Setting up the MobileFirst development environment

Overview

IBM MobileFirst Foundation is made up of several components: the client SDKs, adapter archetypes, security checks, and authentication tools.

These components are available from online repositories and can be installed using package managers. These online repositories provide the latest release of each component. The same component is also available to download from the MobileFirst Foundation Developer Kit for local use. Note that the version that is available from the Developer Kit represents the version that was available at the time the specific Developer Kit build was released, and that downloading a new Developer Kit build will be required in order to use the latest.

Continue reading to learn more about the components of MobileFirst Foundation.

> To evalute MobileFirst Foundation all that is needed is to spin an instance of MobileFirst Server on Bluemix using the Mobile Foundation Bluemix service. See the Using Mobile Foundation (../../../bluemix/using-mobile-foundation/) tutorial for instructions. You may also choose to install the Developer Kit for a local installation.

Jump to:

Installation guide

Read the installation guide (installation-guide) to quickly setup MobileFirst Foundation in your workstation.

MobileFirst Foundation Development Kit

The Developer Kit provides a ready-for-development environment with minimal configuration needed. The kit consists of the following components: MobileFirst Server & MobileFirst Operations Console, MobileFirst Developer Command-line Interface (CLI), as well as optionally provides client SDKs and adapter tooling for download.

> **Note:** If you need to set up your development environment on a computer that has no internet access, you can install components offline. See How to set up an offline IBM MobileFirst development environment (file:////home/travis/build/MFPSamples/DevCenter/_site/blog/2016/03/31/howto-set-up-an-offline-ibm-mobilefirst-8-0-development-environment).

Developer Kit Installer

The Installer packages the components for local installation where Internet connectivity is not available. The components are available through the MobileFirst Operations Console's Download Center.

> To download the installer, visit the downloads (file:////home/travis/build/MFPSamples/DevCenter/_site/downloads/) page.

MobileFirst Foundation components

MobileFirst Server

As part of the Developer Kit, the MobileFirst Server is provided pre-deployed on a WebSphere Liberty profile application server. The server is pre-configured with an "mfp" runtime and uses a filesystem-based Apache Derby database.

In the Developer Kit's root directory, the following scripts are available:

- `run.[sh|cmd]`: Run the MobileFirst Server with trailing Liberty Server messages
  - Add the `-bg` flag to run the process in the background
- `stop.[sh|cmd]`: Stop the current MobileFirst Server instance
- `console.[sh|cmd]`: Open the MobileFirst Console

Adding the MobileFirst Server to Eclipse

The MobileFirst Server can be integrated into the Eclipse IDE.

1. From the **Servers** view in Eclipse, select **New → Server**.
2. If an IBM folder option does not exist, click on "Download additional server adapters".
3. Select **WebSphere Application Server Liberty Tools** and follow the on-screen instructions.
4. From the **Servers** view in Eclipse, select **New → Server**.
5. Select **IBM → WebSphere Application Server Liberty**.
6. Provide a server **name** and **hostname** and click **Next**.
7. Provide the path to the server's root directory, and select a JRE version to use.
8. Click **Next** followed by clicking **Finish**.

You can now start and stop the MobileFirst Server from the Eclipse IDE "servers" view.

MobileFirst Operations Console

The MobileFirst Operations Console exposes the following functionalities.
A developer can:

- Register and deploy applications and adapters
- Optionally download native/Cordova application and adapter starter code templates
- Configure an application's authentication and security properties
- Manage applications:
  - Application Authenticity
  - Direct Update
  - Remote Disable/Notify
- Send Push Notifications to iOS and Android devices
- Generate DevOps scripts for continuous integration workflows and faster development cycles

> Learn more about the MobileFirst Operations Console in the Using the MobilFirst Operations Console (../../../product-overview/components/console/) tutorial.

MobileFirst Command-line Interface

You can use the IBM MobileFirst Foundation Command Line Interface (CLI) to develop and manage applications, in addition to using the IBM MobileFirst Platform Operations Console. The CLI command are prefixed with `mfpdev` and support the following types of tasks:

- Registering apps with the MobileFirst Server
- Configuring your app
- Creating, building, and deploying adapters
- Previewing and updating Cordova apps

> To download and install the MobileFirst CLI, visit the downloads (file:////home/travis/build/MFPSamples/DevCenter/_site/downloads/) page.
> Learn more about the various CLI commands in the Using CLI to manage MobileFirst artifacts (../../../application-development/using-mobilefirst-cli-to-manage-mobilefirst-artifacts/) tutorial.

MobileFirst Foundation client SDKs and adapter tooling

MobileFirst Foundation provides client SDKs for Cordova applications as well as for Native platforms (iOS, Android and Windows 8.1 Universal & Windows 10 UWP). Adapter tooling for adapters and security checks development is available as well.

- To use the MobileFirst client SDKs, visit the Adding the MobileFirst Foundation SDK (../../../application-development/sdk/) tutorials category.
- To develop adapters, visit the Adapters (../../../adapters/) tutorials category.
- To develop security checks, visit the Authentication and security (../../../authentication-and-security/) tutorials category.

Applications and adapters development

Applications

- Cordova applications require NodeJS and the Cordova CLI. Read more about setting up the Cordova development environment (../cordova).

  You can use your preferred code editor, such as Atom.io, Visual Studio Code, Eclipse, IntelliJ and others, to implement applications and adapters.

- Native applications require either Xcode, Android Studio or Visual Studio. Read more about setting up the iOS/Android/Windows development environment (../).

Adapters

Adapters require Apache Maven to be installed. Refer to the Adapters (../../../adapters/) category to learn more about adapters and how to create, develop and deploy.

Tutorials to follow next

Visit the All Tutorials (../../../all-tutorials/) page and select a tutorials category to follow next.