# Java HTTP Adapter

### **Overview**

This tutorial is a continuation of Java Adapter (../../server-side-development/java-adapter/) and assumes previous knowledge of the concepts described there.

Java adapters provide free reign over connectivity to your backend. It is therefore your responsibility to ensure best practices regarding performance and other implementation details.

This tutorial shows an example of a Java adapter that connects to an RSS feed by using a Java HttpClient.

#### Topics:

- RSSAdapterApplication
- RSSAdapterResource
- Results

## **RSSAdapterApplication**

RSSAdapterApplication extends MFPJAXRSApplication and is a good place to trigger any initialization required by your application.

```
@Override
protected void init() throws Exception {
   RSSAdapterResource.init();
   logger.info("Adapter initialized!");
}
```

### **RSSAdapterResource**

```
@Path("/")
public class RSSAdapterResource {
}
```

RSSAdapterResource is where we handle the requests to your adapter.

```
@Path("/") means that the resources will be available at the URL
http(s)://host:port/ProjectName/adapters/AdapterName/.
```

### **HTTP Client**

```
private static CloseableHttpClient client;
private static HttpHost host;
public static void init() {
    client = HttpClients.createDefault();
    host = new HttpHost("developer.ibm.com");
}
```

Because every request to your resource will create a new instance of RSSAdapterResource, it is important to reuse objects that may impact performance. In this example we made the Http client a static object and initialized it in a static init() method, which gets called by the init() of RSSAdapterApplication as described above.

#### **Procedure resource**

```
@GET
@Produces("application/json")
public void get(@Context HttpServletResponse response, @QueryParam("tag") String
tag) throws ClientProtocolException, IOException, IllegalStateException, SAXExcep
tion {
    if(tag!=null && !tag.isEmpty()){
        execute(new HttpGet("/mobilefirstplatform/tag/"+ tag +"/feed"), response);
    } else{
        execute(new HttpGet("/mobilefirstplatform/feed"), response);
    }
}
```

Our adapter exposes just one resource URL which allows to retrieve the RSS feed from the backend service.

- @GET means that this procedure only responds to HTTP GET requests.
- @Produces("application/json") specifies the Content Type of the response to send back. We chose to send the response as a JSON object to make it easier on the client-side.
- @Context HttpServletResponse response will be used to write to the response output stream. This enables us more granularity than returning a simple string.
- @QueryParam("tag") String tag enables the procedure to receive a parameter. The choice of QueryParam means the parameter is to be passed in the query (/RSSAdapter/? tag=MobileFirst\_Platform). Other options include @PathParam, @HeaderParam, @CookieParam, @FormParam, etc.
- throws ClientProtocolException, ... means we are forwarding any exception back to the client. The client code is responsible for handling potential exceptions which will be received as HTTP 500 errors. Another solution (more likely in production code) is to handle exceptions in your server Java code and decide what to send to the client based on the exact error.
- execute(new HttpGet("/mobilefirstplatform/feed"), response). The actual HTTP request to the backend service is handled by another method defined later.

Depending if you pass a tag parameter, execute will retrieve a different build a different path and retrieve a different RSS file.

### execute()

```
public void execute(HttpUriRequest req, HttpServletResponse resultResponse) th
rows ClientProtocolException, IOException,
  IllegalStateException, SAXException {
    HttpResponse RSSResponse = client.execute(host, reg);
    ServletOutputStream os = resultResponse.getOutputStream();
    if (RSSResponse.getStatusLine().getStatusCode() == HttpStatus.SC OK){
        resultResponse.addHeader("Content-Type", "application/json");
        String json = XML.toJson(RSSResponse.getEntity().getContent());
        os.write(json.getBytes(Charset.forName("UTF-8")));
    } else {
        resultResponse.setStatus(RSSResponse.getStatusLine().getStatusCode());
        RSSResponse.getEntity().getContent().close();
        os.write(RSSResponse.getStatusLine().getReasonPhrase().getBytes());
    }
    os.flush();
    os.close();
}
```

- HttpResponse RSSResponse = client.execute(host, req). We use our static HTTP client to execute the HTTP request and store the response.
- ServletOutputStream os = resultResponse.getOutputStream(). This is the output stream to write a response to the client.
- [resultResponse.addHeader("Content-Type", "application/json")]. As mentioned before, we chose to send the response as JSON.
- String json = XML.toJson(RSSResponse.getEntity().getContent()). We used org.apache.wink.json4j.utils.XML to convert the XML RSS to a JSON string.
- [os.write(json.getBytes(Charset.forName("UTF-8")))] the resulting JSON string is written to the output stream.

The output stream is then flushed and closed.

If RSSResponse is not 200 OK, we write the status code and reason in the response instead.

### **Results**

Use the testing techniques described in Java Adapter (../#testing) to test your work.

The adapter should return the RSS feed converted to JSON.

```
Aumai III
              ],
              "commentRss": "https:\/\/developer.ibm.com\/mobilefirstplatform\/2
015\/09\/01\/integrating-mqa-into-xamarin-android-app\/feed\/",
              "comments": [
                  "https:\/\/developer.ibm.com\/mobilefirstplatform\/2015\/09\/01
\/integrating-mga-into-xamarin-android-app\/#comments",
              ],
              "creator": "Vidyasagar MSC",
              "description": "The post <a rel=\"nofollow\" href=\"https:\/\/d
eveloper.ibm.com\/mobilefirstplatform\/2015\/09\/01\/integrating-mqa-into-xamarin
-android-app\">Integrating MQA into Xamarin.Android app\\"> appeared first on <
a rel=\"nofollow\" href=\"https:\/\/developer.ibm.com\/mobilefirstplatform\">IBM
MobileFirst Platform<\/a>.<\/p>",
              "encoded": "It all started when I received an email seeking he
lp on using MQA or to be more precise integrating MQA into Xamarin based android
app. Before jumping into addressing the problem, let's define MQA.<\/p>
\n<h4>What is MQA?<\/h4>\nMQA stands for & amp; #8220; Mobile Quality Assurance&a
mp;#8221; and is part of the IBM MobileFirst Platform.<\/p>\n<br/>blockquote><em><s
pan style=\"line-height: 1.5\">IBM MQA provides line of business professionals and
development teams with insightful and streamlined quality feedback and metrics fr
om both pre-production and production, enabling them to prioritize and take action
to support a dynamic mobile app strategy.<\/span><\/em><\/bookquote>\nTh
e Features of MQA are<\/p>\n<div style=\"width: 1058px\" class=\"wp-caption alignc
enter\"><a href=\"http:\/\/vidyasagarmsc.com\/wp-content\/uploads\/2015\/09\/MQA1
.png\"><img class=\"size-full wp-image-65\" src=\"http:\/\/vidyasagarmsc.com\/wp-c
ontent\/uploads\/2015\/09\/MQA1.png\" alt=\"Features of Mobile Quality Assurance.
\" width=\"1048\" height=\"350\" \/><\/a>Features of
Mobile Quality Assurance.<\/p><\/div>\n<em><strong>Note<\/strong><\/em>: To un
derstand more about MQA, visit <a href=\"http:\/\/www-03.ibm.com\/software\/prod
ucts\/en\/ibm-mobilefirst-platform-quality-assurance\">IBM Mobile Quality Assuran
ce<\/a><\/p>\nSo, by now we should be good with the first part of our blog titl
e that is MQA. So, the next question is <\/p>\n<h4>What is Xamarin.Android? < <math>\/h4>\n
Xamarin is a platform to create nativeÄ iOS, Android, Mac and Windows apps in C
#. Xamarin.Android allows us to create native Android applications using the sam
e UI controls we would in Java, except with the flexibility and elegance of a mode
rn language (C#).<\/p>\nAs we are good with the definitions, let&amp;#8217;s ad
dress the problem.<\/p>\n<strong>What&amp;#8217;s the problem in integrating MQ
A into Xamarin Android app?<\/strong><\/p>At the time of this blog post, the
available MQA SDKs are iOS native SDK, Android native SDK and Javascript  SDK.<\/
p>\nSo, we have to find a workaround to address this use-case. The initial step
is to download the Android MQA SDK and see what& #8217; s provided. you can down
```

