



**Computing & Information Sciences**  
FLORIDA INTERNATIONAL UNIVERSITY

# Mobile Application Development

## lecture4

Spring 2012- COP 4655 U1

M/W 6:25pm – ECS 138

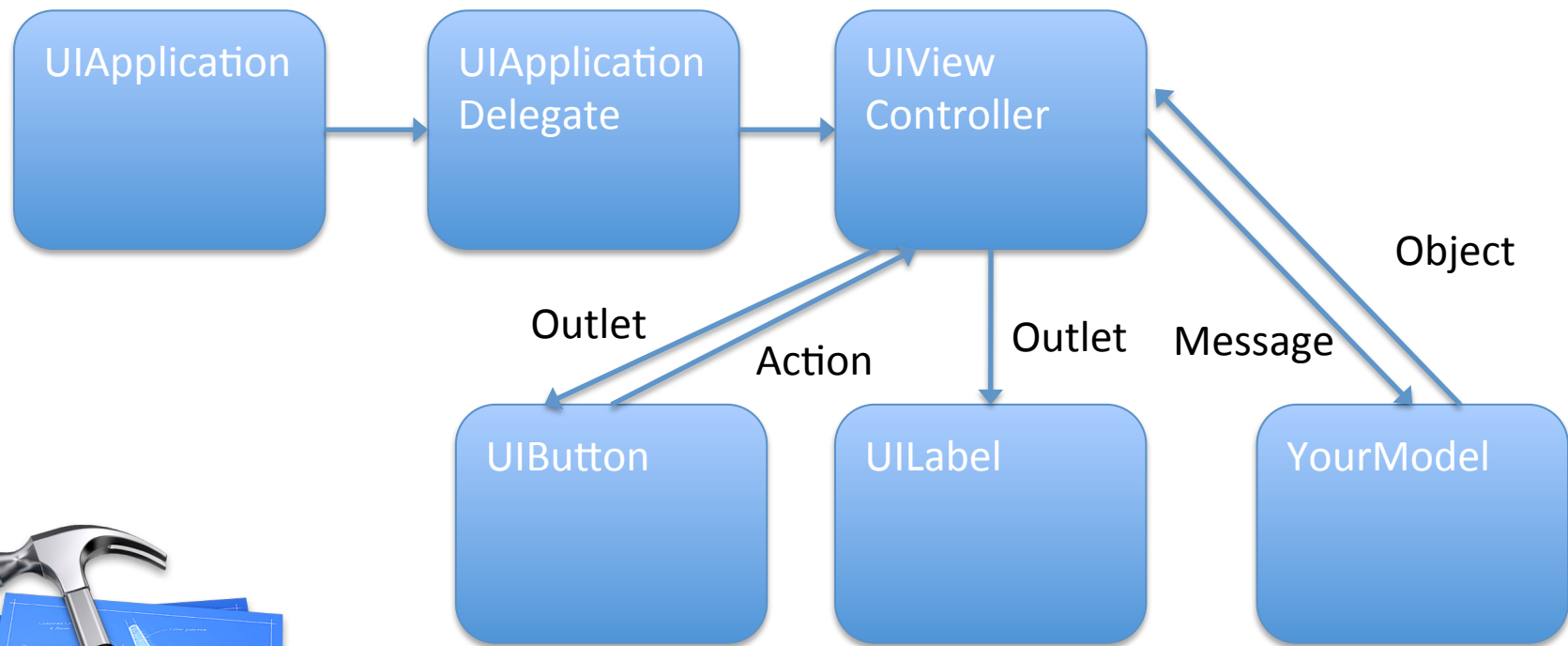
Steve Luis

# Agenda

- Build View Controller Based Application
- Programming assignment #1



# View Based Application: Simplified



Build a View Based Application from Project Templates

