

EE443 / EE593
Mobile Application Development

Jeremy Grajeda

Lecture 8: Chapter 11: Navigation Controllers

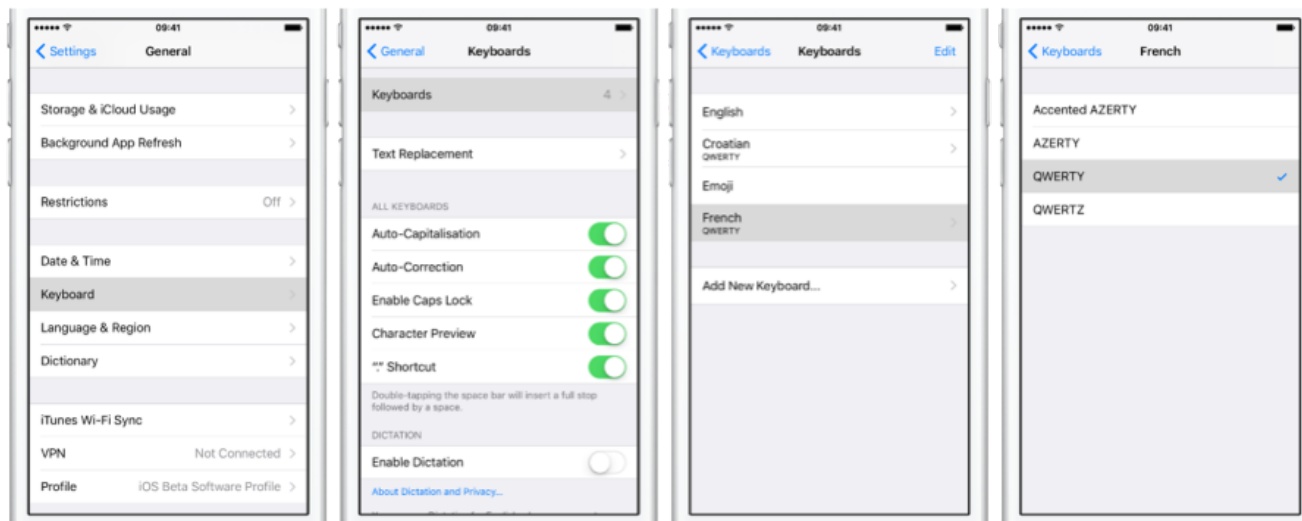
Jeremy Grajeda

Outline

- Navigation Controller
- Programing To-Do List

Navigation Controller

- Manages a stack of View Controllers
 - Remembers the order in which View Controllers are accessed
 - Allows for a “drill-down interface for hierarchical content”



View Controllers pushed onto stack →
← View Controllers popped from stack

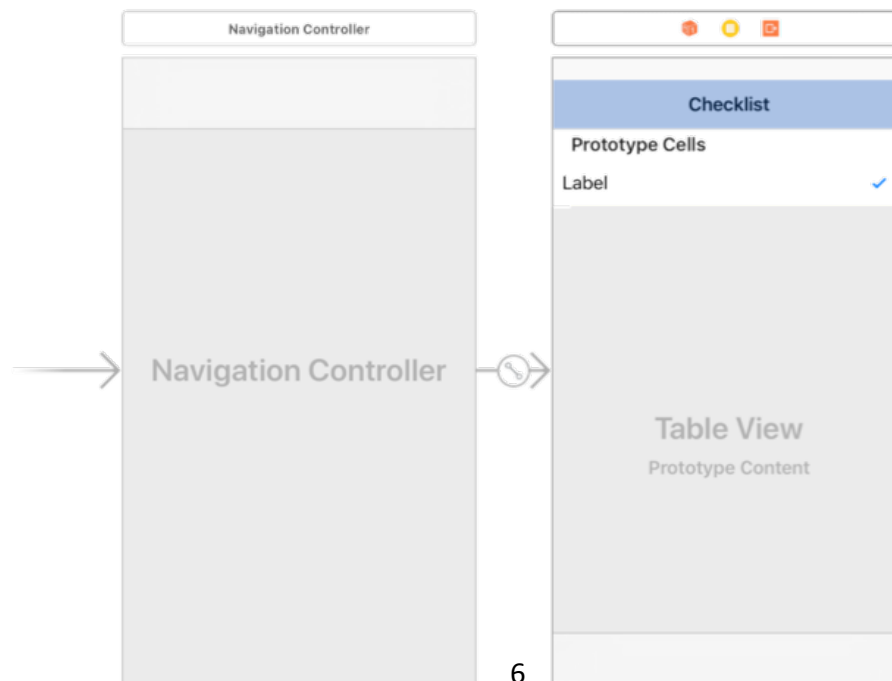
Programing To-Do list

Chapter 11

- Create Navigation Controller
- Add Item to Checklist
- Delete Item on Checklist
- Add Item Screen
- Navigate on Add Item Screen