load it from <a href=\"http:\/\/www-01.ibm.com\/support\/knowledgecenter\/#!\/SSJM L5\_6.0.0\/com.ibm.mqa.uau.saas.doc\/topics\/c\_AndroidSDKsForDownload.html\">here< \/a>. Once successfully downloaded and unzipped, we should see a jar file namely <strong><em>MQA-Android-library-&amp;lt;version number&amp;gt;.jar<\/em> <\/str ong>under lib folder<strong>.<\/strong><\/p>\n<div style=\"width: 634px\" class=\"wp-caption aligncenter\"><a href=\"http:\/\/vidyasagarmsc.com\/wp-content\/upload s\/2015\/09\/MQA2.png\"><img class=\"size-full wp-image-70\" src=\"http:\/\/vidyas agarmsc.com\/wp-content\/uploads\/2015\/09\/MQA2.png\" alt=\"MQA Android SDK \" wi dth=\"624\" height=\"440\" \/><\/a>MQA Android SDK \" wi dth=\"624\" height=\"440\" \/><\/a>MQA Android SDK<\\/p><\/div>\n>We have <strong>Xamarin is C# based, What can we do with this jar file?<\/p>\n>We have <strong>Xamarin bindings<\/strong> to our rescue, which helps using in consuming .JARs from C#.<\/p>\n><strong><em>Note<\/em>:<\/strong> Steps to consume MQA Android JAR in a Xamarin.Android app is mentioned <a href=\"https:\/\/developer.xamarin.com\/guides\/android\/advanced\_topics\/java\_integration\_overview\/b inding a class of the consume of the class of the consume of the consumery of the

inging a java library (.jar)\/\">nere<\/a><\/p>\n<giv style=\"wigth: 25/px\" clas s=\"wp-caption aligncenter\"><a href=\"http:\/\/vidyasagarmsc.com\/wp-content\/up  $loads\/2015\/09\/MQA31.png\"><img class=\"wp-image-72 size-full\" src=\"http:\//$  $vidy as a garmsc.com \wp-content \uploads \2015 \09 \MQA31.png \ alt=\"" width=\"24" \end{tikzpicture}$ 7\" height=\"303\" \/><\/a>Xamarin binding project w ith MQA Android .JAR file<\/p><\/div>\nThe files of our interest here are <stro ng>MQA-Android-library-2.7.4.jar<\/strong> (Version number may vary) and <strong>  $Metadata.xml.<\/p>\n
MQA-Android-library-2.7.4.jar file will h$ ave all the MQA related classes and methods required for us to start an Android MQ A session.<\/li>\nMetadata.xml- <em>Allows changes to be made to the final API , such as changing the namespace of the generated binding. $<\/em><\/li>\n<\/ul>\n<p$ >Based on the errors thrown while building the project, Metadata.xml in my case lo oks like this<\/p>\n&amp;lt;metad ata>\n <!--\n This sample removes the class: android.support.v4.c ontent.AsyncTaskLoader.LoadTask:\n <remove-node path=&amp;quot;\/api\/pac kage[@name='android.support.v4.content']\/class[@name='AsyncTaskLoader.LoadTask'] " \/>\n \n This sample removes the method: android.support.v4.c ontent.CursorLoader.loadInBackground:\n <remove-node path=&amp;quot;\/api \/package[@name='android.support.v4.content']\/class[@name='CursorLoader']\/metho d[@name='loadInBackground']" \/>\n -->\n\n <remov e-node path="\/api\/package[@name='ext.com.google.inject.spi']\/class[@n  $ame='InjectionPoint.Factory.1'] \& amp; quot; \\ \\ (\& amp; gt; \\ n \& amp; lt; remove-node path=\& lt; remove-node p$ amp;quot;\/api\/package[@name='ext.com.google.inject.spi']\/class[@name='Injectio nPoint.Factory.2']"\/>\n <remove-node path=&amp;quot;\/ap i\/package[@name='com.applause.android.log']\/interface[@name='LoggerInterface']& amp;quot;\/>\n <remove-node path=&amp;quot;\/api\/package[@name='e xt.com.google.inject.internal']"\/&qt;\n <remove-node path=& ;\n <remove-node path=&amp;quot;\/api\/package[@name='com.applause.android .util']\/class[@name='AbstractRequest']"\/>\n <remove-nod e path="\/api\/package[@name='ext.com.google.inject.spi']\/class[@name=' Elements.RecordingBinder']\/method[@name='bind' and count(parameter)=1 and paramet er[1][@type='ext.com.google.inject.Key']]"\/>\n\n<attr pat h="\/api\/package[@name='com.applause.android.messages']\/class[@name='M essage']\/field[@name='message']" name="managedName"&a mp;gt;Message1<\/attr&amp;gt;\n&amp;lt;attr path=&amp;quot;\/api\/package[ @name='com.applause.android.log']" name="managedName"&  $amp;gt;log\<\\/attr\&amp;gt;\\n\&amp;lt;\\/metadata\&amp;gt;\\n\\n<\\/pre>\\nOnce$ all the errors are fixed and your binding project builds successfully, add a new Xamarin Android project (if you haven& #8217; t added yet). Now, add MQA bindin g project reference in our Xamarin android app. <em><strong>Note:<\/strong><\/em> Both your binding project and Xamarin. Android project should be of same <strong>t arget framework. <\/strong>You can verify this by right clicking on your project -> Options -> General.