Android SDK Integration Document

**1. Prerequisites:**

# Purchase Mobile Plan

Clients need to purchase mobile membership using server-to-server API calls. [[OnBoard Customer api calls](https://drive.google.com/file/d/1tJHUscwaYismIs6jSKaqMyiJ-vdtWzno/view?usp=sharing)](it will be provided via mail)

# Oauth Token

Get SDK OAuth token using generateToken Api

[[generateToken API](https://docs.google.com/document/d/1nCwsdW5NwQrSU7IHUuYiiS80rXeZLv80nhHLqt9ovv8/edit?usp=sharing)]. (it will be provided via mail)

## Activation Code

To activate any mobile membership use Activation Code from onboardCustomer Api and lead status Api API. (document will be provided via mail)

**2. Getting Started**

## Installation

OneAssist Android SDK .aar is available on github. Use the following link to get the latest version of SDK.

<https://github.com/oneassist/sdk-fraud-detection-android> Add library .aar file dependency to Project.

[[How to add .aar to Android Project](https://developer.android.com/studio/projects/android-library#AddDependency)]

**USED GRADLE VERSION : 3.6.4**

**COMPILESDKVERSION : 29 MINSDKVERSION : 16 TARGETSDKVERSION: 29**

## Initialization

Step 1: Initialize SDK from onCreate method of Activity

OASdkCredential credential = OASdkCredential.build()

.setUserToken(<OAUTH\_TOKEN>)

.setSdkTestingEnv(<SDK\_ENV>)

.isDeveloperModeEnabled

(<true|false>); try {

OneAssistSDK.initialize(<Activity Context>, credential, new OneAssistSDK.InitState() {

@Override

public void onSuccess(OAAction action) {

//.... SDK initialization success

}

@Override

public void onFail(OAError error) {

//.... SDK initialization error

}

});

} catch (Exception e) {

// Custom Exception

}

Replace OAUTH\_TOKEN with oauth token generated using `generateToken API`. Replace SDK\_ENV with SDK environment domain :

|  |  |
| --- | --- |
| <https://uat1.1atesting.in/> | Staging |
| <https://api.oneassist.in/> | Production(Default) |

Set SDK mode using isDeveloperModeEnabled :

true - for staging environment

false - for Production

For Staging set `isDeveloperModeEnabled` true and pass staging URL in setSdkTestingEnv . if `isDeveloperModeEnabled` is fal se, SDK will ignore `setSdkTestingEnv` value and use production URL for API Calls.

Replace <Activity Context> with Activity Context(AppCompatActivity)

OneAssistSDK.InitState callbacks provide status with success/fail error messages.

## 2. Membership Activation:

1. In Activity which contain user membership details Override onActivityResult method.

@Override

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

super.onActivityResult(requestCode, resultCode, data); OneAssistSDK.setOnActivityResult(requestCode, resultCode, data);

//....

}

1. **Event Call back Listener:** To get events/next step and error messages from SDK implements membership call back listener OneAssistSD K.SDKCallbacks:

*Call back methods :*

1. onSuccess(OAAction action Get provided task status with next step require to complete Activation flow. OAAction contains getStatu s(),getNextAction() and getNextActionDesc() methods.
2. onProcess(boolean status) Show SDK task progress to user. When SDK doing any background processing return true and on task complete false.
3. onError(OAError oaError) Get error message if any occurred during SDK processing. OAError contains getErrorCode(), getErrorMsg() methods

## SDK Capabilities

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Task** | **Method Name** | **Description** |
| 1 | Start Membership Activation flow or update user basic details | OneAssistSDK. MembershipActivateRequestBui lder() | Use this method to initiate membership Activation process |
| 2 | Update user basic details | .setUserBasicDetails (<OAUserBasicInfo>) | Use this method of MembershipAc tivateRequestBuilder to provide user details and move membership next step |
| 3 | Send Doc upload link using SMS on user other devices | OneAssistSDK. OASendDocUploadLinkRequestBu ilder() | Use this method for sending SMS with doc upload link to user other device . using this link user will upload device front or back panel images |
| 4 | Open SDK Secure screen | .isOpenSecureScreen (true|false) | Use this method to open SDK secure screen, user need to capture front panel image |

