iOS DeCal : Lecture 3

Multiview Applications

February 21, 2017

Overview : Today's Lecture

Multiview Applications + Segues

Passing data between Views

Tab Bars and Controllers

Navigation Bars and Controllers

Announcements

Lab 3 due date pushed to this Thursday (11:59pm)

No new lab will be assigned this week

Instructors and TA's will still be at lab this week to help you with Lab 3 and the Hangman Project

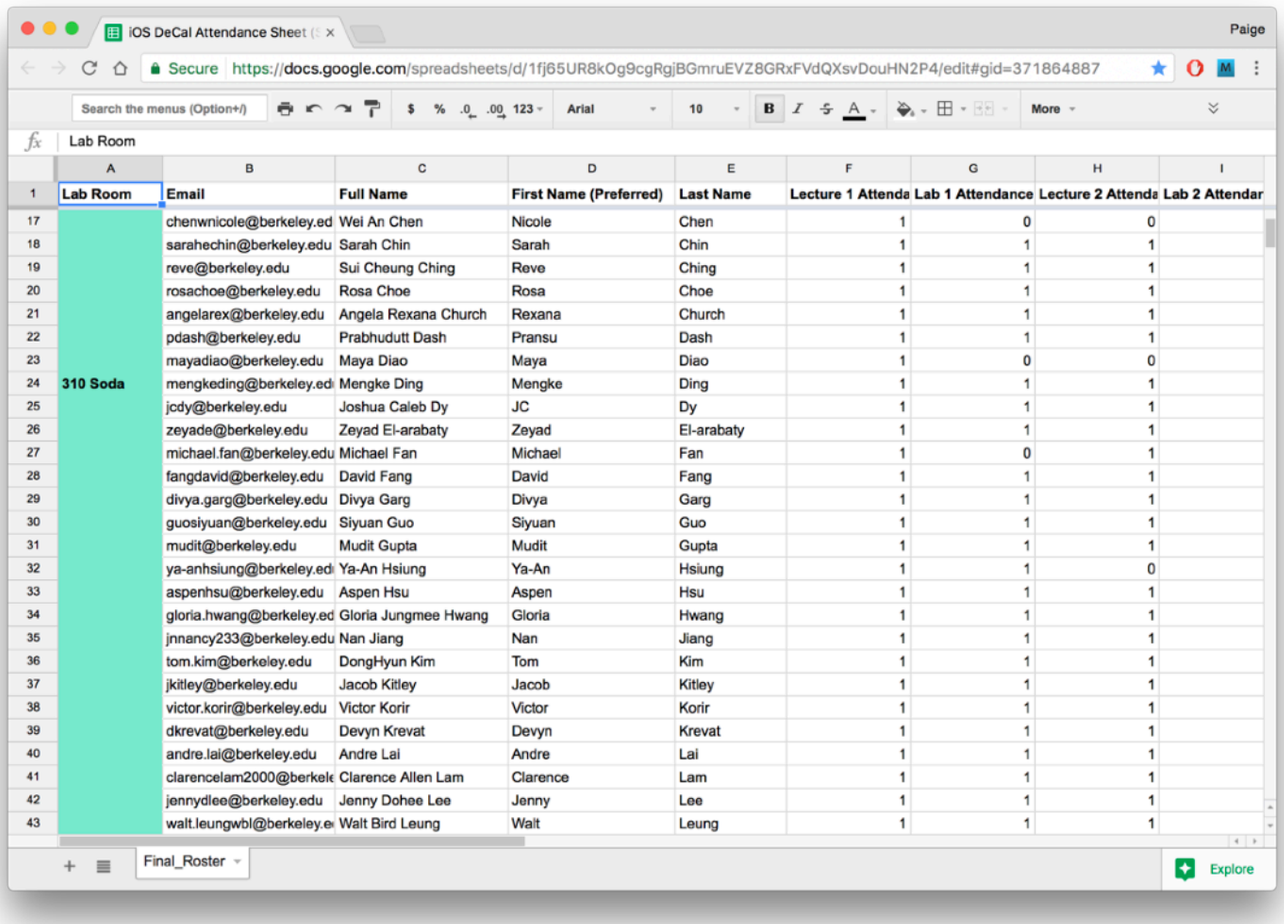
Make sure you submit to Gradescope even if you were checked off

Project 1 due next Tuesday

Submit via Gradescope

Lab 2 Grades posted on Github

Announcements



	A	B	C	D	E	F	G	H	I
1	Lab Room	Email	Full Name	First Name (Preferred)	Last Name	Lecture 1 Attenda	Lab 1 Attendance	Lecture 2 Attenda	Lab 2 Attendar
17		chenwnicole@berkeley.ed	Wei An Chen	Nicole	Chen	1	0	0	
18		sarahechin@berkeley.edu	Sarah Chin	Sarah	Chin	1	1	1	
19		reve@berkeley.edu	Sui Cheung Ching	Reve	Ching	1	1	1	
20		rosachoe@berkeley.edu	Rosa Choe	Rosa	Choe	1	1	1	
21		angelarex@berkeley.edu	Angela Rexana Church	Rexana	Church	1	1	1	
22		pdash@berkeley.edu	Prabhudutt Dash	Pransu	Dash	1	1	1	
23		mayadiao@berkeley.edu	Maya Diao	Maya	Diao	1	0	0	
24	310 Soda	mengkeding@berkeley.ed	Mengke Ding	Mengke	Ding	1	1	1	
25		jcdy@berkeley.edu	Joshua Caleb Dy	JC	Dy	1	1	1	
26		zeyade@berkeley.edu	Zeyad El-arabaty	Zeyad	El-arabaty	1	1	1	
27		michael.fan@berkeley.edu	Michael Fan	Michael	Fan	1	0	1	
28		fangdavid@berkeley.edu	David Fang	David	Fang	1	1	1	
29		divya.garg@berkeley.edu	Divya Garg	Divya	Garg	1	1	1	
30		guosiyuan@berkeley.edu	Siyuan Guo	Siyuan	Guo	1	1	1	
31		mudit@berkeley.edu	Mudit Gupta	Mudit	Gupta	1	1	1	
32		ya-anhsiung@berkeley.ed	Ya-An Hsiung	Ya-An	Hsiung	1	1	0	
33		aspenhsu@berkeley.edu	Aspen Hsu	Aspen	Hsu	1	1	1	
34		gloria.hwang@berkeley.ed	Gloria Jungmee Hwang	Gloria	Hwang	1	1	1	
35		jnnancy233@berkeley.edu	Nan Jiang	Nan	Jiang	1	1	1	
36		tom.kim@berkeley.edu	DongHyun Kim	Tom	Kim	1	1	1	
37		jkitley@berkeley.edu	Jacob Kitley	Jacob	Kitley	1	1	1	
38		victor.korir@berkeley.edu	Victor Korir	Victor	Korir	1	1	1	
39		dkrevat@berkeley.edu	Devyn Krevat	Devyn	Krevat	1	1	1	
40		andre.lai@berkeley.edu	Andre Lai	Andre	Lai	1	1	1	
41		clarencelam2000@berkele	Clarence Allen Lam	Clarence	Lam	1	1	1	
42		jennydlee@berkeley.edu	Jenny Dohee Lee	Jenny	Lee	1	1	1	
43		walt.leungwbl@berkeley.e	Walt Bird Leung	Walt	Leung	1	1	1	

Attendance records published: link

Let us know if we made any issues

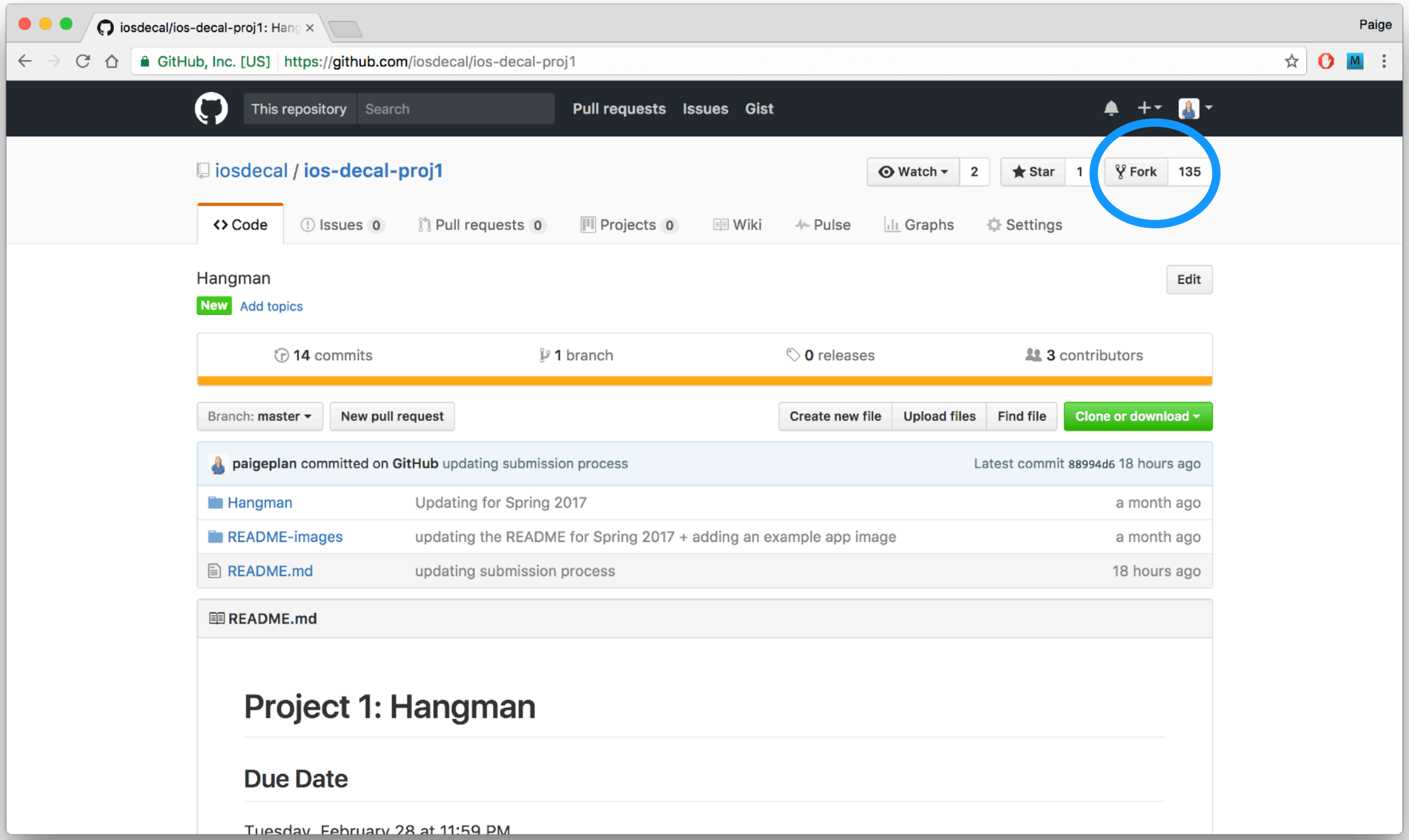
Remember that students with more than 3 unexcused absences will receive an NP

