Protocol for WEB API

Initial Public Offer System

Version 2.0.0

October 2026

National Stock Exchange of India Ltd
Exchange Plaza, Plot No. C/1, G Block,
Bandra-Kurla Complex, Bandra (E),
Mumbai - 400 051.

Non-Confidential

Preface

Purpose

This document describes the protocol to be used for Web Frontend to communicate with the Initial Public Offering (IPO) System and thus serves as a development guide.

Target Audience

The document is written for system designers and programmers of user organizations and third-party software developers who are responsible for the development of software to interact with the IPO System of the National Stock Exchange.

Organization of This Document

This document is organized as follows:

Chapters	Description
Chapter 1	Provides a brief introduction
Chapter 2	Describes the general guidelines for the designers and programmers who develop. It details the data types used and discusses the headers that is prefaced with all the structures.
Chapter 3	Describes how a trader logs on to the host system. It covers the log on request, the system responses, entering fresh orders, modifying mismatch order along with notifications.
Chapter 4	Describes the call back API to be exposed by member/bank system over internet. This will be an optional feature, wherein, exchange host system will inform the member/bank system about any changes in DP Verification status of UPI Payment status by invoking this API
Appendix	Lists the error messages.

Non-Confidential

Table of Contents

Preface 2
Purpose 2
Target Audience 2
Organization of This Document 2
Introduction 5
General Instructions 5
API Security 6
Basic Security (v3) 6
Advance Security (v2) 6
Encryption 6
Data Integrity 7
Authentication 7
Decryption 7
Verifying Authentication 8
Verifying Integrity 8
Summary 8
APIs 9
Summary 9
API Reference 11
POST /<version>/login 11
GET /<version>/logout 12
POST /<version>/changePassword 12
GET /<version>/heartbeat 13
POST/<version>/transactions/add 14
POST/<version>/transactions/addbulk 20
GET /<version>/transactions/{{time}} 24
GET /<version>/transactions/slice/{{fromtime}}/{{totime}} 25
POST /<version>/transactions/fetch 25
GET /<version>/notifications/{{time}} 27
GET /<version>/mismatches/{{symbol}} 28
GET /<version>/ipomaster 29
GET /<version>/allotment/{{fromtime}}/{{totime}} 32
POST /<version>/allotment/fetch 34
GET /<version>/allotment/report/{{symbol}} 36

Non-Confidential

POST/<version>/eforms/add 36
POST/<version>/eforms/query 41
GET /<version>/ncbmaster 43
POST/<version>/ncb/add 46
GET /<version>/ncb/{{time}} 48
POST /<version>/ncb/fetch 49
GET /<version>/ncb/mismatches/{{symbol}} 50
GET /<version>/tendermaster 52
POST/<version>/tender/add 53
GET /<version>/tender/{{time}} 57
POST /<version>/tender/fetch 59
GET /<version>/sgbmaster 60
POST/<version>/sgb/add 63
GET /<version>/sgb/{{time}} 65
POST /<version>/sgb/fetch 66
GET /<version>/sgb/mismatches/{{symbol}} 67
GET /<version>/sgb/incomplete/{{symbol}} 69
POST/<version>/sgb/incomplete/update 69
GET /<version>/holidaymaster/{{fromdate}}/{{todate}} 70
Callback APIs 72
Summary 72
General Instructions 72
API Reference 72
POST /<version>/appdpstatus 72
POST /<version>/apppaystatus 73
POST /<version>/notification 74
Appendix A -Error Messages 76
SGB API - List of Incomplete Codes 77
Appendix B – Base URL 78
Appendix C – EIPO- API rate limits 78

Non-Confidential

Introduction

This document describes the Web API's to programmatically access and exchange data with NSE eIPO system (HOST system). The document outlines the messaging protocols and structures for developing such interface.

General Instructions

1. Following headers need to be provided in all API calls
    - **Content-Type**: Header value should be "application/json"
    - **User-Agent**: Header value should be application client information.
2. Path parameters and query parameters in the URL's must be encoded using percentage encoding. (Refer http://www.w3schools.com/tags/ref_urlencode.asp for details)
3. All request and response messages are in JSON (Javascript Object Notation) format. (Refer http://www.json.org/ for details).
4. Some of the key specifications related to JSON and standards followed for the API's are as follows
    - JSON is built on 2 structures. Map containing key value pairs and an ordered list of values.
    - A value could be boolean (true / false), number, decimal, String or a structure (List or Object).
    - Object or key value pair structure consists of keys which are strings and values of any of the above types. E.g. {"name":"Amit", "age":25}
    - List contains list of values. E.g. ["Amit", "Ajay", "Vikas"]
    - A Boolean has only 2 values true or false.
    - String values are enclosed in single quote or double quotes. E.g. "name", "Amit", "Pending"
    - Numbers and decimals are represented without any thousand – separator character. Decimal indicator is dot (".")
    - All dates, times and datetimes are represented as strings and in Indian standard time. Dates are formatted using format "dd-MM-yyyy". Time are formatted as "hh24:mm:ss". Date times are formatted as "dd-MM-yyyy hh24:mm:ss".
5. Token which is generated at a time of login will be expired in 1 hour if session is idle and no activity is performed on it.
6. API Rate-Limit Throttling feature:
    - The API throttling/limiting feature implemented within the API server itself by intercepting every API request.
    - The validation implemented on list of APIs that need to be limit based on the define size of window and request limit. The validation shall be applicable per user session.
    - The user will get rejection in case request exceeds the defined rate limit.
    - Rejection Reason: API limit reached for API :< API Name> user :< User id>

