CSC 471 / 371 Mobile Application Development for iOS

Prof. Xiaoping Jia
School of Computing, CDM
DePaul University
xjia@cdm.depaul.edu
@DePaulSWEng

1

Building UI with Segues

2

Outline

- Multiple view apps
- Segues

3

An App with Two Scenes

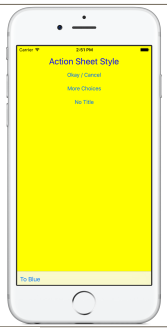
4

An App with Two Scenes – Using Segue

- Multiple scenes in a storyboard
- Using segue to switch between scenes
- The architecture
  - A storyboard with two scenes
    - The *Blue* and *Yellow* view controller
    - Each view controller has a toolbar and a button
  - The *Blue* view controller is the initial scene
  - A segue from the *Blue* scene to the *Yellow* scene

5

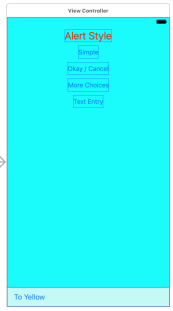
The Blue and Yellow Screens



6

The Initial Scene in Storyboard

- Start with a single view app
- Edit the initial scene in the storyboard
  - Set the background color
  - Add a *Toolbar* and a *Bar Button Item*
  - Add a *Label* and *Buttons*
  - Add constraints

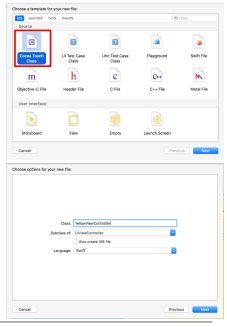


DEPAUL UNIVERSITY 7

7

Add a New View Controller

- New | File … | iOS Source | Cocoa Touch Class
- Choose options for the file
  - Class: `YellowViewController`
  - Subclass of: `UIViewController`
  - Uncheck "Also create XIB file"
  - Language: Swift



DEPAUL UNIVERSITY 8

8

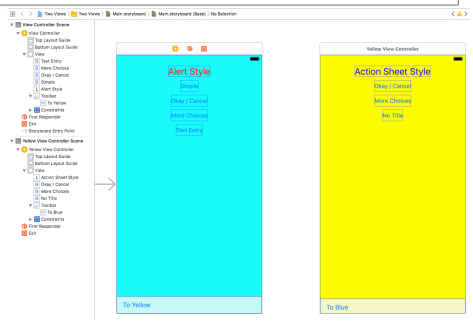
Add a New Scene

- Add a scene in the storyboard
  - Drag a *view controller*
- Set class to `YellowViewController`
- Edit the scene
  - Set the background color
  - Add a *Toolbar* and a *Bar Button Item*
  - Add a *Label* and *Buttons*
  - Add constraints

DEPAUL UNIVERSITY 9

9

The Storyboard with Two Scenes

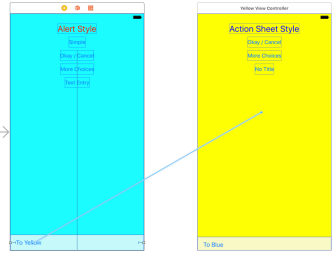


DEPAUL UNIVERSITY 10

10

Add a Segue – Connect the Scenes

- Control-drag from the "*To Yellow*" button to the the *Yellow* scene

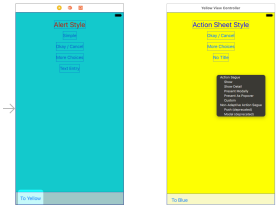


DEPAUL UNIVERSITY 11

11

Add a Segue – Choose Action Segue

- Control-drag from the "*To Yellow*" button to the the *Yellow* scene
  - Action segue: *Show*
- Presentation styles
  - *Show*: default
    - Card view in iOS 13
    - Swipe down to dismiss
  - *Present Modally*
    - Control the modality of presentation, e.g., full screen

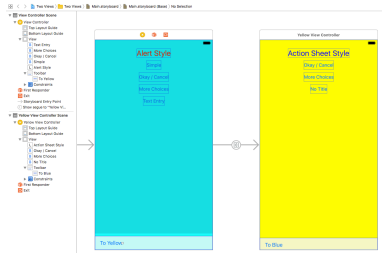


Stay tuned

DEPAUL UNIVERSITY 12

13

Storyboard with a Segue



- Run, press *To Yellow* – perform the segue
  - but no return (yet).