**Required Data-Types for SDK methods:**

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Data Type** | **Mandatory(M) or Optional(O)** |
| setActivationCode | ACTIVATION\_CODE as String | M |
| setUserAddress | USER\_ADDRESS as String | M |
| setUserPinCode | USER\_ADDRESS\_PINCODE as String | M |
| setVerificationNo | VERIFICATION\_MOBILE\_NUMBER as String | M |
| isOpenSecureScreen | boolean | M |

* 1. Use OneAssistSDK.MembershipActivateRequestBuilder() to initiate membership activation process:

try {

OneAssistSDK.MembershipActivateRequestBuilder()

.setActivationCode(<ACTIVATION\_CODE>)

.setEventListener(this)

.activate(this);

} catch (Exception e) {

// error message from sdk

}

Replace ACTIVATION\_CODE with membership activation code . ACTIVATION\_CODE is String type.

* 1. Use OneAssistSDK.MembershipActivateRequestBuilder with setUserBasicDetails method to provide user basic details:

OAUserBasicInfo userBasicInfo = new OAUserBasicInfo(); userBasicInfo.setUserAddress(<USER\_ADDRESS>); userBasicInfo.setUserPinCode(<USER\_ADDRESS\_PINCODE>); try {

OneAssistSDK.MembershipActivateRequestBuilder()

.setActivationCode(<ACTIVATION\_CODE>)

.setEventListener(this)

.setUserBasicDetails(userBasicInfo)

.activate(this);

} catch (Exception e) {

// error message from sdk

}

Replace USER\_ADDRESS with user address,USER\_ADDRESS\_PINCODE with user address Pincode and ACTIVATION\_CODE with membership activation code.

USER\_ADDRESS, USER\_ADDRESS\_PINCODE and `ACTIVATION\_CODE` is String type.

* 1. Use OneAssistSDK.OASendDocUploadLinkRequestBuilder() to send doc upload link on other device:

try {

OneAssistSDK.OASendDocUploadLinkRequestBuilder()

.setActivationCode(<ACTIVATION\_CODE>)

.setVerificationNo (<VERIFICATION\_MOBILE\_NUMBER>)

.setEventListener(this)

.sendDocUploadLink(this);

} catch (Exception e) {

// error message from sdk

}

Replace ACTIVATION\_CODE with membership activation code and VERIFICATION\_MOBILE\_NUMBER with verification mobile number on which user want to receive upload doc link.

* 1. Use OneAssistSDK.MembershipActivateRequestBuilder() with isOpenSecureScreen method to open SDK secure screen for user. user need to capture this screen as front panel image if required.

try {

OneAssistSDK.MembershipActivateRequestBuilder()

.setActivationCode(<ACTIVATION\_CODE>)

.setEventListener(this)

.isOpenSecureScreen(true)

.activate(this);

} catch (Exception e) {

// error message from sdk

}

## Normal UI Flow to use OneAssist SDK:

1. Show membership info to user with following actions using membership details:

|  |  |  |
| --- | --- | --- |
| **Membership Status** | **Possible actions for SDK** | **UI/UX** |
| POSTDTLPEND ING | User basic details pending. ask address and pinCode to user and call Me mbershipActivateRequestBuilder with setUserBasicDetails() method | Show UI to user ask pinCode and address |
| PENDING | No Action required from user | In this stage OneAssist is processing user membership. show processing related custom message. |
| POSTDTLCOMP LETE | Front and back panel images pending. | Show following actions for membership:   1. Show action for Send SMS 2. Show open secure screen action |
| REUPLOAD | Provided details are rejected by OneAssist. user need to provide details again | Show following actions for membership:   1. Show action for Send SMS 2. Show open secure screen action |
| **CANCELLED, REJE CTED, APPROVED, QUEUED** | No Action required from user |  |

