

# Using a MobileFirst application as a container for server-generated pages

This tutorial covers the following topics:

- Migrating applications to IBM MobileFirst Platform Foundation
- Running your application on the Android emulator
- Running your application on the iOS emulator
- Sample application

## Migrating applications to IBM MobileFirst Platform Foundation

- By using mobile web technology, you can deploy applications to the widest variety of devices.
- The existence of public application stores, such as Apple iTunes and Google Play, changes the way applications are hosted and marketed. These changes make traditional methods of distribution less relevant.
- IBM MobileFirst Platform Foundation provides the solution to build cross-platform applications that can be distributed through the application stores by using the hybrid application programming model.
- In the hybrid model, developers typically package the application HTML, CSS, and JavaScript™ code as part of the application that is deployed to the application store.
- This tutorial shows the remote loading of dynamic content capability, where the HTML, CSS, and JavaScript code is hosted externally from the natively packaged hybrid.

## Creating MobileFirst applications

### CLI

From a terminal window, use the following CLI commands to add a project and application:

```
mfp create UsingApplicationAsAContainer
cd UsingApplicationAsAContainer
mfp add hybrid UsingApplicationAsAContainer
```

Next, add a required environment, for example Android:

```
cd UsingApplicationAsAContainer
mfp add environment android
```

### Studio

In MobileFirst Studio, create a new project, "UsingApplicationAsAContainer"



A project might host multiple applications. However, in this tutorial, you use only one app: UsingApplicationAsAContainer



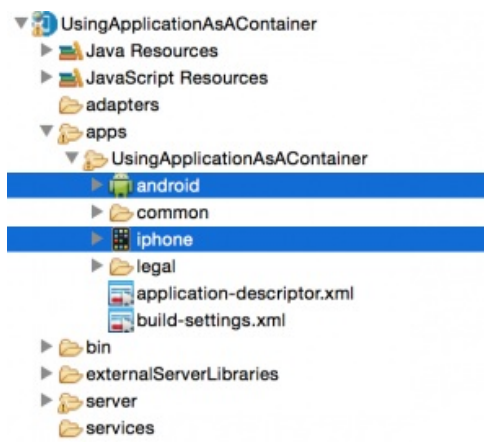
You can set the target environments either while you are working in the **MobileFirst Project** wizard, or later.



## IBM MobileFirst environments

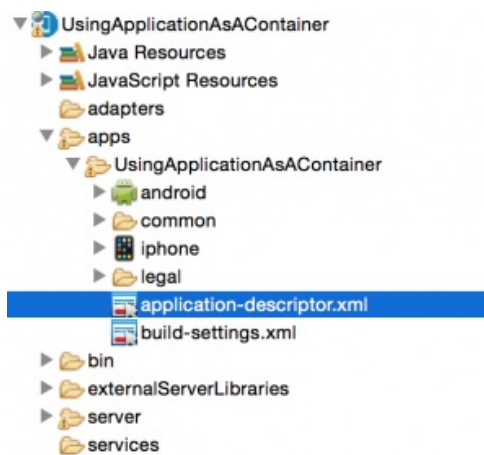
- IBM MobileFirst Foundation provides a model for organizing the application project structure for each target environment (for example, Android, iPhone, iPad).
- You select your target environment through the **MobileFirst Environment** wizard.





## IBM MobileFirst Common environment

- The simplest way to use MobileFirst apps as containers for server-generated pages is through the Common environment.
- Open the application-descriptor.xml file and edit the mainFile tag to point to <http://m.ibm.com>.



```

1 <?xml version="1.0" encoding="UTF-8" standalone="no"?>
2 <application xmlns="http://www.worklight.com/application-descriptor" id="UsingApplicationAsAContainer" platformVersion="6.3.0.00.2014"
3   <displayName>UsingApplicationAsAContainer</displayName>
4   <description>UsingApplicationAsAContainer</description>
5   <author>
6     <name>application's author</name>
7     <email>application author's e-mail</email>
8     <homepage>http://mycompany.com</homepage>
9     <copyright>Copyright My Company</copyright>
10  </author>
11  <mainFile>http://m.ibm.com</mainFile>
12  <features/>
13  <thumbnailImage>common/images/thumbnail.png</thumbnailImage>
14  <iphone bundleId="com.UsingApplicationAsAContainer" version="1.0">
15    <worklightSettings include="false"/>
16    <security>
17      <encryptWebResources enabled="false"/>
18      <testWebResourcesChecksum enabled="false" ignoreFileExtensions="png, jpg, jpeg, gif, mp4, mp3"/>
19    </security>
20  </iphone>
21  <android version="1.0">
22    <worklightSettings include="false"/>
23    <security>
24      <encryptWebResources enabled="false"/>
25      <testWebResourcesChecksum enabled="false" ignoreFileExtensions="png, jpg, jpeg, gif, mp4, mp3"/>
26      <publicSigningKey/>
27      <packageName/>
28    </security>
29  </android>
30 </application>

```

## Running your application on the Android emulator

1. Build the Android environment.
2. Deploy the application.
3. Right-click the generated Android project and click **Run As > Android Application**.

You can see that the `http://m.ibm.com` URL is displayed in your Android emulator.





## Running your application on the iOS emulator

1. Deploy the application to your iOS emulator.
2. Right-click the IOS environment and click **Run As > Xcode project**.



## Sample application

Click to download

(<http://public.dhe.ibm.com/software/products/en/MobileFirstPlatform/docs/v700/UsingWorklightApplicationAsAContainerProject.zip>)  
the Studio project.

