Windows 8.1 Universal and Windows 10 UWP end-to-end demonstration

fork and edit tutorial (https://github.ibm.com/MFPSamples/DevCenter/tree/master/tutorials/en/foundation/8.0/quick-start/windows-8-10/index.md) | report issue (https://github.ibm.com/MFPSamples/DevCenter/issues/new)

Overview

The purpose of this demonstration is to experience an end-to-end flow where an application and an adapter are registered using the MobileFirst Operations Console, an "skeleton" Visual Studio project is downloaded and edited to call the adapter, and the result is printed to the log - verifying a successful connection with the MobileFirst Server.

Prerequisites:

- Configured Visual Studio 2013/5
- MobileFirst Developer CLI (download (file:////home/travis/build/MFPSamples/DevCenter/_site/downloads))
- *Optional*. Stand-alone MobileFirst Server (download (file:////home/travis/build/MFPSamples/DevCenter/_site/downloads))

1. Starting the MobileFirst Server

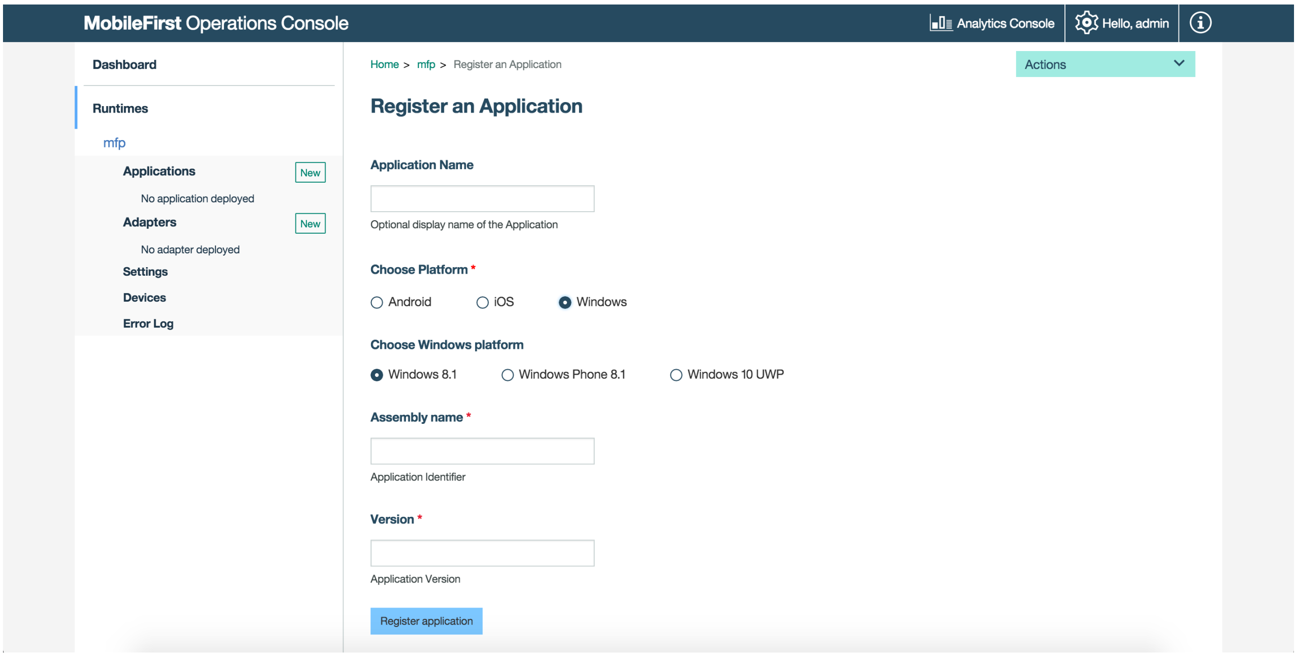
> If a remote server was already set-up, skip this step.

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

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 "New" button next to **Applications** and select the desired *platform*, *identifier* and *version* values.



