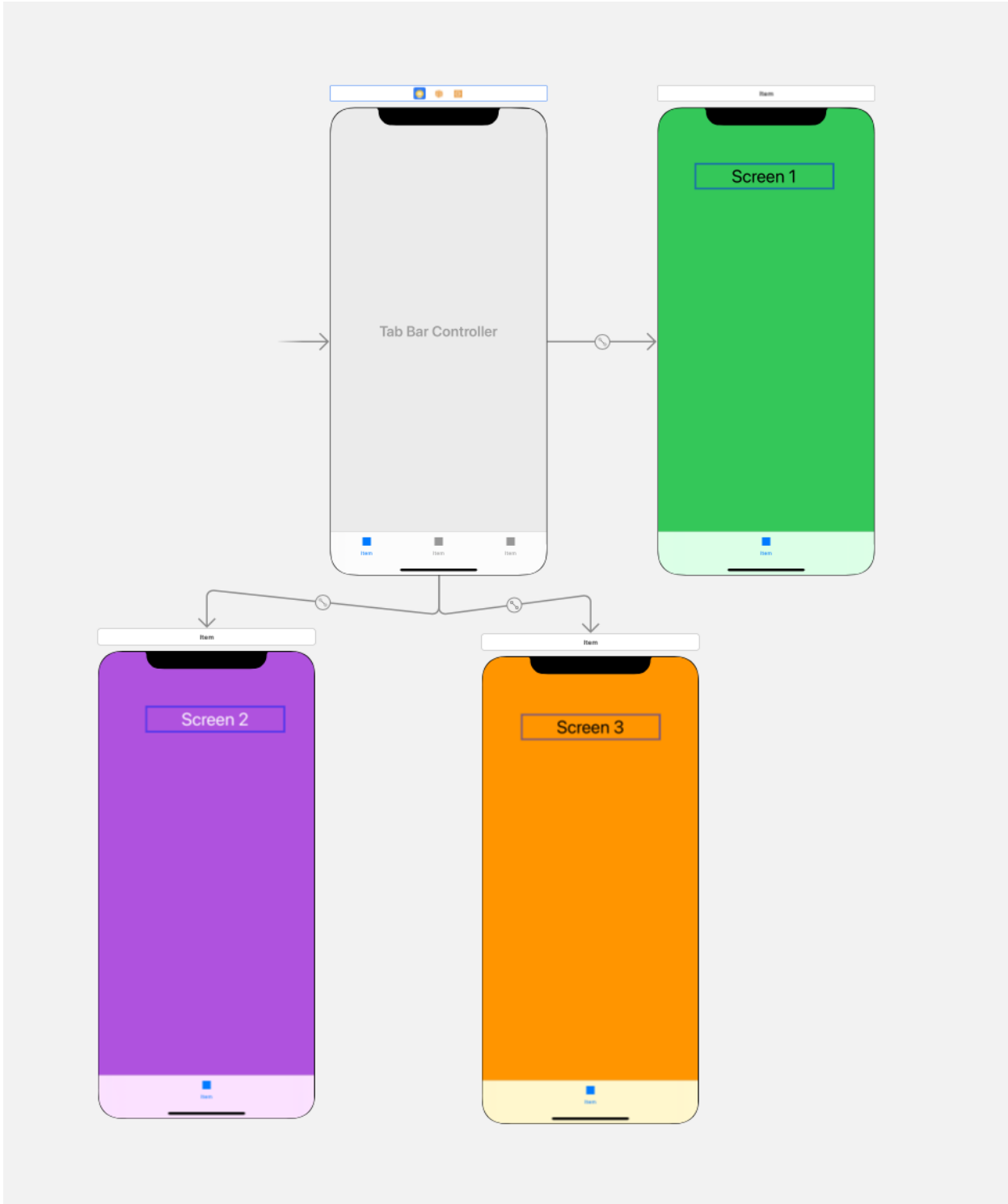

Expected


1:11
◀ Multiscree...
Screen 1
Item

A

Add another screen

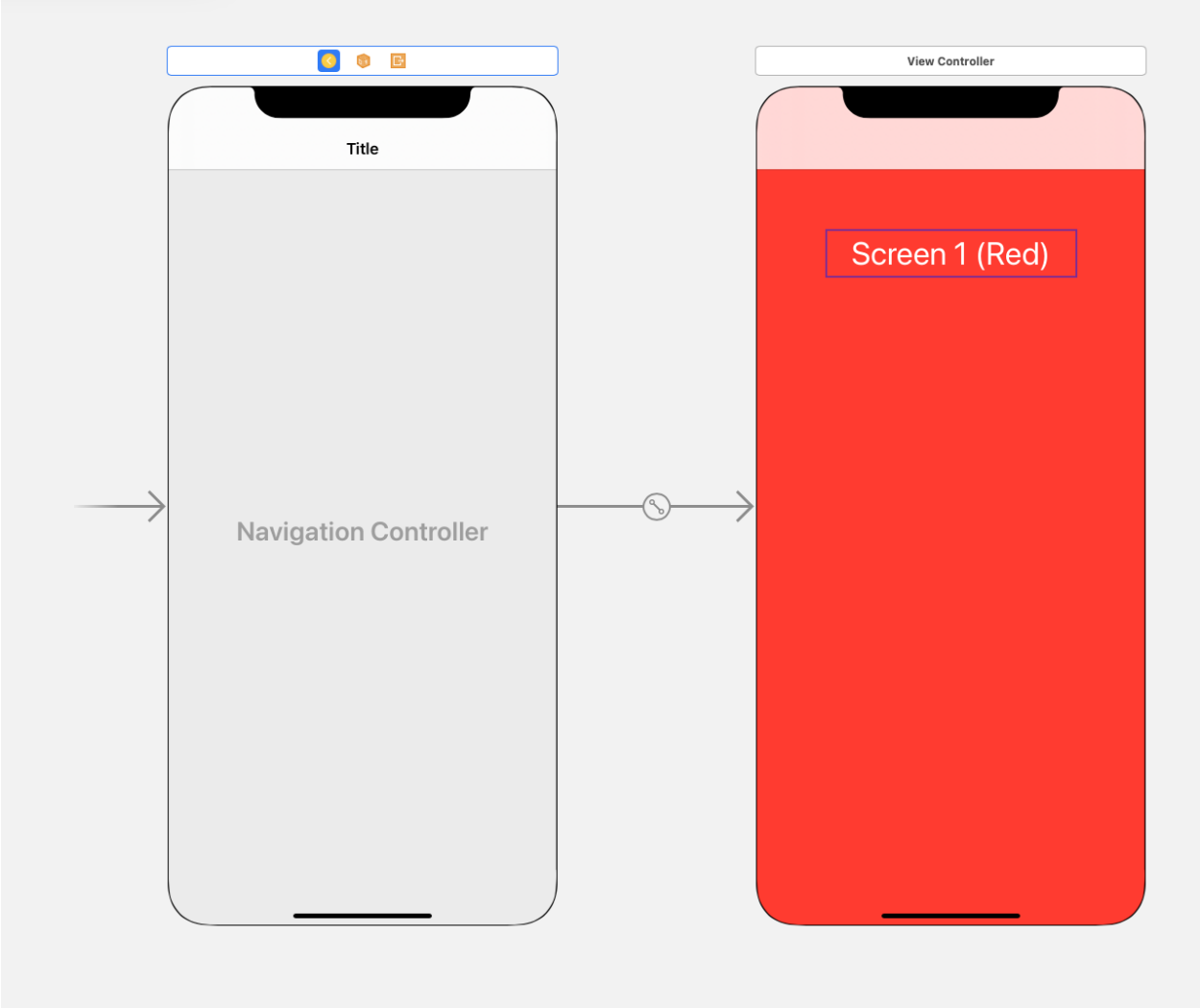



Tab Bar Controller
Item
Screen 1
Item
Item
Screen 2
Item
Item
Screen 3
Item

## 2/ Tab Bar Controller[2]


[2] Image from: https://guides.codepath.com/ios/Tab-Bar-Quickstart

## Navigation Controller


Title
Navigation Controller
View Controller
Screen 1 (Red)