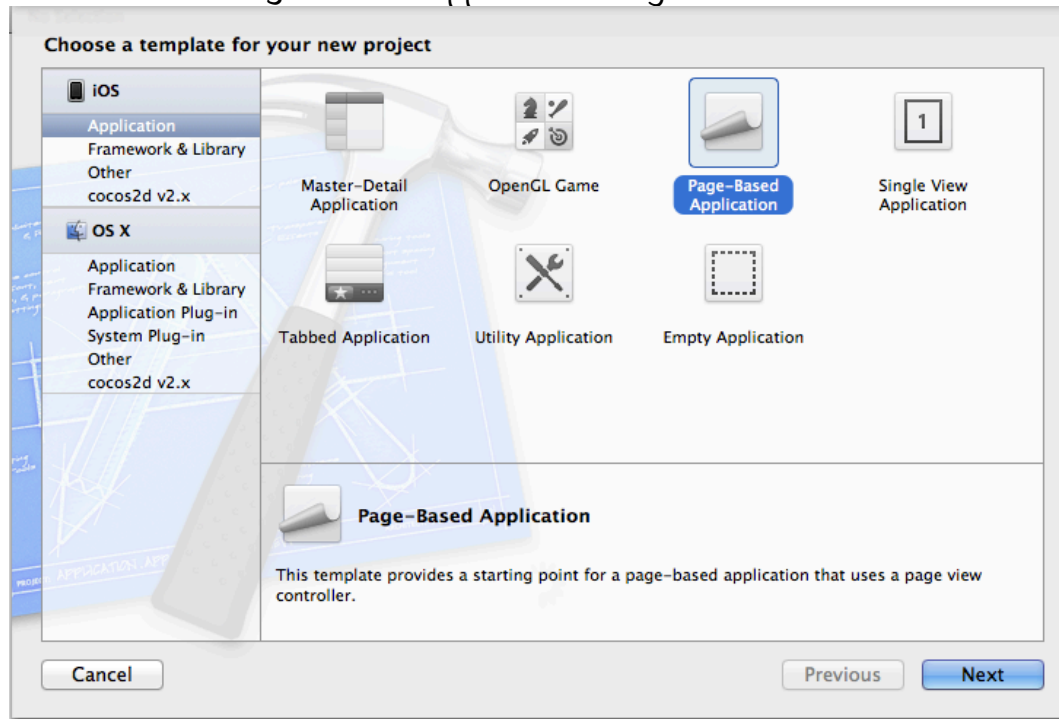


Create a Page View App to use local webpage, external webpage, and local rtf file

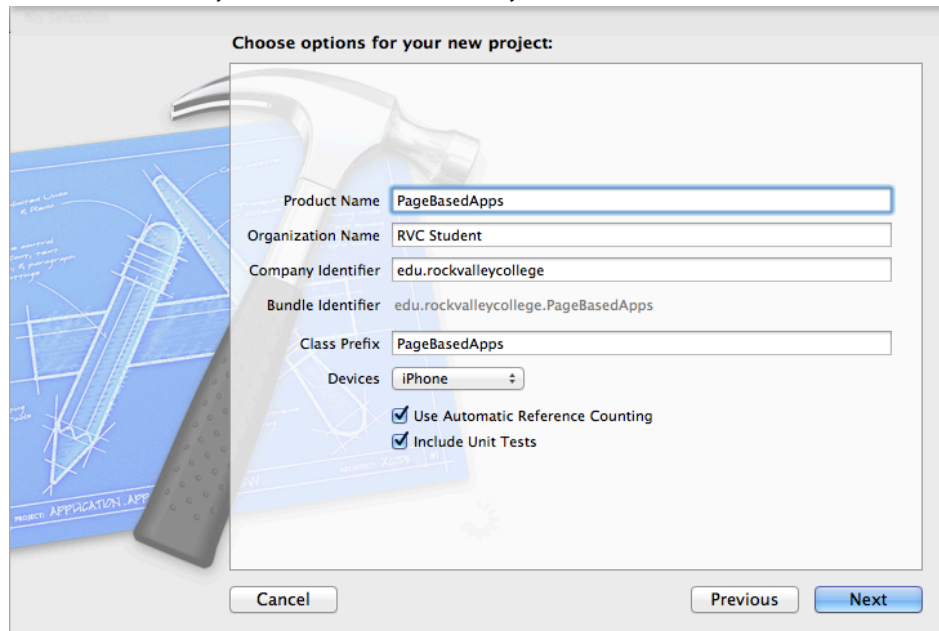
Podcast <http://www.youtube.com/watch?v=US77JX5TiH8&feature=youtu.be>

This type of app is very popular for businesses and events to use.