    Exchange would review the said limits periodically and would make changes in the limits as and when required. Participants are required to align their system accordingly.
    Refer: - Appendix C (Page-80)

Non-Confidential

7. The third-party application should follow the JSON data type and field length as per the API specification when invoking exchange APIs. In case of any violation, user will get rejection with HTTP status 403 Forbidden.

API Security

The APIs will support 2 modes of security

- Basic Security - Identified by API version “v3”. Provides authentication of API invoker using Access-Token.
- Advanced Security - Identified by API version “v2”. Provides authentication, encryption and data integrity checks.

Exchange would host both versions of the API’s (“v3” or “v2”) which can be consumed by the Member/Bank participant.

Basic Security (v3)

JWT Authentication (v3)

Authentication

- Client invokes POST/v3/login with valid credentials.
- Exchange validates credentials
- Exchange generates a JWT token and returns it to the client
- Client must include JWT token in all subsequent API requests
- This JWT token is not required in case of login API.

Authorization Header

Authorization: Bearer <JWT_TOKEN>

Header Key	**Description**
Authorization	JWT Access Token used for authenticating API requests **<JWT_TOKEN>** **Append Bearer<space> before token**

Advance Security (v2)

Exchange will generate a 2048-bit RSA key pair for every Member/Bank and will share the public key of the pair with the Member/Bank participant and securely keep the private key.

Similarly, Member/Bank will also generate a 2048-bit RSA key pair for exchange and will share the public key of the pair with the exchange and securely keep the private key.

Encryption

Non-Confidential

- The entity invoking the API should encrypt the request data (in case of POST method only) and add as request body in the HTTP request.
- The API response data should also be encrypted and sent as HTTP response.
- Encryption should be done in 2 steps.
- In first step the data should be encrypted using AES algorithm using 256-bit key.
- In second step the encrypted binary output should then be base64 encoded.
- The AES key should be randomly generated for every request or for a time window of say 15 minutes.
- The AES key should be encrypted using counter entity's public key and then base64 encoded. The encrypted and encoded key should be inserted as a header in request or response. Header key name should be "**Key1**".

Data Integrity

- The API invoker should insert "**Sig**" header in the HTTP request representing a message digest.
- In case of POST method the digest should be constructed by computing SHA256 of the request body followed by base64 encoding. Note that the request body contains encrypted and encoded data.
- In case of GET method which do not have a request body, a randomly generated string not more than 50 characters should be set as digest. Note that the digest would not be required for integrity check but will be required for authenticating the API invoker. (See below)
- In case of responses by API provider, the digest should be constructed in similar fashion using the encrypted and encoded response body and should be added in response header.

Authentication

- The entity (Exchange or Member/Bank) which is invoking an API will identify itself by providing a digital signature. A digital signature would comprise of a digest and signature obtained by encrypting the digest. Following steps should be performed.
- The API invoking entity should insert a header "**member/loginid**". This will be a unique identifier to identify the entity invoking the API. The identifier for National Stock Exchange will be NSE. In case of Member/Bank - the 5 character entity code and login id will be used as identifier.
- Additionally a header "**key3**" should also be inserted in the request. Signature should be computed by first encrypting the digest (computed for Integrity check) using entity's private key and then encoding the encrypted result using base64.
- The API responder is not expected to add headers "member/loginid" and "key3" in the responses.

Decryption

- The HTTP request received by API provider should be decrypted.

Non-Confidential

- Additionally the HTTP response received by API invoker should also be decrypted.
- Header "**key1**" should be read and decrypted first by base64 decoding the header value and then decrypting using the self private key corresponding to the counter entity. This will yield an AES key.
- The encrypted data should then be decrypted using the key and AES algorithm.

Verifying Authentication

- The API provider entity should authenticate API invoker entity using following steps
- Read the header "**member/loginid**" and fetch corresponding public key provided by the counter entity.
- Read the header "**key3**" for signature. Base64 decode it and then decrypt it using the public key fetched in above step.
- Read the header "**Sig**" for digest and match it with the text decrypted in the above step. A successful match would indicate successful authentication.

