

iOS Privacy Policy Documentation

The Privacy Policy is made up of 3 major pieces:

- **PrivacyViewController**
- **AboutWebViewController**
- **Privacy.html**

Of these 3 **PrivacyViewController** is the most important. This is the view that will be linked to within the rest of your source and is the view that opens the webview that contains the privacy policy. This is also where the majority of the image assets and text labels sit.

PrivacyViewController

The **PrivacyViewController** is comprised of 3 major elements: the top bar, the 4 FAQ accordion views, and the final link to the full privacy policy web view.

It functions as a **UIViewController** and can be used simply:

```
[[PrivacyViewController alloc]
initWithNibName:@"PrivacyViewController" bundle nil];
or
[[PrivacyViewController alloc]
initWithNibName:@"PrivacyViewController_iPad" bundle nil];
```

The top bar includes the **navbar-bg.png** (@2x & @3x) image which can be styled any way. In the example it is a color gradient but can be made to look flat as well. It should be 1x44 at base size. There is also the iPad version navbar-bg-ipad.png (@2x) which at base would be 2x148.

The Back Button (**back-btn-about.png**) can be swapped out or simply restyled. The image should be changed to reflect whichever view it came from. If a variable label is required that will have to be built.

privacy-certified.png can be swapped out as necessary for branding. This is the major image in the view.

```
#define COLLAPSED_VIEW_HEIGHT_PHONE 33
#define COLLAPSED_VIEW_HEIGHT_PAD 48

#define VIEW1_HEIGHT_PHONE 140
#define VIEW2_HEIGHT_PHONE 102
#define VIEW3_HEIGHT_PHONE 122
#define VIEW4_HEIGHT_PHONE 157
```

```
#define VIEW1_HEIGHT_PAD 176
#define VIEW2_HEIGHT_PAD 132
#define VIEW3_HEIGHT_PAD 128
#define VIEW4_HEIGHT_PAD 192
```

These values can be changed to alter the heights of the various views. These work in all iPhone and iPad screens. However, they will need to be altered if a larger image is wanted for collapsed views or the text/assets within the expanded views change.

```
- (IBAction)more1Pressed:(id)sender {
    AboutWebViewController *vc = [[AboutWebViewController alloc]
initWithNibName:@"AboutWebViewController" bundle nil];
    vc.titleText = @"Mobile Privacy Policy";
    vc.htmlFile = @"privacy";
    vc.anchor = @"q1";
    [self.navigationController pushViewController:vc
animated:YES];
}
```

Above is the code for a “more” button pressed method. Here **vc.anchor** can be changed to have the new web view automatically open to another anchor. Alternatively the anchor can be moved in the **privacy.html** file. Both the title in **vc.titleText** and the html file in **vc.htmlFile** can also be altered as needed to accommodate new html code or a new web view name.

The final full link is identical to the FAQ accordion links but lacks the **vc.anchor**.

To change any of the text on the page itself the Labels within the **PrivacyViewController.xib** and **PrivacyViewController_iPad.xib** files will have to be changed. The only exceptions are the “About” and “Privacy Policy” texts in the nav-bar. These are images as stated above and labels will either have to be created or new images will need to be provided.

AboutWebViewController

This web view controller is fairly straightforward. So long as the html file is correctly in **Supporting Files/html/** then it will function properly. Once again **navbar-bg.png** and **back-btn-about.png** are present and will have to be changed if any different branding is wished. However, the Title Label is set in the creation method within the **PrivacyViewController** and is not an image.

Privacy.html

This is a simple web page in html. This can be built from scratch and in the example is very simple. Keep in mind that any assets used within the html (e.g. images, separate html pages) are best kept within the same **Supporting Files/html/** folder. In the sample **privacy_seal.png** is such an asset.

Note: These views only support a portrait view. Adjustments will have to be made to allow for them to support Landscape as well.

Also, there are no mock images for iPhone6 Plus (@3x) images within the source. These will have to be added by the developer.

There is a known issue with the main app view overlapping into the status bar. This is due to the age of the source code and will have to be addressed by the developer.