

# Level 4

UINavigationController  
& Custom Buttons





## 01 Introduction to the UINavigationController

02 Creating a Custom Image Button

03 Creating View Controllers on the Fly

04 Using Both Navigation and TabBar Controllers

# How else do we navigate ViewControllers?



SettingsViewController

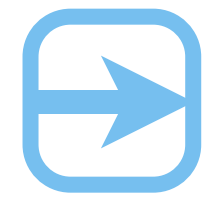


GeneralViewController

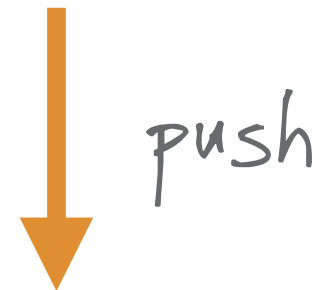


AboutViewController

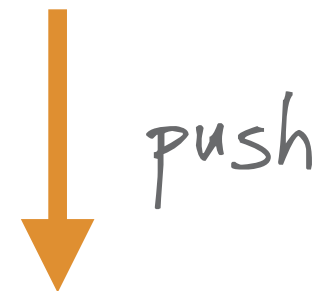
# UINavigationController



SettingsViewController



GeneralViewController



AboutViewController

# This is Our Old Code



## AppDelegate.m

```
#import "AppDelegate.h"
#import "FeedViewController.h"

@implementation AppDelegate

- (BOOL) application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    FeedViewController *feedViewController = [[FeedViewController alloc] init];