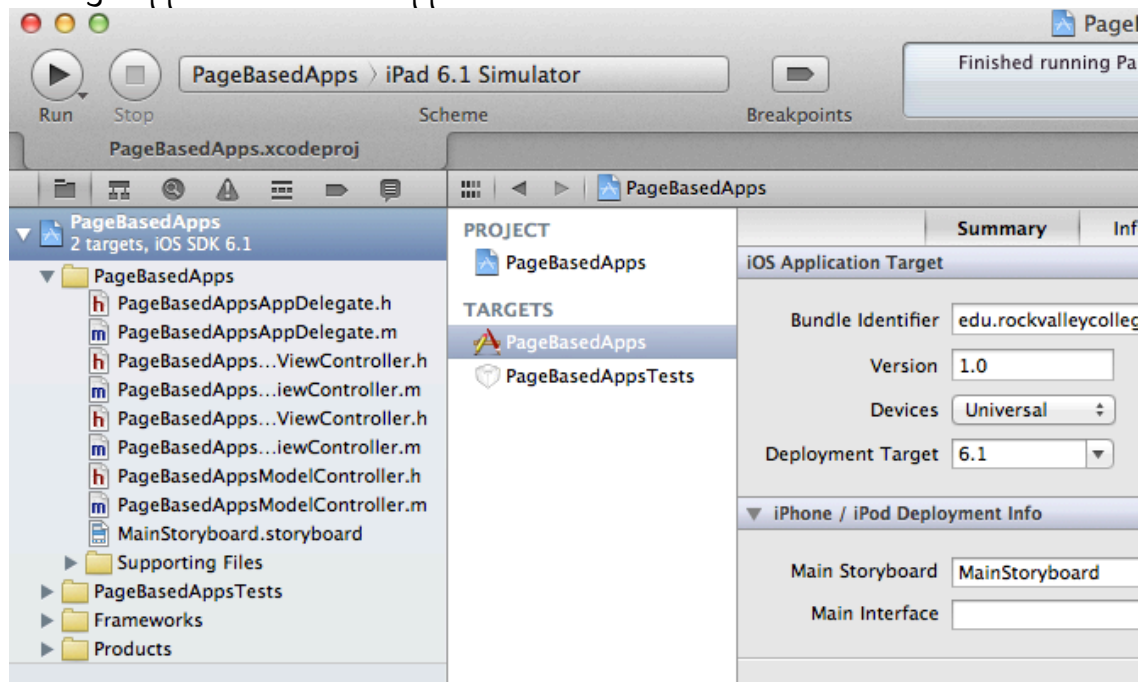
1) Create a new Page-Based Application Project in XCode > NEXT



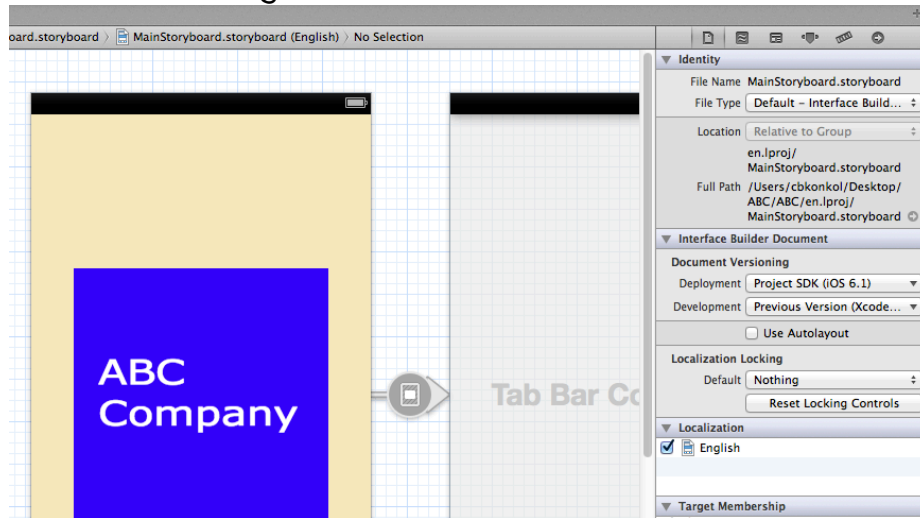
2) Select iPhone, Use as Reference, and Unit Test > NEXT



