Exposing an IBM BPM process to a mobile device with IBM MobileFirst

Starting a business process from a mobile device

Paul Pacholski  
Cristina L. Tecsa  
Luis Felipe Guzman  

November 04, 2015

Learn how to generate an IBM MobileFirst application that interacts with IBM Business Process Manager (BPM) through a secure adapter so that authorized users of your process application can activate, work on, and complete tasks from their mobile devices.

Introduction

In today's dynamic business environments, users expect to be connected to their critical business processes while on the go. The value of mobile integration is significant, expanding the use-case scenarios for IBM BPM processes to reach customers that are rapidly moving towards a new era of mobility.

Integrating with mobile technology provides business agility with real-time access to IBM BPM from remote locations. By enabling mobile workers to interact with IBM BPM processes, enterprises can significantly optimize business responsiveness, increase worker productivity, and improve processing times. A secure mobile integration means that the IBM BPM portfolio can expand to include business-to-consumer customer scenarios.

Starting with IBM BPM V8.5.6, the IBM MobileFirst development license is bundled with IBM BPM. To support MobileFirst as a mobile platform, IBM BPM V8.5.6 includes the following new mobile capabilities:

- Plug-in adapters for the MobileFirst development environment make it easier to build mobile applications with secure REST APIs that access IBM BPM.
- Technical demonstrations of the Responsive Federated Portal and the Responsive Coach Toolkit are available as samples on the IBM BPM Developer Center to see how to use the federation and mobile capabilities. You can generate a MobileFirst mobile application directly from a human service. This tutorial explores generating an IBM MobileFirst application that interacts with IBM BPM through a secure adapter, so that authorized users can activate, work
on, and complete tasks from their mobile devices. The tutorial shows an example of how you can create mobile applications to start process instances by reusing a human service that is already part of a process.

The first part explores a completed solution, in particular the mobile application that gets generated, and its interaction with a process. The second part describes in detail how the mobile application was built, deployed, and made secure.

Exploring the example solution

The business scenario that is used in this tutorial is shown in Figure 1. Order Fulfillment Process is a process to manage vendor interactions in a supply chain. It focuses on obtaining and approving quotes from third-party suppliers.

Figure 1. Order Fulfillment Process example business process

The process can be started through the Process Portal in IBM BPM, as shown in Figure 2, where the Create Order task is assigned and completed by a member of the Analysts team. The next step, Verify Order, is then assigned to the Managers team.

Figure 2. Order Fulfillment Process started from Process Portal
Alternatively, the process instance can be started by a member of the Analyst team, by using a mobile app from a mobile device that is connected to the public cloud. See Figure 3.

**Figure 3. Order Fulfillment Process started from a mobile device in public cloud**

![Image of Order Fulfillment Process](image)

In this case, the Create Order step is skipped, and the process instance starts at the Verify Order step.

The example mobile app that starts the IBM BPM process from mobile device has the following functional requirements:

- The user interface to provide order details (Create Order task) must be reused. The mobile app to start the process instance and the Create Order task user interface that is running on a desktop must be implemented by using the same client-side human service.
- The mobile app must use responsive user interfaces that can accommodate different display sizes.
- The mobile app must run in public cloud, but it must be able to start a process instance on the IBM Process Server that is running behind the corporate firewall.

IBM BPM V8.5.6 and later releases include the following features to fulfill those three requirements:

- To reuse the original user interface that is defined for the desktop, the tutorial takes advantage of the capability to generate a MobileFirst mobile application directly from a human service. To ensure that the same user interface is usable on both large and small display sizes, the tutorial uses the Responsive Settings in Coach View Editor. See the [Responsive settings for coach views](https://www.ibm.com/support/knowledgecenter/SSSSTX_8.5.6/com.ibm.bpm.tools.doc/topics/respsettings.html) topic in the IBM BPM V8.5.6 documentation on IBM Knowledge Center.
- To ensure that the mobile application is lightweight and responsive, the tutorial uses Responsive settings for coach views from the Responsive Coaches Toolkit V8.5.6. See [bpm-responsive-coach](https://www.ibm.com/support/knowledgecenter/SSSSTX_8.5.6/com.ibm.bpm.tools.doc/topics/tools/reference/tools/bmrt.html).
- Access to the Process Server running behind a firewall from a mobile app that is running in a public cloud is not secure. To avoid direct connection from the mobile app to Process Server, the tutorial uses IBM MobileFirst Servers. The tutorial also uses the IBM BPM Generic Adapter, which acts as a proxy between the mobile app and the Process Server, as shown in Figure 4.
Starting the Order Fulfillment process instance

This section examines the user experience of starting the Order Fulfillment Process from a mobile app and contrasts it with the on-premises user experience with the Responsive Federated Portal web application. With responsive settings in the coach view editor, it is possible to create user interfaces that provide different user experiences with the same coach – depending on the device size.

There are two ways that you can start the Order Fulfillment process instance: from a mobile device or from the Responsive Federated Portal.

**Starting the process instance from a mobile app**

To start a process instance from a mobile device, open the IBM MobileFirst Application. In Figure 5, the application is called **CreateOrder Mobile**.

**Figure 5. Example mobile app that opens the Order Fulfillment process instance**

Starting the process instance from the Responsive Federated Portal

To open a process instance from an on-premises web browser, start the IBM BPM Responsive Federated Portal web application, as shown in Figure 6.
Figure 6. Responsive Federated Portal that opens the example Order Fulfillment process instance

Working with the Order Fulfillment process instance user interface

1. To start a process instance from a mobile device, the mobile user opens the Create Order mobile app, as shown in Figure 7.

Figure 7. Example of opening the mobile app from a mobile device

The mobile app starts. After the mobile user logs in, the Introduction screen opens, as shown in Figure 8.

Figure 8. Introduction screen example
2. After the mobile user selects **Next**, the step that is part of a three-step order creation opens, as shown in Figure 9.

The first step, called **Review**, is highlighted in the status bar. The mobile user can enter order information: **Material, Customer, Priority, Issue Date**, and **Planner Comments**. The responsive coach views in IBM BPM make it possible to enable different user interface behavior on devices with different screen sizes.

**Figure 9. Review step for the Create Order mobile application**

![Review step for the Create Order mobile application](image.png)

The views are responsive and tailored to mobile usage, as shown in Figure 10. For example, the **Order Details** section is collapsed by default on mobile devices and expanded on desktops.
Figure 10. Comparison of the mobile device view and the desktop view (responsive behavior)

3. Selecting **Find Material** opens the Available Materials screen where the mobile user can select a material, as shown in Figure 11.
4. Selecting **Select and Return** returns to the Order Information screen with the Material field completed based on the selection on the Available Materials screen.

5. After the order information is completed, the mobile user can collapse that section and expand the **Order Details** section, as shown in Figure 12. Then, the mobile user can provide the required quantity for the order, the delivery date for each quantity, and the pricing information and comments that justify the order details.
6. Selecting **Next** brings the mobile user to the second step of the Create Order application, with the ability to select vendors to compete in the bidding process, as shown in Figure 13. The Vendors step is highlighted in the status bar. Selecting the **Columns** configuration allows table columns customization, including more columns on the desktop where there is space for more columns that are presented on the screen.
7. Selecting **Next** opens the last step of the Create Order human service, which is Confirm, shown in Figure 14. The Order Information and Order Details sections contain read-only information, based on the input that is provided in the previous steps. By default, these sections are collapsed on mobile devices and expanded on desktops.
After the mobile user completes the application, a process is automatically started on the IBM BPM side, passing the Order business object as a process input variable. This business object encapsulates all the information that is entered by the mobile user from the Create Order mobile application. At the same time, the startedFromMobile flag is set to true. Therefore, the token is automatically moved to the Verify Order task in the Managers swimlane. Review the Order Fulfillment process in Figure 3.