<\/p>\n<div id=\"attachment 83\" style=\"widt h: 270px\" class=\"wp-caption aligncenter\"><a href=\"http:\/\/vidyasagarmsc.com\  $\mbox{wp-content}\poline{100} \mbox{0015}\poline{100} \mbox{0015}\poline{100$ c=\"http:\/\/vidyasagarmsc.com\/wp-content\/uploads\/2015\/09\/MQA5.png\" alt=\"Xamarin Android project with added reference to MQA\" width=\"260\" height=\"652\" \/><\/a>Xamarin Android project with added reference to MQA<\/p><\/div>\nNow, let&amp;#8217;s start MQA android session in our Coun t.Android app. Before doing this, we should create a MQA service on IBM Bluemix. You can follow the instructions mentioned atA <a href=\"https:\/\/www.ng.bluemix.n et\/docs\/#services\/MobileQualityAssurance\/index.html#MobileQualityAssurance\"> Getting started with Mobile Quality Assurance- Bluemix<\/a>Â or watch this video. <\/p>\n<span class='embed-youtube' style='text-align:center; display: block;'> <iframe class='youtube-player' type='text\/html' width='980' height='582' src='ht</pre> tps:\/\/www.youtube.com\/embed\/zHRfGatcKPM?version=3&rel=1&fs=

```
1&showsearch=0&showinfo=1&iv load policy=1&wm
ode=transparent' frameborder='0' allowfullscreen='true'><\/iframe><\/span><\/p>\n
Starting a\hat{A} < span class = \"ph\">< span id = \"d6087e24\" class = \"ph\">Mobile Qualit
y Assurance<\/span><\/span>Â session with the Android SDK entails three steps. Fi
rst, build a configuration to define how <span class=\"ph\"><span id=\"d6087e24-d
6083e11a1310\" class=\"ph\">Mobile Quality Assurance<\/span><\/span>Â works with
your app. Second, start the session itself. Third, add tracking to your activitie
s. Open <strong>MainActivity.cs<\/strong> file (Android Project) and paste the cod
e provided below<\/p>\nre class=\"brush: csharp; title: ; notranslate\">using S
ystem;\n\nusing Android.App;\nusing Android.Content;\nusing Android.Runtime;\nusi
ng Android.Views;\nusing Android.Widget;\nusing Android.OS;\n\/\/MQA references\n
using Com.Ibm.Mqa.Config;\nusing Com.Ibm.Mqa;\n\nnamespace Count.Android\n{\n\t
[Activity (Label = "Count.Android", MainLauncher = true, Icon =
"@drawable\/icon")]\n\tpublic class MainActivity : Activity\n\t
{\n\t count = 1;\n\t //Use your own generated APP KEY\n\t string AP}
P KEY="1g59b7d884f9fdf5426162e5cb1f87a700648bce4fg0g1g379e0d3a"
;\n\t\tprotected override void OnCreate (Bundle bundle)\n\tt\th\tt\tbase.OnCreat
e (bundle);\n\t\t\Android session configuration \n\t\Configuration co
nfiguration = new Configuration.Builder(this)\n\t\t\t.WithAPIKey(APP KEY) \/\/P
rovides the quality assurance application APP KEY\n\t\t.WithMode(MQA.Mode.Qa)
\/\/Selects the quality assurance application mode\n\t\t\t.WithReportOnShakeEna
bled(true) \/\/Enables shake report trigger\n\t\t\t.WithDefaultUser("d
efault user@email.com") \/\/Sets a default user and user selection\n\t\t
\t\t.Build();\n\n\t\t\\/\/Starting MQA Android Session\n\t\t\tMQA.StartNewSession
(this, configuration);\n\t\t\\/\/ Set our view from the "main"
layout resource\n\t\t\SetContentView (Resource.Layout.Main);\n\n\t\t\/\/ Get ou
r button from the layout resource,\n\t\t\/\/ and attach an event to it\n\t\t\B
utton button = FindViewById<Button&amp;gt; (Resource.Id.myButton);\n\t\t\t
\n\t\t\tbutton.Click += delegate {\n\t\t\tbutton.Text = string.Format (&quot
;\{0\} clicks!", count++);\n\t\t\t};\n\t\t}\n\h\n\n\n\n\n<\/pre>\nNow,
MQA is integrated into Xamarin. Android app and we are good to go.<\p>\nWhat we
have implemented above is just a drop in the Ocean of MQA, to know more about MQA
and its features – Visit <a href=\"http:\/\/www-01.ibm.com\/support\/kn
owledgecenter\/?lang=en#!\/SSJML5 6.0.0\/com.ibm.mqa.uau.saas.doc\/mqa600saas wel
come.html\" target=\"_blank\">MQA Knowledge Centre<\/a><\/p>\nHappy Coding !!!
<\/p>\nThe post <a rel=\"nofollow\" href=\"https:\/\/developer.ibm.com\/mobile
firstplatform\/2015\/09\/01\/integrating-mqa-into-xamarin-android-app\/\">Integra
ting MQA into Xamarin.Android app<\/a> appeared first on <a rel=\"nofollow\" href=
\"https:\/\/developer.ibm.com\/mobilefirstplatform\">IBM MobileFirst Platform<\/a
>.<\/p>",
              "guid": {
                 "content": "https:\/\/developer.ibm.com\/mobilefirstplatform\/?
p=16964",
                 "isPermaLink": "false"
              "link": "https:\/\/developer.ibm.com\/mobilefirstplatform\/2015\/0
9\/01\/integrating-mqa-into-xamarin-android-app\/",
              "pubDate": "Tue, 01 Sep 2015 20:27:07 +0000",
              "title": "Integrating MQA into Xamarin.Android app"
           },
           {
              "category": [
                 "Uncategorized",
                 "MobileFirst Platform"
              "commentRss": "https:\/\/developer.ibm.com\/mobilefirstplatform\/2
015\/08\/19\/try-on-bluemix-and-buy-mfp\/feed\/",
```

