

Mobile MKAbeFook 0.1

mike@pointlesscrap.net

Introduction	2
What's Different	2
<i>Login Window</i>	<i>2</i>
<i>Parsing</i>	<i>3</i>
Class Descriptions	3
<i>MMKFacebook</i>	<i>3</i>
<i>MMKLoginViewController</i>	<i>4</i>
<i>MMKFacebookRequest</i>	<i>4</i>
Compile Requirements	5
<i>Other Linker Flags</i>	<i>5</i>
<i>Header Search Paths</i>	<i>5</i>
<i>Required Classes and Categories</i>	<i>6</i>
<i>Required Frameworks</i>	<i>6</i>
TODO	6

Introduction

Mobile MKAbeFook is still very much a work in progress. The goal of the project is to convert the desktop version into usable code for the iPhone and iPod Touch. There are methods and features left over from the desktop version that have not been tested or implemented yet and there are features that have been temporarily removed or disabled.

Untested methods and features:

- fetchFacebookData: - synchronous request (is there any reason to use it at all?)
- setShouldUseSynchronousLogin: - (is there any reason to use it at all?)
- infinite sessions - (are NSUserDefaults working in the iPhone simulator?)

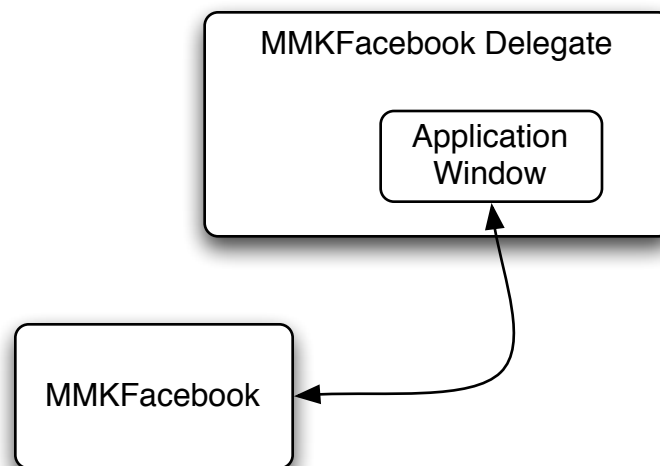
Disabled or temporarily removed features:

- image uploads
- request queue
- grant extended permissions
- framework format

What's Different

Login Window

In Cocoa applications it's easy to create a new NSWindow and display it above all other windows. Within the iPhone environment (Cocoa Touch), instead of creating multiple windows there is one application window and subviews are added, inserted, removed, manipulated, etc... In order to transition in / out the Facebook login window and the loading views the MMKFacebook class needs access to the application window. To accomplish this the MMKFacebook delegate object must implement the MMKFacebook protocol method `-(UIView *)applicationView`. The UIView object returned by this method should be the application window containing of all other controls and views.



If there is a better way to handle displaying the login window and or loading views please let me know.

Parsing

CXML classes used instead of NSXML thanks to <http://code.google.com/p/touchcode/wiki/TouchXML>

JSON support will probably be added thanks to <http://code.google.com/p/touchcode/wiki/TouchJSON>