The rest of the following steps in the process are identical, whether the process was started from a mobile device or from a desktop browser that uses Process Portal:

- The Vendors input their bidding proposal and send a response back to the sender.
- The Analysts analyze the response and select one vendor that best fulfills the order requirements, based on the bidding data.
- The Manager evaluates the proposal and the selected vendor response, and then approves or rejects the order fulfillment.
- If the order is rejected, the process flow returns to the Analysts to select a fulfillment option.
- If the order is approved, the contract with the selected vendor proceeds, and the process is successfully completed.

This section showed the scenario from a user interaction perspective. It highlights how mobile workers can securely create orders in the public cloud from mobile devices and start IBM BPM processes that process the orders.
You can download and work with a process app file named `Mobile_Enabled_Order_Fulfillment.twx` that contains the solution that is described in this tutorial. From the Download section of this tutorial, download the `Mobile_Enabled_Order_Fulfillment.twx` file and import it to your Process Center.

**Building the mobile solution**

This section describes how to author a mobile application (targeting the Android operating system) from the client-side human service.

**Solution components**

The following components that are shown in Figure 15 are part of the solution:

- IBM BPM
  - Author the client-side human service.
  - Generate the project file for the client-side human service.
- IBM MobileFirst Studio
  - Generate application artifacts from the project file.
  - Test application artifacts.
- IBM MobileFirst Test/Production Server
  - Deploy the application here by using the application artifacts.
  - Connect to IBM BPM by using the secure application artifacts.
- Android Device
  - Connect to the MobileFirst Test/Production Server.
  - Run the client-side human service.

**Figure 15. Deployment flow of a client-side human service as a mobile application**

![Application Flow Diagram]

**Required components**

The following required components include references if configuration is required:
• IBM BPM V8.5.6
  • Responsive Coaches Toolkit, which can be downloaded and added as dependency to the process application that you are using.
  • Mobile-ready client-side human service, which is included in the example Mobile_Enabled_Order_Fulfillment.twx file as the Create Order client-side human service.

• IBM MobileFirst 6.3
  • MobileFirst Studio with Android SDK & Android Development Tools
  • MobileFirst Test/Production Server with an existing server configuration

Security configurations

Some configurations are required so that IBM BPM, MobileFirst, and the mobile device can connect securely to each other. Security certificates must be exchanged.

To import the IBM BPM certificate to the MobileFirst Test/Production server, first export the default certificate from the IBM BPM server to a file. From the administrative console, click Security > SSL certificate and key management > Keystores and certificates > NodeDefaultKeyStore > Personal certificates.

Next, import the certificate into the MobileFirst Test/Production server. Run the following command from the Java installation context that is associated with MobileFirst:

```
path_to_MobileFirst_server_jre\jre\bin\keytool -importcert -file location_of_certificate -keystore path_to_MobileFirst_server_jre\jre\lib\security\cacerts -storepass password
```

End-to-end deployment flow

The flow begins with creating the client-side human service, and then you run it on the MobileFirst Development Server. You also must configure the application for deployment on the MobileFirst Test/Production server and then deploy it.

Authoring the client-side human service

When you author a client-side human service with the goal of running it as a MobileFirst application, ensure that you meet the following requirements:

• The client-side human service must be compatible with use on multiple devices. To enable this compatibility, select Intended for use on multiple devices in the Overview tab of the client-side human service.
• The coaches in the client-side human service must use coach views from the Responsive Coaches Toolkit.

After the requirements are met, the client-side human service can be exported as a MobileFirst project. To export the client-side human service, expose it as a URL. Then, from the Overview tab, export it as a MobileFirst project, as shown in Figure 16.
Exporting generates a .zip file that encapsulates all the code in a form of a MobileFirst project. Also, this project can be imported in the MobileFirst Studio and then deployed to the MobileFirst environments.