Announcements - Gradescope

Please submit using the GitHub or BitBucket feature on Gradescope

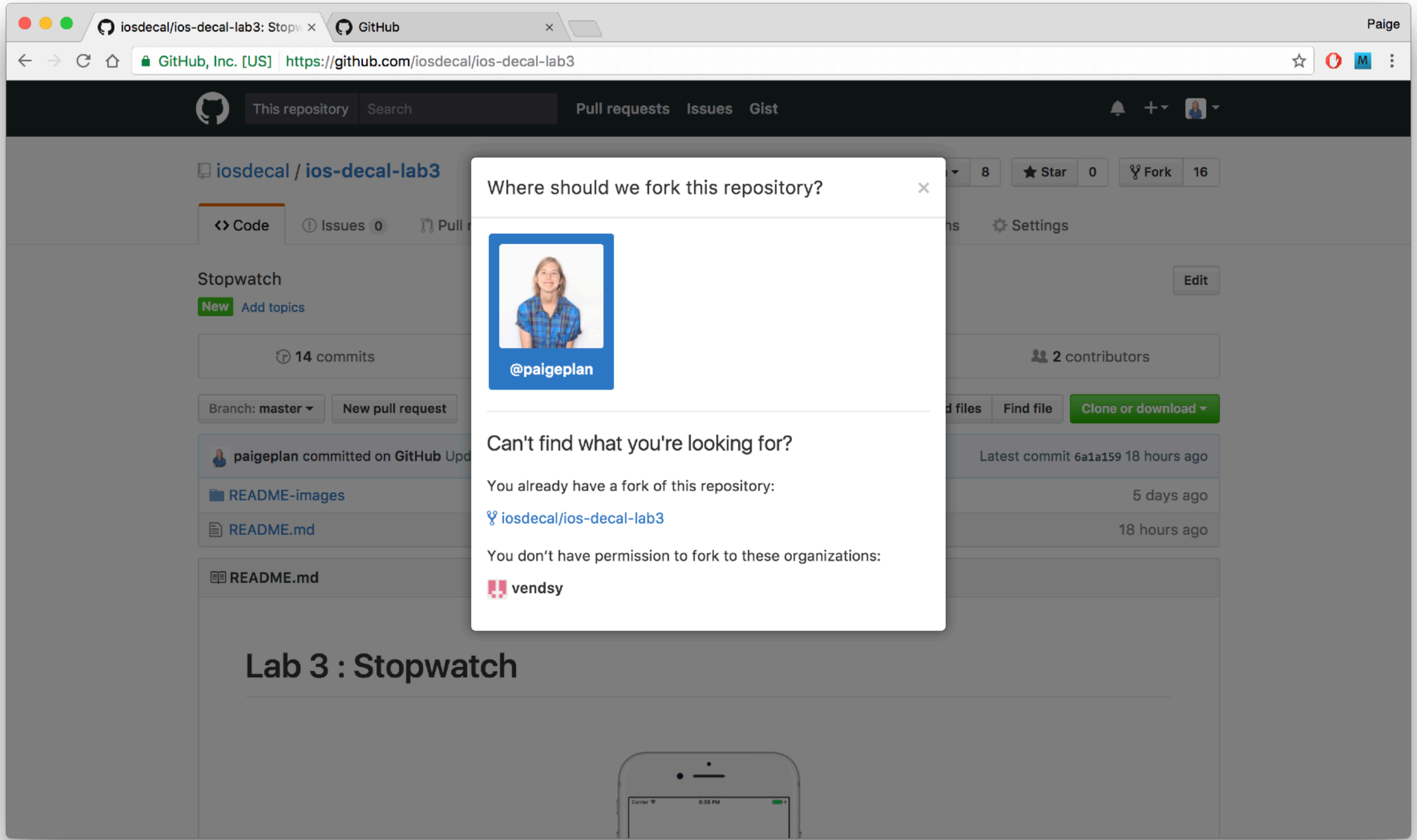
Though we will still allowing zip submissions, we've been finding more bugs in student submissions that use this feature (so keep in mind we may have to ask you re-submit)

How to submit to Gradescope with GitHub



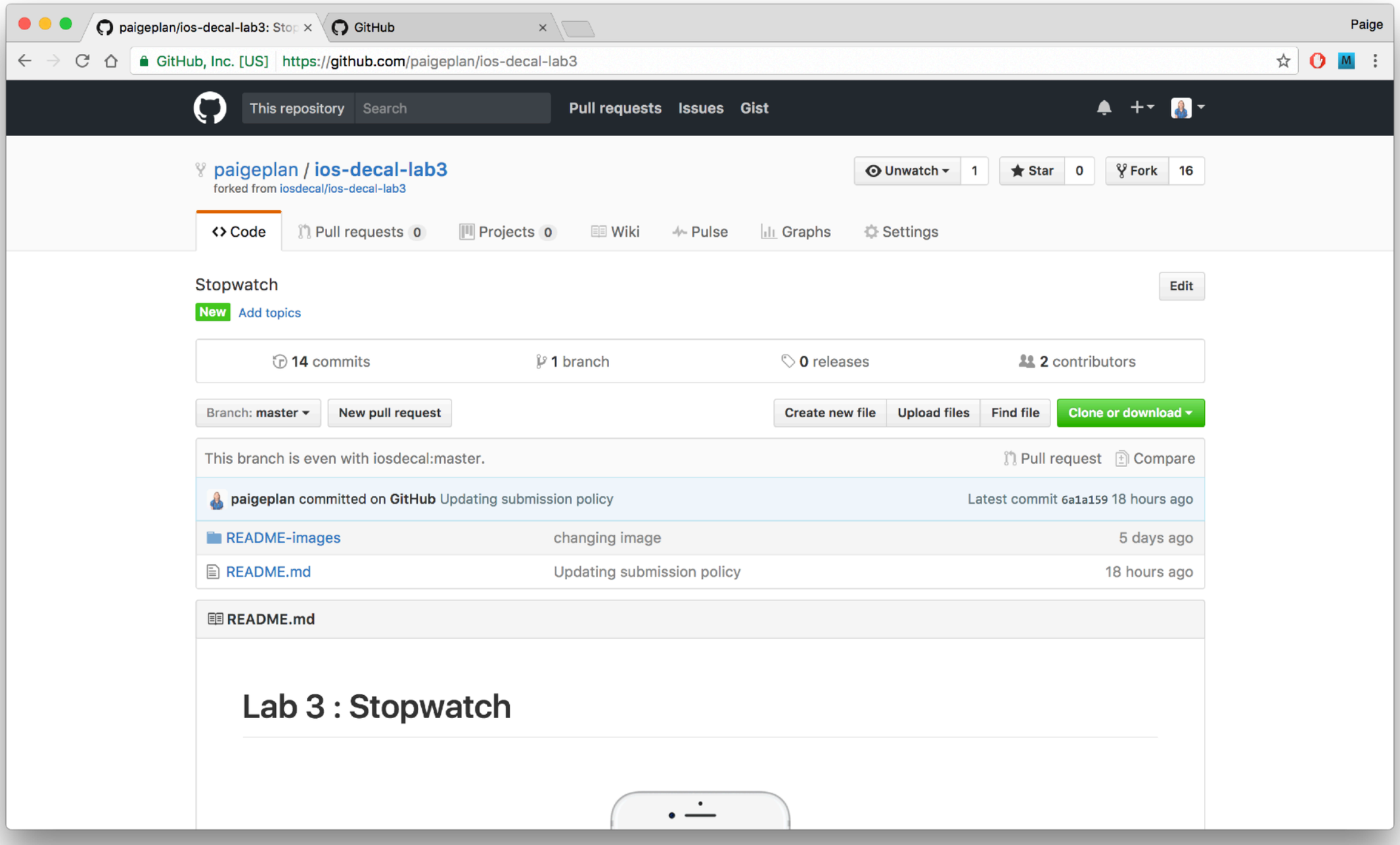
Step 1: Fork the project repository (tap the fork button)

How to submit to Gradescope with GitHub



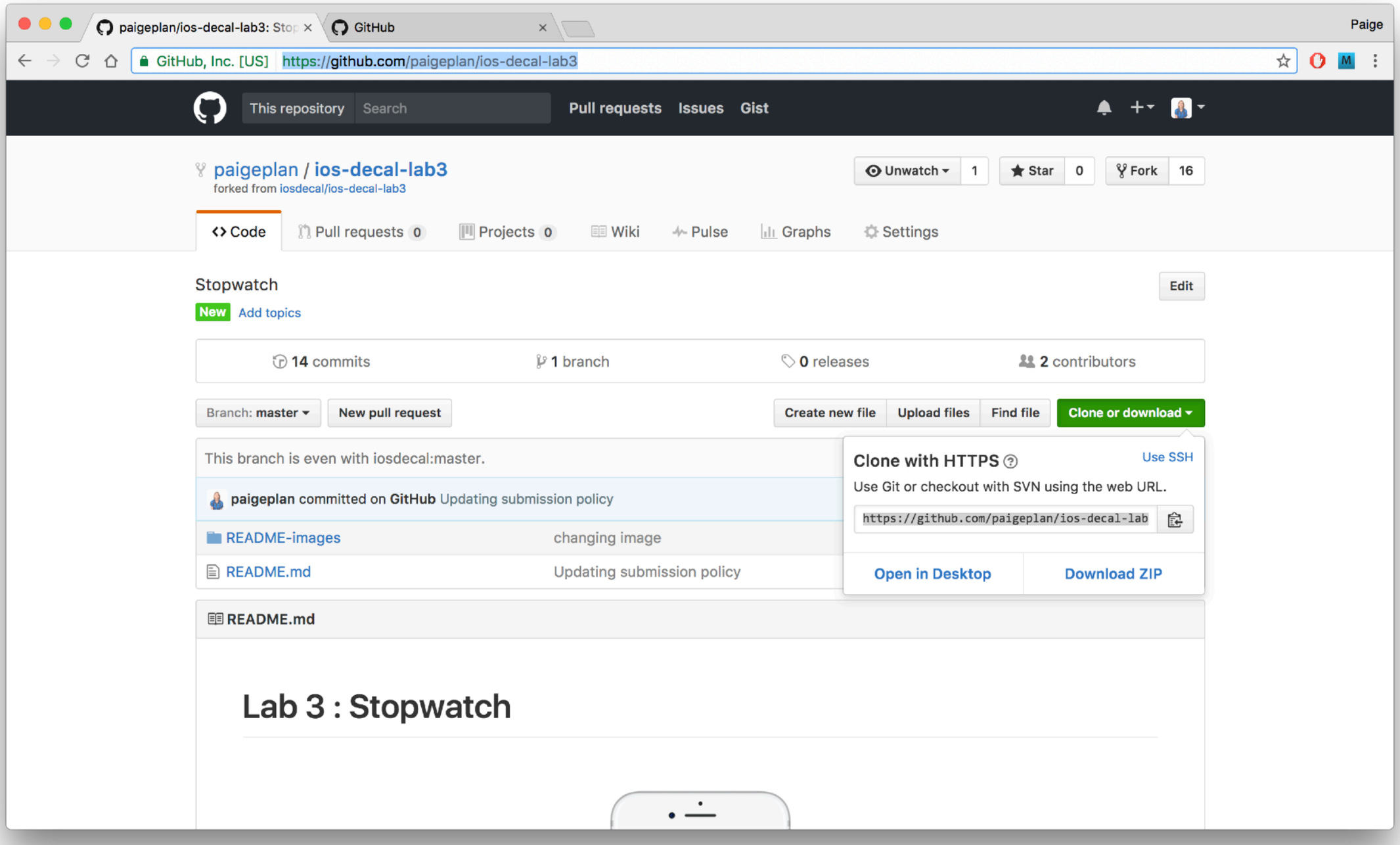
Step 2: Click on your account you wish to submit from