Verifying Integrity

- Integrity verification should be performed on request body data received in POST method.
- Integrity verification should also be done for all API responses received from the API provider.
- A digest should be computed first by SHA256 encoding the original encrypted and encoded data received in request body or response and then base64 encoding the same.
- The computed digest should be matched with the header value "**Sig**" received. A successful match would indicate a successful integrity verification.

Summary

Header Key	Description	Computation Logic	Required in Request	Required in Response
member	Identifier for the entity invoking the API	Member/Bank Code The identifier for National Stock Exchange will be NSE	Yes	No
loginId	Identifier for the entity invoking the API	Login Id in case of Member/Bank The identifier for National Stock Exchange will be NSE	Yes	No
key1	RSA Encrypted randomly generated AES key used for encrypting request payload or response	BASE64Encode(RSAEncrypt(AESKey, CounterPublicKey)	Yes	Yes
Sig	Message digest by hashing POST request payload or response. For GET request a random alphanumeric string (<=50	BASE64Encode(SHA256(Encrypted POST Request/Response Payload))	Yes	Yes

Non-Confidential

	chars) should be set as message digest			
key3	Signature by RSA encrypting the message digest (Sig) using self private key. In case of GET request where no	BASE64Encode(RSAEncrypt(ipo-key-2, SelfPrivateKey))	Yes	No

Following key sizes and cipher transformations to be used

Algorithm	Key Strength	Cipher Transformation
RSA	2048	RSA/NONE/OAEPWithSHA1AndMGF1Padding
AES	256	AES/GCM/NoPadding

Following list of APIs that support Advanced Security (v2)

API	Description
GET /<version>/ipomaster	This API allows users to download ipo master
POST /<version>/transactions/add	Add / Modify / Cancel application and bids within the application
POST /<version>/transactions/addbulk	Add / Modify / Cancel application and bids within the application in bulk
GET /<version>/transactions/{{time}}	Get Transactions entered after given time. Used to synchronize vendor application database
GET /<version>/transactions/slice/{{fromtime}}/{{totime}}	Get Transactions entered for a given time window slice. Used to synchronize vendor application database
POST /<version>/transactions/fetch	Get Transactions for the given filter criteria
GET /<version>/allotment/{{fromtime}}/{{totime}}	This API allows users to download transactions for which allotment is done.
POST /<version>/ allotment /fetch	This API allows user to download allotment transactions for the given filtered criteria
POST /<version>/eforms/add	Add applications and bids for eForm Pdf generation
POST /<version>/eforms/query	Get uploaded eforms for given filter criteria

APIs

Summary

API	Description
POST /<version>/login	Login
GET /<version>/logout	Log out
POST /<version>/changePassword	Change password

Non-Confidential

GET /<version>/heartbeat	This API allows user to maintain active session to the host in case session is idle or no activity is performed.
POST /<version>/transactions/add	Add / Modify / Cancel application and bids within the application
POST /<version>/transactions/addbulk	Add / Modify / Cancel application and bids within the application in bulk
GET /<version>/transactions/{{time}}	Get Transactions entered after given time. Used to synchronize vendor application database
GET /<version>/transactions/slice/{{fromtime}}/ {{totime}}	Get Transactions entered for a given time window slice. Used to synchronize vendor application database
POST /<version>/transactions/fetch	Get Transactions for the given filter criteria
GET /<version>/notifications/{{time}}	Get notifications (exchange alerts / news / session) after given time
GET /<version>/mismatches/{{symbol}}	Get all mismatch transactions for the given symbol.
GET /<version>/ipomaster	This API allows user to download ipo master
GET /<version>/allotment/{{fromtime}}/{{totime}}	This API allows user to download transactions for which allotment is done.
POST /<version>/ allotment /fetch	This API allows user to download allotment transactions for the given filtered criteria
GET /<version>/allotment/report/{{symbol}}	This API allows user to download symbol wise allotment file only for IPO (SME/Mainboard)
POST /<version>/eforms/add	Add applications and bids for eForm Pdf generation
POST /<version>/eforms/query	Get uploaded eforms for given filter criteria
GET/<version>/ncbmaster	This API allows user to download ncb master
POST/<version>/ncb/add	Add / Modify / Cancel application
GET /<version>/ncb/{{time}}	Get Transactions entered after given time. Used to synchronize vendor application database
POST /<version>/ncb /fetch	Get Transactions for the given filter criteria
POST /<version>/ncb /mismatches	Get Transactions for the given filter criteria
GET /<version>/tendermaster	This API allows user to download tender master
POST/<version>/tender/add	Add application
GET /<version>/tender/{{time}}	Get Transactions entered after given time. Used to synchronize vendor application database
POST /<version>/tender /fetch	Get Transactions for the given filter criteria
GET/<version>/sgbmaster	This API allows user to download SGB master
POST/<version>/sgb/add	This API allows user to add / modify / cancel an application
GET /<version>/sgb/{{time}}	This API allows user to download SGB transactions.
POST /<version>/sgb/fetch	This API allows user to download SGB transactions for the given filtered criteria

Non-Confidential

POST /<version>/sgb/mismatches	This API allows user to download mismatches
POST /<version>/sgb/incomplete	This API allows user to download Incomplete Details
POST /<version>/sgb/incomplete/update	This API allows user to modify incomplete investor details for an application
GET /<version>/holidaymaster/{{fromdate}}/{{todate}}	This API allows user to download holiday master.