To-Do: Create Navigation Controller

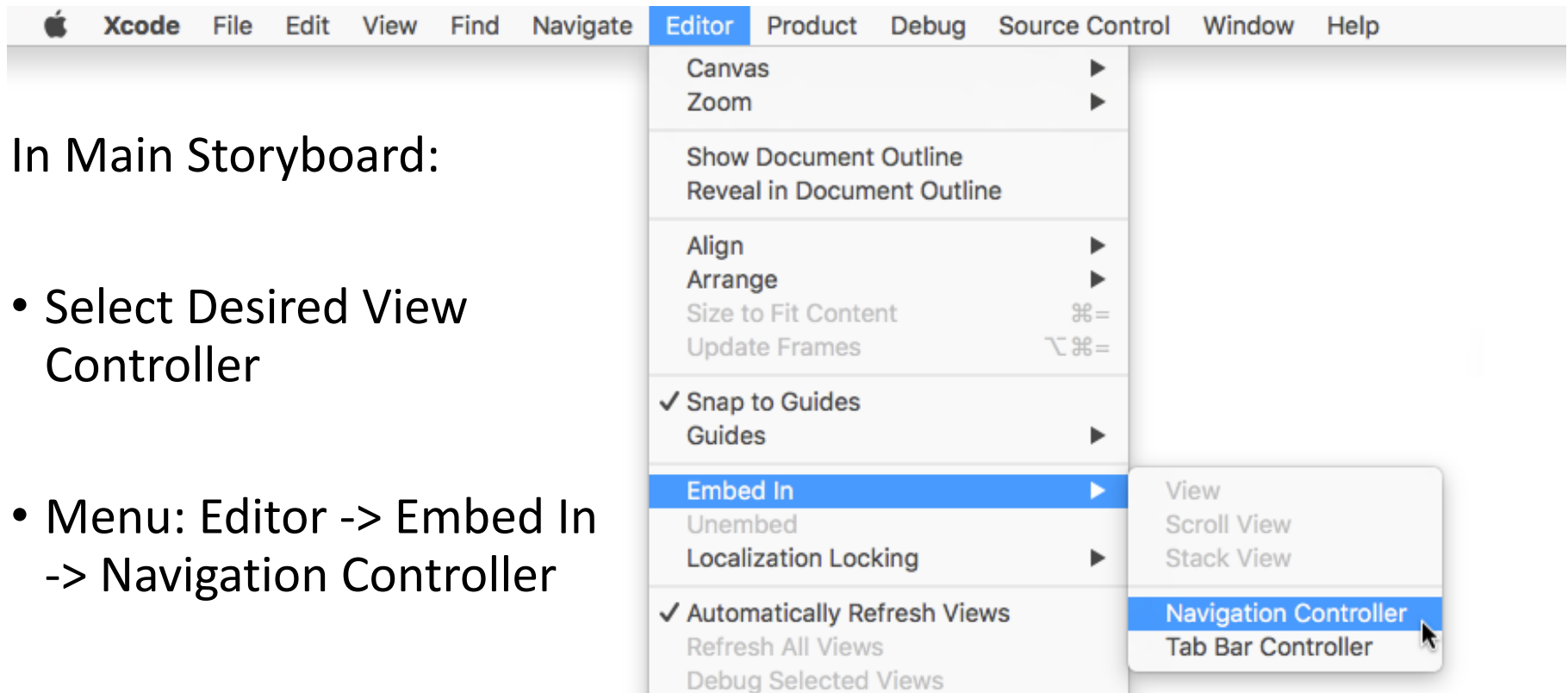
- Embed Checklist View Controller into a Navigation Controller
- Customize the Checklist View Controller's Navigation Bar



Embed in Navigation Controller

In Main Storyboard:

- Select Desired View Controller
- Menu: Editor -> Embed In -> Navigation Controller



Behind the Scenes Xcode:

- Create Navigation Controller
- Move Storyboard Entry Point to Navigation Controller
- Connect Navigation Controller to Checklist Controller
 - Relationship Segue -> Root View Controller

The Navigation Controller now serves as a container, holding the Checklist View Controller within it as a “child view controller”

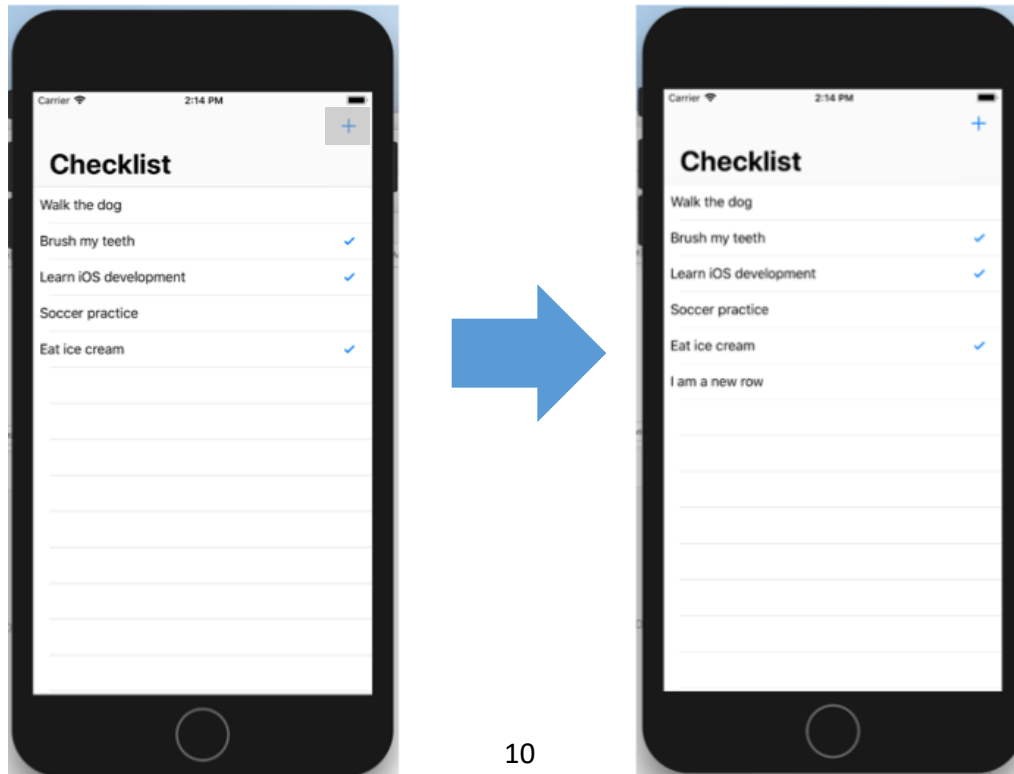
Customize the Checklist View Controller's Navigation Bar

- Navigation Item can be given a Title in the Attributes Inspector
- Enable large titles, in viewDidLoad() add

```
navigationController?.navigationBar.prefersLargeTitles = true
```