"description": "The post <a rel=\"nofollow\" href=\"https:\/\/d eveloper.ibm.com\/mobilefirstplatform\/2015\/08\/19\/try-on-bluemix-and-buy-mfp\/\
">Try on Bluemix and migrate to on-prem MobileFirst Platform<\/a> appeared first o n <a rel=\"nofollow\" href=\"https:\/\/developer.ibm.com\/mobilefirstplatform\">I
BM MobileFirst Platform<\/a>.<\/p>",

"encoded": "Contributed By : Chethan Kumar SN (chethankumar.sn@ in.ibm.com) and Vittal Pai (vittalpai@in.ibm.com)<\/p>\nWith the release of Mob ileFirst Platform v7.1, one can now migrate any existing iOS app built for MobileS ervices on Bluemix to MobileFirst Platform with just a handful of simple steps.<\ /p>\nTo elucidate the process, lets look at how to migrate a simple Bluemix iOS app.<\/p>\nTo migrate an existing iOS app built for MobileServices on Bluemix t o run on MobileFirst Platform, follow the steps below.<\/p>\n\n<a href=\" #migrateexisting\">Existing Bluemix Server Application<\/a><\/li>\n<a href=\"</pre> #migrateblu\">Existing Bluemix Client Application<\/a><\/li>\n<a href=\"#conf</pre> igureclient\">Migration of Client Application<\/a><\/li>\n<a href=\"#migratem" fp\">Migration of JAX-RS Application to JAVA Adapter<\/a><\/li>\n<a href=\"#c onfigoauth\">Configuring Custom-OAuth<\/a><\/li>\n<a href=\"#configurepush\"> Configuring Push Capability<\/a><\/li>\n<a href=\"#sample\">Sample and Source Code<\/a><\/li>\n<\/ul>\n<h2 id=\"migrateexisting\">Existing Bluemix Server Appli cation<\/h2>\nThe Bluemix app has the following functionality:<\/p>\n> On the client side, the application stores a list of items and provides a way to add more items to the list. Each item can able to store Name, Store, Price and ima ge of the product. The App& #8217; s are protected by Custom Authenticator via A MA security service provided by bluemix.<\/li>\nli>On the server side, the App co ntains a JAX-RS class to store and manipulate the data. It also contains the serve r side AMA security implementation.<\/li>\n<\/ul>\nOn BlueMix we have applicati on with the following configuration:<\/p>\n\nLiberty Runtime : which used to run JAX-RS application on Bluemix<\/li>\nAdvance Mobile Access service : w hich gives mobile application security and monitoring functionality<\/li>\nPu sh Service for iOS 8 : which provides the capability to use iOS Push features<\/li >\n<\/ul>\n<h3> Liberty Runtime <\/h3>\n\nLiberty contains two projects wi th JAX-RS service (i.e Custom-oauth-java for Custom Authentication and Localstore Adapter for storing items). The service include the protected resource and the cu stom identity provider code. The liberty server is configured with TAI.\n<\/li> li>Trust Association Interface (TAI) is a service provider API that enables the i ntegration of third-party security services with a Liberty profile server. For mo re info on TAI : <a href=\"http:\/\/www-01.ibm.com\/support\/knowledgecenter\/was beta liberty\/com.ibm.websphere.wlp.nd.multiplatform.doc\/ae\/twlp dev custom tai .html\" target=\" blank\">click here<\/a>\n<\/li>The custom identity provid er authenticates a user by sending challenges to the client. However, custom iden tity providers do not communicate directly with clients. They send challenges and receive responses to the challenges by means of the Advanced Mobile Access service . When a custom identity provider successfully authenticates the user, it provides the user identity information to Advanced Mobile Access. For more information on c ustom authentication refer bluemix documentation : <a href=\"https:\/\/www.ng.blu emix.net\/docs\/services\/mobileaccess\/security\/id provs\/index-gentopic2.html# custom id prov\" target=\" blank\">click here<\/a>\nThe custom identity provid er code is defined by two http API:<\/p>\nre class=\"brush: plain; title: ; not ranslate\">\/startAutorization<\/pre>\n and\n<pre class=\"brush: plain; title: ; notranslate\">\/handleChallengeAnswer<\/pre>\n ; notranslate\"> @POST\n\t@Consumes ("application\/ison")\n\t@Pa

```
th(\& quot; \/\{tenantId\}\/customAuthRealm\_3\/startAuthorization\& quot; )\/n\t@Partial (\& quot; )\/n\makebox{\colored}
roduces(MediaType.APPLICATION JSON)\n\tpublic JSONObject startAuthorization(Strin
g payload,\n\t\
\t@PathParam("realmName") String realmName) throws Exception {\
n\t\tJSONObject returnJson = (JSONObject) JSON.parse(CHALLENGE JSON);\n\t\treturn
returnJson;\n\t}\n\t@POST\n\t@Consumes ("application\/json"
)\n\t@Path("\/{tenantId}\/customAuthRealm 3\/handleChallengeAnswer&q
uot;)\n\t@Produces(MediaType.APPLICATION JSON)\n\tpublic JSONObject handleChlleng
eAnswer(String payload,\n\t\t\@PathParam("tenantId") String dev
iceId,\n\t\t\@PathParam("realmName") String realmName) throws
Exception \{ \n \t \t \SONObject userStoreJson = (JSONObject) JSON.parse(USER ST \)
ORE_JSON);\n\t\tJSONObject failedResponseJson = (JSONObject) JSON.parse(FAILURE_J
SON); h\t t f(payload == null || payload.isEmpty()) {h\t t return failedRe}
sponseJson; \n\t\t\}\n\t\tJSONObject payloadJson = (JSONObject) JSON.parse(payload)
;\n\t\tJSONObject challengeAnswer = (JSONObject) payloadJson.get("challe
"userName");\n\t\
("password");\n\t\t\n\t\tif (userName == null || userName.isEmpt
y() || password == null || password.isEmpty()) {\n\t\treturn failedResponseJson;
\n\t\t}\n\t\tif (userStoreJson.containsKey(userName)) {\t\n\t\t\tJS0N0bject
userInfoJson = (JSONObject) userStoreJson.get(userName);\n\t\t\tString userPasswo
rd = (String) userInfoJson.get("password");\n\t\tString userD
isplayName = (String) userInfoJson.get("displayName");\n\t\t\
n\t \in \{n \in \mathbb{N} \mid n \in \mathbb{N} \}
JSONObject(); \n\t\t\t\JSONObject\ userIdentityJson = new\ JSONObject(); \n\t\t\t\t
serIdentityJson.put("userName", userName);\n\t\t\tuserIdentit
yJson.put("displayName", userDisplayName);\n\t\t\t\t\t\t\t\t\t
eturnJson.put("status", "success");\n\t\t\tr
eturnJson.put("userIdentity", userIdentityJson);\n\t\t\t\treturn
re>\nThe Localstore adapter contains few http API's to perform some
