GetLoanDisburseStatus API GATEWAY DOC

Ujjivan Small Finance Bank Limited
Grape Garden, No.27, 3rd A Cross, 18th Main, 6th Block, Koramangala, Bengaluru – 560 095

Table of Contents

1. INTRODUCTION

1.1 Design Document Purpose

The purpose of this document is to provide a detailed specification of the **GetLoanDisburseStatus** in sufficient depth to:

- Enable the component to be built and tested.
- Ensure that it can be enhanced, supported and maintained by other areas of the organization after initial implementation.

1.2 Design Reviews

The service design will be reviewed within Middleware team and security testing team as needed. Once finalized, design resource will provide an overview to other teams such as front-end application development team and various project resources.

2. API CONNECT COMPONENTS

- API Connect is used to expose the service to front end applications.
- Within API Connect, there are multiple Products. The **GetLoanDisburseStatusReq** API is exposed within **GetLoanDisburseStatus Product.**
- Service Consumers must subscribe to the API. A unique application Identification (client-id) and a secret will be generated.
- The assigned Client-id must be supplied in the header for each API.
- URLs for invoking the services can be found in API Connect Developer portal and mentioned in the below section.

3. TO ACCESS & SUBSCRIBE IN PORTAL

Refer Subscription User manual shared during initial on board.

4. API AUTHENTICATION

JWT Access token to be passed in JSON wrapper as string value in “JWTokenValue” field. The Token can be generated by subscribing to **TOKEN API**.

3.1 TOKEN API

- **Overview:** To retrieve access token.

- **METHOD**: GET

- **Mandatory HTTP Headers**:
  - X-IBM-Client-Id: xxxxxxxxxxxxxxx
  - X-IBM-Client-Secret: xxxxxxxxxxxxxxxxx

- **SIT URL**: https://apiuat.ujjivansfb.in/ujjivan/development/v1/tokens

5. GetLoanDisburseStatus API Details

- **Overview:**

  Service is designed to get the loan disbursement status based on the parameters. The API Gateway makes a backend ESB call which in turn connects to provider system **CBS** of service name **executeFinacleScript.**

- **METHOD**: POST

- **API URL**:

  UAT:
  https://apiuat.ujjivansfb.in/ujjivan/development/v2/exeFinScriptForLoanDisbStatus/getLoanDisburseStatus
  PROD :
  https://api.ujjivansfb.in/ujjivan/development/v1/exeFinScriptForLoanDisbStatus/getLoanDisburseStatus

- **Mandatory HTTP Headers**:
  - X-IBM-Client-Id: xxxxxxxxxxxxxxx

- **Accept**: application/json

- **Content-type**: application/json

- **REQUEST PAYLOAD:**

At API Gateway the request has to be passed as JSON wrapper. The sample payload is as below,

```
 {
"RequestEncryptedValue": "",
 "TransactionId": "",
 "JWTokenValue": ""
 }
```

The individual fields of the Standard JSON Request Payload are described below-

- **RequestEncryptedValue** : This will contain the encrypted value of original REST-JSON request sample.
- **TransactionId** : External partners need to set a transaction ID to uniquely identify every request, in order to retrieve it from an audit trail at a later date.
- **JWTokenValue** : This value can be obtained by invoking Bank's token generator service.

(For more details on request encryption Refer UjjivanSFB_API_Integration_TechnicalProcess_document)

- **Actual REST-JSON Input Parameter:**

Element Name	M/O /C	Data Type	Size	Description	Validation Rules
getLoanDisburseStatusReq/reqHdr	Complex				
reqHdr/consumerContext/applicationId	M	String	3	Application id from which request originated. Example IB, MB,HHD, BRN	
reqHdr/ serviceContext/ uniqueMsgId	M	Stirng		Unique request message id for each message generated from consumer for tracking purpose.	