    UITabBarController *tabBarController = [[UITabBarController alloc] init];

    self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];

    self.window.rootViewController = tabBarController;
}
```

*old code*

# Using the UINavigationController



## AppDelegate.m

```
#import "AppDelegate.h"
#import "FeedViewController.h"

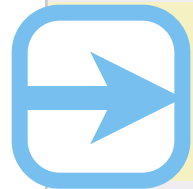
@implementation AppDelegate

- (BOOL) application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    FeedViewController *feedViewController = [[FeedViewController alloc] init];

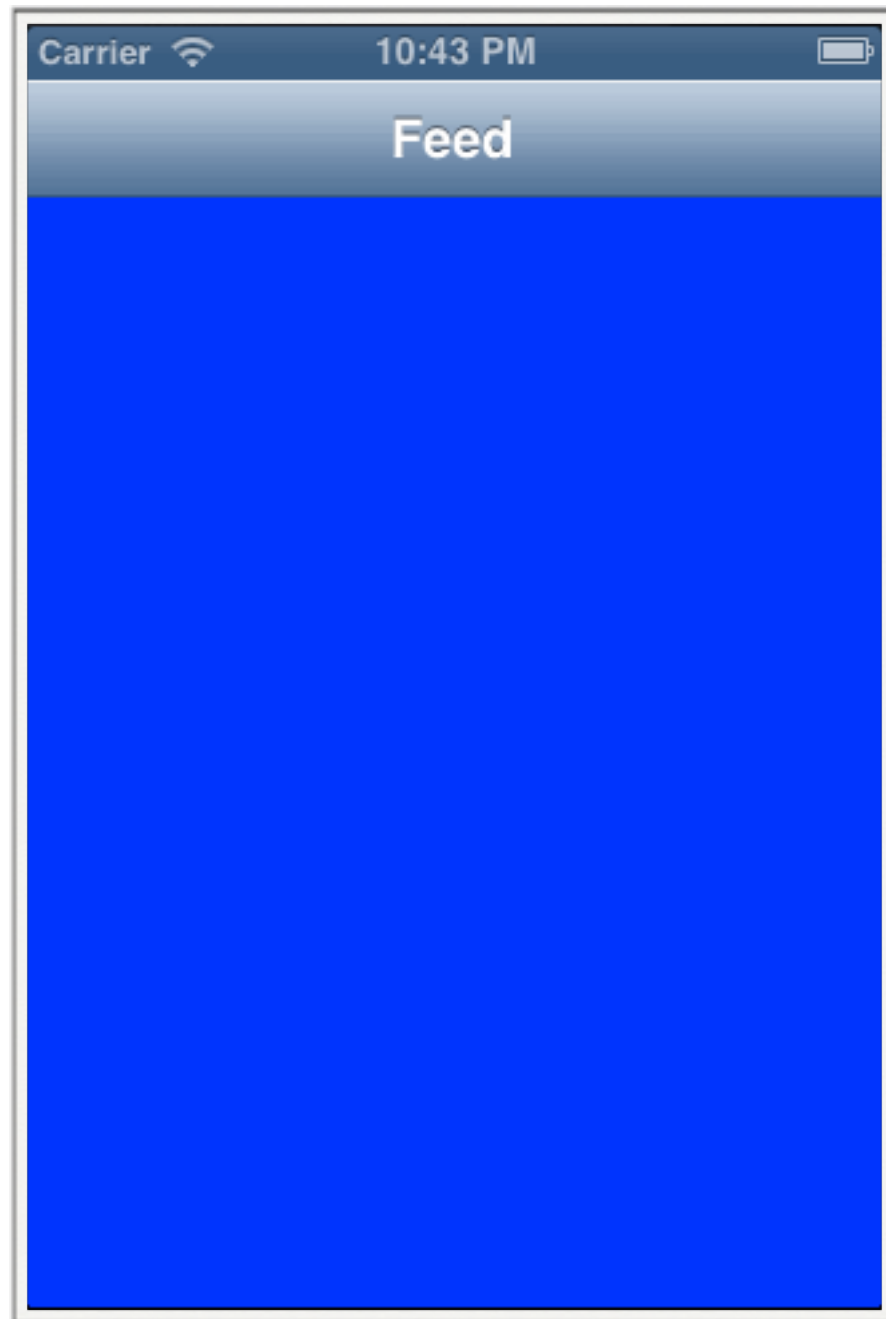
    UINavigationController *navController = [[UINavigationController alloc]
                                           initWithRootViewController:feedViewController];

    self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];

    self.window.rootViewController = navController;
}
```