API Reference

POST /<version>/login

This API authenticates the user and creates a session for the user in the Host System.

Method	POST
URL	https://<baseurl>/<version>/login
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
member	String(10)	Member/Bank code
loginId	String(30)	Login Id
password	String(15)	Password

Sample Request

```
{
   "member": "M0001",
   "loginId": "U0001",
   "password": "Zcs@44556677"
}
```

Response JSON

Field	Type	Description
member	String(10)	Member/Bank code. As in request
loginId	String(30)	Login Id. As in request
status	String(30)	success / failed / passwordExpired
currentTime	DateTime	Host end date time at the time of login
reason	String(100)	Login failure reason text. Only if “status” = “failed”
token	String(50)	Login Token. Only if “status” = “success”. In subsequent API calls this token should be passed in HTTP header “Access-Token” NOTE :- Token which is generated will be expired in 1 hour if session is idle and no activity is performed on it.

Sample Response – Login Success

```
{
   "member": "M0001",
   "loginId": "U0001",
   "status": "success",
```


Non-Confidential

```
    "currentTime": "01-12-2015 14:30:45",
    "token": "ssadfX3ddf4SZd"
}
```

Sample Response – Login Failed

```
{
    "member": "M0001",
    "loginId": "U0001",
    "status": "failed",
    "currentTime": "01-12-2015 14:30:45",
    "reason": "User disabled"
}
```

GET /<version>/logout

This API closes the current session in the HOST system

Method	GET
URL	https://<baseurl>/<version>/logout
Request	NONE
Response	JSON

Response JSON

Field	Type	Description
status	String(30)	success / failed
currentTime	DateTime	Host end date time at the time of login
reason	String(100)	Failure reason text. Only if "status" = "failed"

Sample Response

```
{
    "status": "success",
    "currentTime": "01-12-2015 15:30:45"
}
```

POST /<version>/changePassword

This API allows user to change password.

Method	POST
URL	https://<baseurl>/<version>/changePassword
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
member	String(10)	Member/Bank code
loginId	String(30)	Login Id
password	String(15)	Password
newPassword	String(15)	New Password. Should follow following guidelines • Minimum length of 12 characters and maximum length of 15 characters. • The password shall be case sensitive and should contain at least one each of the following characters with no space:

Non-Confidential

		a) Upper case alphabet: A to Z or Lower case alphabet: a to z b) Number: 0 to 9 c) Special Characters: `!@#$%^&*()-_=+{}\;:'",./<>? • Password will expire after 365 Days. • Maximum 5 Attempts in case of wrong password entered. • Last 5 passwords are not allowed. • Password to be changed upon first logon for each newly created user.

Sample Request

```
{
   "member": "M0001",
   "loginId": "U0001",
   "password": "Zcs@44556677",
   "newPassword": "rtY#67U56677"
}
```

Response JSON

Field	Type	Description
member	String(10)	Member/Bank code. As in request
loginId	String(30)	Login Id. As in request
status	String(30)	success / failed
reason	String(100)	Failure reason text
currentTime	DateTime	Host end date time

Sample Response - Success

```
{
   "member": "M0001",
   "loginId": "U0001",
   "status": "success",
   "currentTime": "01-12-2015 13:30:45"
}
```

Sample Response - Failed

```
{
   "member": "M0001",
   "loginId": "U0001",
   "status": "failed",
   "reason": "Incorrect old password",
   "currentTime": "01-12-2015 13:30:45"
}
```

GET /<version>/heartbeat

This API allows user to maintain active session to the host in case session is idle or no activity is performed.

Method	GET
URL	https://<baseurl>/<version>/heartbeat
Request	NONE
Response	JSON

Non-Confidential

Response JSON

Field	Type	Description
status	String (10)	success / failed
currentTime	Number	Host current time in milliseconds since epoch
reason	String (100)	Failure reason text. Only if “status” = “failed”

Sample Response - Success

```
{
   "status": "success",
   "currentTime": 1654691767800
}
```

POST/<version>/transactions/add

This API allows user to add / modify / cancel an application and its bids

Method	POST
URL	https://<baseurl>/<version>/transactions/add
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
category	String(5)	Category code.
clientName	String(50)	Optional Client Name – Name of client
depository	String(5)	Depository Name (NSDL, CDSL)
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
nonASBA	Boolean	Flag to indicate application is Non ASBA. Applicable only for DEBT segment true – Non ASBA false – ASBA (default)
chequeNumber	Number	Cheque number Mandatory and valid if nonASBA = true
pan	String(10)	PAN No.- Permanent Account Number
referenceNumber	String(60)	Input ASBA Blocking Ref No- Prefix D for Direct ASBA (Bank User) Prefix S for Syndicate ASBA (Bank User) Prefix T for 3IN1 account type bid (Member User. NON-MANDATORY for UPI bid. Refer ASBA Block Ref No Rules given below
allotmentMode	String(10)	demat / physical

