IODClient Library for Android

Version 1.0

Contents

1. Overview

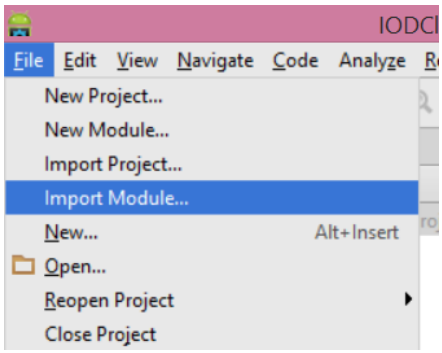
IODClient library for Android is a lightweight Java based API, which helps you easily integrate your Android app with HP IDOL OnDemand Services.

IODClient library requires a minimum Android API level 10.

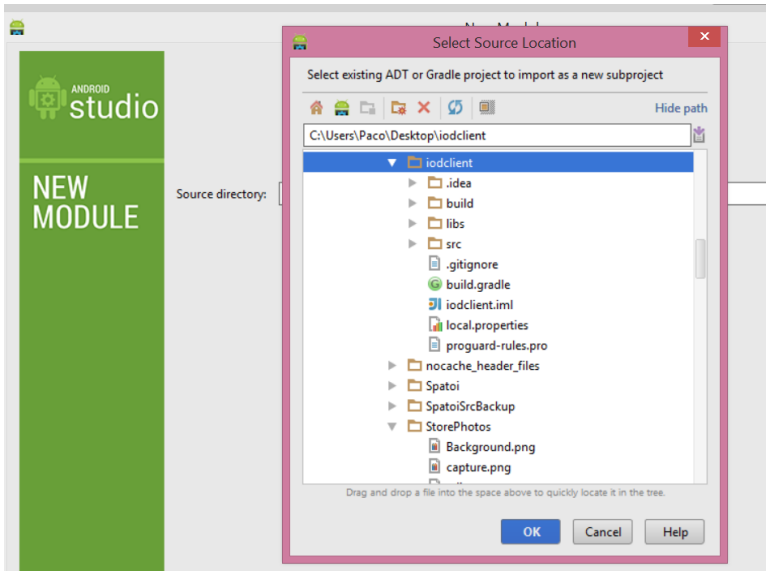
IODClient library uses the Apache httpmime-4.3.2 and httpcore-4.3.2 libraries. For your convenience, the library project included the httpmime-4.3.2.jar and httpcore-4.3.jar files. The dependent components are licensed under the Apache License 2.0. Please see the files called LICENSE.txt and NOTICE.txt for more information.

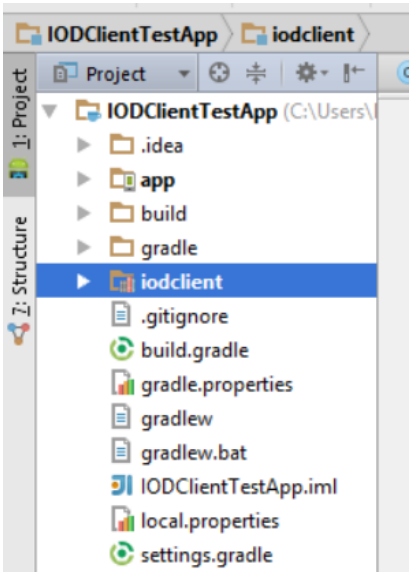
2. Integrate IODClient into Android project

a) Download the IODClient library project for Android.
b) Create a new or open an existing Android project
c) Select the app main folder and click on the File menu then choose Import Module option.



d) Browse to IODClient folder and click OK. The IODClient folder should be created into your project.





e) Open the main project build.gradle and add packaging options and the dependency as follows:

```
android {
        packagingOptions {
                exclude 'META-INF/DEPENDENCIES'
                exclude 'META-INF/NOTICE'
                exclude 'META-INF/LICENSE'
                exclude 'META-INF/LICENSE.txt'
                exclude 'META-INF/NOTICE.txt'
                exclude 'META-INF/ASL2.0‘
        }
}
dependencies {
        compile fileTree(dir: 'libs', include: ['*.jar'])
        compile project (':iodclient')
}
```

3. IODClient API reference

3.1. Constructors

3.1.1 IODClient(String apiKey, IIODClientCallback callback);

Description:

Creates and initializes an IODClient object

Parameters:

apiKey: your developer apikey.

callback: class that implements the IIODClientCallback interface.