To-Do: Add Item to Checklist

- Add ability to create new Items in checklist



Add ability to create new Items in checklist

- We need a UI Bar Button Item in the Checklist Navigation Bar

- New Method:

```
@IBAction func addItem() {
    let newRowIndex = items.count

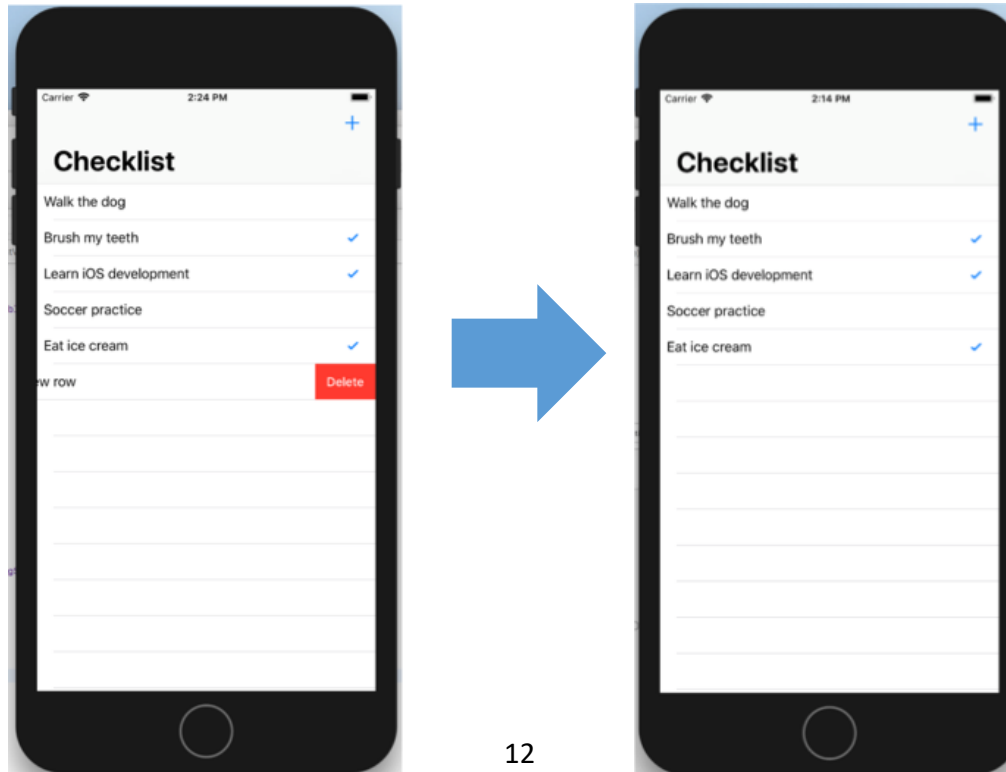
    let item = ChecklistItem()
    item.text = "I am a new row"
    item.checked = false
    items.append(item)

    let indexPath = IndexPath(row: newRowIndex, section: 0)
    let indexPaths = [indexPath]
    tableView.insertRows(at: indexPaths, with: .automatic)
}
```

*We won't be keeping this function

To-Do: Delete Items from Checklist

- Add ability to remove an item from checklist



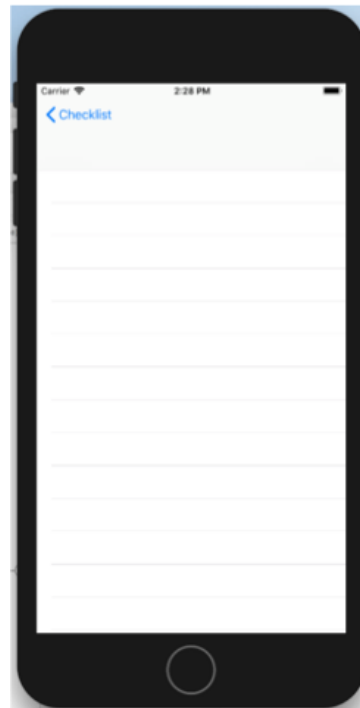
Add ability to remove item from checklist

- Use tableView Commit to remove rows from table

```
override func tableView(  
    _ tableView: UITableView,  
    commit editingStyle: UITableViewCellEditingStyle,  
    forRowAt indexPath: IndexPath) {  
  
    // 1 Remove Checklist Item  
    items.remove(at: indexPath.row)  
  
    // 2 Delete Row  
    let indexPaths = [indexPath]  
    tableView.deleteRows(at: indexPaths, with: .automatic)  
}
```

To-Do: Add Item Screen

- Create Add Item Screen



Create Add Item Screen

- Since we are adding a new screen, we need another View Controller
- Create an Action Segue (choose Show, so the view controller is pushed onto the stack)
- We can now get rid of the addItem method

To-Do: Navigate on Add Item Screen

- Create Navigation Item



Create Navigation Item

- Since we want a more complex navigation bar, add a UI Navigation Item and the UI Bar Button Items
- It is probably best to use the small titles, add to viewDidLoad():

```
navigationItem.largeTitleDisplayMode = .never
```

Cancel and Done

- We will not be adding new item to the Checklist just yet, so for now, let's just use the UI Bar Buttons to pop back to the previous screen.

```
@IBAction func cancel() {  
    navigationController?.popViewController(animated: true)  
}
```

```
@IBAction func done() {  
    navigationController?.popViewController(animated: true)  
}
```