Non-Confidential

upiFlag	String(1)	Mandatory and valid if logged in user is a member Y = UPI Based Blocking N = Non UPI Based Blocking (ASBA)
upi	String(45)	UPI of the investor client Mandatory and valid if logged in user is a bank and upiFlag=Y
bankCode	String(6)	Bank Code Mandatory and valid if logged in user is a member and upiFlag = N
locationCode	String(6)	Location Code Mandatory and valid if logged in user is a member and upiFlag = N
bankAccount	String(45)	Bank Account number Mandatory and valid if logged in user is a bank
ifsc	String(11)	IFSC code Mandatory and valid if logged in user is a bank
subBrokerCode	String(20)	
timestamp	DateTime	Not required in case of new application. In case of application modification timestamp received in the last response for the application
bids	List of Objects	List of bid details. See structure below

Bid Detail JSON

Field	Type	Description
activityType	String(10)	Activity Type new / modify / cancel
bidReferenceNumber	Number	Bid Reference Number. Not required in case of “activityType” = “new”
series	String(4)	Valid series for the issue (applicable for Debt market issues only, Ignored otherwise)
quantity	Number	Bid Quantity
atCutOff	Boolean	Cut off Indicator true - if opting for cut off false- if not opting for cut off
price	Decimal	Bid Price. Mandatory and valid only if “atCutOff” = false. Should be greater than 0 and maximum 2 decimal places.
amount	Decimal	Bid Amount. Should be greater than 0 and maximum 2 decimal places.
remark	String(30)	Optional bid level remarks. Applications may populate their bid unique identifier in this field

	ASBA BLOCK REF NO RULES	
	IPO/SME IPO	**RIGHTS/CALL MONEY**

Non-Confidential

Bank Login	Mandatory & start with 'D' or 'S'	Mandatory & start with 'D'
	Special character (~, |%&) will not be allowed in reference number	Special character (~, |%&) will not be allowed in reference number
Member Login	Mandatory & start with 'T' in case of 3-in-1 account facility investor bid	Mandatory & start with 'T' in case of 3-in-1 account facility investor bid (Applicable only for CALL MONEY)
	Special character (~, |%&) will not be allowed in reference number	Special character (~, |%&) will not be allowed in reference number
	Non-Mandatory in case of UPI & Non-3-in-1 account type bid	

Sample Request

```
{
   "symbol": "TEST",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "clientName": "Sumit Aggarwal",
   "depository": "NSDL",
   "dpId": "33445566",
   "clientBenId": "12345678",
   "nonASBA": false,
   "pan": "AFAKA2323L",
   "referenceNumber": "MYREF0001",
   "allotmentMode": "demat",
   "upiFlag": "Y",
   "upi": "9393939393@upi",
   "bankCode": null,
   "locationCode" : null,
   "timestamp" : "01-12-2015 13:40:55",
   "bids": [
         {
               "activityType" : "new",
               "quantity": 100,
               "atCutOff": false,
               "price": 55.30,
               "amount": 5530.00,
               "remark": "BD/000001"
         },
         {
               "activityType" : "modify",
               "bidReferenceNumber": 2015120100000122,
               "quantity": 110,
               "atCutOff": false,
               "price": 54.30,
               "amount": 5863.00,
               "remark": "BD/000002"
         },
         {
               "activityType" : "cancel",
               "bidReferenceNumber": 2015120100000123,
               "quantity": 95,
               "atCutOff": true,
               "remark": "BD/000003"
```


Non-Confidential

```
                }
            ]
        }
```

Response JSON

Field	Type	Description
Symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
category	String(5)	Category code.
clientName	String(50)	Optional Client Name – Name of client
depository	String(5)	Depository Name (NSDL, CDSL)
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
nonASBA	Boolean	Flag to indicate application is Non ASBA. Applicable only for DEBT segment true – Non ASBA false – ASBA (default)
chequeNumber	Number	Cheque number Mandatory and valid if nonASBA = true
pan	String(10)	PAN No.- Permanent Account Number
referenceNumber	String(60)	Input ASBA Blocking Ref No- Prefix D for Direct ASBA (Bank User) Prefix S for Syndicate ASBA (Bank User) Prefix T for 3IN1 account type bid (Member User. NON-MANDATORY for UPI bid.
allotmentMode	String(10)	demat / physical
upiFlag	String(1)	Mandatory and valid if logged in user is a member Y = UPI Based Blocking N = Non UPI Based Blocking (ASBA)
upi	String(45)	UPI of the investor client Mandatory and valid if logged in user is a bank and upiFlag=Y
bankCode	String(6)	Bank Code Mandatory and valid if logged in user is a member and upiFlag = N
locationCode	String(6)	Location Code Mandatory and valid if logged in user is a member and upiFlag = N
bankAccount	String(45)	Bank Account number Mandatory and valid if logged in user is a bank
Ifsc	String(11)	IFSC code Mandatory and valid if logged in user is a bank
subBrokerCode	String(20)	
timestamp	DateTime	Last update time of the application