# First Screen of our UINavigationController ➡

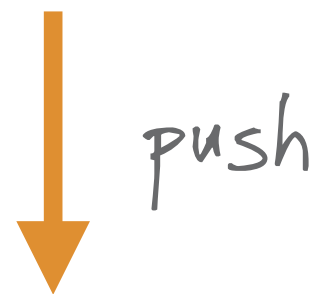


← Takes the name of the controller by default

**We can push to a new controller with**

```
[self.navigationController  
    pushViewController:<New View Controller> animated:YES];
```

**FeedViewController**



**FavoritesViewController**

# Navigating to the FavoritesViewController



## FeedViewController.m

```
...
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.
    self.view.backgroundColor = [UIColor blueColor];

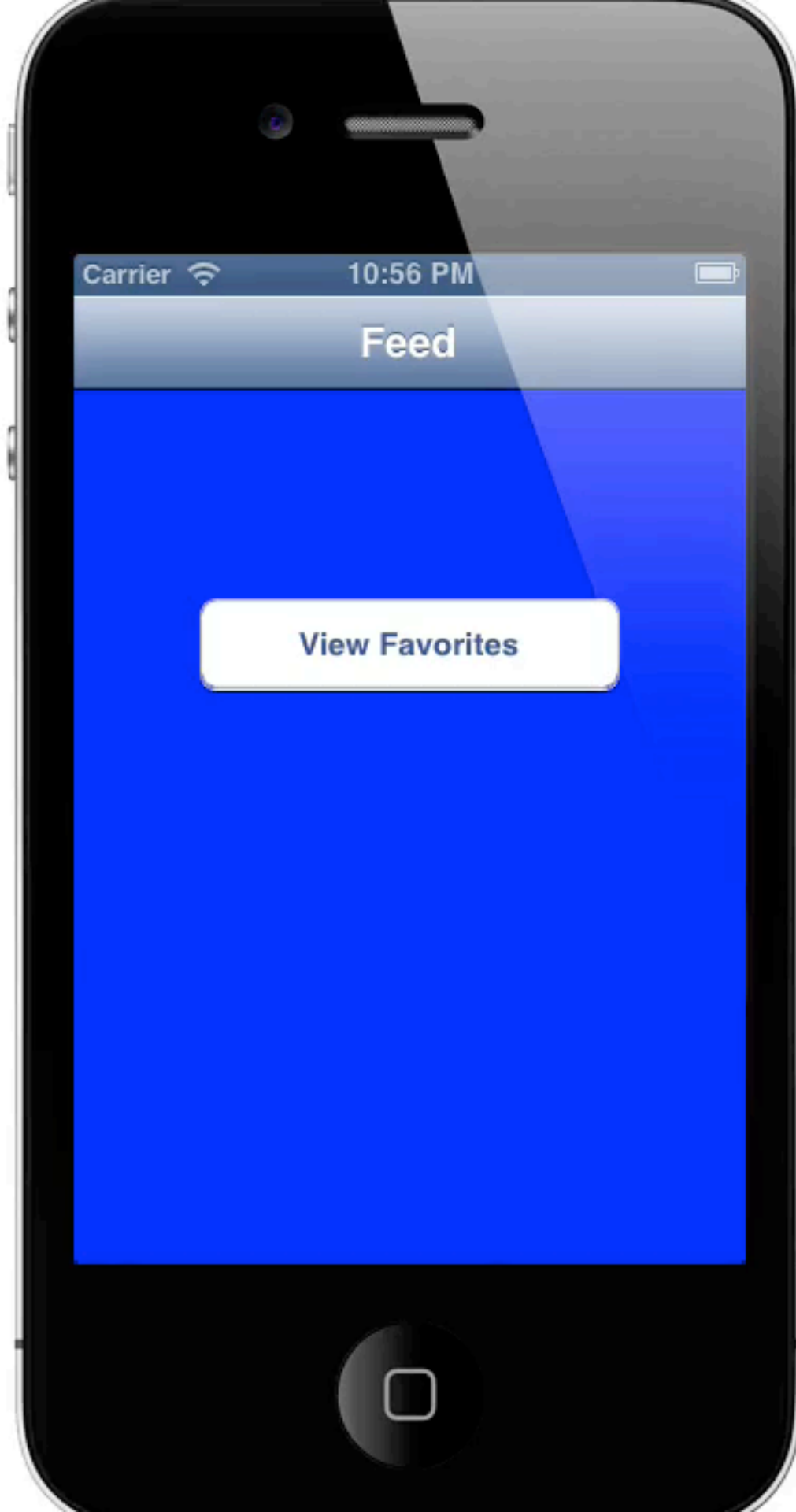
    UIButton *favoritesButton = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    favoritesButton.frame = CGRectMake(60, 100, 200, 44);
    [favoritesButton setTitle:@"View Favorites" forState:UIControlStateNormal];
    [self.view addSubview:favoritesButton];

    [favoritesButton addTarget:self action:@selector(showFavorites:)
                          forControlEvents:UIControlEventTouchUpInside];
}

- (void)showFavorites:(UIButton *)sender
{
    FavoritesViewController *favoritesViewController =
        [[FavoritesViewController alloc] init];
    [self.navigationController pushViewController:favoritesViewController
                          animated:YES];
}
```