basic operations like Add, Update, Create and Delete in client application.<\/p>\
n @GET\n\t@Path("\/getA
llItems")\n\tpublic String getAllItems() throws IOException{\n\t\tinit()
;\n\t\tJsonArray jsonArray = new JsonArray();\n\t\tfor(Object key : props.keySet(
)){\n\t\tjsonArray.add(parser.parse(props.getProperty((String) key)).getAsJson0
bject());\n\t\treturn jsonArray.toString();\n\t}\n\n\t@PUT\n\t@Path(&q
uot;\/addItem")\n\tpublic void addItem(String itemJson) \n\t\t\throws IO
Exception, URISyntaxException{\n\t\ttry{\n\t\tinit();\n\t\tint newKey = props
.keySet().size()+1;\n\t\t\tprops.put(String.valueOf(newKey), itemJson);\n\t\t\tUR
L url = this.getClass().getClassLoader().getResource("data.properties&am
p;quot;); \n\t\tFile file = new File(url.toURI().getPath());\n\t\tFileOutputS
tream foStream = new FileOutputStream(file);\n\t\t\tprops.store(foStream, &quo
t; saving new item& quot;); \n\t\t
addAllItems")\n\tpublic String addAllItems(String itemsJson) \n\t\t\tthr
ows URISyntaxException, IOException{\n\t\ttry{\n\t\tinit();\n\t\t\clearAllData
();\n\t\tJsonArray jsonArr = parser.parse(itemsJson).getAsJsonArray();\n\t\t\tf
or(int i=0; i\&amp; amp; t; jsonArr.size(); i++){\n\t\t\props.put(String.valueOf(i+))}
1), jsonArr.get(i).toString());\n\t\t\t\tURL url = this.getClass().getClass
Loader().getResource("data.properties"); \n\t\t\tFile f
ile = new File(url.toURI().getPath());\n\t\tFileOutputStream foStream = new File
OutputStream(file);\n\t\tprops.store(foStream, "saving new item&amp
;amp;quot;);\n\t\tfoStream.close();\n\t\treturn "true&quo
t;;\n\t\t}catch(IOException ioe){\n\t\t\tioe.printStackTrace();\n\t\t}\n\t\tretur
```

n &.amn.quot.falca&.amn.quot..\n\t\\n\n\+@DFLFTF\n\+@Path/&.amn.quot.\/claarAll&.amn.

```
וו ממוויף, קעוטנ, ומנג פעמוויף, קעוטנ,, אוואנגן אוואנגעפורב בוב אוואנער פנווא עמוויף, קעוטנ, או כנפפו אנגעפוויף,
quot;)\n\tpublic String clearAllData() \n\t\tthrows MissingConfigurationOptionE
xception, URISyntaxException, IOException{\n\t\tinit();\n\t\tprops.clear();\n\
t\t\tSystem.out.println("Size : "+props.size());\n\t\tURL url
= this.getClass().getClassLoader().getResource("data.properties&quot
;); \n\t\tFile file = new File(url.toURI().getPath());\n\t\tFileOutputStream
foStream = new FileOutputStream(file);\n\t\t\tprops.store(foStream, "cle
aring all data");\n\t\t\tfoStream.close();\n\t\treturn "clear
of server directory server\/usr\/extensions<br \/>\nTAI Extension Link : Download
the extension.zip from <a href=\"https:\/\/hub.jazz.net\/project\/chethan\/parksto
re-bluemix-server\/overview\" target=\" blank\">here<\/a>\n<\/li>\nAdd TAI Se
curity constraint in web.xml file for both the projects.\npre class=\"brush: xml;
title: ; notranslate\"><security-constraint&amp;gt;\n
                                                     \t<web-reso
urce-collection>\n
                          \t
                              & lt; web-resource-name& gt; LocalstoreAp
plication<\/web-resource-name&amp;gt;\n
                                           \t
                                                & lt; url-pattern&
qt;\/apps\/*<\/url-pattern&amp;qt;\n
                                      \t& lt; \/web-resource-collection
             \t<auth-constraint&amp;gt;\n
                                                    & lt; role-name&a
mp;gt;TAIUserRole<\/role-name&amp;gt;\n
                                           \t<\/auth-constraint&a
mp;gt;\n<\/security-constraint&amp;gt;\n&amp;lt;security-role id=&amp;quot;
SecurityRole TAIUserRole" &qt;\n
                                           & lt; role-name& gt; TAIU
serRole<\/role-name&amp;gt;\n&amp;lt;\/security-role&amp;gt;<\/pre>\n<\/li
>\nAdd OAuthTai feature in server.xml\n
otranslate\"><feature&amp;gt;usr:0AuthTai-1.0&amp;lt;\/feature&amp;gt;<\/p
re>\n<\/li>\nProtect the Url&amp;#8217;s using TAI by adding following code
in server.xml\n <usr OAut
hTAI id="myOAuthTAI" realmName="imfRealm"&amp
;qt;\n\t\t<securityConstraint httpMethods=&amp;quot;GET, POST&amp;quot; sec
uredURLs="\/LocalstoreAdapter\/*"\/>\n\t\t<securi
tyConstraint httpMethods="GET, POST" securedURLs="\/cu
stom-oauth-java\/*"\/>\n\t<\/usr OAuthTAI&amp;gt; \n\n
<webApplication id=&amp;quot;custom-oauth-java&amp;quot; location=&amp;quot
;custom-oauth-java.war" name="custom-oauth-java"&g
       <application-bnd&amp;gt;\n\t\t&amp;lt;security-role name=&amp;quo
t;TAIUserRole"&qt;\n\t\t\<special-subject type=&amp;quot;ALL
AUTHENTICATED USERS"\/>\n\t\t<\/security-role&amp;gt;\n\t
<\/application-bnd&amp;gt; \n\t&amp;lt;\/webApplication&amp;gt;
mp;lt;webApplication id="LocalstoreAdapter" location="L
ocalstoreAdapter.war" name="LocalstoreAdapter">\
       <application-bnd&amp;gt;\n\t\t&amp;lt;security-role name=&amp;quo
n
t;TAIUserRole">\n\t\t\t<special-subject type=&amp;quot;ALL
_AUTHENTICATED_USERS"\/>\n\t\t<\/security-role&amp;gt;\n\t
<\/application-bnd&amp;gt; \n\t&amp;lt;\/webApplication&amp;gt;<\/pre>\n<\
/li>\nSpecify the IMF Auth Url inside Server.env file in liberty.\n<pre class
=\"brush: xml; title: ; notranslate\">imfServiceUrl=https:\/\/imf-authserver.ng.b
luemix.net\/imf-authserver<\/pre>\n<\/li>\Create a server package which con
tains above two applications using following command.\n<pre class=\"brush: plain;
title: ; notranslate\">.\/server package ${server name} --include=usr<\/pre>\n<\/
li>\nPush the newly created server package to bluemix using following comman
d.\ncf push ${app name} -p ${pa
th to server package zip}<\/pre>\n<\/li>\n<\/ul>\n<h3>Advance Mobile Access servi
ce<\/h3>\n\nBind the pushed application to Advance Mobile Access Service
.\n<a href=\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/upl</pre>
oads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-17-at-3.28.04-pm.png\"><img src=\"h
ttps:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2
015\/07\/Screen-Shot-2015-07-17-at-3.28.04-pm-1024x346.png\" alt=\"Advance Mobile
Access\" width=\"980\" height=\"331\" class=\"alignnone size-large wp-image-14882
```

