iOS end-to-end demonstration

Overview

The purpose of this demonstration is to experience an end-to-end flow where an application is quickly created using the MobileFirst Operations Console and connectivity is verified with the MobileFirst Server.

Prerequisites:

- Configured Xcode
- *Optional* Stand-alone MobileFirst Server and MobileFirst CLI (download (file:////home/travis/build/MFPSamples/DevCenter/_site/downloads))
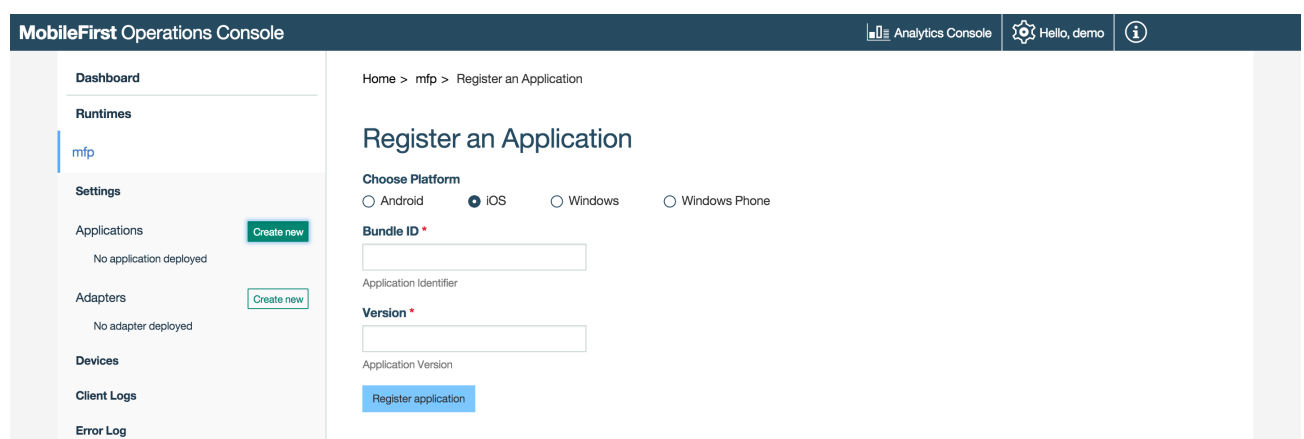
1. Starting the MobileFirst Server

> If a remote server was already set-up, skip this step.

1. From a **Command-line** window, navigate to the server's **scripts** folder and run the command: `./start.sh`.

2. Creating an application

In a browser window, open the MobileFirst Operations Console by loading the URL: `http://your-server-host:server-port/mfpconsole`. If running locally, use: http://localhost:9080/mfpconsole (http://localhost:9080/mfpconsole). The username/password are *admin/admin*.

1. Click on the "Create new" button next to **Applications** and select the desired *platform*, *identifier* and *version* values.

2. Click on the **Get Starter Code** tile and select to download the iOS Starter Code.

⟨ Back

Home ＞ mfp ＞ MyBankApp ＞ iOS 1.0   Delete version

mfp

Applications   Create new   **MyBankApp iOS v 1.0**

    MyBankApp   ✓   Your application is now registered!   ✕

Platform
    ⌃ iOS   1   Next Steps   ✕
       1.0

Push

Get Starter Code	Set Up Security	Set Up Push	Get CLI

3. Editing application logic

1. Open the Xcode project project

2. Select the **[project-root]/AppDelegate.m** file and paste the following code snippet:

In Objective-C:

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOpt
ions:(NSDictionary *)launchOptions {
    WLAuthorizationManager.sharedInstance().obtainAccessTokenForScope(nil)
{
        (token, error) -> Void in

        if (error != nil) {
            NSLog(@"Access token not granted.");
        } else {
            NSLog(@"Access token granted.");
        }
    }

    return YES;
}
```

In Swift:

```
func application(application: UIApplication, didFinishLaunchingWithOptions
launchOptions: [NSObject: AnyObject]?) -> Bool {
    WLAuthorizationManager.sharedInstance().obtainAccessTokenForScope(nil)
{
        (token, error!)->Void

        if error != nil {
            NSLog("Access token not granted.")
        }
        else {
            NSLog("Access token granted.")
        }

        return true
    }
}
```

4. Testing the application

1. In Xcode, press the **Play** button.

Next steps

- To add an adapter follow the Adapter end-to-end demonstration (../adapter)
- Review All Tutorials (../../all-tutorials)