How to submit to Gradescope with GitHub



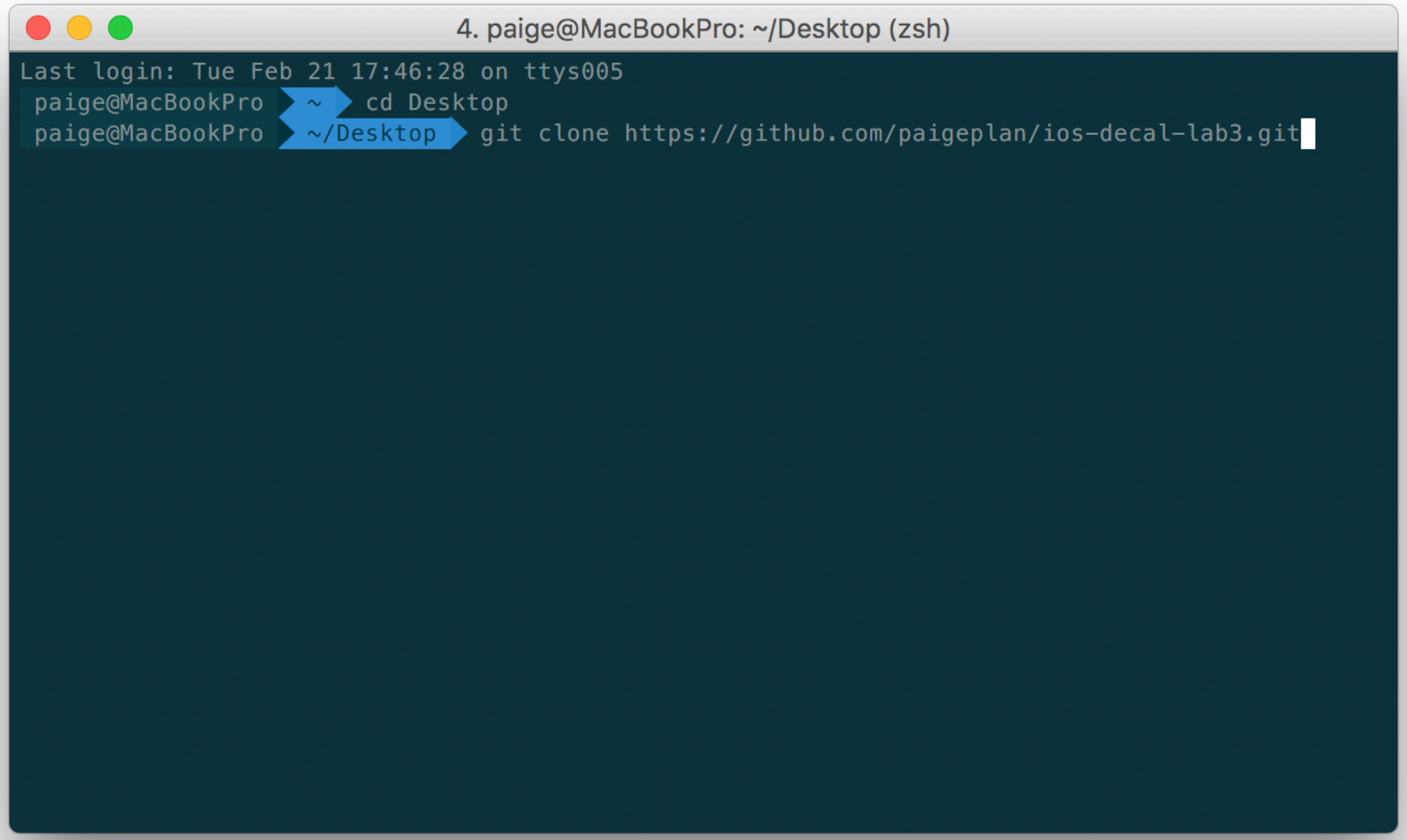
Step 3: Now you have your own repository containing the assignment.

How to submit to Gradescope with GitHub



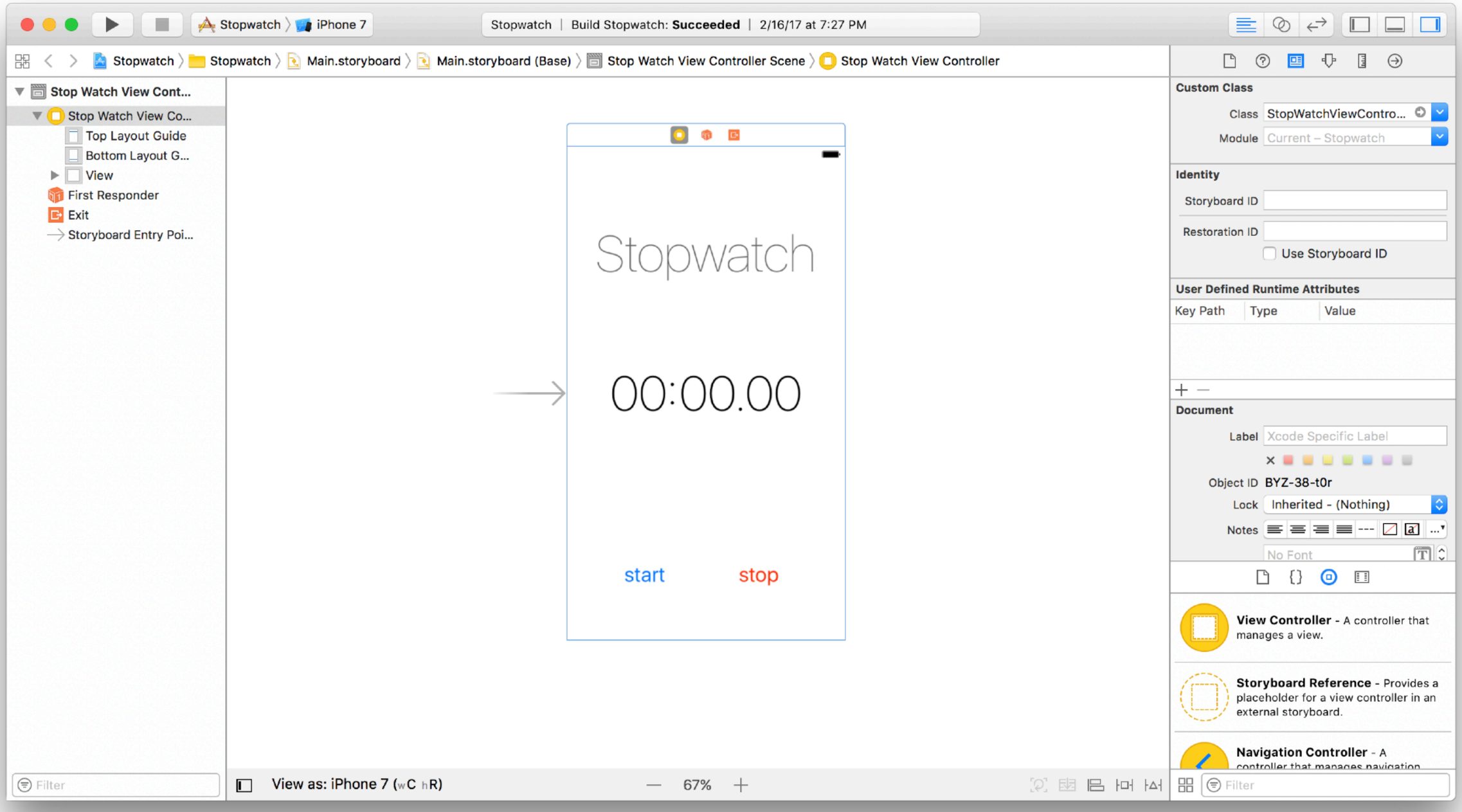
Step 4: Tap on “Clone or Download” and copy the link.

How to submit to Gradescope with GitHub



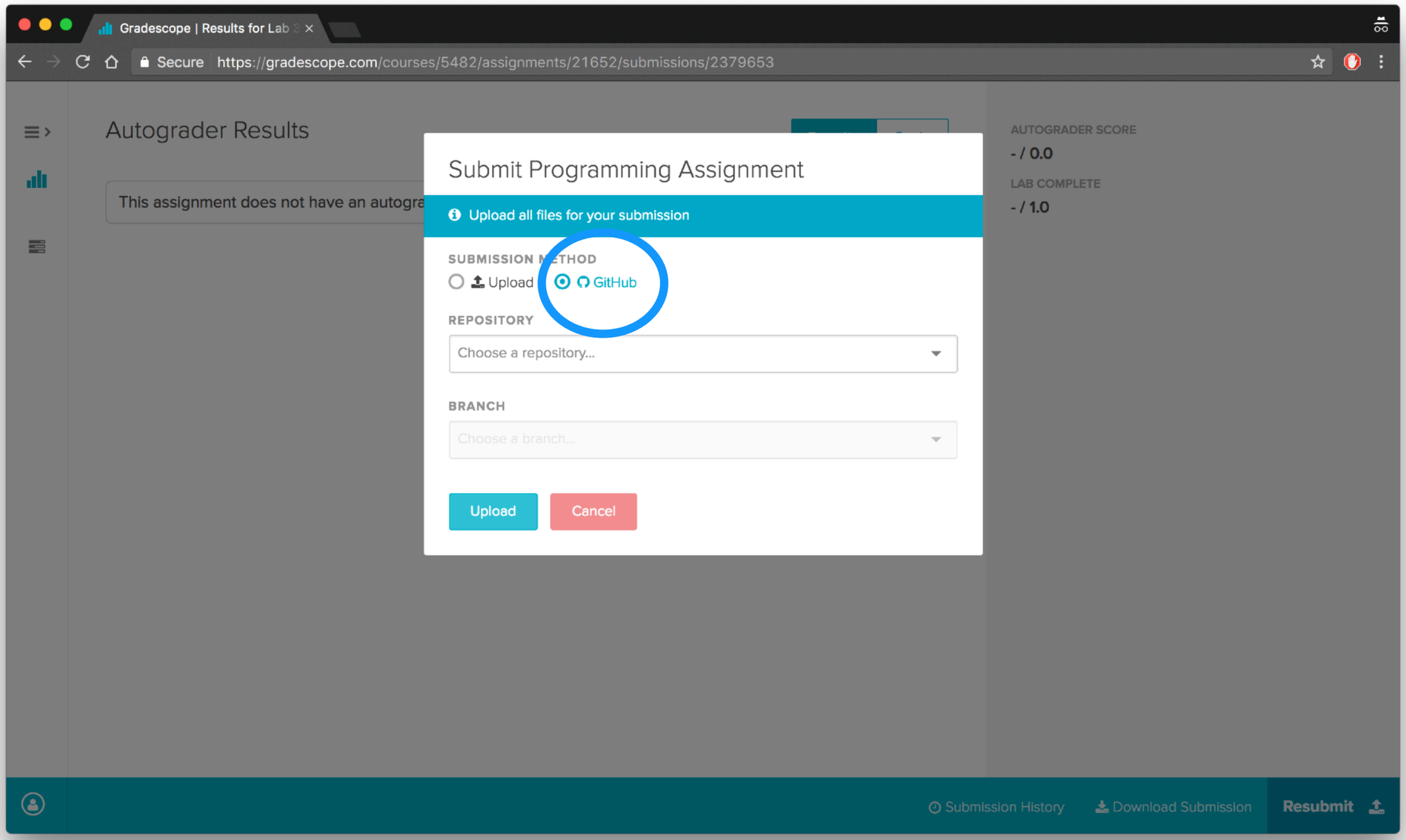
Step 5: In terminal, type git clone [link copied from step 4] to get a copy of this repository on your computer