ore info refer documentation : <a href=\"https:\/\/www.ng.bluemix.net\/docs\/servi  $ces\/mobileaccess\/index.html\" target=\"\_blank\">click here<\/a>\n<a href=\"h$  $ttps: \/\developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2$ 015\/07\/screen-Shot-2015-07-17-at-3.42.32-pm.png\"><img src=\"https:\/\/developer .ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/ $32\/2015\/07\/Screen-S$ hot-2015-07-17-at-3.42.32-pm.png\" alt=\"AMA Client Registration\" width=\"935\" h eight=\"452\" class=\"alignnone size-full wp-image-14883\" \/><\/a>\n<\/li> AMA provides Facebook, Google, or a custom identity provider to authenticate acces s to protected resources. Add Custom identity provider feature as it can be migrat ed to MFPF and specify the corresponding jax-rs custom authentication application url and realm name.<br \/>\n<a href=\"https:\/\/developer.ibm.com\/mobilefirstplat</pre> pm.png\"><img src=\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/</pre>  $uploads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-17-at-4.03.21-pm.png\" alt=\"Cus"$ tom Auth AMA\" width=\"955\" height=\"375\" class=\"alignnone size-full wp-image-1 4890\" \/><\/a>\n<\/li> Add the following code inside didFinishLaunchingWith Options function in AppDelegate of client application which will register the real m and initialize connection with Bluemix Application.\n<pre class=\"brush: plain; title: ; notranslate\"> IMFClient.sharedInstance().registerAuthenticationDelegate (customAuthDelegate, forRealm: "customAuthRealm 3")\nIMFClient.s haredInstance().initializeWithBackendRoute("https:\/\/parkstore.mybluemi x.net", backendGUID: "5e3ad88d-dd48-469d-b46f-2c4ad66b5345& quot;)<\/pre>\n<\/li> The following is the sample code to invoke the Rest ur l's in client application.\n late\">var request: IMFResourceRequest = IMFResourceRequest(path: "https: \/\/parkstore.mybluemix.net\/LocalstoreAdapter\/apps\/5e3ad88d-dd48-469d-b46f-2c4a d66b5345\/localstore\/getAllItems", method: "GET")\n request.sendWithCompletionHandler { (wlResponse:IMFResponse!, err:NSError!) -&amp ;gt; Void in<\/pre>\n<\/li>\n<\/ul>\n<h3>Push Service for iOS 8<\/h3>\n\n Bind the application with Push Service for iOS 8<br/>
| No<a href=\"https:\/\/deve  $loper.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2015\/07\/Scr$ een-Shot-2015-07-17-at-4.07.01-pm.png\"><img src=\"https:\/\/developer.ibm.com\/mo bilefirstplatform\/wp-content\/uploads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-1 7-at-4.07.01-pm-1024x367.png\" alt=\"Push AMA\" width=\"980\" height=\"351\" class =\"alignnone size-large wp-image-14891\" \/><\/a>\n<\/li>\Configure Apple Pu sh Notification service (APNs) which requires Apple Developer Account and Generat e pl2 certificates. Documentation link : <a href=\"https:\/\/www.ng.bluemix.net\/d ocs\/services\/mobilepush\/index.html#certificates\" target=\" blank\">click here <\/a>\n<\/li>\nUpload the generated pl2 certificate in Push service dashboard \n<a href=\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uplo</pre> ads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-12-at-6.47.14-pm.png\"><img src=\"ht tps:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/20 15\/07\/Screen-Shot-2015-07-12-at-6.47.14-pm-1024x377.png\" alt=\"Push Service\" w idth="980" height=\"361\" class=\"alignnone size-large wp-image-14816\" \/><\/a >\n<\/li>\nAdd the following code inside didFinishLaunchingWithOptions functi on in AppDelegate of client application which will register notifications in clien t app.\n let notificationTypes : UIUserNotificationType = UIUserNotificationType.Badge | UIUserNotificationType. Alert | UIUserNotificationType.Sound\n let notificationSettings: UIUserNot ificationSettings = UIUserNotificationSettings(forTypes: notificationTypes, categ ories: nil)\n application.registerUserNotificationSettings(notif \n icationSettings)\n application.registerForRemoteNotifications()<\/pre>\n<\</pre> /li>\nAdd the following code inside didRegisterForRemoteNotificationsWithDevi ceToken function in AppDelegate of client application which will register pushcl ient and subscribe to tag in client app.\nnpre class=\"brush: plain; title: ; notr anslate\">IMFPushClient.sharedInstance().registerDeviceToken(deviceToken, completi

```
onHandler: { (response, error) -> Void in\n
                                                       if error != nil {\n
println("Error during device registration \\(error.description)&quot
                                                 println(& quot; Response
;)\n
                           else {\n
during device registration json: \\(response.responseJson.description)")
                var tags = ["parkstore"]\n
PushClient.sharedInstance().subscribeToTags(tags, completionHandler: { (response:
IMFResponse!, err:NSError!) -> Void in\n
                                                            if err != nil
                        println("There was an error while subscribing
to tag")\n
                                   }else{\n
                                                               println(&
amp;quot;Successfully subscribe to tag parkstore")\n
                               }<\/pre>\n<\/li>\Add the following functi
                 })\n
on inside Appdelegate which triggers when push notification arrived in client app
.\nfunc application(application
: UIApplication, didReceiveRemoteNotification userInfo: [NSObject : AnyObject]) {
         println("Got remote Notification. Data : \\(userInfo.descripti
\n
                     let info = userInfo as NSDictionary\n
on)")\n
nfo.objectForKey("aps")?.objectForKey("alert"
) as! NSDictionary\n
                         let userData = data.objectForKey("body&q
                       let alertView = UIAlertView(title: "WishList!&
uot;) as! String\n
amp;quot;, message: "\\(userData)", delegate: nil, cancelButtonT
itle: "OK")\n
                                   alertView.show()\n
                                                         \n}<\/pre>\n<\/l</pre>
i>\n<\/ul>\n<h2 id=\"migrateblu\">Existing Bluemix Client Application<\/h2>\nA
dd the following Code snippets to the existing Bluemix Client Application and name
the application with same name which you have registered in Advance Mobile Access
Dashboard.<\/p>\nAdd the following code inside didFinishLaunchingWithOp
tions function in AppDelegate of client application which will register the realm
and initialize connection with Bluemix Application.\n<pre class=\"brush: plain; ti
tle: ; notranslate\"> IMFClient.sharedInstance().registerAuthenticationDelegate(c
ustomAuthDelegate, forRealm: "customAuthRealm 3")\nIMFClient.sh
aredInstance().initializeWithBackendRoute("https:\/\/parkstore.mybluemix
.net", backendGUID: "5e3ad88d-dd48-469d-b46f-2c4ad66b5345&qu
ot;)<\/pre>\n<\/li> The following is the sample code to invoke the Rest url&
amp;#8217;s in client application.\n<pre class=\"brush: plain; title: ; notransla
te\">var request: IMFResourceRequest = IMFResourceRequest(path: "https:\/
\/parkstore.mybluemix.net\/LocalstoreAdapter\/apps\/5e3ad88d-dd48-469d-b46f-2c4ad
66b5345\/localstore\/getAllItems", method: "GET")\n
request.sendWithCompletionHandler { (wlResponse:IMFResponse!, err:NSError!) -&amp
;gt; Void in<\/pre>\n<\/li>\Add the following code inside didFinishLaunchingW
ithOptions function in AppDelegate of client application which will register notif
ications in client app.\n let
notificationTypes: UIUserNotificationType = UIUserNotificationType.Badge | UIUser
NotificationType.Alert | UIUserNotificationType.Sound\n
                                                        let notificationSe
ttings: UIUserNotificationSettings = UIUserNotificationSettings(forTypes: notific
ationTypes, categories: nil)\n
                                 \n
                                           application.registerUserNotificat
ionSettings(notificationSettings)\n
                                      application.registerForRemoteNotificat
ions()<\/pre>\n<\/li>Add the following code inside didRegisterForRemoteNoti
ficationsWithDeviceToken function in AppDelegate of client application which wil
l register pushclient and subscribe to tag in client app.\npre class=\"brush: pla
in; title: ; notranslate\">IMFPushClient.sharedInstance().registerDeviceToken(dev
iceToken, completionHandler: { (response, error) -> Void in\n
f error != nil {\n
                              println("Error during device registrat
ion \\(error.description)")\n
                                            }\n
                                                          else {\n
println("Response during device registration json: \\(response.responseJ
son.description)")\n
                                        var tags = ["parkstore&
quot;]\n
                     IMFPushClient.sharedInstance().subscribeToTags(tags, comp
letionHandler: { (response:IMFResponse!, err:NSError!) -> Void in\n
if err != nil {\n
                                    println(& quot; There was an error whi
```

```