Example code:

```
public class MyActivity extends Activity implements
IIODClientCallback {
      String apiKey = “my-api-key”;
      IODClient iodClient = new IODClient(my-api-key, this);
      @Override
      public void requestCompletedWithJobID(String response){ }
```

```
        @Override
        public void requestCompletedWithContent(String response){ }
        @Override
        public void onErrorOccurred(String errorMessage){ }
    }
```

3.1.2 IODClient(String apiKey, String version, IIODClientCallback callback);

Description:

Creates and initializes an IODClient object

Parameters:

apiKey: your developer apikey.

callback: class that implements the IIODClientCallback interface.

version: IDOL OnDemand API version. Currently it only supports version 1. Thus, the value is "v1".

Example code:

```
public class MyActivity extends Activity implements
IIODClientCallback {
        String apiKey = "my-api-key";
        String ver = "v1";
        IODClient iodClient = new IODClient(my-api-key, ver, this);
        @Override
        public void requestCompletedWithJobID(String response){ }
        @Override
        public void requestCompletedWithContent(String response){ }
        @Override
        public void onErrorOccurred(String errorMessage){ }
}
```

3.2. GetRequest(Map<String,Object> params, String iodApp, REQ_MODE mode)

Description:

Sends a GET request to an IDOL OnDemand API.

Parameters:

params: a HashMap object containing key/value pair parameters to be sent to an IDOL OnDemand API, where the keys are the parameters of an IDOL OnDemand API.

Note:

In the case of a parameter type is an array<>, the key must be defined as "arrays" and the value must be a Map<String,String> object with the key is the parameter name and the values separated by commas ",". E.g.:

```
Map<String, String> entity_array = new HashMap<String, String>();

entity_array.put("entity_type", "people_eng,places_eng");

params.put("arrays", entity_array);
```

iodApp: a string to identify an IDOL OnDemand API. E.g. "extractentities". Current supported apps are listed in the IODApps class.

mode [REQ_MODE.SYNC | REQ_MODE.ASYNC]: specifies API call as Asynchronous or Synchronous.

Return: void.

Example code:

```
String iodApp = IODApps.ENTITY_EXTRACTION;

Map<String,Object> params =  new HashMap<String,Object>();

params.put("url", "http://www.cnn.com");

Map<String, String> entity_array = new HashMap<String, String>();

entity_array.put("entity_type", "people_eng,places_eng");

params.put("arrays", entity_array);

iodClient.GetRequest(params, iodApp, IODClient.REQ_MODE.SYNC);
```

Response:

- If the mode is "ASYNC", response will be returned via the requestCompletedWithJobID(String response) callback function.
- If the mode is "SYNC", response will be returned via the requestCompletedWithContent(String response) callback function.
- If there is an error occurred, the error message will be sent via the onErrorOccurred(String errorMessage) callback function.

3.3. PostRequest(Map<String,Object> params, String iodApp, REQ_MODE mode)

Description:

Sends a POST request to an IDOL OnDemand API.

Parameters:

params: a HashMap object containing key/value pair parameters to be sent to an IDOL OnDemand API, where the keys are the parameters of an IDOL OnDemand API.

Note:

In the case of a parameter type is an array<>, the key must be defined as "arrays" and the value must be a Map<String,String> object with the key is the parameter name and the values separated by commas ",". E.g.:

```
Map<String, String> entity_array = new HashMap<String, String>();

entity_array.put("entity_type", "people_eng,places_eng");

params.put("arrays", entity_array);
```

iodApp: a string to identify an IDOL OnDemand API. E.g. "ocrdocument". Current supported apps are listed in the IODApps class.

mode [REQ_MODE.SYNC | REQ_MODE.ASYNC]: specifies API call as Asynchronous or Synchronous.

Return: void.

Example code:

```
String iodApp = IODApps.OCR_DOCUMENT;

Map<String,Object> params =  new HashMap<String,Object>();

params.put("file", "full/path/filename.jpg");

params.put("mode", "document_photo");

iodClient.PostRequest(params, iodApp, IODClient.REQ_MODE.ASYNC);
```

Response:

- If the mode is "ASYNC", response will be returned via the requestCompletedWithJobID(String response) callback function.
- If the mode is "SYNC", response will be returned via the requestCompletedWithContent(String response) callback function.
- If there is an error occurred, the error message will be sent via the onErrorOccurred(String errorMessage) callback function.

3.4. GetJobResult(String jobID)

Description:

Sends a request to IDOL OnDemand to retrieve the content identified by the jobID.

Parameter:

jobID: the job ID returned from an IDOL OnDemand API upon an asynchronous call.

Response:

Response will be returned via the requestCompletedWithContent(String response)

Example code:

```
@Override
public void requestCompletedWithJobID(String response) {
      try {
            JSONObject mainObject = new JSONObject(response);
            if (!mainObject.isNull("jobID")) {
                  jobID = mainObject.getString("jobID");
                  iodClient.GetJobResult(jobID);
            }
      } catch (Exception ex) { }
}
```