How to submit to Gradescope with GitHub



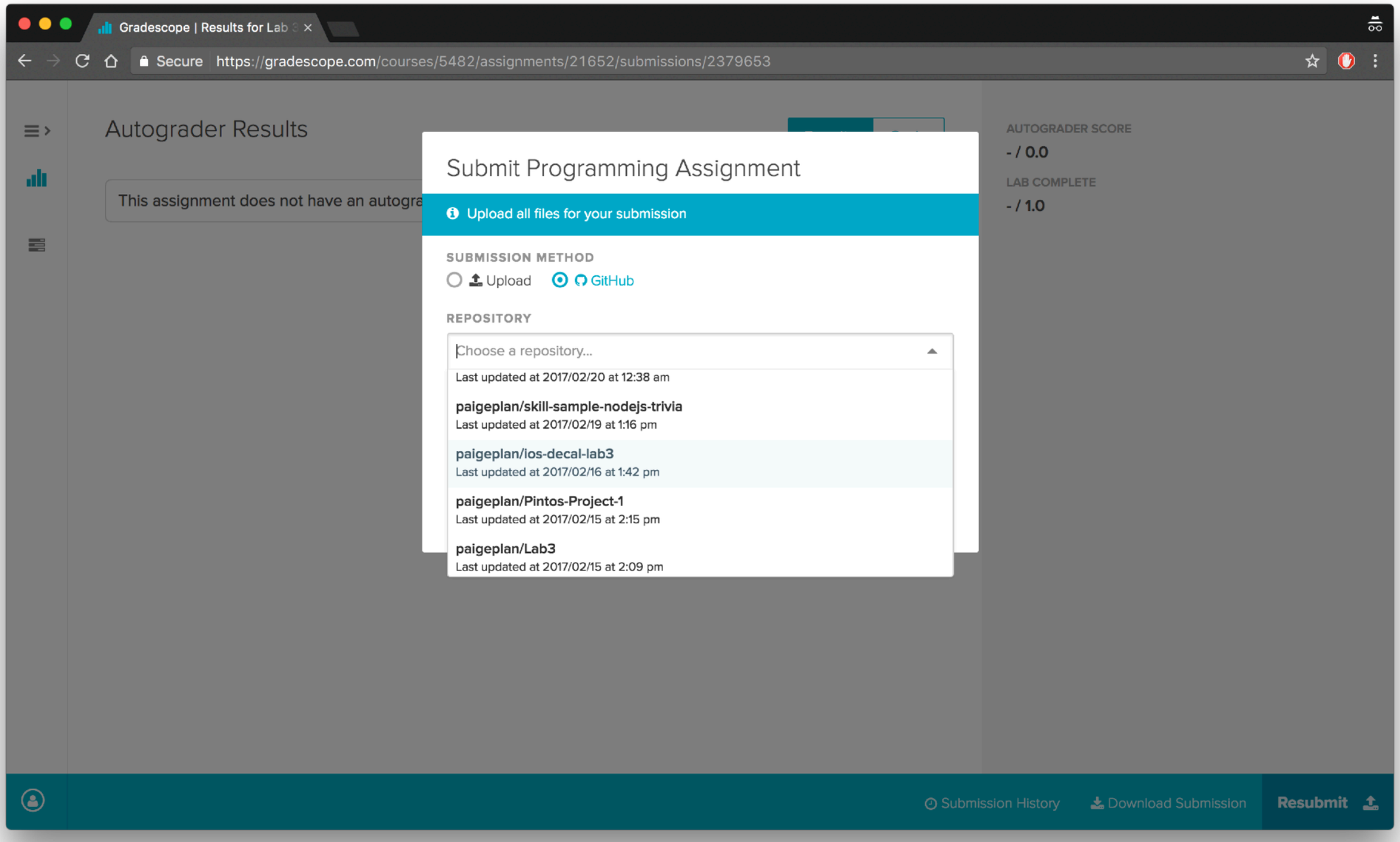
Step 6: Start working on the assignment, making commits and pushing them as you go

How to submit to Gradescope with GitHub



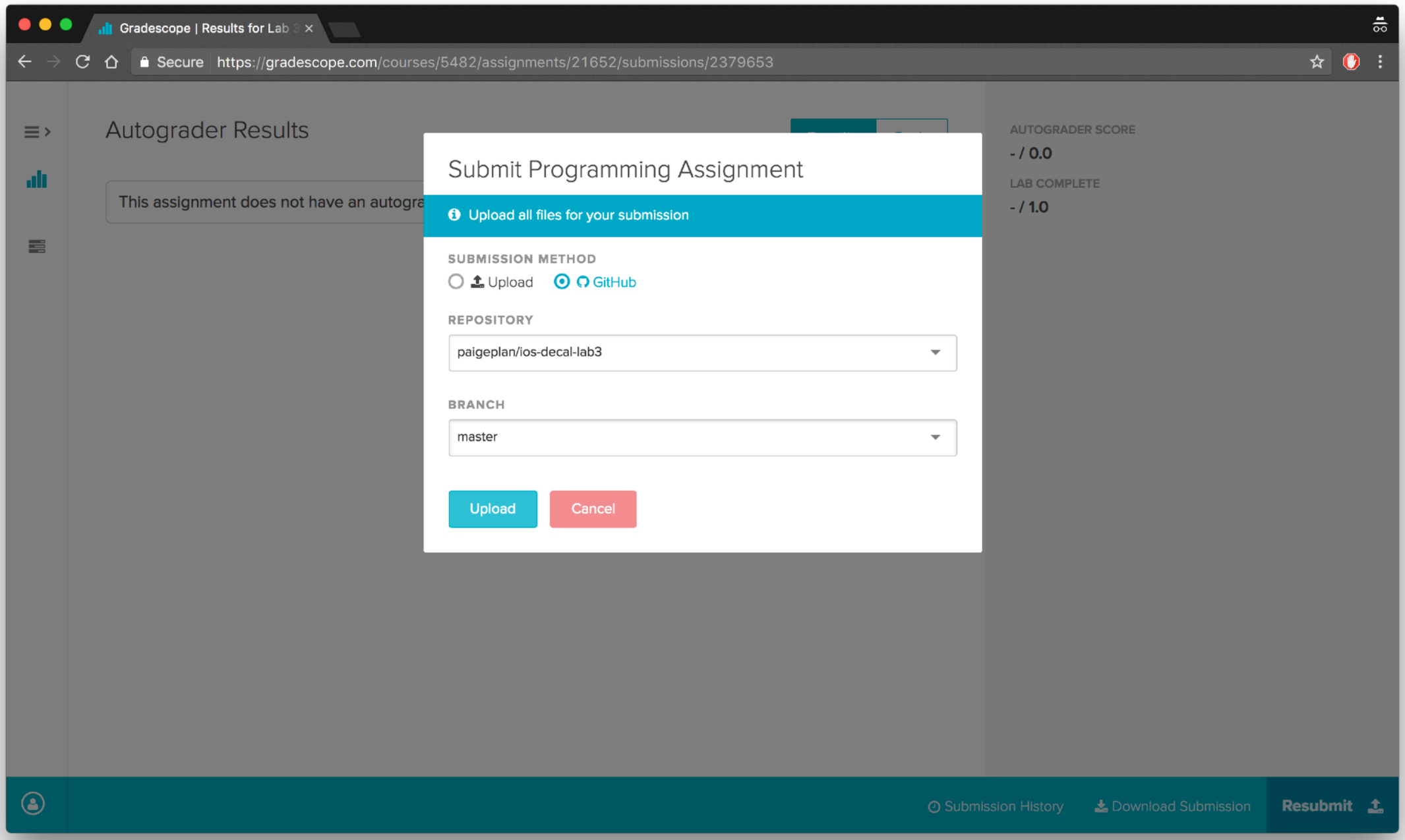
Step 7: Once done, commit/push your changes and go to Gradescope. Select the Github Submission option

How to submit to Gradescope with GitHub



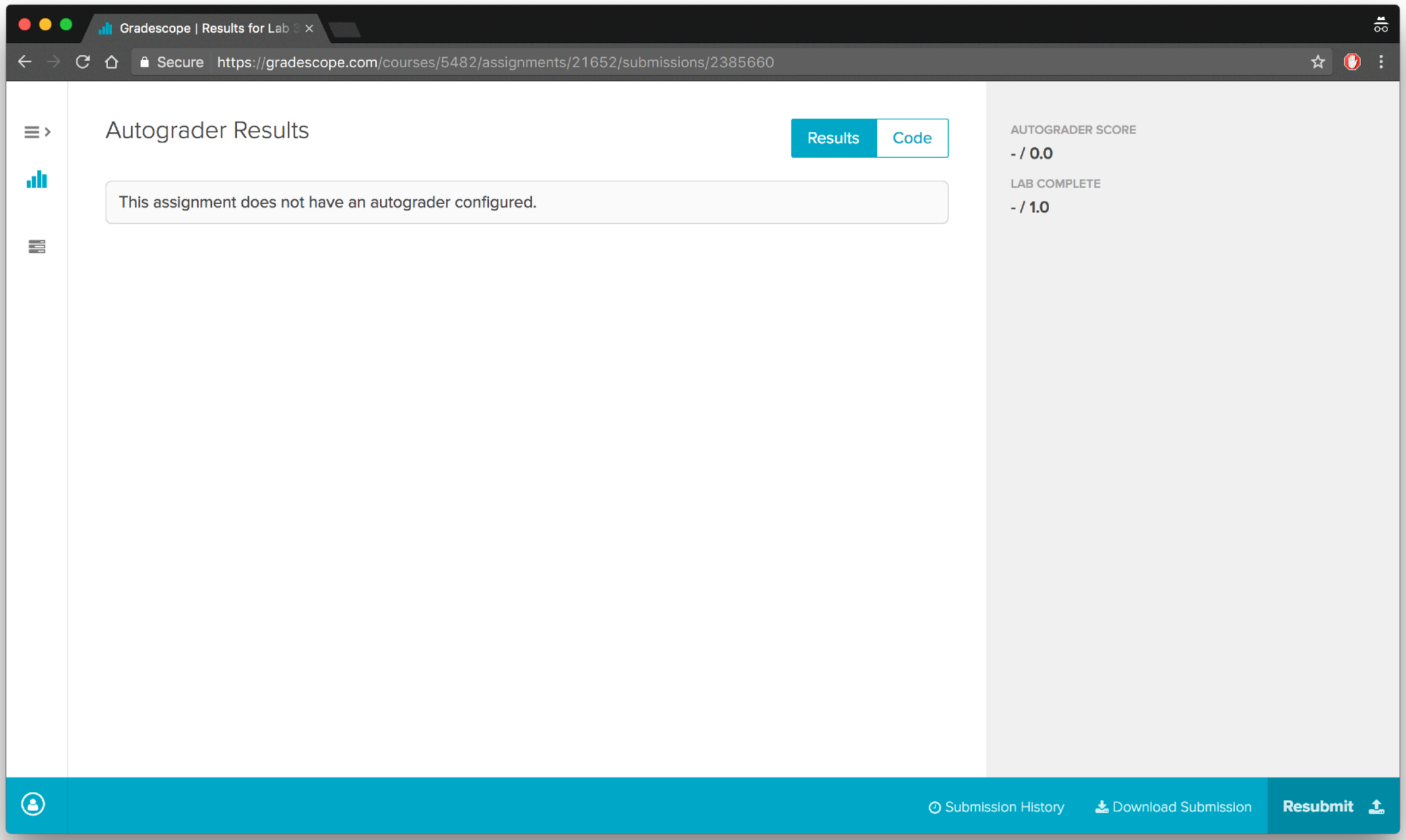
Step 8: Select your forked repository in the drop down