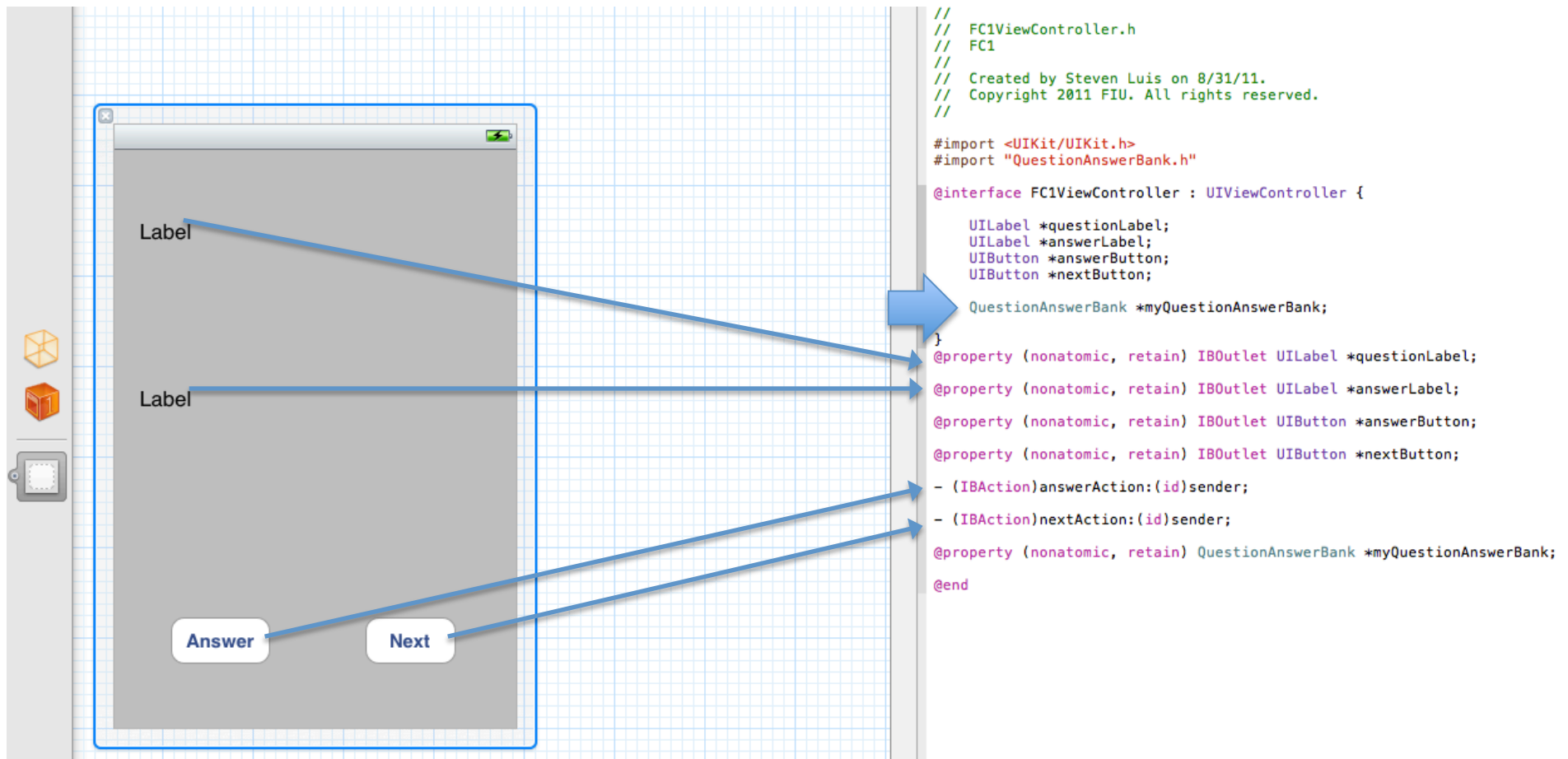
# Demo: Build a Button Counter

1. Choose a View Based Application from Project Templates, in iOS section.
2. Using Interface Builder, create the UI for the application.
3. Link the objects created in the Interface Builder to the code.
4. Add the code that updates the counter.