Non-Confidential

dpVerStatusFlag	String(1)	DP Verification Status for PAN DP combination check P = Pending S = Success F = Failed
dpVerFailCode	String(10)	Latest DP Verification failure code given by depository
dpVerReason	String(250)	DP Verification failure reason.
upiPaymentStatusFlag	Number	Last UPI Payment status received from sponsor bank for the given application This field only be visible in response if upi payment status is received from sponsor bank ends. 0=Request Sent 1=Request Failed 10=Request Accepted By Sponsor Bank 11=Rejected due to invalid UPI 12=Rejected by Sponsor Bank 13=Unsupported Investor Bank 21=Rejected by Investor Bank 22=Block/Release Request rejected by Client Bank due to Technical Reason 31=Rejected by Investor 100=Accepted by Investor 110=Block Released (Due to cancellation of order)
upiAmtBlocked	Decimal	Amount blocked in case of applications with upiFlag=Y
status	String	success/failed
reasonCode	Number	Fail reason code. Valid only if “status” = “failed”
reason	String(250)	Reason text
bids	List of Objects	List of bid details. See structure below

Bid Detail JSON

Field	Type	Description
activityType	String(10)	Activity Type new / modify / cancel
bidReferenceNumber	Number	Bid Reference Number.
series	String(4)	Valid series for the issue (applicable for Debt market issues only, Ignored otherwise)
quantity	Number	Bid Quantity
atCutOff	Boolean	Cut off Indicator true - if opting for cut off false- if not opting for cut off
price	Decimal	Bid Price. Mandatory and valid only if “atCutOff” = false. Should be greater than 0 and maximum 2 decimal places.
amount	Decimal	Bid Amount. Should be greater than 0 and maximum 2 decimal places.
remark	String(30)	Same as input
status	String	success/failed

Non-Confidential

reasonCode	Number	Fail reason code. Valid only if “status” = “failed”
reason	String(250)	Reason text

Sample Response


```
{
   "symbol": "TEST",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "clientName": "Sumit Aggarwal",
   "depository": "NSDL",
   "dpId": "33445566",
   "clientBenId": "12345678",
   "nonASBA": false,
   "chequeNumber": null,
   "pan": "AFAKA2323L",
   "referenceNumber": "MYREF0001",
   "allotmentMode": "demat",
   "upiFlag": "Y",
   "upi": "9393939393@upi",
   "bankCode": null,
   "locationCode" : null,
   "subBrokerCode": "",
   "dpVerStatusFlag” : “P”,
   "upiAmtBlocked” : 0.00,
   "timestamp" : "01-12-2015 14:55:32",
   "status" : "success",
   "bids": [
         {
               "activityType" : "new",
               "bidReferenceNumber": 2015120100000382,
               "quantity": 100,
               "atCutOff": false,
               "price": 55.30,
               "amount": 5530.00,
               "remark": "BD/000001",
               "status" : "success"
         },
         {
               "activityType" : "modify",
               "bidReferenceNumber": 2015120100000122,
               "quantity": 110,
               "atCutOff": false,
               "price": 54.30,
               "amount": 5863.00,
               "remark": "BD/000002",
               "status" : "success"
         }

   ]
}
```


Sample Response - Failed


```
{
   "symbol": "TEST",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "clientName": "Sumit Aggarwal",
```

```
      "depository": "NSDL",
      "dpId": "33445566",
      "clientBenId": "12345678",
      "nonASBA": false,
      "chequeNumber": null,
      "pan": "AFAKA2323L",
      "referenceNumber": "MYREF0001",
      "allotmentMode": "demat",
      "upiFlag": "Y",
      "upi": "9393939393@upi",
      "bankCode": null,
      "locationCode" : null,
      "timestamp" : "01-12-2015 14:55:32",
      "status" : "failed",
      "reasonCode" : 501,
      "reason" : "Error in bid",
      "bids": [
            {
                  "activityType" : "new",
                  "bidReferenceNumber": 2015120100000382,
                  "quantity": 100,
                  "atCutOff": false,
                  "price": 55.30,
                  "amount": 5530.00,
                  "remark": "BD/000001",
                  "status" : "failed",
                  "reasonCode" : 91,
                  "reason" : "Market timings over"
            },
            {
                  "activityType" : "modify",
                  "bidReferenceNumber": 2015120100000122,
                  "quantity": 110,
                  "atCutOff": false,
                  "price": 54.30,
                  "amount": 5863.00,
                  "remark": "BD/000002",
                  "status" : "success"
            }

      ]
   }
```


POST/<version>/transactions/addbulk

This API allows user to add / modify / cancel one or more applications and their corresponding bids

Method	POST
URL	https://<baseurl>/<version>/transactions/addbulk
Request	JSON
Response	JSON

Request JSON

JSON containing list of elements with each element representing an application. The structure of the element in the list will be same as "Request JSON" in API POST/<version>/transactions/add.