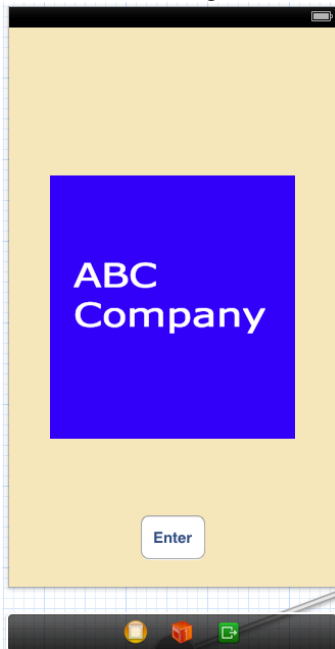
3) Change app to Universal App



#### 4) Uncheck AutoLayout



#### 5) Click on StoryBoard and add a button to PageView

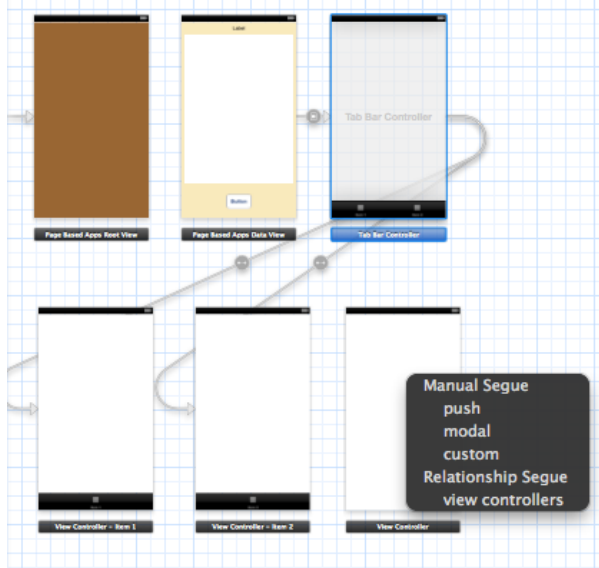


6) Now, add a tab bar controller and ctrl drag button to new tab bar controller



7) Run to see how it works!

8) To Add a new page just add a new **View Controller** to storyboard. Ctrl drag from tab bar controller to new **View Controller** and select **View Controllers** under Segue.

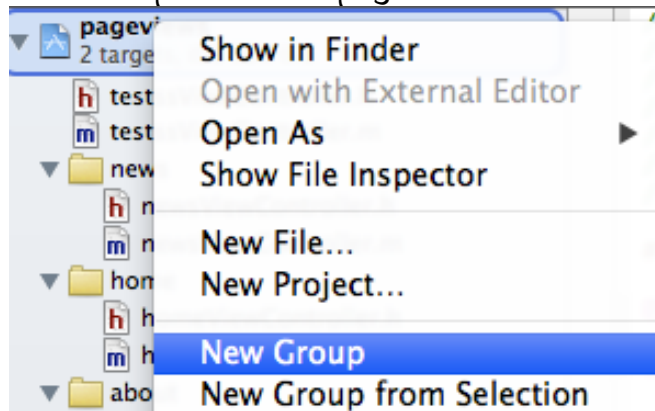


9) Edit button titles on all ViewControllers

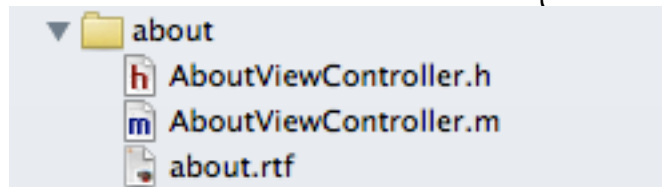
10) Add UIWebView to each ViewController. Check **Scale page to fit**

- 11) Decide if page is going to use external URL, Local html, or Local RTF file
- a. External URL – can be edited after deployment. Good for pages you know can change often
  - b. Local html – updates have to occur before users see changes. Good for static content
  - c. Local RTF – updates have to occur before users see changes. Good for static content that allows for more control of formatting.

12) Add Groups for each page



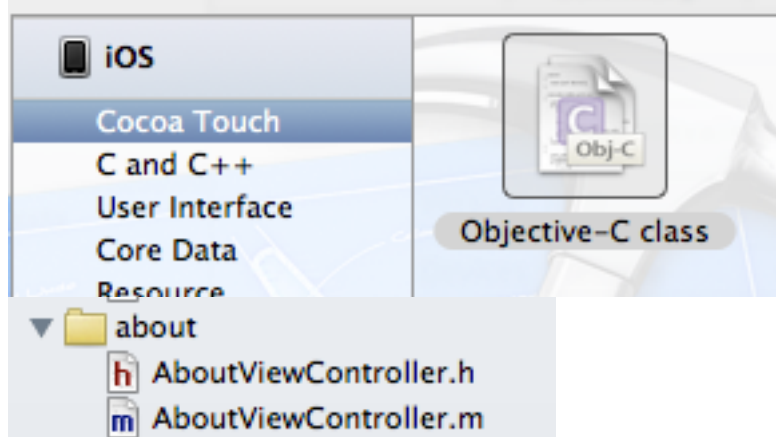
13) Create local html and rtf files and place in corresponding group