**Running the client-side human service on a MobileFirst Development server**

The exported project file can be imported into MobileFirst Studio as an archive project file.

To bring the MobileFirst-generated application into the workspace, open MobileFirst studio and click **File > Import > General: Existing Projects Into Workspace > Select archive file** and then browse to the generated .zip file.

To run it on the local development server, the following configurations can be made in MobileFirst Studio:

- The MobileFirst Development server can be configured by opening it from the Servers tab. By default, the host name is set to localhost. It is recommended that the host name is set to the actual IP address of the machine.
- The *worklight.properties* file configures the application connections. Find the file in the */server/conf* folder of the project. By default, the application is set to connect through the HTTPS protocol to the IBM BPM server. For now, change *bpm.server.protocol* to http and modify *bpm.server.port* to disable security.

To run the application as a preview on the Android simulator, complete the following steps:

1. Right-click the application, inside the */apps* folder of the project, and select **New > MobileFirst Environment**. Create an Android environment, as shown in Figure 17.
Figure 17. Creating an Android environment

Figure 18 shows a view of the application inside the workspace of the MobileFirst Studio:

Figure 18. MobileFirst Studio workspace with the generated application

2. Right-click the **CreateOrderMobile** application and select **Run As > Run on MobileFirst Application Server** to generate the artifacts and deploy them to the local MobileFirst development server.
3. Deploy the adapter that is used to connect to the IBM BPM server. From the adapter/IBM_BPM folder, right-click and select **Run as > Deploy MobileFirst Adapter**.

4. To run the app on a browser simulator, right-click and select **Run as > Preview**. Here, the client-side human service can be tested with multiple screen sizes.

5. To run the app on the Android emulator, select **Run As > Android Application** from the generated Android project. An Android application file (.apk) is created and run. Ensure that at least one virtual device from the Android Device Manager is already set up. See Figure 19 for a comparison of the client-side human service that is running as an application on a mobile browser simulator and on the Android emulator with the Responsive Coaches Toolkit.

**Figure 19. Comparison of an application on a mobile browser simulator and an Android emulator that uses the Responsive Coaches Toolkit**

![Mobile app comparison]

**Configuring the application for deployment to the MobileFirst Test/Production server**

1. To configure the application for the Test/Production Server, make the following changes to the worklight.properties file. Configure a secure connection to the IBM BPM server by completing the following steps:
   - Set `publicWorklightHostname`, `publicWorklightProperties`, and `publicWorklightPort`, to match the MobileFirst Production server.
   - Under **DB Settings**, uncomment the URL for the database to be used, and specify `wl.db.username` and `wl.db.password`.
   - Lastly, modify the `bpm.server.protocol` to `https`, and specify the secure port of the IBM BPM server.

2. Next, set the deployment target of the application. Right-click and select **Run As > Build Settings & Deploy Target** and make the following changes, as shown in Figure 20:
   - Select **Build the application to work with a different MobileFirst server**.
   - Under the Server label, enter the URL and port of the MobileFirst server.
   - Use the same context path as the previous path that you used on the development server.
3. After you clean the bin folders of both projects, right-click and select Run As > Build All Environments. Next, redeploy the IBM BPM adapter, and regenerate the .apk file. The bin folders now contain all the artifacts that are required to deploy the application to the MobileFirst Test/Production server, with the updated properties specified in the worklight.properties file.

The following artifacts are needed when you deploy to the MobileFirst Test/Production Server:

- \textit{APP\_NAME}-all.wlapp
- \textit{APP\_NAME}.war
- IBM\_BPM.adapter
- \textit{APP\_NAME}.apk: (the application file to install on Android devices)

The role of these artifacts is defined in the following section:

The \texttt{wlapp} file makes the connection between the requests that come from the mobile app that is installed on the mobile device as an .apk file and the web UI code that is deployed as a .war file in the IBM MobileFirst server. The .war file connects to IBM BPM back-end system through secure REST API calls included in the IBM\_BPM.adapter. The interaction between these components is elaborated further through the following deployment steps.

