Resource Request from Native Android Applications

Overview

MobileFirst applications can access resources using the `WLResourceRequest` REST API.
The REST API works with all adapters and external resources.

Prerequisite: Ensure you have added the MobileFirst Platform SDK (../../adding-the-mfpf-sdk/adding-the-mfpf-sdk-to-android-applications) to your Native Android project.

WLResourceRequest

The `WLResourceRequest` class handles resource requests to adapters or external resources.

1. Define the URI of the resource:

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

   - For JavaScript adapters, use `/adapters/{AdapterName}/{procedureName}`
   - For Java adapters, use `/adapters/{AdapterName}/{path}`
   - To access resources outside of the project, use the full URL
2. Create a `WLResourceRequest` object and choose the HTTP Method (GET, POST, etc):

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

3. Add the required parameters:
   - In JavaScript adapters, which use ordered nameless parameters, pass an array of parameters with the name `params`:

     ```
     request.setQueryParameter("params","['param1', 'param2']");
     ```

   - In Java adapters or external resources, use the `setQueryParameter` method for each parameter:

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

4. CHANGE according to the new sample Call the resource by using the `.send()` method.
   Specify a `MyInvokeListener` class instance:

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

> See the user documentation to learn more about `WLResourceRequest` and other signatures for the `send` method, which are not covered in this tutorial.

The response

CHANGE according to the new sample When the resource call is completed, the framework calls one of the methods of the `MyInvokeListener` class.

1. CHANGE according to the new sample Specify that the `MyInvokeListener` class implements the `WLResponseListener` interface:

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

2. Implement the `onSuccess` and `onFailure` methods.
   If the resource call is successful, the `onSuccess` method is called. Otherwise, the `onFailure` method is called. Use these methods to get the data that is retrieved from the adapter.
   The `response` object contains the response data and you can use its methods and properties to retrieve the required information.

   ```
   public void onSuccess(WLResponse response) {
        String responseText = response.getResponseText();
        AndroidNativeApp.updateTextView("Successfully called the resource\n" + responseText);
   }

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

For more information

> For more information about WLResourceRequest, refer to the user documentation.

Sample application

Click to download (https://github.com/MobileFirst-Platform-Developer-Center/ResourceRequestAndroid/tree/release80) the Native project. Click to download (https://github.com/MobileFirst-Platform-Developer-Center/Adapters/tree/release80) the adapter Maven project.

Sample usage

- Make sure to update the **mfpclient.plist** file in the Xcode project with the server properties.
- The sample uses the `JavaAdapter` contained in the Adapters Maven project.
- Use either Maven or MobileFirst Developer CLI to build and deploy the adapter (../../creating-adapters/).

SCREENSHOT