

Social Media Applications in ios

Twitter has been integrated in **ios 5.0** and Facebook has been integrated in **ios 6.0** Our tutorial focuses on using the classes provided by Apple and the deployment targets for Twitter and Facebook are iOS 5.0 and iOS 6.0 respectively.

Steps involved

Step 1. Create a simple view-based application.

Step 2. Select your project file, then select targets and then add Social.Framework and Accounts.framework in **choose frameworks**.

Step 3 – Add two buttons named facebookPost and twitterPost and create ibActions for them.

Step 4 – Update **ViewController.h** as follows –

```
#import <Social/Social.h>#import <Accounts/Accounts.h>#import <UIKit/UIKit.h>

@interface ViewController : UIViewController

-(IBAction)twitterPost:(id)sender;-(IBAction)facebookPost:(id)sender;

@end
```

Step 5 – Update **ViewController.m** as follows –

```
#import "ViewController.h"

@interface ViewController ()@end

@implementation ViewController

- (void)viewDidLoad {
    [super viewDidLoad];
}

- (void)didReceiveMemoryWarning {
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

-(IBAction)facebookPost:(id)sender {
    SLComposeViewController *controller = [SLComposeViewController
composeViewControllerForServiceType:SLServiceTypeFacebook];
```

```

SLComposeViewControllerCompletionHandler myBlock =
    ^(SLComposeViewControllerResult result){

        if (result == SLComposeViewControllerResultCancelled) {
            NSLog(@"Cancelled");
        } else {
            NSLog(@"Done");
        }

        [controller dismissViewControllerAnimated:YES completion:nil];
    };

controller.completionHandler = myBlock;

//Adding the Text to the facebook post value from iOS
[controller setInitialText:@"My test post"];

//Adding the URL to the facebook post value from iOS
[controller addURL:[NSURL URLWithString:@"http://www.test.com"]];

//Adding the Text to the facebook post value from iOS
[self presentViewController:controller animated:YES completion:nil];

-(IBAction)twitterPost:(id)sender {
    SLComposeViewController *tweetSheet = [SLComposeViewController
composeViewControllerForServiceType:SLServiceTypeTwitter];
    [tweetSheet setInitialText:@"My test tweet"];
    [self presentViewController:tweetSheet animated:YES];} @end

```

Integrate a social media application with iOS's Calendar and Address Book

To integrate a social media application with iOS's Calendar and Address Book (Contacts) using WiFi, you'll need to leverage iOS's built-in frameworks and potentially third-party APIs for social media platforms. This involves setting up accounts for the relevant services in Settings, and then using the appropriate frameworks (like EventKit for Calendar and AddressBook for Contacts) to access and interact with the data within your application.

Here's a more detailed breakdown:

1. Setting up Accounts:

Calendar:

Go to Settings > Calendar > Accounts > Add Account and choose your provider (iCloud, Google, etc.).

Address Book:

Go to Settings > Contacts > Accounts > Add Account and choose your provider (iCloud, Google, etc.).

2. Using Frameworks:

EventKit (Calendar):

This framework allows you to access calendar data, create, edit, and delete events.

AddressBook (Contacts):

This framework provides access to the address book, allowing you to search, retrieve, and modify contact information.

3. Integrating with Social Media Application:

Social Media APIs:

Most social media platforms have APIs that allow you to interact with their data, including events and contacts.

Data Exchange:

Your app can use these APIs to:

1. Share calendar events with social media (e.g., invite friends to an event).
2. Integrate contact information from your app with social media profiles.

3. Fetch events or contact information from social media and display them within your app.

4. WiFi Dependency:

Connectivity:

Your application will need a WiFi connection to access the internet and interact with social media APIs and your cloud-based accounts.

Data Transfer:

Data transfer to and from the social media platforms and your cloud-based accounts will occur over WiFi.

Example Scenario:

Imagine you want to allow users to invite friends to a social media event from within your app. You could:

1. **Access User Contacts:** Use AddressBook to get the user's contacts.
2. **Present Contacts:** Display these contacts in your app's interface.
3. **Select Contacts to Invite:** Allow the user to select which friends to invite.
4. **Create an Event:** Use EventKit to create a calendar event.
5. **Share Event:** Use the social media API to share the event with the selected contacts.