

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

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 folder and run the command: `run.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**
 - Select a **Windows** platform
 - Enter **MFPStarterCSharp.Windows** as the **application identifier** for Windows, or **MFPStarterCSharp.WindowsPhone** for Windows Phone
 - Enter **1.0.0** as the **version** value
 - Click on **Register application**

MobileFirst Operations Console

Home > mfp > Register an Application

Register an Application

Application Name
MFPSarterCSharpWindows ✓
Optional display name of the Application

Choose Platform *
☐ Android
 ☐ iOS
 ☒ Windows

Choose Windows platform
☐ Windows 8.1
 ☐ Windows Phone 8.1
 ☒ Windows 10 UWP

Package Identity Name *
MFPSarterCSharp.Windows ✓
Application Identifier

Version *
1.0 ✓
Application Version

Register application

- Click on the **Get Starter Code** tile and select to download the Windows 8.1 or Windows 10 mobile app scaffold.

MobileFirst Operations Console

Home > mfp > MyApp > Windows 10 UWP 1.0

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

✓ Your application is now registered!

Next Steps

Get Starter Code Set Up Authenticity Set Up Push Get CLI

Management Authenticity Security Log Filters Configuration Files

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

Application Access

Status: *
☒ Active
 ☐ Active and Notifying
 ☐ Access Disabled

3. Editing application logic

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

```

WorklightAccessToken accessToken = await Worklight.WorklightClient.CreateInstance().AuthorizationManager.ObtainAccessToken("");

if(accessToken.IsValidToken && accessToken.Value != null && accessToken.Value != "")
{
    try
    {
        IWorklightClient _newClient = WorklightClient.CreateInstance();

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

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

        WorklightResponse resp= await rr.send();

        if (resp.success)
        {
            Debug.WriteLine("Success: " + resp.ResponseText);
        } else
        {
            Debug.WriteLine("Failure: " + resp.error);
        }
    } catch(Exception e)
    {
        Debug.WriteLine(e.StackTrace);
    }
}

```

4. Creating an adapter

Download this prepared .adapter artifact (../javaAdapter.adapter) and deploy it from the MobileFirst Operations Console using the **Actions → Deploy adapter** action.

Alternatively, click on the "New" button next to **Adapters**.

1. Select the **Actions → Download sample** option. Download the "Hello World" **Java** adapter sample.

If Maven and MobileFirst Developer CLI are not installed, follow the on-screen **Set up your development environment** instructions.

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

```
mfpdev adapter build
```

3. 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.