le subscribing to tag")\n
                                                                          }else{\n
println("Successfully subscribe to tag parkstore")\n
                                                }<\/pre>\n<\/li>\Add the following functi
}\n
                          })\n
on inside Appdelegate which triggers when push notification arrived in client app
.\nfunc application(application
: UIApplication, didReceiveRemoteNotification userInfo: [NSObject : AnyObject]) {
             println("Got remote Notification. Data : \\(userInfo.descripti
\n
on)")\n
                                 let info = userInfo as NSDictionary\n
                                                                                               let data = i
nfo.objectForKey("aps")?.objectForKey("alert"
                                      let userData = data.objectForKey("body&q
) as! NSDictionary\n
uot;) as! String\n
                                   let alertView = UIAlertView(title: "WishList!&
amp;quot;, message: "\\(userData)", delegate: nil, cancelButtonT
itle: "OK")\n
                                                     alertView.show()\n
                                                                                        }\n}<\/pre>\n<\/l</pre>
i>\nThe following are the screenshots of client application.<br \/>\n<a href=
\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/3
2\/2015\/07\/IMG_0020.jpg\"><img src=\"https:\/\/developer.ibm.com\/mobilefirstple"
atform\/wp-content\/uploads\/sites\/32\/2015\/07\/IMG 0020-169x300.jpg\" alt=\"IM
G 0020\" width=\"169\" height=\"300\" class=\"alignnone size-medium wp-image-1491
7\" \/><\/a><a href=\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-conten
t\\rho sites/32\/2015\/07\/IMG 00211.jpg\"><img src=\"https:\/\/developer.i"
bm.com\mbox{mobilefirstplatform\wp-content\uploads\sites\32\2015\07\MG 00211-
169x300.jpg\" alt=\"IMG 0021\" width=\"169\" height=\"300\" class=\"alignnone siz
e-medium wp-image-14918\" \/><\/a><a href=\"https:\/\/developer.ibm.com\/mobilefi
rstplatform\\/wp-content\\/uploads\\/sites\\/32\\/2015\\/07\\/IMG\_0025.jpg\\"><img\ src=\\"
https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\
/2015\/07\/IMG_0025-169x300.jpg\" alt=\"IMG_0025\" width=\"169\" height=\"300\" clearly constructed by the construction of t
ass=\"alignnone size-medium wp-image-14920\" \/><\/a><a href=\"https:\/\/develope
r.ibm.com\mbox{mobilefirstplatform}/\mbox{wp-content}/\mbox{uploads}/\sites//32\/2015\/07\/IMG 0024
.jpg\"><img src=\"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/u
ploads\/sites\/32\/2015\/07\/IMG_0024-169x300.jpg\" alt=\"IMG_0024\" width=\"169\)
" height=\"300\" class=\"alignnone size-medium wp-image-14919\" \/><\/a><a href=\
"https:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32
\/2015\/07\/IMG 0026.jpg\"><img src=\"https:\/\/developer.ibm.com\/mobilefirstplat
form\/wp-content\/uploads\/sites\/32\/2015\/07\/IMG 0026-169x300.jpg\" alt=\"IMG
0026\" width=\"169\" height=\"300\" class=\"alignnone size-medium wp-image-14921\
"\/><\/a>\n<\/li>\n<\/ul>\n<h2>Migration to On-Prem<\/h2>\n<h3 id=\"configureclie
nt\">Migration of Client Application<\/h3>\nMigration of Client Application inc
ludes following two steps<\/p>\nConfiguring Cocoapods<\/li>\nClient App Mi
gration<\/li>\n<h3 id=\"cocoapods\">Configuring Cocoapods<\/h3>\nIf CocoaPods
has not been installed on a specific computer:<\/p>\n\nFollow the &amp;#82
20;Getting Started" guide for CocoaPods installation: http:\/\/guides.co
coapods.org\/using\/getting-started.html<\/li>\nOpen &amp;#8220;Terminal&amp;
#8221; at the installation location and run the & #8220; pod init& #8221; com
mand<\/li>\n<\/ul>\nThe following steps assume that the client application is
working with CocoPods. If not, follow this & #8220; Using CocoaPods& #8221;
documentation : <a href=\"http:\/\/guides.cocoapods.org\/using\/using-cocoapods.h</pre>
tml\" target=\" blank\">click here<\/a><\/p>In both cases, the instructions
below explain how to edit the & #8220; Podfile& #8221; file.<\/p>\n\n>
Open the & #8220; Podfile& #8221; file located in the root of your XCode proj
ect in a favourite text editor.<\/li>\nComment out or remove the existing cont
ent.<\/li>\nAdd the following lines:\nnpre class=\"brush: plain; title: ; not
ranslate\">source 'https:\/\/github.rtp.raleigh.ibm.com\/imflocalsdks\/imf-client
-sdk-specs.git'\npod 'IMFCompatibility'<\/pre>\n<\/li>Open &amp;#8220;Termi
nal" at the location of "Podfile".<\/li>\nVerify
that the XCode project is closed.<\/li>\nRun the &amp;#8220;pod install&amp;#8
221; command.<\/li>\n<\/ol>\nOpen the [MyProject].xcworkspace file in XCode. Th
is file is located side by side with [MvProiectl xcodenroi <hr \/>\nAn usual Coco
```

```
IS TICC IS COCACCA SIAC by SIAC WICH [Hyproject].Accacproj.Ast (// (Hum asaac coco
aPods-based project is managed as a workspace containing the application (the exe
cutable) and the library (all project dependencies brought by the CocoaPods manag
er).<\/p>\nIn Xcode&amp;#8217;s Build Settings, search for &amp;#8220;0ther Li
nker Flags" and insert ${inherited} (if -ObjC is defined in this field,
you can just delete it, since it is configured in the CocoaPod project).<\/p>\n<h3
>Client App Migration<\/h3>\n\nSearch for bluemix dependency imports like
\npre class=\"brush: plain; title: ; notranslate\">#import <IMFCore\/IMFC
ore.h>\n#import <IMFPush\/IMFPush.h&amp;gt;<\/pre>\nReplace the a
bove imports with <\/p>\n#impor
t <IMFCompatibility\/IMFCompatibility.h&amp;gt;<\/pre>\n<\/li>\nLook fo
r a call to the & #8220; initializeWithBackendRoute& #8221; method and repla
ce the route URL with your on-premise server URL. For example:\npre class=\"brush
: plain; title: ; notranslate\">IMFClient.sharedInstance().initializeWithBackendR
oute("https:\/\/parkstore.mybluemix.net", backendGUID: &quo
t;5e3ad88d-dd48-469d-b46f-2c4ad66b5345"<\/pre>\nshould be replaced wit
h your on-premise MFP server URL<\/p>\n<pre class=\"brush: plain; title: ; notran
slate\">IMFClient.sharedInstance().initializeWithBackendRoute("http:\/\/
localhost:10080\/ParkStoreMFP", backendGUID: "5e3ad88d-dd48-469
d-b46f-2c4ad66b5345\"\nNote, that backendGUID parameter is igno
red and can be empty. Look for all instantiations of IMFResourceRequest class and
update it<\/li>\nLook for all instantiations of IMFResourceRequest class and u
pdate the request URL with absolute or relative path to the resource. For example:
\npre class=\"brush: plain; title: ; notranslate\">var request: IMFResourceReque
st = IMFResourceRequest(path: "https:\/\/parkstore.mybluemix.net\/Locals
toreAdapter\/apps\/5e3ad88d-dd48-469d-b46f-2c4ad66b5345\/localstore\/getAllItems&
amp;quot;, method: "GET")<\/pre>\nshould be replaced with<\/p</pre>
>\nvar request: IMFResourceRequ
est = IMFResourceRequest(path: "http:\/\/localhost:10080\/ParkStoreMFP\/a
dapters\/LocalstoreAdapter\/localstore\/getAllItems", method: "
GET")<\/pre>\n<\/li>Add the following code inside didRegisterForRe
moteNotificationsWithDeviceToken function in Appdelegate of Client application.\n<
pre class=\"brush: plain; title: ; notranslate\"> WLPush.sharedInstance().tokenFr
omClient = deviceToken.description<\/pre>\n<\/li>All on-premise application
s require the "worklight.plist" file to be present in the appl
ication resources. In the <code>IBMMobileFirstPlatformFoundationNativeSDK<\/code>
pod we supply a file named <strong>sample.worklight.plist<\/strong>.\n\nL
ocate the "sample.worklight.plist" file in the â€~IBMMobileFir
stPlatformFoundationNativeSDK' pod.<\/li>\nCopy this file to the parent (app
lication) project and rename it to "worklight.plist".<\/li>
li>Edit the & #8220; worklight.plist& #8221; file by setting the & #8220;
application id" key to the name of your application deployed to the on-p
remise MFPF server<\/li>\n<\/li>\n<\/ol>\n<h3 id=\"migratemfp\">Migration
of JAX-RS Application to JAVA Adapter<\/h3>\n\nTo migrate JAX-RS applicat
ion to on-prem (MobileFirst Foundation) server we need to do the following steps f
                 Create MobileFirst Project & amp; #8211; & amp; gt; Create native A
or server:\n
PI app for iOS<br \/>\n
                            â€<â€<<br \/>\n<a href=\"https:\/\/developer.ibm.c
om\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2015\/07\/Screen-Shot-201
5-07-12-at-6.50.04-pm.png\"><img src=\"https:\/\/developer.ibm.com\/mobilefirstpla
tform\/wp-content\/uploads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-12-at-6.50.04
-pm.png\" alt=\"Screen Shot 2015-07-12 at 6.50.04 pm\" width=\"595\" height=\"596 ^{\circ}
:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2015
\/07\/Screen-Shot-2015-07-12-at-6.51.13-pm.png\"><img src=\"https:\/\/developer.ib
m.com\mbox{mobilefirstplatform}/\mbox{wp-content}/\mbox{uploads}/\mbox{sites}/32\mbox{2015}/07\mbox{Screen-Shot-ploads}
2015-07-12-at-6.51.13-pm.png\" alt=\"Screen Shot 2015-07-12 at 6.51.13 pm\" width
=\"598\" height=\"590\" class=\"alignnone size-full wp-image-14818\" \/><\/a><\/p
```