**Deploying the application to the MobileFirst Test/Production server**

The artifacts that are generated in IBM MobileFirst Studio are now ready for deployment.

1. Open the Server Configuration Tool from the test/production server and select a server configuration to use (if one does not exist, \textit{create one}).
2. To deploy the .war file to the IBM MobileFirst server, complete the following steps to add a MobileFirst Runtime Environment for the application that you are deploying:
• Under MobileFirst runtime environment, point to the .war file that is generated in MobileFirst Studio and set the context root to the root from Build Settings & Deploy Target in MobileFirst Studio. This context root usually matches the format of the /ProjectName, which in this tutorial is /CreateOrderMobileProject.
• Under Database Additional Settings, change the database name to match the name in the worklight.properties file.

**Figure 21. Example of creating a runtime environment with the Server Configuration Tool for the MobileFirst Test/Production server**

3. Now, deploy the runtime environment, and restart the server, so the runtime deployment changes are captured.

The .wlapp and .adapter artifacts must be deployed through the IBM MobileFirst Platform Operations Console, as shown in Figure 22. Go to the console's URL, http://host_name:port/worklightconsole, and select the deployed runtime environment that you created. Deploy the artifacts APP_NAME-all.wlapp and IBM_BPM.adapter. After both artifacts are deployed, selecting **Preview as common resource** allows running the mobile application from the browser.
The .wlapp and .adapter artifacts, deployed through the MobileFirst Platform Operations Console

The .apk file also can be uploaded to the application center of the production/test server, http://host_name:port/appcenterconsole/. A mobile device that runs the application must be able to connect to the MobileFirst server for the application to run. After the application is uploaded, access the application center URL from a mobile device, then download and install the app. You are now ready to run the client-side human service as a mobile application.

Additional information about building the solution

Consider the following information about running the client-side human service on iOS and deployed on the IBM BPM Process Server.

Running the client-side human service on iOS

Generating an application from the client-side human service and running it on iOS is a similar process to that of running it on Android. The key difference is how the application file is created. Instead of MobileFirst Studio generating the .apk file for Android, Xcode on Mac is required to generate the .ipa file. Also, the .ipa file must be signed by using Apple provisioning profiles and security certificates. After the application is signed, it can be installed on iOS iPhones and iPads.

Running the client-side human service that is deployed to IBM BPM Process Server

There are some minor adjustments to make when you run a client-side human service from a process application that is deployed to the Process Server. The project file that is exported from
the client-side human service must be exported from the same snapshot that is deployed to the Process Server. By default, the exported project file is configured to connect to the Process Center. Modify the worklight.properties file with the URL of the Process Server instead.

Now you understand the setup and development information that is needed to build the mobile application that interacts with IBM BPM, and you can deploy and run it securely from mobile devices.

**Conclusion**

This tutorial showed how to integrate IBM MobileFirst and IBM BPM V8.5.6 by starting a process from a mobile device that uses a generated client-side human service application.

By combining the powerful mobile platform that is provided by IBM MobileFirst with the convenient way of generating mobile applications from client-side human services, IBM BPM developers can quickly deliver mobile apps that start IBM BPM processes.

**Acknowledgments**

The authors would like to thank John Green and John Mourra from the IBM BPM management team for their leadership and support.
## Downloadable resources

<table>
<thead>
<tr>
<th>Description</th>
<th>Name</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code sample</td>
<td>Mobile_Enabled_Order_Fulfillment.twx</td>
<td>5995KB</td>
</tr>
</tbody>
</table>
Related topics

- Configuration of MobileFirst applications on the server (IBM Knowledge Center)
- IBM Business Process Manager documentation on IBM Knowledge Center
- Setting up your Android development environment (IBM MobileFirst Platform Developer Center)
- Integrating IBM Business Process Manager with a hybrid MobileFirst application (developerWorks)

© Copyright IBM Corporation 2015

Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)