14) Add Objective-C Classes for each group

(use name that matches page name. Example: **AboutViewController** for about page)

- right-click on projectname > New File > Cocoa Touch > Objective-C Class



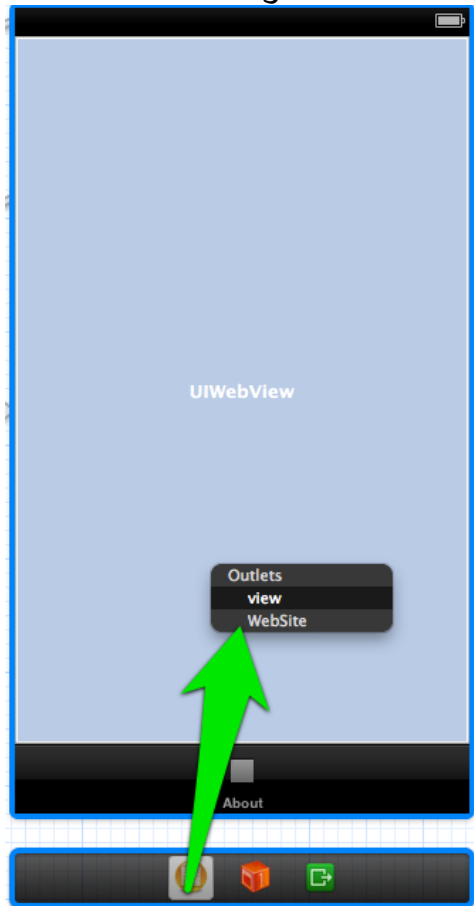
15) Bind ViewControllers to new class you create for each page

- Click on ViewController > Click on BlackBar at top of ViewController
- Click on Identity Inspector on right side of xcode
- Type in new Class under **Custom Class** (example: AboutViewController)
- Save

16) Add code into .h

(See Below **[Code]**)

17) Ctrl + Drag ViewController to **UIWebView** and Select **Website**



18) Add Code to .m file

(See Below **[Code]**)

19) **Add icons using pixlr for ipad and iphone**

20) Create new images for tabbar controller

The main two things to know are:

Your images shouldn't be bigger than 30px by 45px (Apple suggests a 30px by 30px monochrome picture).

## [CODE]

### .h file

```
@property (retain, nonatomic) IBOutlet UIWebView *WebSite;
```

### .m file

```
@synthesize WebSite=WebSite;
```

```
//Three Options for viewDidLoad
```

```
// 1) Add code to load web content in UIWebView of local .htm file
```

```
- (void)viewDidLoad
```

```
{
```

```
    [super viewDidLoad];
```

```
    // Add code to load web content in UIWebView of local .htm file
```

```
    NSURL *url = [NSURL fileURLWithPath:[NSBundle mainBundle]pathForResource:@"news.htm" ofType:nil];
```

```
    NSURLRequest *request = [NSURLRequest requestWithURL:url];
```

```
    [WebSite loadRequest:request];
```

```
}
```



// 2) Add code to load web content in UIWebView to load external website

- (void)viewDidLoad

{

[super viewDidLoad];

// Do any additional setup after loading the view.

// Add code to load web content in UIWebView to load external website

NSURL \*url = [NSURL URLWithString:@"http://rockvalleycollege.edu/"];

NSURLRequest \*request = [NSURLRequest requestWithURL:url];

[WebSite loadRequest:request];

}

//3) Load rtf file in UIWebView

- (void)viewDidLoad

{

[super viewDidLoad];

//Load rtf file in UIWebView

NSString \*filePath = [[NSBundle mainBundle] pathForResource:@"about.rtf"];

NSURL \*fileURL = [NSURL alloc] initWithFileURLWithPath:filePath];

NSURLRequest \*requestObj = [NSURLRequest requestWithURL:fileURL];

[WebSite loadRequest:requestObj];

}

**Add to bottom of .M file / Paste the load code from viewDidLoad. This allows content to reload when you leave and come back to button**

```
-(void)viewWillAppear:(BOOL)animated {
    [super viewWillAppear:animated];
    // Add code to load web content in UIWebView
    NSString *filePath = [[NSBundle mainBundle] pathForResource:@"about.rtf"];
    NSURL *fileURL = [[NSURL alloc] initWithFileURLWithPath:filePath];
    NSURLRequest *requestObj = [NSURLRequest requestWithURL:fileURL];
    [WebSite loadRequest:requestObj];
}
```