ns an hunsf-1 ||https:////davalanaw ibm aam//mahilafiuntulatfaum//www.aautaut//wwl

```
>\n<a nret=\"nttps:\/\/qeveloper.ipm.com\/mopiletirstplattorm\/wp-content\/upl</pre>
oads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-12-at-6.52.28-pm.png\"><img src=\"h
ttps:\/\/developer.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2
015\/07\/Screen-Shot-2015-07-12-at-6.52.28-pm.png\" alt=\"Screen Shot 2015-07-12
at 6.52.28 \text{ pm}" width=\"717\" height=\"424\" class=\"alignnone size-full wp-image
-14819\" \/><\/a><\/li>\nAdd two adapters for Custom Authentication and Locals
tore and migrate the JAX-RS code as shown in the following example.<\/li>
nCopy the JAX-RS BlueMix code and paste it in the newly created Localstore Jav
a adapter JAX-RS file.hAdd and remove the following changes in your adap
ter code.<\p>\n\nremove <code>\{\tenantId}\/<\/code><\/\li>\nremove
the <code>@PathParam -> PathParam(\"tenantId\") String deviceId<\/code> an
d <code>@PathParam(\"realmName\") String realmName<\/code><\/li>\nAdd scope
to the all http api resource <code>@OAuthSecurity (scope=\"customAuthRealm 3\")<\/
code > \langle li > n < \beta \rangle 
: plain; title: ; notranslate\">\n\t@GET\n\t@OAuthSecurity (scope="custo
mAuthRealm 3")\n\t@Path("\/getAllItems")\n\tpublic Str
ing getAllItems() throws MissingConfigurationOptionException{\n\t\tinit();\n\t\tJs
onArray jsonArray = new JsonArray();\n\t\tfor(Object key : props.keySet()){\n\t\t
\tjsonArray.add(parser.parse(props.getProperty((String) key)).getAsJsonObject());
\n\t\ \n\t\treturn jsonArray.toString();\n\t\\n\n\t@PUT\n\t@OAuthSecurity (scope
="customAuthRealm 3")\n\t@Path("\/addItem")\n
\tpublic void addItem(String itemJson) \n\t\t\throws MissingConfigurationOptionE
xception, URISyntaxException, IOException{\n\t\ttry{\n\t\tinit();\n\t\tint new
Key = props.keySet().size()+1;\n\t\tprops.put(String.valueOf(newKey), itemJson)
;\n\t\t\tURL url = this.getClass().getClassLoader().getResource("data.pr
operties"); \n\t\tFile file = new File(url.toURI().getPath());\n\t\t
FileOutputStream foStream = new FileOutputStream(file);\n\t\t\tprops.store(foStrea
m, "saving new item");\n\t\t\tfoStream.close();\n\n\t\t}catch(I
OException ioe){\n\t\tioe.printStackTrace();\n\t\\\n\n\t}\n\n\t@POST\n\t@OAuthS
ecurity (scope="customAuthRealm 3")\n\t@Path("\/addAll
Items")\n\tpublic String addAllItems(String itemsJson) \n\t\t\throws Mis
singConfigurationOptionException, URISyntaxException, IOException{\n\t\ttry{\n\t\
t\tinit();\n\t\t\tclearAllData();\n\t\tJsonArray jsonArr = parser.parse(itemsJs
on).getAsJsonArray();\n\t\ti=0;i<jsonArr.size();i++){\n\t\t
\tprops.put(String.valueOf(i+1), jsonArr.get(i).toString());\n\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\\t
url = this.getClass().getClassLoader().getResource("data.properties&
amp;amp;quot;); \n\t\tFile file = new File(url.toURI().getPath());\n\t\tFile0
utputStream foStream = new FileOutputStream(file);\n\t\tprops.store(foStream, &a
mp;amp;quot;saving new item");\n\t\t\foStream.close();\n\t\t\retur
n "true";\n\t\t}catch(IOException ioe){\n\t\t\tioe.prin
tStackTrace(); \n\t\t\}\n\t\terurn \& amp; quot; false\& amp; quot;; \n\t\n\n\t@D
ELETE\n\t@OAuthSecurity(enabled=false)\n\t@Path("\/clearAll")\n
\tpublic String clearAllData() \n\t\t\throws MissingConfigurationOptionException
, URISyntaxException, IOException{\n\t\tinit();\n\t\tprops.clear();\n\t\tSy
stem.out.println(\&amp;quot;Size : \&amp;quot;+props.size());\n\t\t\tURL url = this
.getClass().getClassLoader().getResource("data.properties"); \n\
t\t\file file = new File(url.toURI().getPath());\n\t\tFileOutputStream foStream
= new FileOutputStream(file);\n\t\tprops.store(foStream, "clearing all
data");\n\t\t\foStream.close();\n\t\t\treturn "cleared&quo
t;;\n\t}\n<\pre>\n<h3 id=\"configoauth\">Configuring Custom-OAuth<\/h3>\n\n<
li>Add realm with same name you had on BlueMix and login module to the authenticat
ionConfig.xml.\n<realm nam
e="customAuthRealm 3" loginModule="customAuthLoginModu
le 3">\n<className&amp;gt;com.worklight.core.auth.ext.Cust
omIdentityAuthenticator<\/className&amp;gt;\t\n&amp;lt;parameter name=&amp
;quot;providerUrl" value="http:\/\/localhost:10080\/ParkStoreMF
P\\adapters\\Customauth\"\\Aamp;gt;\n\<\\realm\&amp;gt;\n\amp;lt;l
```

```
oginModule name="customAuthLoginModule 3" expirationInSeconds=&
amp;quot;3600">\n<className&amp;gt;com.worklight.core.auth
.ext.CustomIdentityLoginModule<\/className&amp;gt;\n&amp;lt;\/loginModule&
amp;gt;<\/pre>\n<\/li>Add Custom-oauth Realm in userIdentityRealms in Appli
cation Descriptor file of iOS Native API\nre class=\"brush: xml; title: ; notra
nslate\"><userIdentityRealms&amp;gt;customAuthRealm_3&amp;lt;\/userIdentit
yRealms\><\/pre>\n<\/li>\n<h3 id=\"configurepush\">Configuring Push
Capability<\/h3>\n\nAdd apns p12 certificate which is generated from Appl
e Developer Account under iOS Native API Folder\n<a href=\"https:\/\/developer
.ibm.com\/mobilefirstplatform\/wp-content\/uploads\/sites\/32\/2015\/07\/Screen-S
hot-2015-07-12-at-6.58.03-pm.png\"><img src=\"https:\/\/developer.ibm.com\/mobilef
irstplatform\/wp-content\/uploads\/sites\/32\/2015\/07\/Screen-Shot-2015-07-12-at-
6.58.03-pm.png\" alt=\"Screen Shot 2015-07-12 at 6.58.03 pm\" width=\"286\" heigh
t=\"171\" class=\"alignnone size-full wp-image-14820\" \/><\/a>\n<\/li> Add
Push configuration in Application Descriptor file of iOS Native API and include th
e password of added apns certificate.\n<pre class=\"brush: xml; title: ; notransla
te\"><pushSender password=&amp;quot;password&amp;quot;\/&amp;gt;\n&amp;lt;
tags>\n <tag&amp;gt;\n
                                       <name&amp;gt;parkstore&amp;lt;\/n
ame\>\n \<\/tag\&amp;gt;\n\&amp;lt;\/tag\&amp;gt;<\/pre>\n<\/li>
Create HTTP Push Adapter with following function code which will send the user pus
h notification to the devices which is subscribed to tag "parkstore&
#8221;.\nre class=\"brush: xml; title: ; notranslate\">function sendTagNotifica
tion(notificationText) {\n
                            var notificationOptions = {};\n
                                                              notificationOpti
ons.message = \{\}; \ 
                      notificationOptions.target = {};\n\n
                                                            notificationOption
s.message.alert = notificationText;\n
                                       notificationOptions.target.tagNames = [&
                                    WL.Server.sendMessage("ParkStoreMF
amp;quot;parkstore"];\n\n
P& quot; , notificationOptions); \n\n
                                        return {\n
                                                         result : "No
tification sent to users subscribed to the tag parkstore."\n
re>\n<\/li>\n<\/ul>\nBy performing above steps one can easily run iOS app built
for Bluemix on MobileFirst Platform and following are the links to samples.<\/p>\n
<h3 id=\"sample\">Sample and Source Code<\/h3>\nBluemix Server : <a href=\"htt
ps:\/\/hub.jazz.net\/git\/chethan\/parkstore-bluemix-server\">Parkstore bluemix s
erver<\/a><br \/>\nBluemix Client : <a href=\"https:\/\/hub.jazz.net\/git\/chetha</pre>
n\/parkstore-bluemix\">Parkstore bluemix<\/a><br \/>\nMFP Server
                                                                  : <a href=\"
https:\/\/hub.jazz.net\/git\/chethan\/parkstore-mfp-server\">Parkstore mfp server
<\/a><br \/>\nMFP Client
                           : <a href=\"https:\/\/hub.jazz.net\/git\/chethan\/pa</pre>
rkstore-mfp\">Parkstore mfp<\//a><\/p>The post <a rel=\"nofollow\" href=\"htt
ps: \/\developer.ibm.com\/\mobilefirstplatform\/\2015\/\08\/\19\/\try-on-bluemix-and-b
uy-mfp\/\">Try on Bluemix and migrate to on-prem MobileFirst Platform<\/a> appeare
d first on <a rel=\"nofollow\" href=\"https:\/\/developer.ibm.com\/mobilefirstplat</pre>
form\">IBM MobileFirst Platform<\/a>.<\/p>",
              "guid": {
                 "content": "https:\/\/developer.ibm.com\/mobilefirstplatform\/?
p=14769",
                 "isPermaLink": "false"
              },
              "link": "https:\/\/developer.ibm.com\/mobilefirstplatform\/2015\/0
8\/19\/try-on-bluemix-and-buy-mfp\/",
              "pubDate": "Wed, 19 Aug 2015 10:36:51 +0000",
              "title": "Try on Bluemix and migrate to on-prem MobileFirst Platfo
rm"
           }
        ],
        "language": "en-US",
        "lastBuildDate": "Tue, 08 Sep 2015 09:22:53 +0000",
        "link": [
           {
```

# **Sample**

The attached sample (https://github.com/MobileFirst-Platform-Developer-Center/JavaAdapters/tree/release71) includes an adapter called RSSReader to test the adapter inside an application.

Last modified on