Sample Request


```
   [
      {
            "symbol": "TEST",
```

```
            "applicationNumber": "1200299929020",
            "category": "IND",
            "clientName": "Sumit Aggarwal",
            "depository": "NSDL",
            "dpId": "33445566",
            "clientBenId": "12345678",
            "nonASBA": false,
            "pan": "AFAKA2323L",
            "referenceNumber": "MYREF0001",
            "allotmentMode": "demat",
            "upiFlag": "Y",
            "upi": "9393939393@upi",
            "bankCode": null,
            "locationCode" : null,
            "timestamp" : "01-12-2015 13:40:55",
            "bids": [
                  {
                        "activityType" : "new",
                        "quantity": 100,
                        "atCutOff": false,
                        "price": 55.30,
                        "amount": 5530.00,
                        "remark": "BD/000001"
                  },
                  {
                        "activityType" : "modify",
                        "bidReferenceNumber": 2015120100000122,
                        "quantity": 110,
                        "atCutOff": false,
                        "price": 54.30,
                        "amount": 5863.00,
                        "remark": "BD/000002"
                  },
                  {
                        "activityType" : "cancel",
                        "bidReferenceNumber": 2015120100000123,
                        "quantity": 95,
                        "atCutOff": true,
                        "remark": "BD/000003"
                  }
            ]
      },
      {
            "symbol": "TEST1",
            "applicationNumber": "12002991111",
            "category": "IND",
            "clientName": "Suresh Kumar",
            "depository": "NSDL",
            "dpId": "33445566",
            "clientBenId": "12345678",
            "nonASBA": false,
            "pan": "AFAKA2333L",
            "referenceNumber": "MYREF0002",
            "allotmentMode": "demat",
            "upiFlag": "Y",
            "upi": "333333@upi",
            "bankCode": null,
            "locationCode" : null,
            "timestamp" : "01-12-2015 13:40:55",
            "bids": [
                  {
                        "activityType" : "new",
```


Non-Confidential

```
                        "quantity": 100,
                        "atCutOff": false,
                        "price": 55.30,
                        "amount": 5530.00,
                        "remark": "BD/000001"
                  },
                  {
                        "activityType" : "modify",
                        "bidReferenceNumber": 2015120100000122,
                        "quantity": 110,
                        "atCutOff": false,
                        "price": 54.30,
                        "amount": 5863.00,
                        "remark": "BD/000002"
                  },
                  {
                        "activityType" : "cancel",
                        "bidReferenceNumber": 2015120100000123,
                        "quantity": 95,
                        "atCutOff": true,
                        "remark": "BD/000003"
                  }
            ]
      }
]
```


Request JSON

JSON containing list of elements with each element response for applications in the request. The structure of the element in the list will be same as "Response JSON" in API POST/<version>/transactions/add

Sample Response


```
[
   {
         "symbol": "TEST",
         "applicationNumber": "1200299929020",
         "category": "IND",
         "clientName": "Sumit Aggarwal",
         "depository": "NSDL",
         "dpId": "33445566",
         "clientBenId": "12345678",
         "nonASBA": false,
         "chequeNumber": null,
         "pan": "AFAKA2323L",
         "referenceNumber": "MYREF0001",
         "allotmentMode": "demat",
         "upiFlag": "Y",
         "upi": "9393939393@upi",
         "bankCode": null,
         "locationCode" : null,
         "subBrokerCode": "",
         "dpVerStatusFlag" : "P",
         "upiAmtBlocked" : 0.00,
         "timestamp" : "01-12-2015 14:55:32",
         "status" : "success",
         "bids": [
               {
                     "activityType" : "new",
                     "bidReferenceNumber": 2015120100000382,
```