reqHdr/ serviceContext/ reqMsgDateTime	M	DateTime		Request time stamp in the format CCYY-MM-DDThh:mm:ss.sss	
reqHdr/ serviceContext/ serviceName	M	String		ServiceName to be provided.	
reqHdr/ serviceContext/ serviceVersion	O	String		Service version to be provided. It’s value is 1.	
reqHdr/ providerContext/ providerId	O/F	String	3	Provide application Id from where data expecting	
reqHdr/ userContext/ appUserID	M	String	20	AppUserId will be provided which is created unique to user. Client need to pass this value each and every request.	
reqHdr/ userContext/ appPassword	M	String	64	AppPassword will be provided which is specific to user. Client need to pass this value each and every request.	
getLoanDisburseStatusReq/ body	M	Complex			
body/ acctNumber	M	String	16	Account Number	

- **RESPONSE PAYLOAD STRUCTURE**

A standard JSON wrapper containing encrypted response will be obtained. The encrypted response is of AES/CBC/256 mode with Initiaization vector concatenated at the start of original JSON response.

The sample response structure is as below,

```
{
  "ResponseOfEncrytptedValue": "<encrytped value>",
  "TransactionId": "162193467244544"
}
```

Using the static key shared by Ujjivan Bank, decryption of the value in tag ResponseEncryptedValue should be performed with AES/CBC/256/IV
TransactionId: This field will contain the transaction ID which was passed in request.

(For more details on response encryption Refer UjjivanSFB_API_Integration_TechnicalProcess_document)

- **ACTUAL OUTPUT PARAMETERS:**

	Response				
getLoanDisburseStatusRes/resHdr	Complex				
resHdr/consumerContext/applicationId	M	String	3	Value will be echoed from request message	
resHdr/consumerContext/ terminalId	O	String		Value will be echoed from request message	
resHdr/ serviceContext/ uniqueMsgId	M	Stirng		Value will be echoed from request message	
resHdr/ serviceContext/ reqMsgDateTime	O	DateTime		Value will be echoed from request message	
resHdr/ serviceContext/ timeZone	O	String	3	Value will be echoed from request message	
resHdr/ serviceContext/ serviceName	M	String		Value will be echoed from request message	

reqHdr/ serviceContext/ serviceVersion	O	String		Value will be echoed from request message	
resHdr/ serviceContext/ conversationId	O	String		Value will be echoed from request message	
resHdr/ serviceContext/ parentMsgId	O	String		Value will be echoed from request message	
resHdr/ providerContext/ providerId	M	String	3	Provider System Id from where data is sending	
resHdr/ providerContext/ responseMsgDateTime	O	DateTime		Response message date and time stamp.	
resHdr/responseStatus/status	M	String		Response status from ESB Status 0 = Successful, 1 = Failure	
resHdr/responseStatus/ esbResDateTime	M	DateTime		Response date time stamp from ESB	
resHdr/responseStatus/ errorSource	O	String		If Status = 1, value will be populated with error system source	Will be populated in case of exceptions thrown by ESB
resHdr/responseStatus/ errorCode	C	String		If Status = 1, value will be populated with error code	
resHdr/responseStatus/ errorDescription	C	String		If Status = 1, value will be populated with error description	
resHdr/responseStatus/ errorType	O	String	1	If Status = 1, value will be populated with error Type	
resHdr/ additionalDetails/ details1	O/F	String		Future Use	
resHdr/ additionalDetails/ details1	O/F	String		Future Use	
resHdr/ additionalDetails/ details1	O/F	String		Future Use	
getLoanDisburseStatusRes/body	M	Complex			
body/ status	M	String	256		
body/ referenceNumber	M	String	15		
getLoanDisburseStatusRes/body /errorInfo	O, Repeating	Complex			Will be populated in case of any business exception from back end
errorInfo/errorCode	M	String		Error code	
errorInfo/errorDescription	M	String		Error Desc	
errorInfo/errorType	O	String		Error Type	
errorInfo/errorSource	O	String			Application code where error generated

- **Error Handling**

HTTP Status Code	HTTP Message	More Information
401	Unauthorized	Access token missing or validation
400	Bad Request	The parameters are invalid or missing.
503	Service Unavailable	The parameters were valid but the request failed.

---**END OF DOCUMENT**--