How to submit to Gradescope with GitHub



Step 9: Choose the branch you have your finished app on

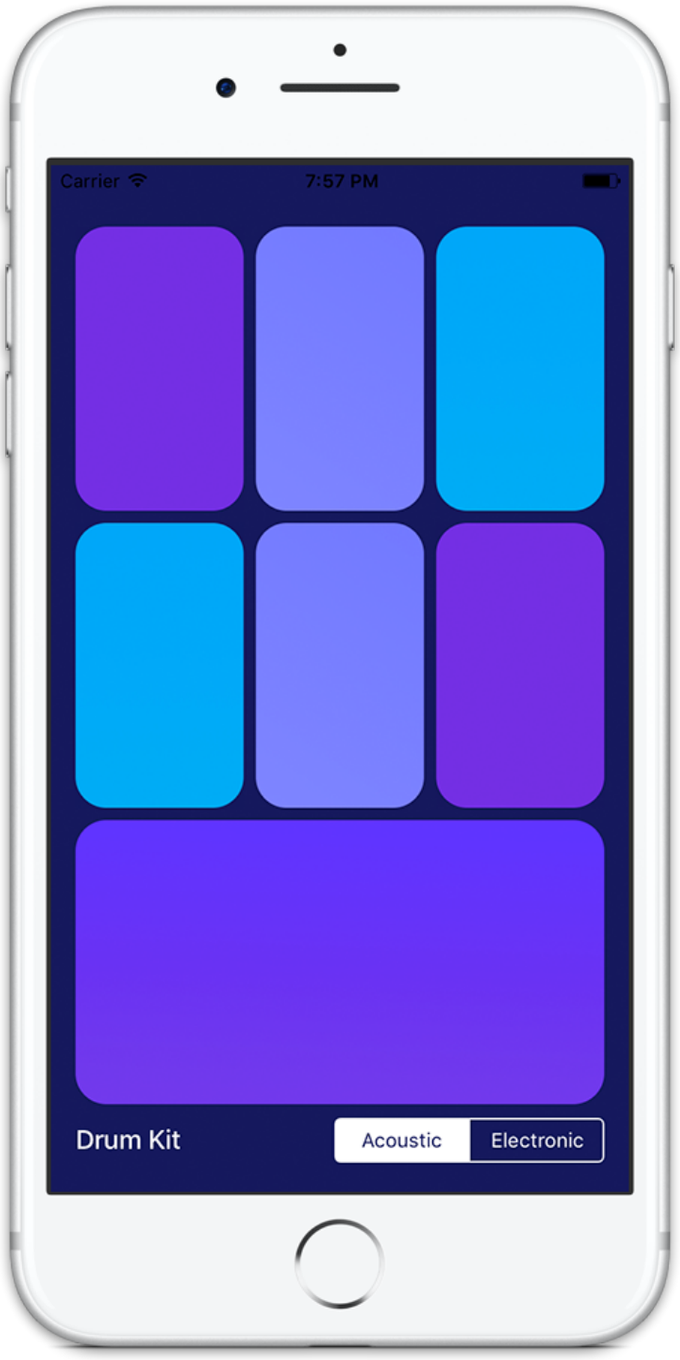
How to submit to Gradescope with GitHub



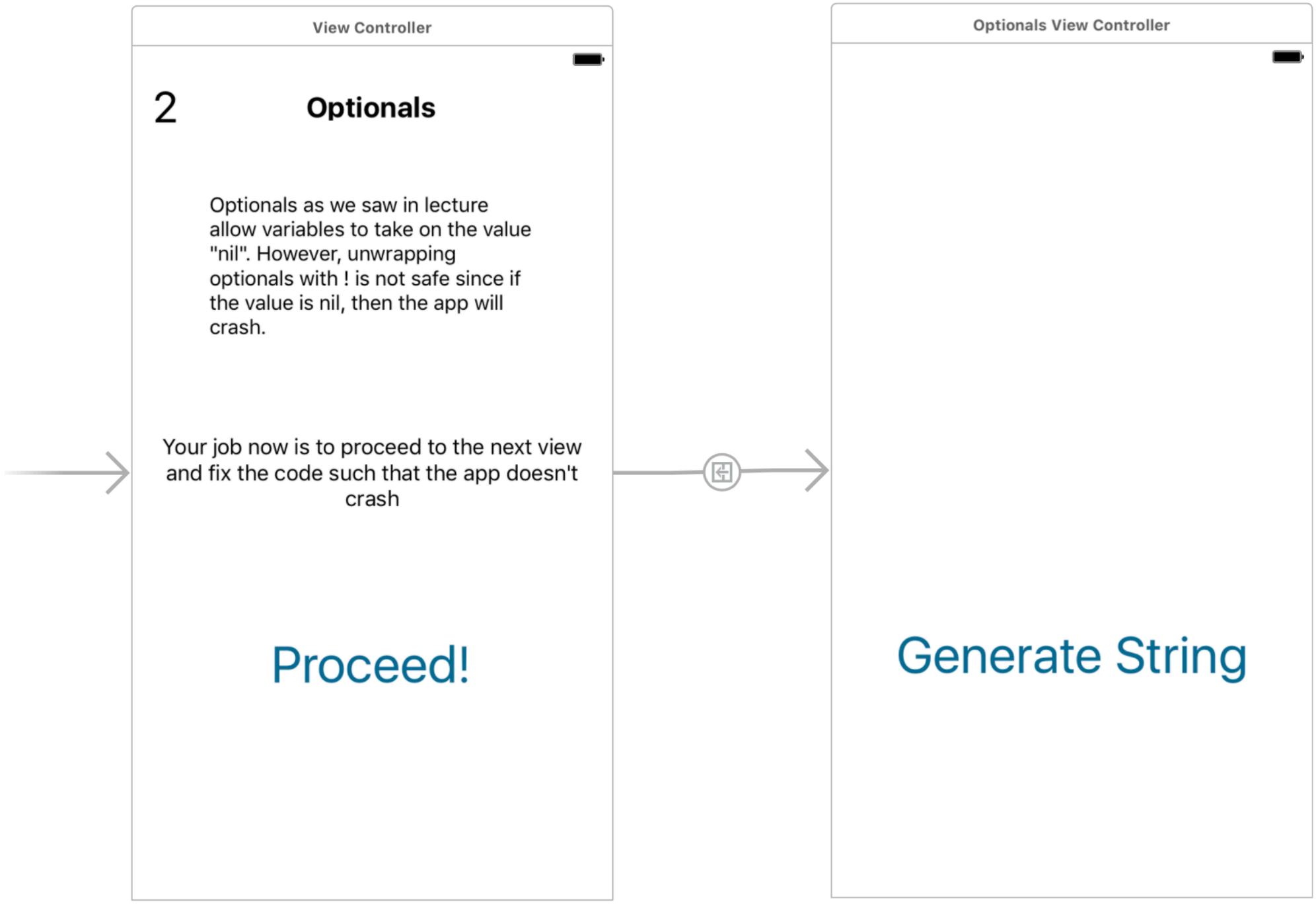
Step 10: Download your submission and make sure it's working (bugs are not rare, so please do!)

Multiview Applications

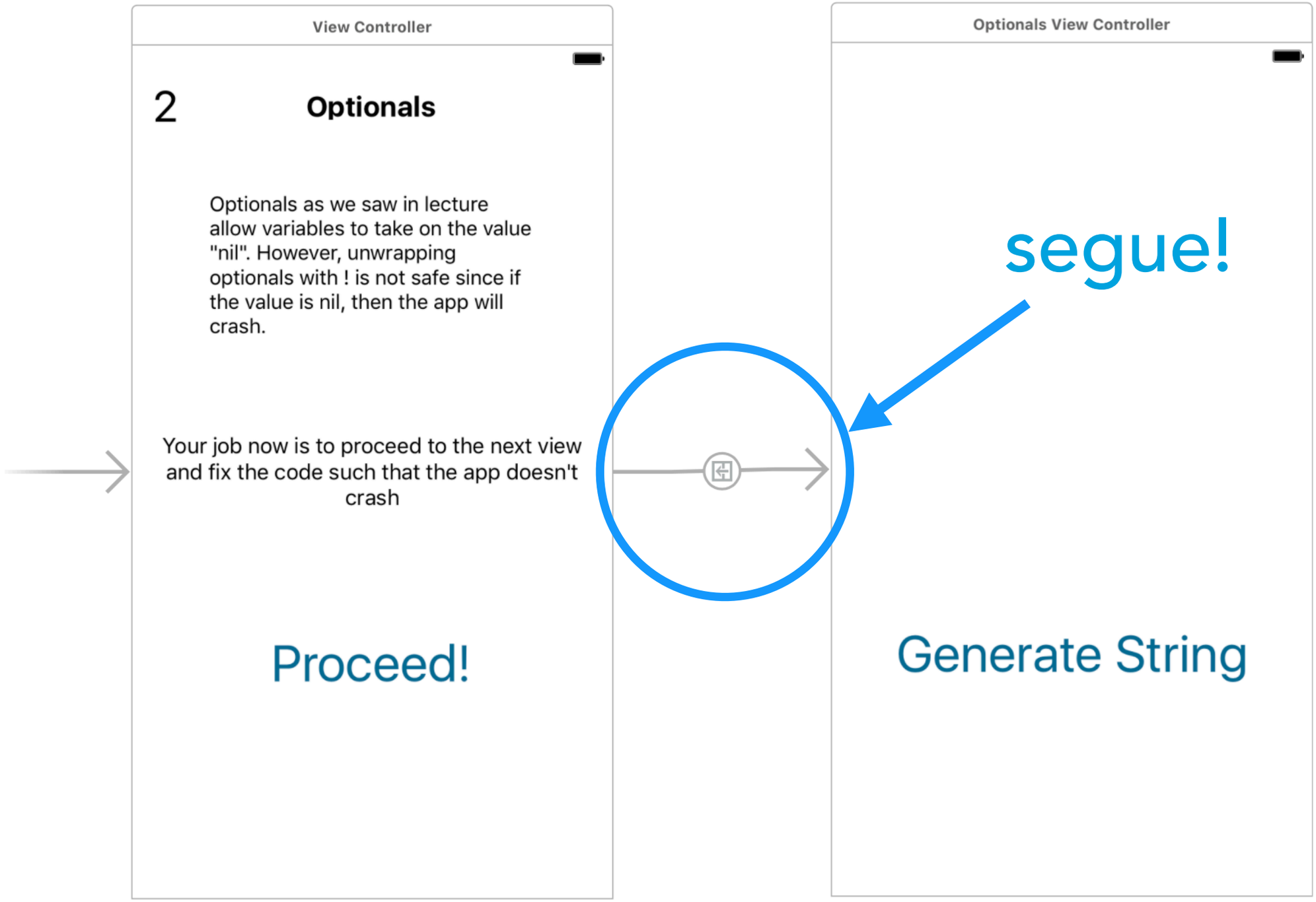
So Far - Single View Applications



Multi-view Applications : (Lab 1)



Multi-view Applications : (Lab 1)