Register an Application

Application Name

Optional display name of the Application

Choose Platform *

○ Android ○ iOS ● Windows

Choose Windows platform

● Windows 8.1 ○ Windows Phone 8.1 ○ Windows 10 UWP

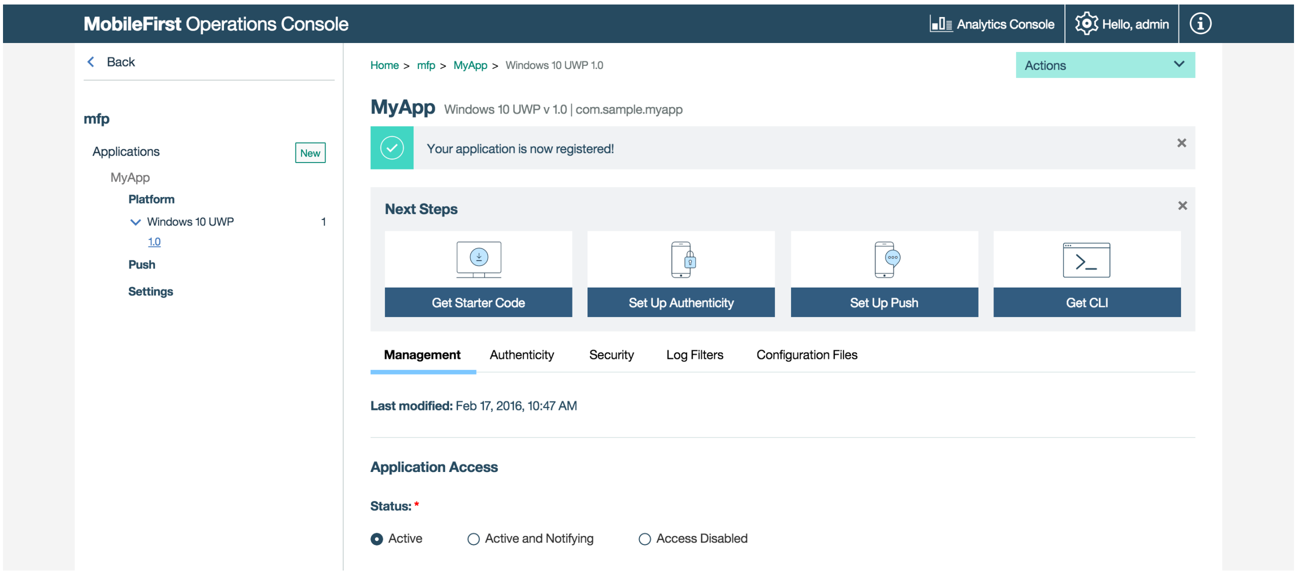
Assembly name *

Application Identifier

Version *

Application Version

2. Click on the **Get Starter Code** tile and select to download Windows 8.1 or Windows 10 Starter Code.



MyApp Windows 10 UWP v 1.0 | com.sample.myapp

Your application is now registered!

Next Steps

Last modified: Feb 17, 2016, 10:47 AM

Application Access

Status: *

● Active ○ Active and Notifying ○ Access Disabled

3. Editing application logic

1. Open the Visual Studio project.
2. Select the solution's **MainPage.xaml.cs** file and paste the following code snippet:

```
IWorklightClient _newClient = WorklightClient.CreateInstance();

StringBuilder uriBuilder = new StringBuilder().Append("/adapters/javaAdapter/users/world");

WorklightResourceRequest rr = _newClient.ResourceRequest(uriBuilder.ToString(), "GET");

WorklightResponse resp= return Task.Run<WorklightResponse> (() => {
   rr.send();
});

if (resp.success) {
  Debug.WriteLine("Success: " + resp.ResponseText);
} else {
  Debug.WriteLine("Failure: " + resp.error);
}
```