Class Descriptions

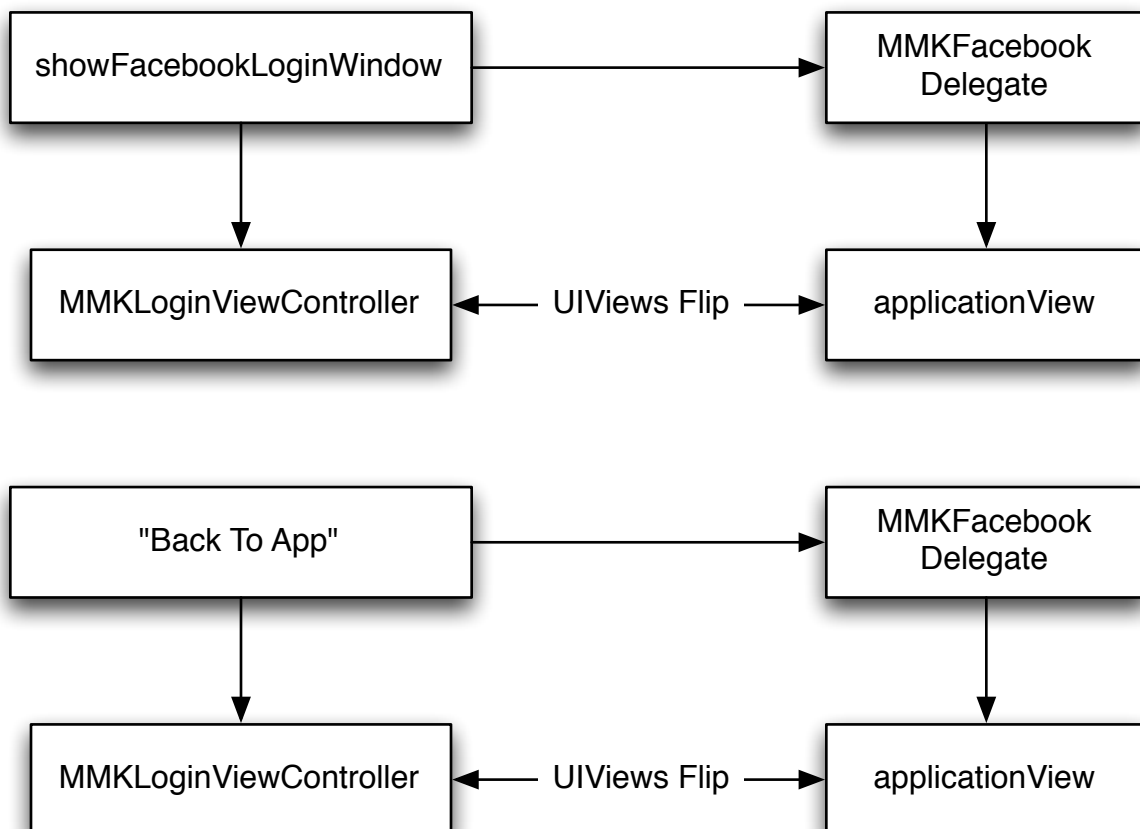
MMKFacebook

The MMKFacebook class is used to set up the initial communication with the Facebook API. It handles the authentication process as described by the Facebook documentation, see the following links for more details:

<http://developers.facebook.com/documentation.php?v=1.0&doc=auth>

http://developers.facebook.com/documentation.php?doc=login_desktop

The biggest change to this class, compared to the desktop version, is related to the showFacebookLoginWindow method and the use of the -(UIView *)applicationView method. Here's what happens when showFacebookLoginWindow is called.



Steps involved with showFacebookLoginWindow:

- A request for an auth token is sent to Facebook
- The MMKLoginViewController and UINavigationController are set up. Note: Use of a UINavigationController and UIViewController here are probably a bit overkill, this might change but shouldn't affect how you use the class.
- The delegate applicationView: method is called and the UIView returned is flipped with the view from the MMKLoginViewController which contains the UIWebView used for logging into Facebook.
- Hopefully by now the request for the auth token has come back from Facebook and the UIWebView has started loading the login page.
- When the "Back To App" button is pressed MMKLoginViewController request for an auth session is sent to Facebook based on the auth token we've received and the MMKLoginViewController view is flipped back to the UIView from applicationView

MMKLoginViewController

This class is a UIViewController subclass responsible for displaying a UIWebView that loads the Facebook login page and a "Back To App" button that returns the user to the application. This class is only used by MMKFacebook and does not need to be used directly. Using a UIViewController subclass here is probably overkill and might change in the future.

MMKFacebookRequest

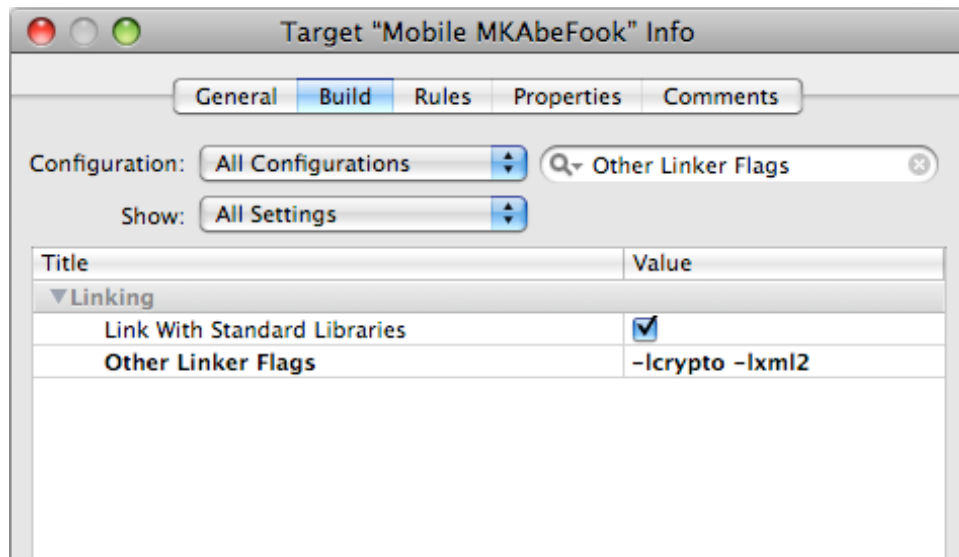
This class uses the applicationView delegate method from the supplied MMKFacebook object to retrieve the application window and either adds a loading subview or transitions with a new user supplied UIView that is displayed while the request is loading. When the request is completed the loading subview is removed or the UIView is transitioned back to the application window.