**SDK Success and Error Codes**

1. **OAError** will contains following error code with messages:

|  |  |  |
| --- | --- | --- |
| **Error** | **Code** | **Description** |
| **ERROR\_EMPTY\_ACTIVATION\_CODE** | **1002** | **ACTIVATION CODE EMPTY** |
| **ERROR\_EMPTY\_ADDRESS** | **1006** | **ADDRESS IS EMPTY** |
| **ERROR\_EMPTY\_MOBILE\_NUMBER** | **1003** | **MOBILE NUMBER IS EMPTY** |
| **ERROR\_EMPTY\_PINCODE** | **1005** | **PINCODE IS EMPTY** |

|  |  |  |
| --- | --- | --- |
| **ERROR\_EMPTY\_TOKEN** | **1001** | **TOKEN EMPTY** |
| **ERROR\_EMPTY\_USER\_DETAILS** | **1004** | **USER DETAILS NOT PROVIDED** |
| **ERROR\_GET\_MEMBERSHIP\_DETAILS** | **3001** | **UNABLE TO GET MEMBERSHIP DETAIL** |
| **ERROR\_GET\_REQ\_DOC\_DETAILS** | **3002** | **UNABLE TO GET MEMBERSHIP DETAIL** |
| **ERROR\_IMEI\_MISSING** | **1007** | **IMEI IS MISSING** |
| **ERROR\_MEMBERSHIP\_APPROVED** | **2003** | **MEMBERSHIP APPROVED** |
| **ERROR\_MEMBERSHIP\_CANCELLED** | **2006** | **MEMBERSHIP CANCELLED** |
| **ERROR\_MEMBERSHIP\_IN\_PROGRESS** | **2001** | **ACTIVATION IN-PROGRESS** |
| **ERROR\_MEMBERSHIP\_PENDING** | **2002** | **ACTIVATION IN-PROGRESS** |
| **ERROR\_MEMBERSHIP\_QUEUED** | **2004** | **MEMBERSHIP QUEUED** |
| **ERROR\_MEMBERSHIP\_REJECTED** | **2005** | **MEMBERSHIP REJECTED** |
| **ERROR\_SEND\_UPLOAD\_LINK** | **3004** | **UNABLE TO SEND SMS** |
| **ERROR\_SMS\_NOT\_SENT** | **1008** | **SEND MESSAGE ON SECONDARY DEVICE FIRST** |
| **ERROR\_UNABLE\_TO\_DO\_OCR** | **4001** | **IMEI NOT RECOGNIZED** |
| **ERROR\_UNABLE\_TO\_PROCESS\_DESC** | **4006** | **UNABLE TO PROCESS. TRY AGAIN!** |
| **ERROR\_UPDATE\_USER\_DETAILS** | **3003** | **UNABLE TO UPDATE USER DETAIL** |
| **ERROR\_MIN\_ANDROID\_VERSION** | **6001** | **THIS FEATURE REQUIRE MINIMUM OREO (ANDROID 8) TO WORK!** |
| **ERROR\_TESTING\_ENV\_NOT\_FOUND** | **6001** | **PROVIDE TESTING ENV BASE URL OR SET DEVELOPER MODE TO FALSE** |
| **ERROR\_REQUIRED\_ACTIVITY\_CONTEXT** | **6003** | **PROVIDE ACTIVITY CONTEXT!!** |
| **ERROR\_UNABLE\_TO\_DO\_OCR** | **4001** | **DEVICE ID NOT RECOGNIZED** |

1. **OAAction** will contains following success code with messages:

|  |  |
| --- | --- |
| **Next Action** | **Next Action Description** |
| **ACTION\_PROVIDE\_USER\_BASIC\_DETAILS** | **USE MEMBERSHIPACTIVATEREQUESTBUILDER WITH SETUSERBASICDETAILS METHOD** |
| **ACTION\_SEND\_UPLOAD\_LINK\_OR\_OPEN\_SECURE\_SCREEN** | **SEND SMS ON OTHER DEVICE OR OPEN SECURE SCREEN** |
| **ACTION\_SEND\_UPLOAD\_LINK** | **SEND SMS ON OTHER DEVICE** |
| **ACTION\_SECURE\_SCREEN\_DISMISSED** | **SECURE SCREEN DISMISSED** |