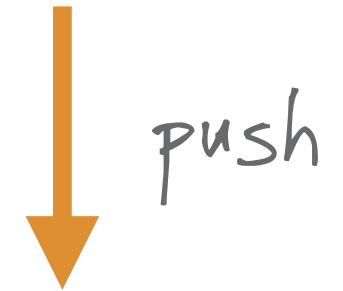






# With UINavigationController

FeedViewController

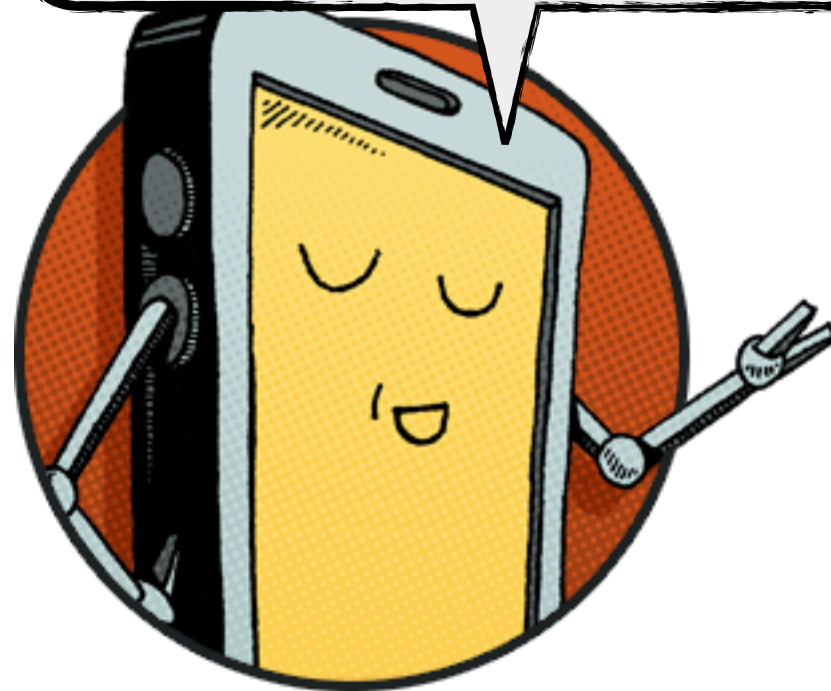


FavoritesViewController

# Custom Buttons



WOULDN'T IT BE NICE IF CLICKING THAT BUTTON LINKED TO A BIGGER IMAGE?



# Changing an Image to a Button

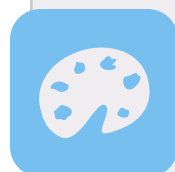


## FavoritesViewController.m

old code



```
...  
- (void)viewDidLoad  
{  
    [super viewDidLoad];  
    // Do any additional setup after loading the view.  
    self.view.backgroundColor = [UIColor blueColor];  
  
    UIImageView *tryiosView = [[UIImageView alloc] initWithImage:  
        [UIImage imageNamed:@"tryios.gif"]];  
    [tryiosView setContentMode:UIViewContentModeScaleAspectFit];  
    tryiosView.frame = self.view.frame;  
    [self.view addSubview:tryiosView];  
}
```