14

Switch Back to the Blue Scene

- Connect an action to the *To Blue* button in the `YellowViewController`
  - Implement the action

```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    dismiss(animated: true, completion: nil)
}
```

- Run,
  To Yellow – perform the segue
  To Blue – invoke the action to dismiss view controller

15

Passing Data Through Segues

16

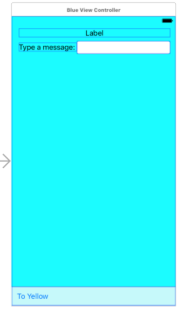
The Segue App

- Three scenes and three view controllers
  - Blue, Yellow, Green
- Segues
  - Blue → Yellow
  - Yellow → Green
- Return action by dismissing view controllers
  - Yellow → Blue
  - Green → Yellow
  - Green → Blue

17

Scenes and View Controllers

- Start with a single view app
- Edit the initial scene in the storyboard
  - Set the background to blue
  - Add a toolbar and button
  - Add labels and a text field
- Change the view controller name to `BlueViewController`



18

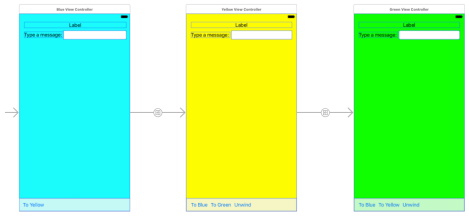
Scenes and View Controllers

- Add two similar scenes to the storyboard and add two new view controller classes
  - `YellowViewController` and `GreenViewController`

19

Add Segues

- Add "Show" segues
  - From the "*To Yellow*" button of the *Blue* scene to the *Yellow* scene
  - From the "*To Green*" button of the *Yellow* scene to the *Green* scene



DePaul University 19

20

Dismiss View Controllers – From Yellow View Controller

- Connect the "*To Blue*" button of the *Yellow* scene to an action in the *YellowViewController*
  - The return action to the *Blue* scene

```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    dismiss(animated: true, completion: nil)
}
```

DePaul University 20

21

Dismiss View Controllers – From Green View Controller

- Connect the "*To Blue*" and "*To Yellow*" buttons of the *Green* scene to actions in the *GreenViewController*
  - The return actions to the *Yellow* scene and the *Blue* scene

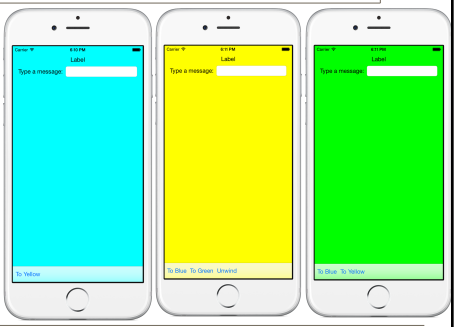
```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    var top: UIViewController = self;
    while top.presentingViewController != nil {
        top = top.presentingViewController!;
    }
    top.dismiss(animated: true, completion: nil)
}
@IBAction func switchToYellow(_ sender: UIBarButtonItem) {
    dismiss(animated: true, completion: nil)
}
```

DePaul University 21

22

Run …

- *To Yellow*
- *To Green*
- And back



DePaul University 22

23

Passing Data Through Segues

- Use the method in the *presenting* view controller, i.e., the *source*

```
func prepare(for segue: UIStoryboardSegue,
             sender: Any?)
```

- Invoked before the segue is performed
- The *destination* view controller can be accessed as `segue.destination`
- The presenting view controller can put data directly in the destination view controller

Will have examples

DePaul University 23

24

Connecting Outlets

- For each view controller
  - Connect outlets to the message label and the text field
  - Connect an action to the "*Did End on Exit*" of the text field to dismiss keyboard

```
class BlueViewController: UIViewController {
    @IBOutlet weak var label: UILabel!
    @IBOutlet weak var textField: UITextField!
    @IBAction func doneEditing(_ sender: UITextField) {
        sender.resignFirstResponder()
    }
    …
}
```

DePaul University 24

25

Prepare for Segue

- Define the method `prepare(for segue: sender:)` in the *BlueViewController*
  - To prepare for the segue from the *Blue* scene to the *Yellow* scene

```
override func prepare(for segue: UIStoryboardSegue,
                      sender: Any?) {
    if let target = segue.destination as?
                    YellowViewController {
        target.label.text = "From BlueViewController"
    }
}
```

