Resource request from native Android applications

Overview

To create and configure an Android native project, first follow the Configuring a native Android application with the MobileFirst Platform SDK (../../hello-world/configuring-a-native-android-application-with-the-mfp-sdk/) tutorial.

MobileFirst applications can access resources using the `WLResourceRequest` REST API.
This tutorial explains how to use the `WLResourceRequest` API with an HTTP adapter.

Initializing WLClient

1. Create an instance of the `WLClient` class.
   The `WLClient` instance requires a reference to the activity in which it is running.

   ```
   WLClient client = WLClient.createInstance(context);
   ```

2. To establish a connection to the MobileFirst Server instance, use the `connect` method by specifying the `MyConnectListener` class instance as a parameter.
   The `WLClient` instance tries to connect to the MobileFirst Server according to the properties of the `wlclient.properties` file.
   After the connection is established, it invokes one of the methods of the `MyConnectListener` class.
3. Specify that the `MyConnectListener` class implements the `WLResponseListener` interface.

   ```
   public class MyConnectListener implements WLResponseListener {
   ```

   The `WLResponseListener` interface defines two methods:
   - `public void onSuccess (WLResponse response) { }`
   - `public void onFailure (WLFailResponse response) { }`
4. Use these methods to process connection success or connection failure.

Invoking an adapter procedure

After the connection is established with a MobileFirst Server instance, you can use the `WLResourceRequest` class to invoke adapter procedures or call any REST resources.

1. Define the URI of the resource. For a JavaScript HTTP adapter: `/adapters/{AdapterName}/{ProcedureName}`

   ```
   URI adapterPath = new URI("/adapters/RSSReader/getFeed");
   ```

2. Create a `WLResourceRequest` object and choose the HTPP Method (GET, POST, etc).

   ```
   WLResourceRequest request = new WLResourceRequest(adapterPath,WLResourceRequest.GET);
   ```

3. Add the required parameters.
   - For JavaScript-based adapters, use the `params` parameter name to set an array of parameters.

     ```
     request.setQueryParameter("params","['MobileFirst_Platform']");
     ```

   - For Java adapters or other resources, you can use `setQueryParameter` for each parameter.

     ```
     request.setQueryParameter("param1","value1");
     request.setQueryParameter("param2","value2");
     ```

   - Trigger the request with `.send()`.
     Specify a `MyInvokeListener` class instance as a parameter.
     You learn how to define this class instance in the next section.

```
request.send(new MyInvokeListener());
```

Other signatures, which are not covered in this tutorial, exist for the `send` method. Those signatures enable you to set parameters in the body instead of the query, or to handle the response with a delegate instead of a completion handler. See the user documentation to learn more.

Receiving a procedure response

After the procedure invocation is completed, the framework calls one of the methods of the `MyInvokeListener` class.

1. Specify that the `MyInvokeListener` class implements the `WLResponseListener` interface.

   ```
   public class MyInvokeListener implements WLResponseListener {
   ```

   The `WLClient` instance invokes the `onSuccess` and `onFailure` methods.

   If the procedure invocation is successful, the `onSuccess` method of `MyInvokeListener` is invoked.

2. Use that method to get the data that is retrieved from the adapter. The `response` object contains the response data.

   You can use its methods and properties to retrieve the required information.

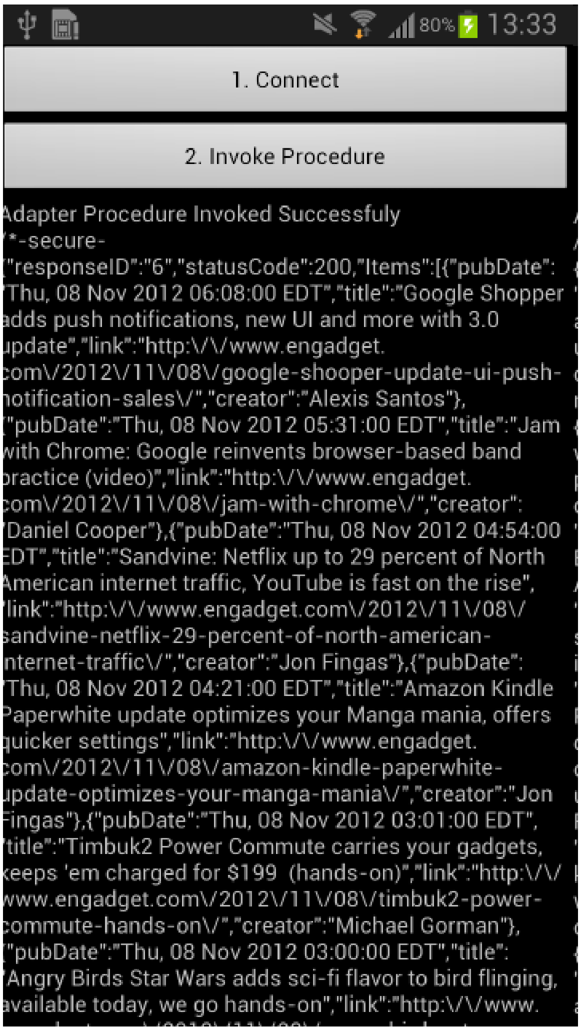
   ```
   public void onSuccess(WLResponse response) {
       String responseText = response.getResponseText();
       AndroidNativeApp.updateTextView("Adapter Procedure Invoked Successfuly\n" + responseText)
   ;
   }

   public void onFailure(WLFailResponse response) {
       String responseText = response.getResponseText();
       AndroidNativeApp.updateTextView("Failed to Invoke Adapter Procedure\n" + responseText);
   }
   ```

Sample application

Click to download (https://github.com/MobileFirst-Platform-Developer-Center/InvokingAdapterProcedures/tree/release71) the MobileFirst project.

Click to download (https://github.com/MobileFirst-Platform-Developer-Center/InvokingAdapterProceduresAndroid/tree/release71) the Native project.

– The `InvokingAdapterProcedures` project contains a MobileFirst Native API to deploy to your MobileFirst Server instance.
– The `InvokingAdapterProceduresAndroid` project contains a native Android application that uses a MobileFirst native API library to communicate with MobileFirst Server.
- Make sure to update the `wlclient.properties` file in the native Android project with the required server settings.



Last modified on