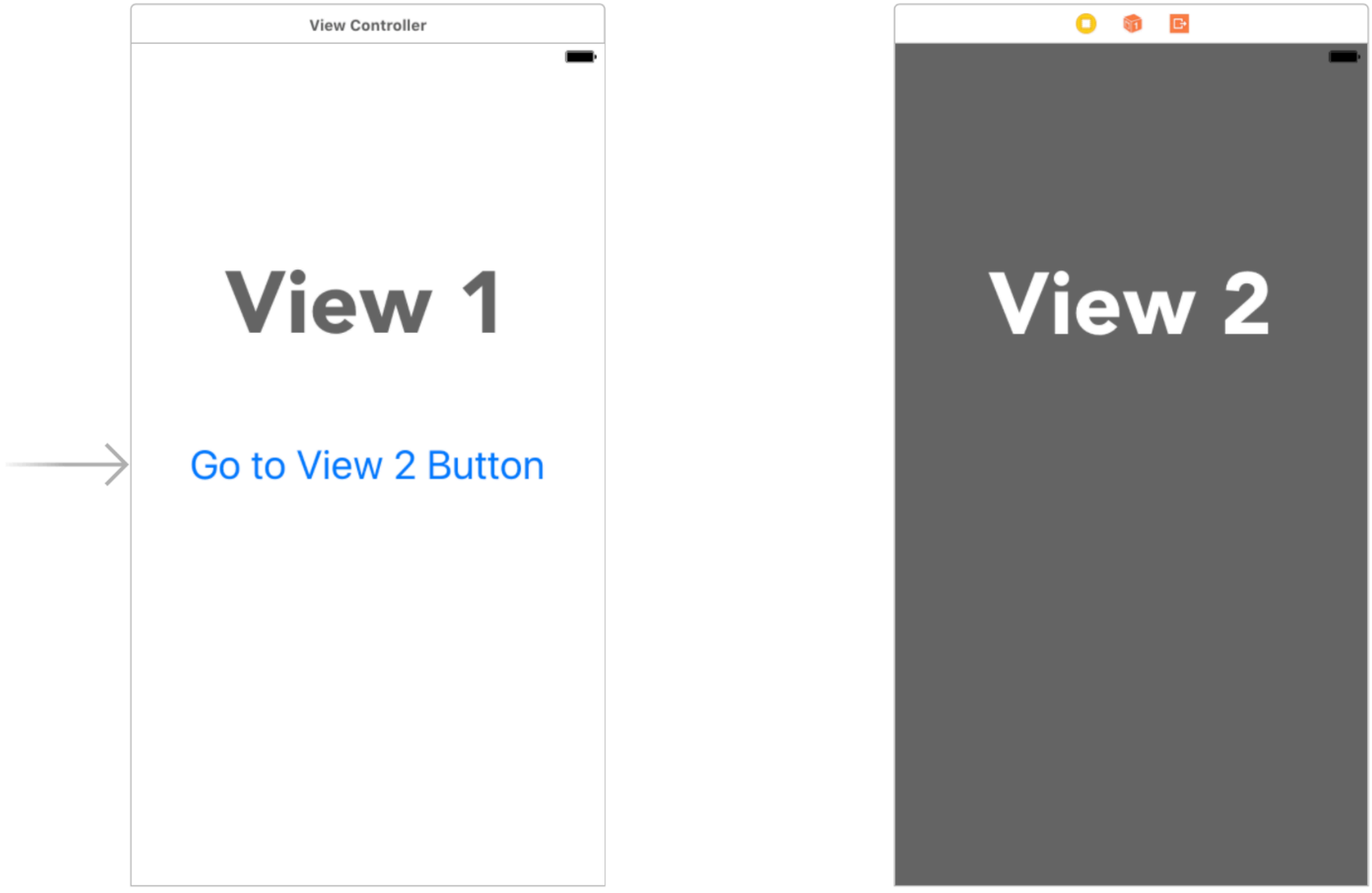


Multi-view Applications

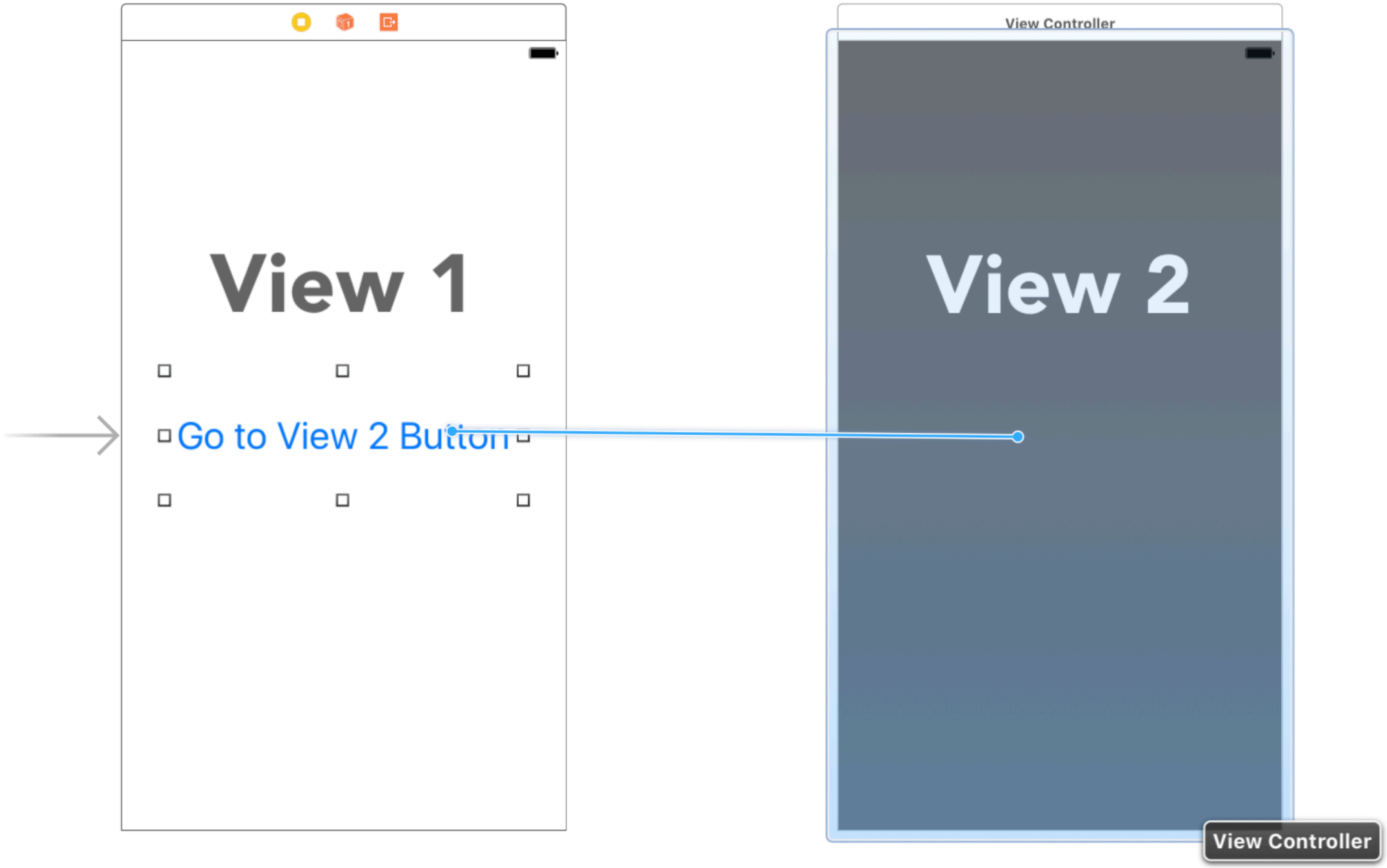
Multi-view applications are made up of multiple MVC's stringed together

To communication and transitioning between MVC's, create **segues** between View Controllers in Storyboard

Each time a user triggers a segue, a **new instance** of an MVC will be created (not an old instance).
More on this later!

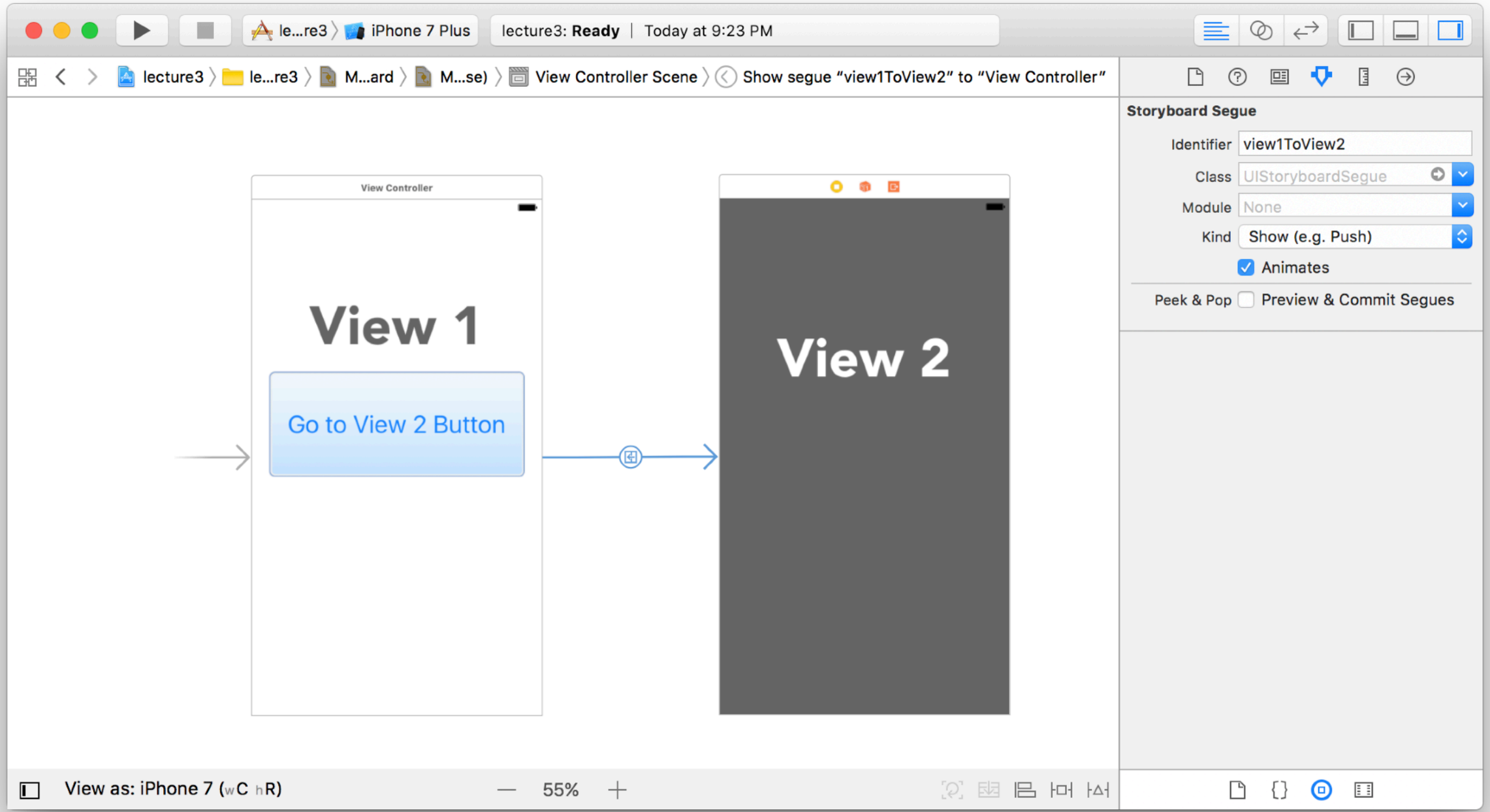


Segue Creation: Control + drag from an instigator (typically a button) from one MVC to another MVC

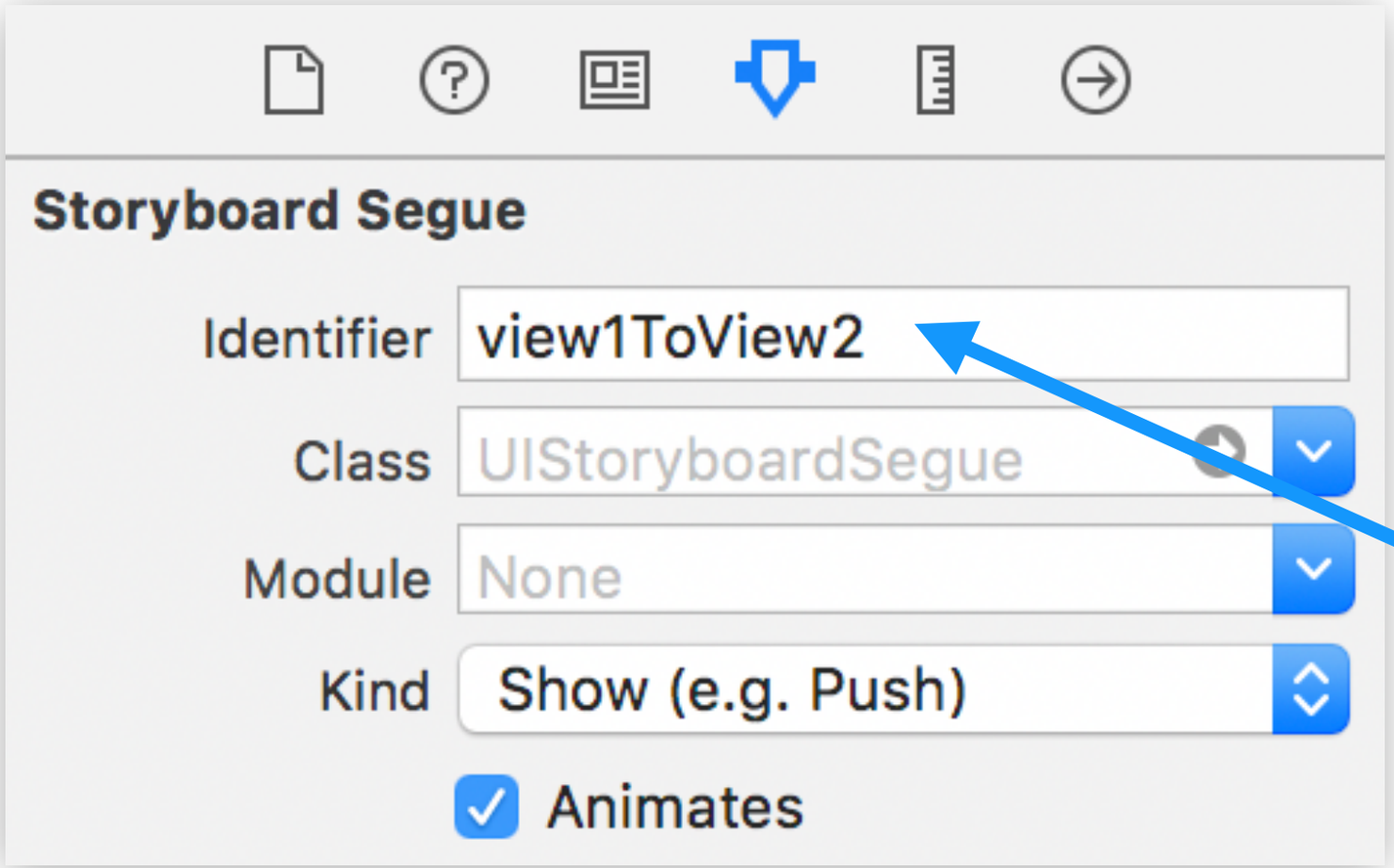


Segue Creation: Control + drag from an instigator (typically a button) from one MVC to another MVC