DePaul University 25

26

Prepare for Segue

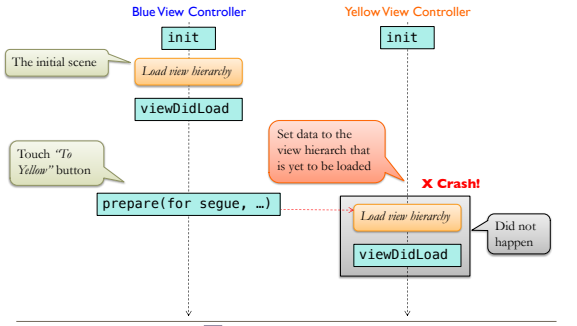
- Let's run the app
  - Press the "*To Yellow*" button
  - Crash!

    fatal error: unexpectedly found nil while unwrapping an Optional value
- Why?
  - The *Yellow* view controller has been initialized, but not its view hierarchy
  - The view hierarchy is initialize when the view becomes visible.
  - **Important:** Access the view objects **only** from within the view controller associated with the view.

DePaul University 26

27

View Controllers and View Hierarchies



DePaul University 27

34

Passing Data

- **Store data in the properties of view controllers**, not in view objects.
- For each view controller,
  - Add a property *message* for storing data
  - Implement the lifecycle callback `viewWillAppear`

```
class BlueViewController: UIViewController {
    @IBOutlet weak var label: UILabel!
    var message: String = "BlueViewController"
    override func viewWillAppear(_ animated: Bool) {
        label.text = message
    }
    …
}
```

DePaul University 28

35

Prepare for Segue, 2nd Attempt

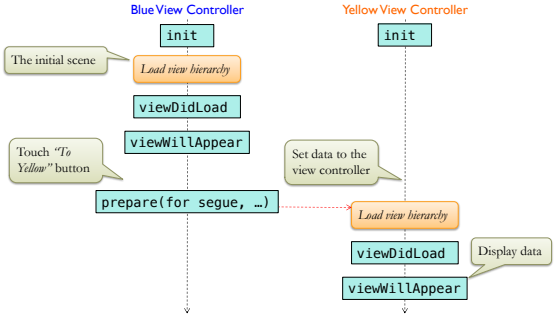
- Define the method `prepareForSegue` in the *BlueViewController*

```
override func prepare(for segue: UIStoryboardSegue,
                      sender: Any?) {
    if let target = segue.destination as?
                    YellowViewController {
        target.message = "From BlueViewController"
        if let text = textField.text  {
            target.message += "\nMessage: \(text)"
        }
    }
}
```

DePaul University 29

36

View Controllers and View Hierarchies

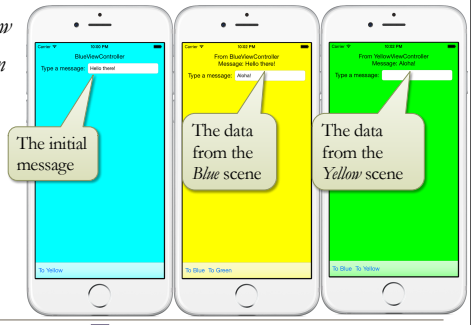


DePaul University 30

45

Run …

- *To Yellow*
- *To Green*



DePaul University 31

48

Passing Data in Return

- Revised dismiss action from the *Yellow* scene to the *Blue* scene
  - The *presenting view controller*

```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    if let from = presentingViewController as?
                  BlueViewController {
        from.message = "From YellowViewController"
        if let text = textField.text {
            from.message += "\nMessage: \(text)"
        }
    }
    dismiss(animated: true, completion: nil)
}
```

DePaul University 32

49

Passing Data in Return

- Revised dismiss actions of the *Green* scene
  - The *presenting view controller*

```
@IBAction func switchToYellow(_ sender: UIBarButtonItem) {
    if let from = presentingViewController as?
                  YellowViewController {
        from.message = "From GreenViewController"
        if let text = textField.text {
            from.message += "\nMessage: \(text)"
        }
    }
    dismiss(animated: true, completion: nil)
}
```

DePaul University 33

50

Passing Data in Return

```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    var top: UIViewController = self;
    while top.presentingViewController != nil {
        top = top.presentingViewController!;
    }
    if let blue = top as? BlueViewController {
        blue.message = "From YellowViewController"
        if let text = textField.text {
            blue.message += "\nMessage: \(text)"
        }
    }
    top.dismiss(animated: true, completion: nil)
}
```