# Lets add a Picture In our Profile



## FavoritesViewController.m

...

```
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.
    self.view.backgroundColor = [UIColor blueColor];
}
```



```
UIButton *tryiosView = [UIButton buttonWithType:UIButtonTypeCustom];

[tryiosView setImage:[UIImage imageNamed:@"tryios.gif"]
               forState:UIControlStateNormal];

[tryiosView setImage:[UIImage imageNamed:@"tryios.gif"]
               forState:UIControlStateHighlighted];

[tryiosView setFrame:CGRectMake(15, 15, 100, 100)];

[self.view addSubview:tryiosView];

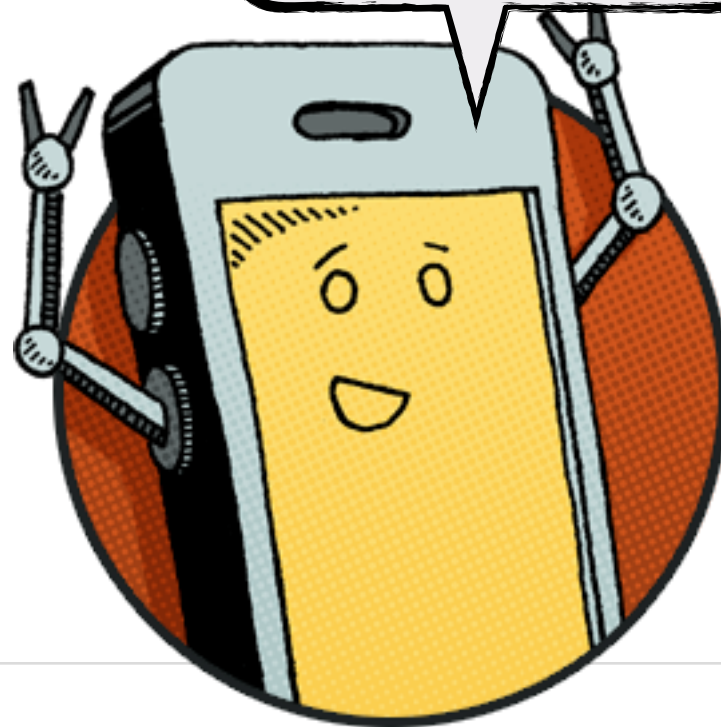
[tryiosView addTarget:self
                 action:@selector(showZoomedPicture:)
               forControlEvents:UIControlEventTouchUpInside];
```

# We have an Image Button



But now we need a `ViewController` filled with an image to pass control to....

BUT HOLD YOUR HORSES,  
YOU DON'T NEED ANOTHER  
SUBCLASS!




# Lets push to a New ViewController




## FavoritesViewController.m

...

– (void)showZoomedPicture:(UIButton \*)sender  
{

 UIViewViewController \*tryiosImageViewController = [[UIViewViewController alloc] init];  
tryiosImageViewController.view.frame = self.view.frame;  
tryiosImageViewController.title = @"Try IOS Logo";

Create a ViewController on the fly!

 UIImageView \*tryiosView = [[UIImageView alloc]  
initWithImage:[UIImage imageNamed:@"tryios.gif"]];  
[tryiosView setContentMode:UIViewContentModeScaleAspectFit];