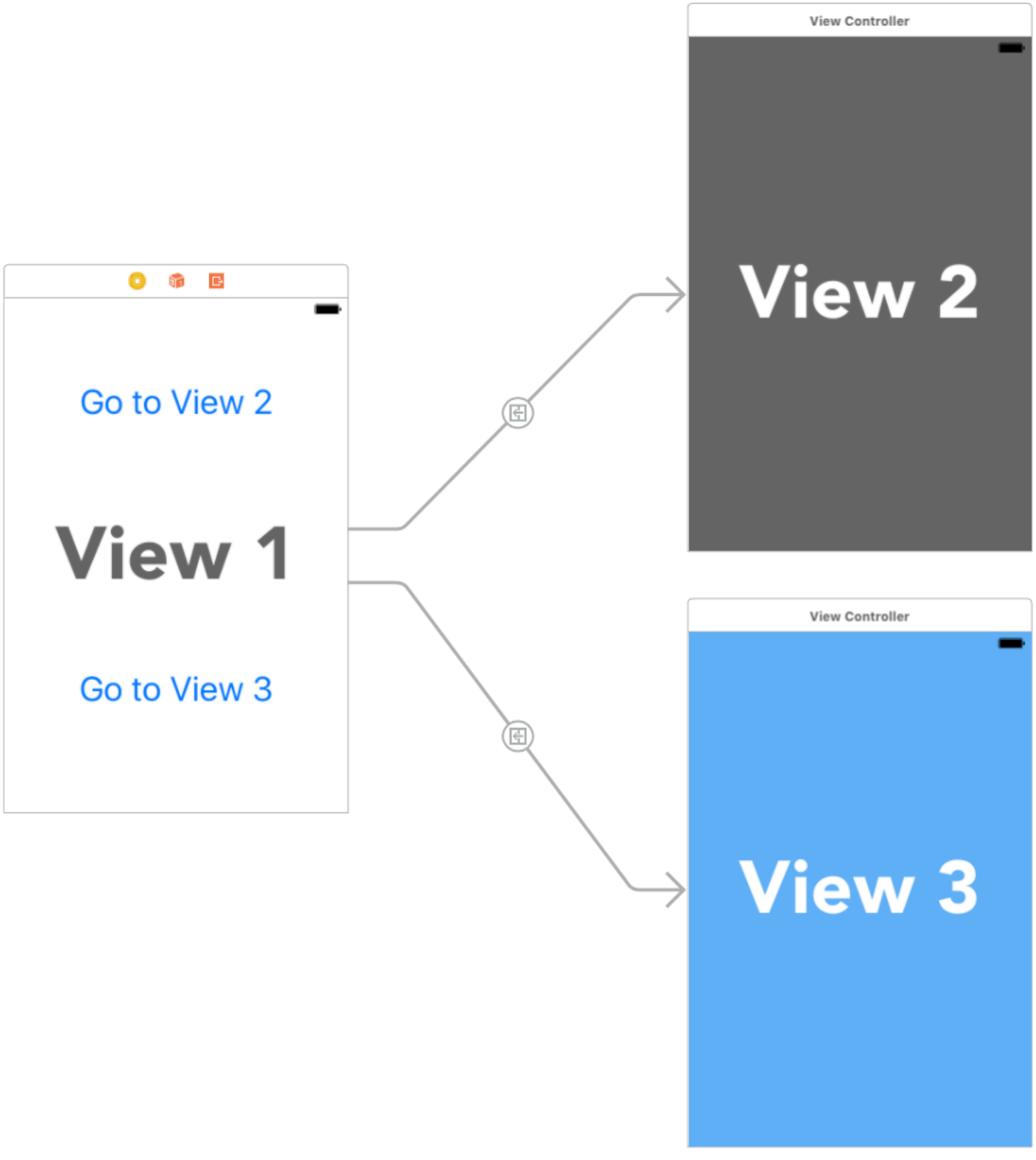
Segue Identification: Tap on a segue in Storyboard, and add a **Identifier** in the **Attributes Inspector**



Segue Identification: Tap on a segue in Storyboard, and add a **Identifier** in the **Attributes Inspector**



you can use this identifier to reference your segue in code



Segue Identifiers are useful to distinguish between which MVC you are going to (one MVC can have a segue to multiple others)

Passing data between MVC's

When the user taps on a button that triggers a segue, the MVC they will transition to will be a **new instance of that MVC**.

Passing data between MVC's

When the user taps on a button that triggers a segue, the MVC they will transition to will be a **new instance of that MVC**.

Therefore, we need to pass data from the first MVC to the new MVC **before** the segue takes place.

How do we do this? —> `prepareForSegue`

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
  }
}
```

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
    if let identifier = segue.identifier {
        if identifier == "goToMainView" {
            if let dest = segue.destination as? MainViewController {
                dest.somePropertyOnMainView = "hi!"
                dest.someMainViewSetupFunc()
            }
        }
        else if identifier == "goToSettings" {
            if let dest = segue.destination as? SettingsViewController {
                // do stuff in the settingsVC before it loads
            }
        }
    }
}
```

segue : The segue just instigated

sender : The segue's instigator (usually a button)

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

First, get the identifier we created in Storyboard by accessing `segue.identifier`

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

If the identifier is "`goToMainView`", we know we are heading to the `MainViewController`

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

We can get a reference to the destination View Controller by accessing `segue.destination`

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

type(segue.destination) is ViewController, we need to cast it as a MainViewController

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

Now we can set properties / call methods in the `MainViewController` before it's view loads

prepareForSegue : Example

```
func prepare(for segue: UIStoryboardSegue, sender: Any?) {
  if let identifier = segue.identifier {
    if identifier == "goToMainView" {
      if let dest = segue.destination as? MainViewController {
          dest.somePropertyOnMainView = "hi!"
          dest.someMainViewSetupFunc()
      }
    }
    else if identifier == "goToSettings" {
      if let dest = segue.destination as? SettingsViewController {
          // do stuff in the settingsVC before it loads
      }
    }
}
```

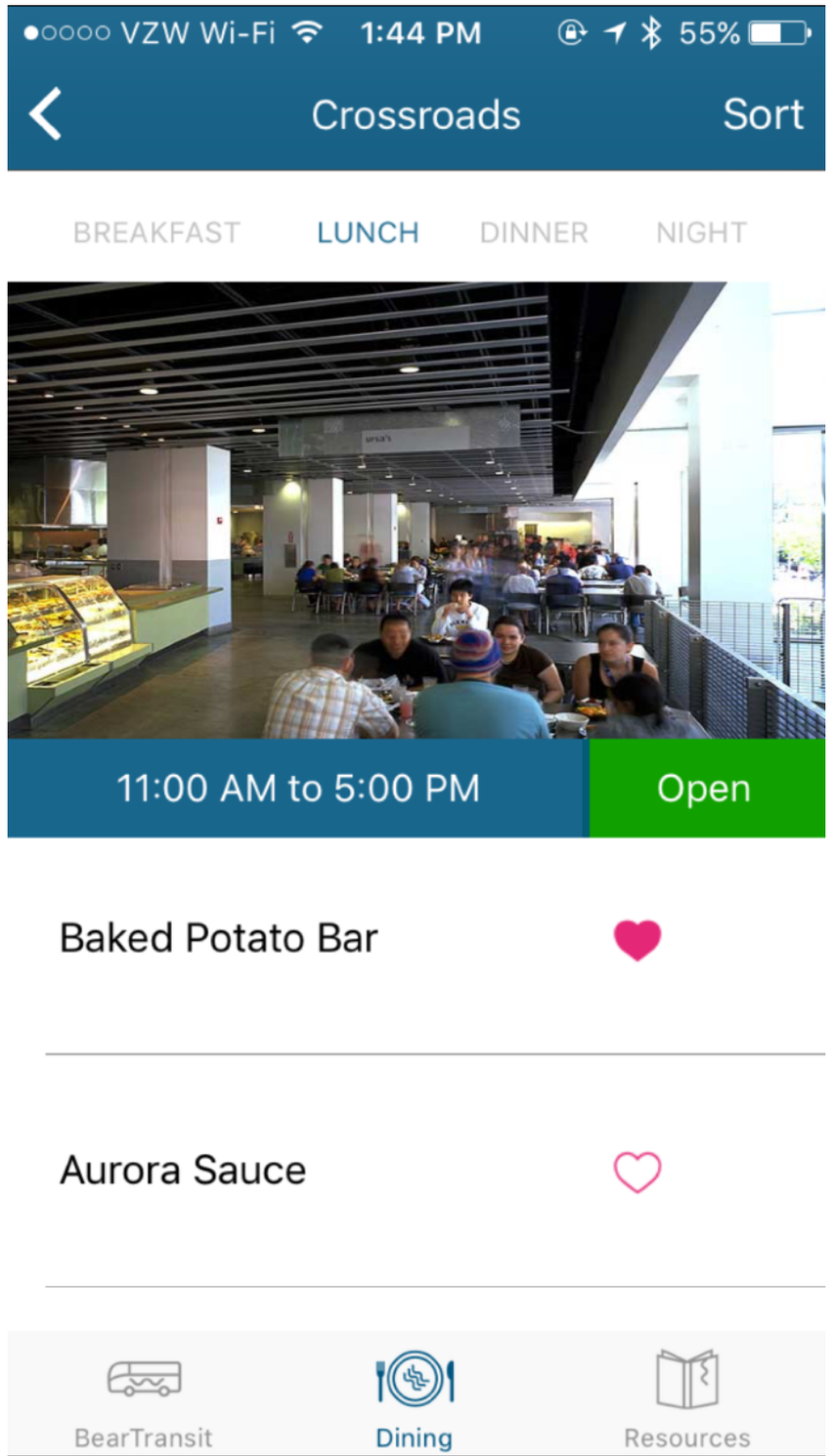
Since MVC's can segue to multiple other MVC's, `segue.identifier` can take on different values