4. Creating an adapter

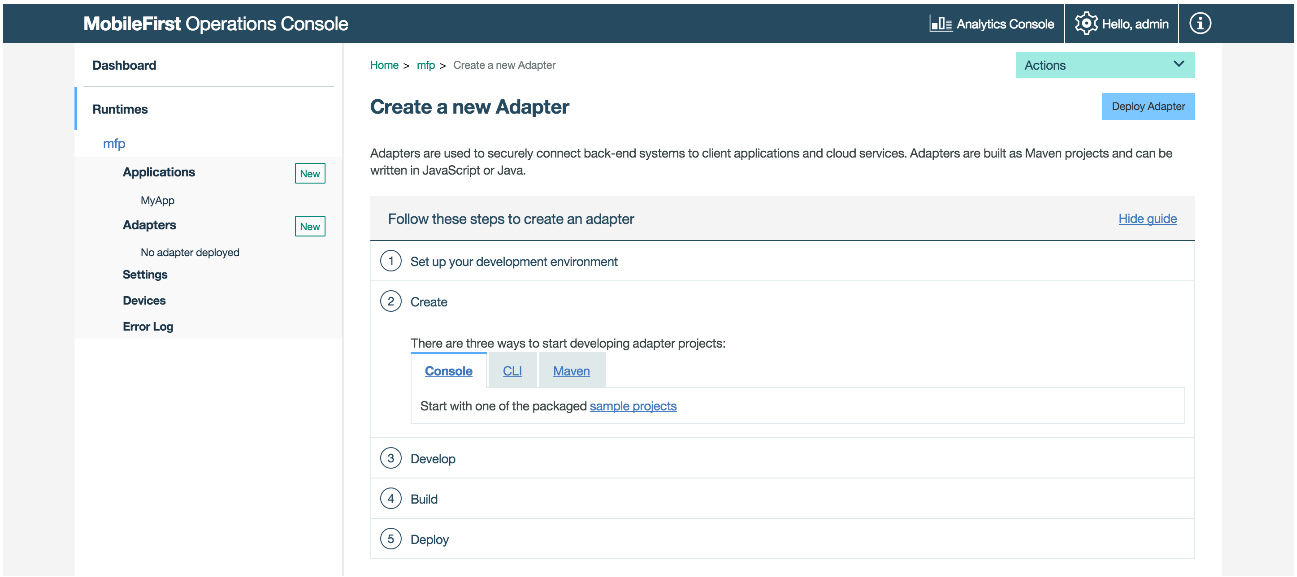
1. Click on the "New" button next to **Adapters**
   - Select the **Actions → Download sample** option. Download the **Java** adapter sample.

     > If Maven and MobileFirst CLI are not installed, follow the on-screen **Setting up your environment** instructions to install.

   - From a **Command-line** window, navigate to the adapter's Maven project root folder and run the command:

     ```
     mfpdev adapter build
     ```

   - When the build finishes, deploy it from the MobileFirst Operations Console using the **Actions → Deploy adapter** action. The adapter can be found in the **[adapter]/target** folder.
   - Alternatively, download this prepared .adapter artifact and deploy it from the MobileFirst Operations Console using the **Actions → Deploy adapter** action.



Create a new Adapter

Adapters are used to securely connect back-end systems to client applications and cloud services. Adapters are built as Maven projects and can be written in JavaScript or Java.

Follow these steps to create an adapter

1 Set up your development environment

2 Create

There are three ways to start developing adapter projects:

Start with one of the packaged sample projects

3 Develop

4 Build

5 Deploy

5. Testing the application

1. In Visual Studio, click on the **Start Debugging** button.

Next steps

Learn more on using adapters in applications, and how to integrate additional services such as Push Notifications, using the MobileFirst security framework and more:

- Review the Server-side development tutorials (../../adapters/)
- Review the Authentication and security tutorials (../../authentication-and-security/)
- Review the Notifications tutorials (../../notifications/)
- Review All Tutorials (../../all-tutorials)