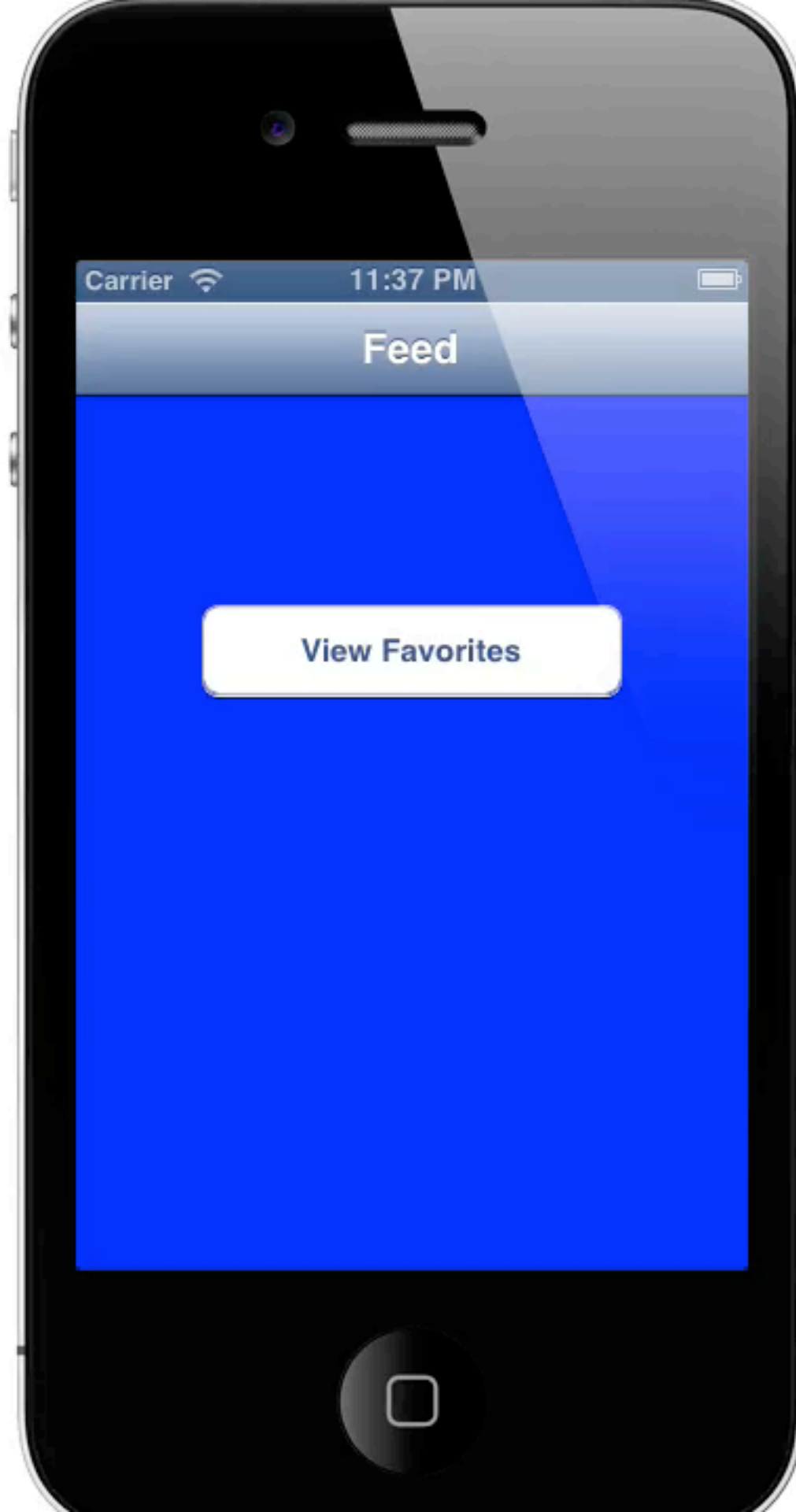
tryiosView.frame = tryiosImageViewController.view.frame;  
[tryiosImageViewController.view addSubview:tryiosView];

 [self.navigationController pushViewController:tryiosImageViewController  
animated:YES];