# Observations

- Interface Builder can create objects and initialize their instance variables.
- Objects created by Interface Builder can be linked to your classes.
- Once linked, you can send messages to those objects, access instance variables, and destroy them.

# Review: Build the Interface



# Build the Model

```
//  
// QuestionAnswerBank.h  
// FC1  
//  
// Created by Steven Luis on 8/31/11.  
// Copyright 2011 FIU. All rights reserved.  
//  
  
#import <Foundation/Foundation.h>  
  
@interface QuestionAnswerBank : NSObject {  
}  
  
- (NSString *) dispenseQuestion ;  
- (NSString *) dispenseAnswer ;  
  
@end
```

```
//  
// QuestionAnswerBank.m  
// FC1  
//  
// Created by Steven Luis on 8/31/11.  
// Copyright 2011 FIU. All rights reserved.  
//  
  
#import "QuestionAnswerBank.h"  
  
@implementation QuestionAnswerBank  
  
- (NSString *) dispenseQuestion {  
    return(@"What is the meaning of life?");  
}  
  
- (NSString *) dispenseAnswer {  
    return(@"42");  
}  
  
@end
```

# Build the View Controller

```
//  
// FC1ViewController.m  
// FC1  
//  
// Created by Steven Luis on 8/31/11.  
// Copyright 2011 FIU. All rights reserved.  
//
```

```
#import "FC1ViewController.h"
```

```
@implementation FC1ViewController  
@synthesize questionLabel;  
@synthesize answerLabel;  
@synthesize answerButton;  
@synthesize nextButton;  
@synthesize myQuestionAnswerBank;
```

```
- (void)dealloc
```

```
{  
    [questionLabel release];  
    [answerLabel release];  
    [answerButton release];  
    [nextButton release];  
    [super dealloc];  
}
```

```
- (void)didReceiveMemoryWarning
```

```
{  
    // Releases the view if it doesn't have a superview.  
    [super didReceiveMemoryWarning];  
  
    // Release any cached data, images, etc that aren't in use.  
}
```

```
#pragma mark - View lifecycle
```

```
// Implement viewDidLoad to do additional setup after loading the view,  
// typically from a nib.
```

```
- (void)viewDidLoad
```

```
{  
    myQuestionAnswerBank = [[QuestionAnswerBank alloc] init];  
    [super viewDidLoad];  
}
```

```
- (void)viewDidUnload
```

```
{  
    [self setQuestionLabel:nil];  
    [self setAnswerLabel:nil];  
    [self setAnswerButton:nil];  
    [self setNextButton:nil];  
    [super viewDidUnload];  
    // Release any retained subviews of the main view.  
    // e.g. self.myOutlet = nil;  
}
```

```
- (BOOL)shouldAutorotateToInterfaceOrientation:(UIInterfaceOrientation)  
interfaceOrientation
```

```
{  
    // Return YES for supported orientations  
    return (interfaceOrientation == UIInterfaceOrientationPortrait);  
}
```

```
- (IBAction)answerAction:(id)sender {  
    answerLabel.text = [myQuestionAnswerBank dispenseAnswer];  
}
```

```
- (IBAction)nextAction:(id)sender {
```

```
    questionLabel.text = [myQuestionAnswerBank dispenseQuestion];  
    answerLabel.text = nil;  
}
```

```
@end
```



# Notes

- You can lookup the Framework classes with the Organizer Window.
- View the inheritance structure.
- Learn more about instance variable and method declaration associated with any class.

# Assignment

- Read in Apple Developer Library:
  - Cocoa Design Patterns
  - UIViewController, UIApplication, UIApplicationDelegate
- Read Kochan:
  - Chapters 4, 5, and 6
- Complete Reading assignments by Monday, January 30<sup>th</sup>.