Compile Requirements

Because things are not packaged as a framework the build settings must be configured accordingly and required files and frameworks must be added.

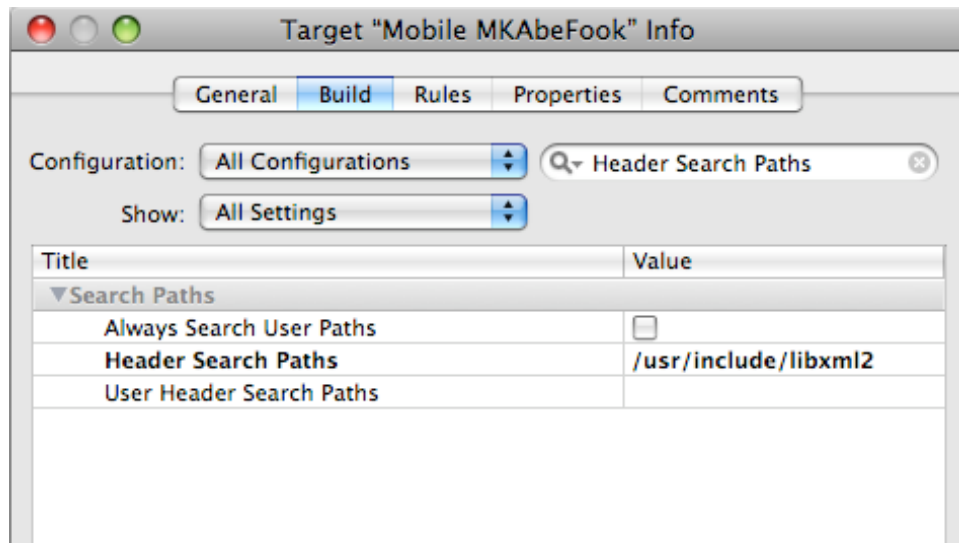
Other Linker Flags

- -lcrypto
- -lxml2

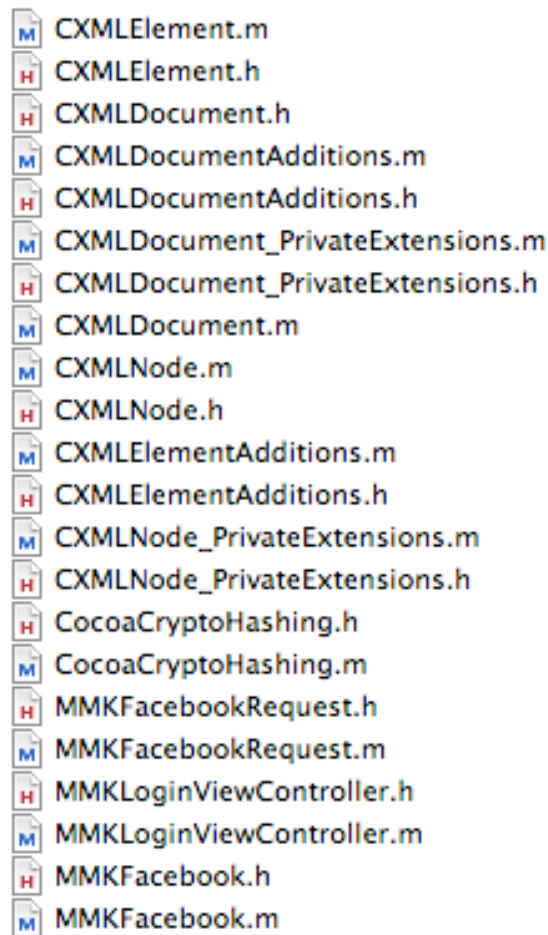


Header Search Paths

- /usr/include/libxml2



Required Classes and Categories



Required Frameworks

- -QuartzCore.framework

TODO

- clean up UIViews used in MMKFacebookRequest
- use properties where applicable
- implement photo uploading
- auto zoom in on login form in UIWebView if possible (at this time Apple documentation is unclear regarding UIWebView and UIScrollView subclass)
- implement infinite sessions (are UserDefaults working in the iPhone simulator?)
- JSON support
- documentation
- code clean up
- examples

Known Issues

- in the simulator, using the “Go” button on the keyboard to submit the login form will cause a crash. User must click done to hide the keyboard then use the Login button within the UIWebView to submit the form.
- animation for widgets and loading sheet from top of screen are out of sync