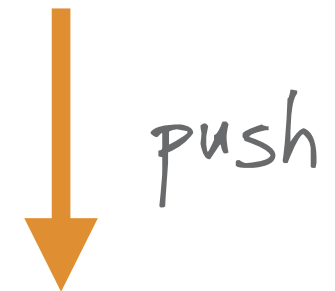
}

Push Control to our new ViewController

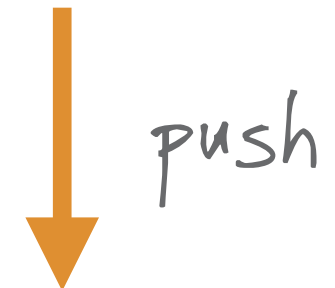
# With Our Custom Button



**FeedViewController**



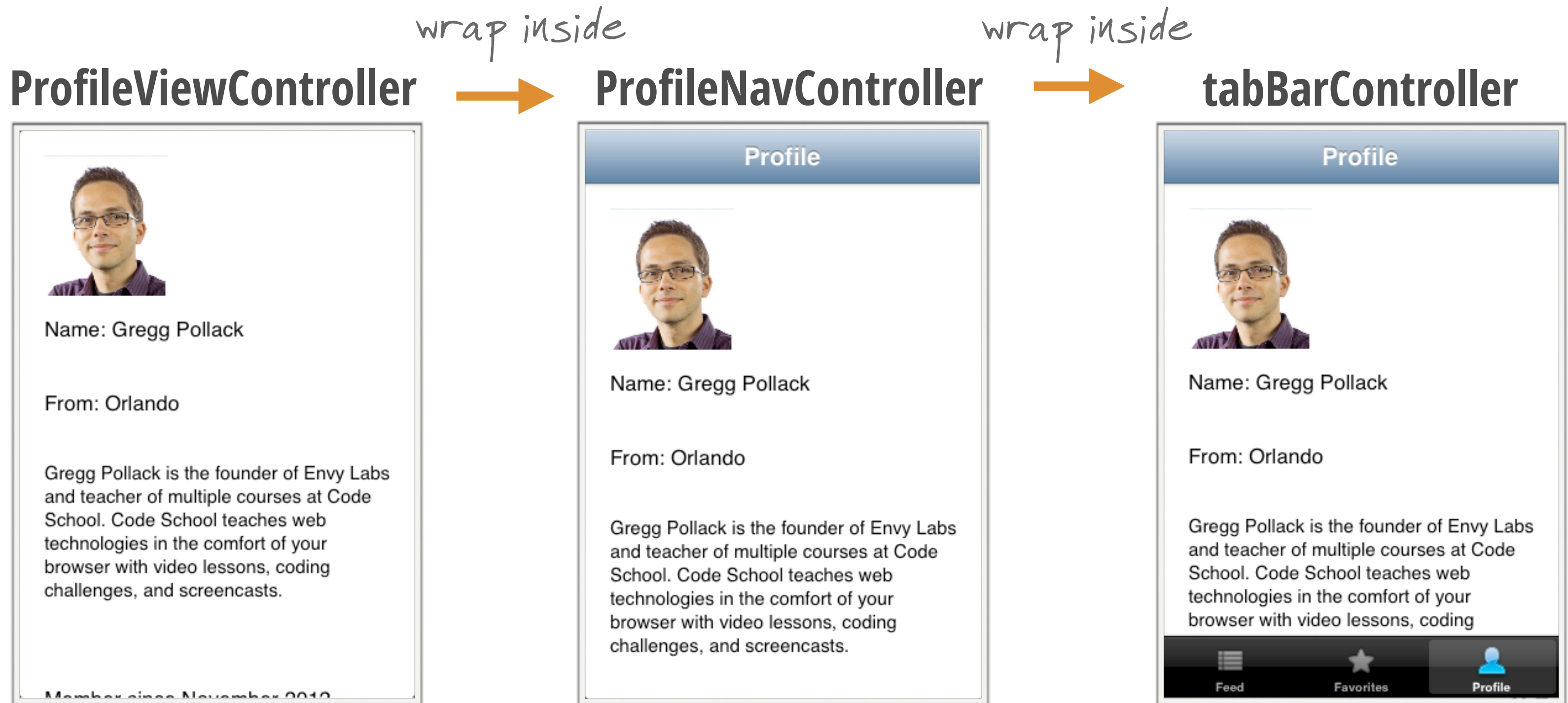
**FavoritesViewController**



**tryiosImageViewController**



# Lets Combine Both Nav + Tab Containers





# Using the UINavigationController




## AppDelegate.m

```
@implementation AppDelegate
```


```
- (BOOL) application:(UIApplication *)application  
didFinishLaunchingWithOptions:(NSDictionary *)launchOptions  
{
```

```
 ProfileViewController *profileViewController = [[ProfileViewController alloc] init];
```

```
 UINavigationController *profileNavController = [[UINavigationController alloc]  
initWithRootViewController:profileViewController];
```