Segue Demo

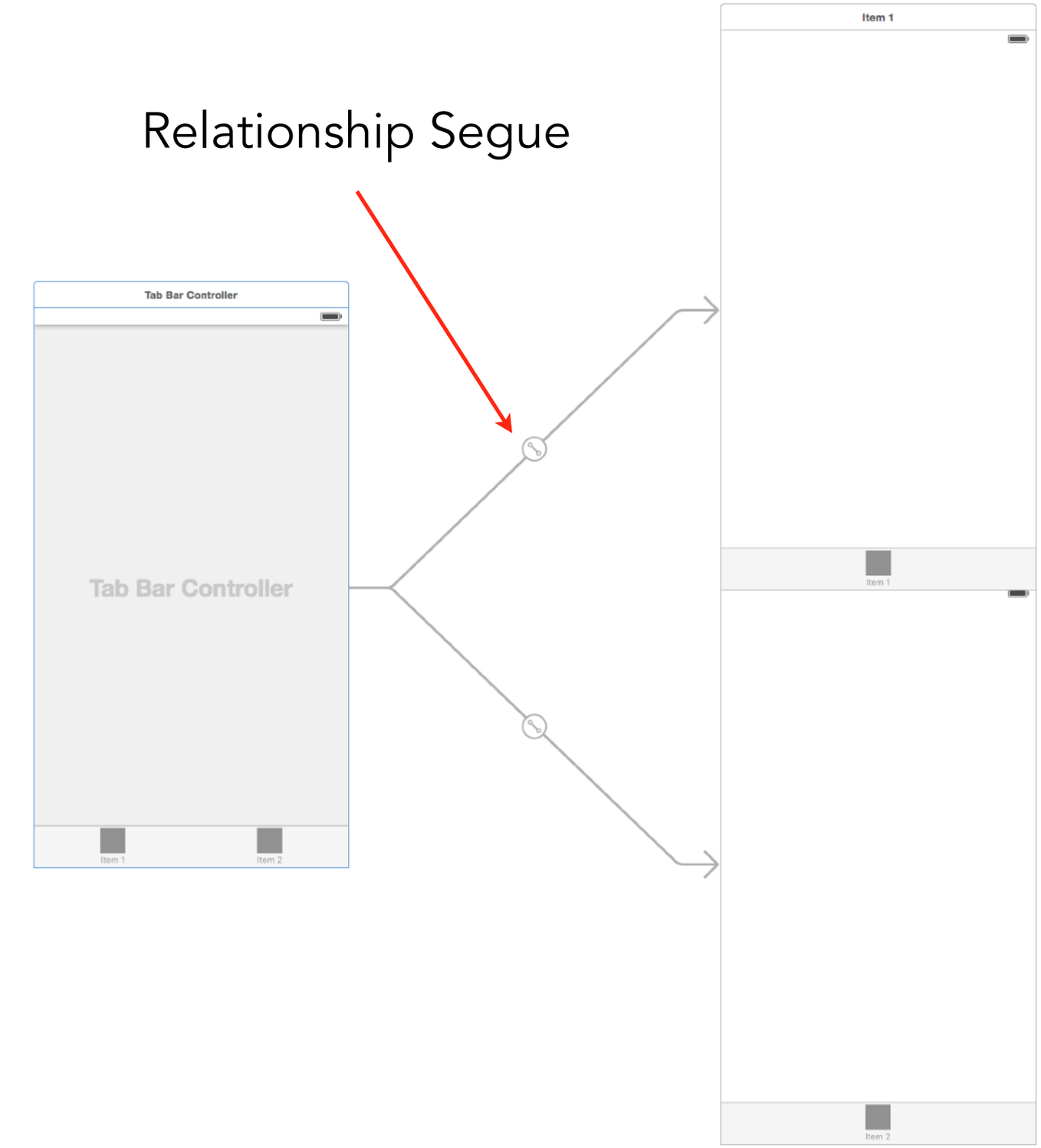
Source code link :

https://github.com/paigeplan/Segue-Demo

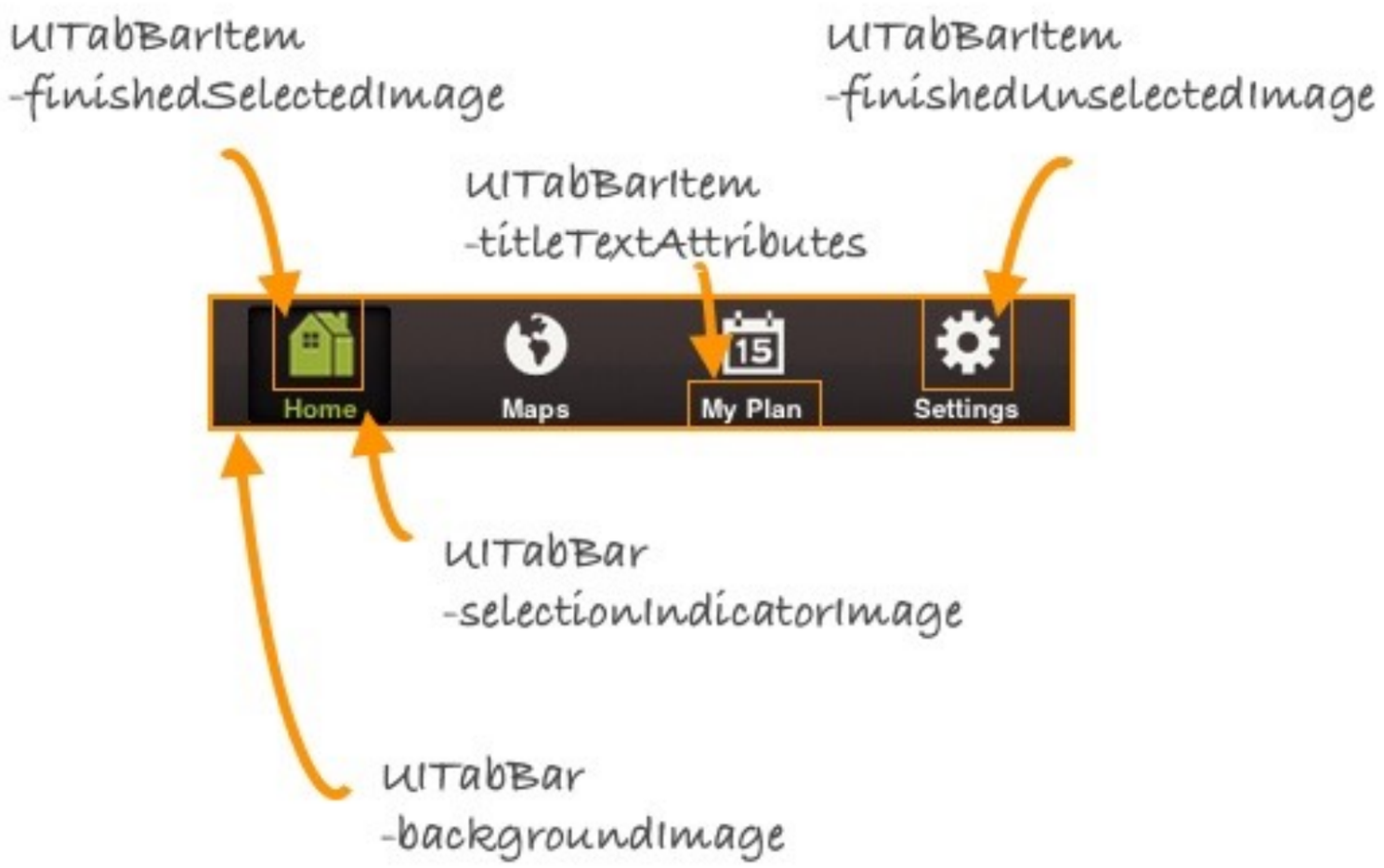
Navigation and Tab Bars


VZW Wi-Fi
1:44 PM
55%
Crossroads
Sort
BREAKFAST
LUNCH
DINNER
NIGHT
11:00 AM to 5:00 PM
Open
Baked Potato Bar
Aurora Sauce
BearTransit
Dining
Resources

Tab Bars


Relationship Segue
Tab Bar Controller
Tab Bar Controller
Item 1
Item 2
Item 1
Item 1
Item 2

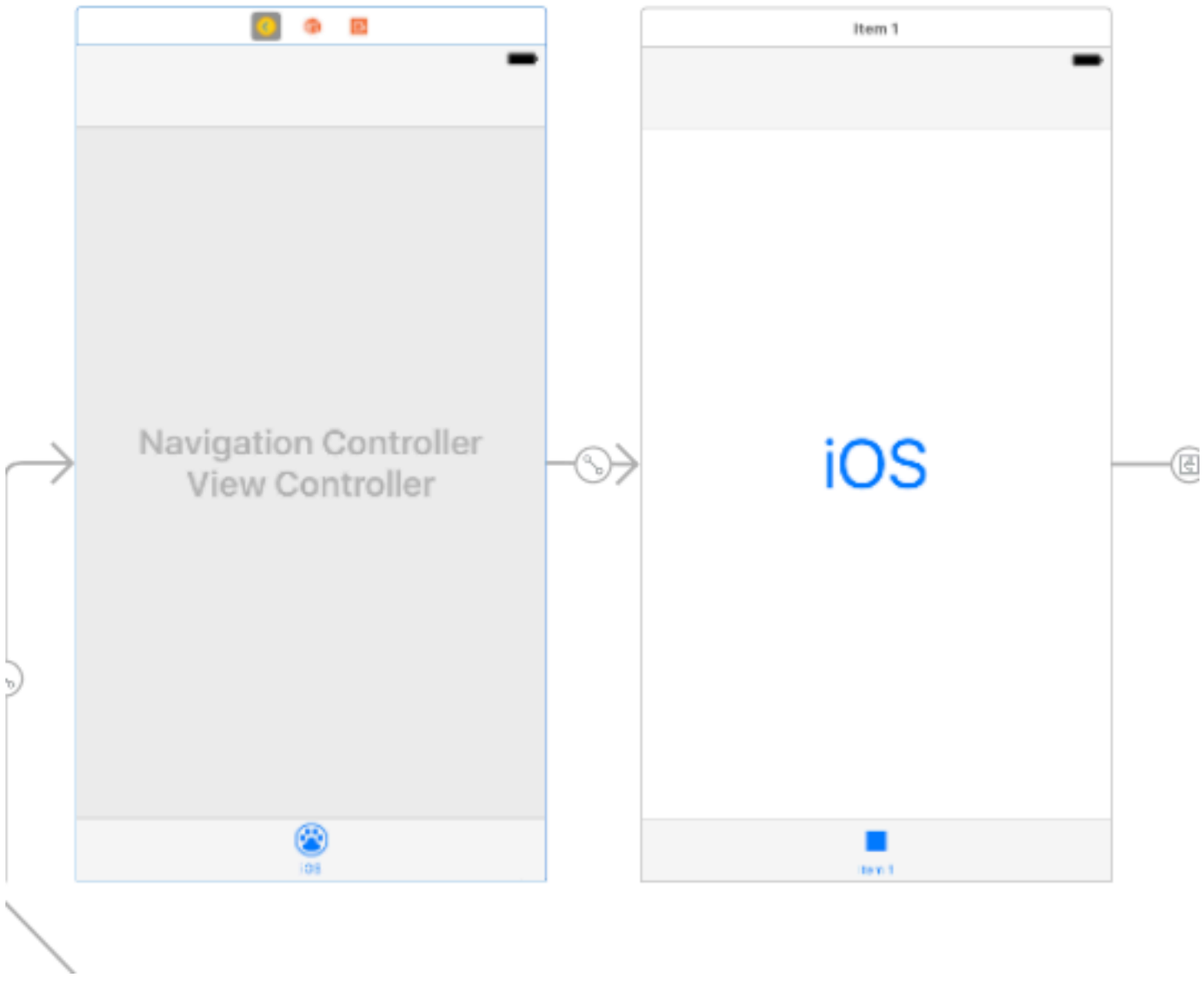
Tab Bars : Customization



Tab Bars : Customization

```
class AppDelegate: UIResponder, UIApplicationDelegate {

    var window: UIWindow?


    func application(_ application: UIApplication,
didFinishLaunchingWithOptions launchOptions:
[UIApplicationLaunchOptionsKey: Any]?) -> Bool {

        UITabBar.appearance().tintColor =
UIColor.lightGray//selected tab color
        UITabBar.appearance().backgroundImage =
UIImage(named:"tabbarbg.png")
        UITabBar.appearance().barTintColor =
UIColor.white

        return true
    }
```

Navigation Bars



Navigation Bars : Customization

```
class ViewController:
UIViewController

func viewDidLoad() {
self.navigationBar.barTintColor = UIColor(colorLiteralRed:
51/255, green: 90/255, blue: 149/255, alpha: 1)

self.navigationBar.titleTextAttributes =
[NSForegroundColorAttributeName: UIColor.white]

self.navigationBar.tintColor = UIColor.white
}
```

View Lifecycle

viewDidLoad()
viewWillAppear()
viewDidAppear()
viewWillDisappear()
viewDidDisappear()

someViewController.swift

Check-Ins

Demo

Even more custom?

- Define your own Tab Bar Class To Be Superview

https://guides.codepath.com/ios/Creating-a-Custom-Tab-Bar

- Remember: View Hierarchy

Lab 3 : Stopwatch

Due **this Thursday** at 11:59pm

(don't forget to submit to Gradescope,
even if you have checked off!)

Project 1 : Hangman

Due **next Tuesday** at 11:59pm

Next Lecture: Table Views and Collection Views