DePaul University 34

51

A Breaking Change in iOS 13

- Show segue is presented as in a card view not full screen
  - The presented view can be dismissed with a swipe down gesture
  - When dismissed, the `viewWillAppear()` method *is not* called on the presenting view controller
- It is necessary to explicitly updating of the presenting view upon return

DePaul University 35

52

Passing Data Back to Presenting View Controller, iOS 13 and Onward

- Define a method in each presenting view controller to explicitly update the view

```
class BlueViewController: UIViewController {
    var message: String = "BlueViewController"

    func updateView() {
        label.text = message
    }

    …
}
```

DePaul University 36

53

Passing Data Back to Presenting View Controller, iOS 13 and Onward

- Call the `updateView()` method explicitly before dismissing the presented view

```
@IBAction func switchToBlue(_ sender: UIBarButtonItem) {
    if let from = presentingViewController as?
                  BlueViewController {
        from.message = "From YellowViewController"
        if let text = textField.text {
            from.message += "\nMessage: \(text)"
        }
        from.updateView()
    }
    dismiss(animated: true, completion: nil)
}
```


54

Full Screen Presentation iOS 13 and Onward

- To present view controllers in full screen mode
  - Segue options:
    - Kind: *Present Modally* instead of *Show* (default)
    - Presentation: *Full Screen*
- Differences in the appearance and *behavior* in life cycle callbacks
  - When a presented view is dismissed, the `viewWillAppear()` method is called on the presenting view controller.
- An alternative version of Segues demo using *Modal* segue presentation is provided

55

Unwind Segues

56

Unwind Segue

- You can *unwind* a segue, i.e., return to the previous scene, using an *unwind segue*
- You can also *unwind* to any previous scenes, one or more steps away
- How:
  - Define one or more *unwind actions*
    - In swift code, the destination of unwinding
  - Add *unwind segues*
    - In storyboard

57

Unwind Action & Unwind Segue

- Add an *unwind action* in the destination view controller, i.e., the view controller you want to unwind to
  - Unwind action is a method with the following signature:
    `@IBAction func name(_ segue : UIStoryboardSegue)`
  - The *name* must be unique within the context of application.
- Add an *unwind segue*
  - Control-drag from the trigger button to the *Exit* icon of the source view controller
  - Select the name of the unwind action

58

Unwind Action

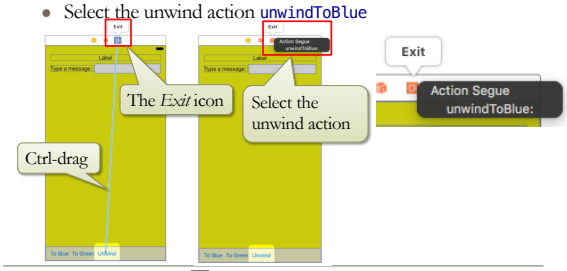
- Define an unwind action in the *BlueViewController*

```
@IBAction func unwindToBlue(_ segue : UIStoryboardSegue) {
    if let from = segue.source as?
                  YellowViewController {
        message = "Unwind from YellowViewController"
        if !from.textField.text.isEmpty {
            message += "\nMessage: \(from.textField.text)"
        }
    }
}
```


59

Add an Unwind Segue

- In storyboard, add an unwind segue in the *YellowViewController*, from the "*Unwind*" button to the *Exit* icon
  - Select the unwind action `unwindToBlue`

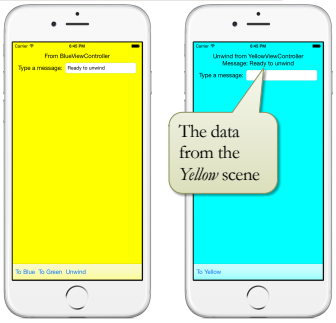


DePaul University 43

60

Run …

- Unwind from *Yellow* to *Blue*
- Pay attention to the variations in *Show* vs. *Modal* presentation of segues



DePaul University 44

61

Sample Code

- Two Views – SB.zip
- Segues – Show – SB.zip
- Segues – Modal – SB.zip

DePaul University 45

62

Next …

- Tabbed views
- Pickers
- Table views, static and dynamic
- Navigation views

✧ iOS is a trademark of Apple Inc.

DePaul University 46

63