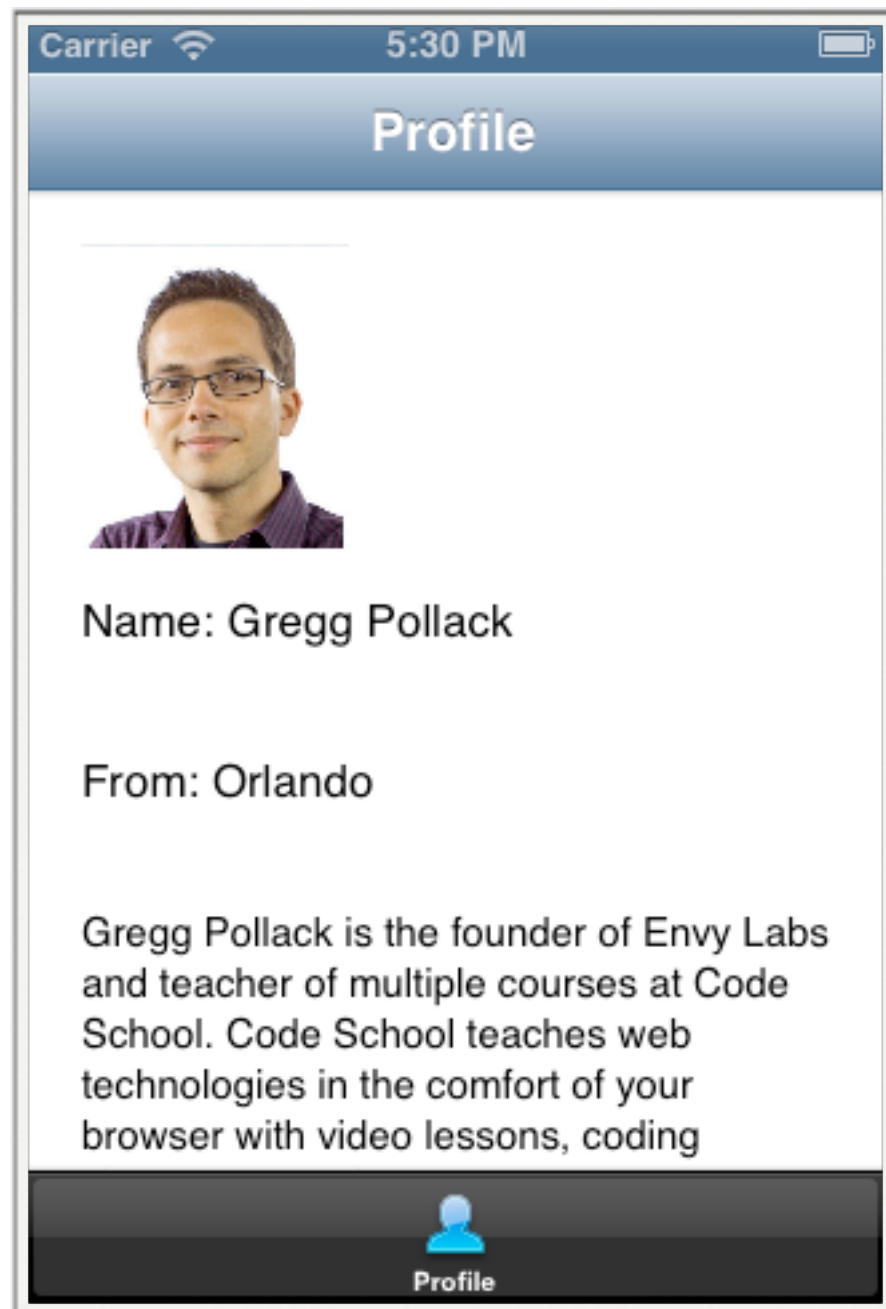
```
self.window = [[UIWindow alloc] initWithFrame:[[UIScreen mainScreen] bounds]];
```

```
UITabBarController *tabBarController = [[UITabBarController alloc] init];
```

```
 [tabBarController setViewControllers:@[profileNavController]];
```

```
self.window.rootViewController = tabBarController;
```

# Both Containers Together!



WHERE DID YOUR  
OTHER TABS GO?