Non-Confidential

```
                        "quantity": 100,
                        "atCutOff": false,
                        "price": 55.30,
                        "amount": 5530.00,
                        "remark": "BD/000001",
                        "status" : "success"
                  },
                  {
                        "activityType" : "modify",
                        "bidReferenceNumber": 2015120100000122,
                        "quantity": 110,
                        "atCutOff": false,
                        "price": 54.30,
                        "amount": 5863.00,
                        "remark": "BD/000002",
                        "status" : "success"
                  }

            ]
      },
      {
            "symbol": "TEST1",
            "applicationNumber": "12002991111",
            "category": "IND",
            "clientName": "Suresh Kumar",
            "depository": "NSDL",
            "dpId": "33445566",
            "clientBenId": "12345678",
            "nonASBA": false,
            "chequeNumber": null,
            "pan": "AFAKA2333L",
            "referenceNumber": "MYREF0002",
            "allotmentMode": "demat",
            "upiFlag": "Y",
            "upi": "9393939393@upi",
            "bankCode": null,
            "locationCode" : null,
            "timestamp" : "01-12-2015 14:55:32",
            "status" : "failed",
            "reasonCode" : 501,
            "reason" : "Error in bid",
            "bids": [
                  {
                        "activityType" : "new",
                        "bidReferenceNumber": 2015120100000382,
                        "quantity": 100,
                        "atCutOff": false,
                        "price": 55.30,
                        "amount": 5530.00,
                        "remark": "BD/000001",
                        "status" : "failed",
                        "reasonCode" : 91,
                        "reason" : "Market timings over"
                  },
                  {
                        "activityType" : "modify",
                        "bidReferenceNumber": 2015120100000122,
                        "quantity": 110,
                        "atCutOff": false,
                        "price": 54.30,
                        "amount": 5863.00,
                        "remark": "BD/000002",
```


Non-Confidential

```
                        "status" : "success"
                    }

                ]
            }
    ]
```

GET /<version>/transactions/{{time}}

This API allows user to download transactions.

Method	GET
URL	https://<baseurl>/<version>/transactions/{{time}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
time	DateTime	Cutoff date time. Transactions entered after this time will be returned in the response

Sample URL with Path Parameters

```
https://<baseurl>/<version>/transactions/01-12-2015%2009:30:00
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction responses. See Response JSON Structure for API "POST /<version>/transactions/add"

Sample Response - Success

```
    {
       "status": "success",
       "transactions" :
             [
              {
                   "symbol": "TEST",
                   "applicationNumber": "1200299929020",
                   "category": "IND",
                   "clientName": "Sumit Aggarwal",
                   "depository": "NSDL",
                   "dpId": "33445566",
                   "clientBenId": "12345678",
                   "nonASBA": false,
                   "chequeNumber": null,
                   "pan": "AFAKA2323L",
                   "referenceNumber": "MYREF0001",
                   "allotmentMode": "demat",
                   "upiFlag": "Y",
                   "upi": "9393939393@upi",
                   "bankCode": null,
                   "locationCode" : null,
                   "dpVerStatusFlag" : "P",
                      "subBrokerCode": "",
```


Non-Confidential

```
                    "upiAmtBlocked" : 0.00,
                    "timestamp" : "01-12-2015 14:55:32",
                    "status" : "success",
                    "bids": [
                                {
                                "activityType" : "new",
                                "bidReferenceNumber": 2015120100000382,
                                "quantity": 100,
                                "atCutOff": false,
                                "price": 55.30,
                                "amount": 5530.00,
                                "status" : "success"
                                },
                                {
                                      "activityType" : "modify",
                                "bidReferenceNumber":2015120100000122,
                                      "quantity": 110,
                                      "atCutOff": false,
                                      "price": 54.30,
                                      "amount": 5863.00,
                                      "status" : "success"
                                }

                                ]
                    }
              ]
    }
```


Sample Response - Failed


```
    {
       "status": "failed",
       "reason": "Xyz"
    }
```


GET /<version>/transactions/slice/{{fromtime}}/{{totime}}

This API allows user to download transactions for a time window

Method	GET
URL	https://<baseurl>/<version>/transactions/slice/{{fromtime}}/{{totime}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
fromtime	DateTime	Transactions entered after this time will be returned in the response
totime	DateTime	Transactions entered before this time will be returned in the response

Sample URL with Path Parameters

https://<baseurl>/<version>/transactions/01-12-2021%2010:30:00/01-12-2021%2011:30:00

Response JSON

Response JSON will be same as that for API GET /<version>/transactions/{{time}}

POST /<version>/transactions/fetch

Non-Confidential

This API allows user to download transactions for the given filter criteria.

Method	POST
URL	https://<baseurl>/<version>/transactions/fetch
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Mandatory Symbol
applicationNumber	String(16)	Mandatory Application No.(of the physical form)

Sample Request

```
{"symbol": "TEST","applicationNumber": "1200299929020"}
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction response. See Response JSON Structure for API "POST /<version>/transactions/add"

Sample Response - Success

```
{
   "status": "success",
   "transactions" :
         [
          {
               "symbol": "TEST",
               "applicationNumber": "1200299929020",
               "category": "IND",
               "clientName": "Sumit Aggarwal",
               "depository": "NSDL",
               "dpId": "33445566",
               "clientBenId": "12345678",
               "nonASBA": false,
               "chequeNumber": null,
               "pan": "AFAKA2323L",
               "referenceNumber": "MYREF0001",
               "allotmentMode": "demat",
               "upiFlag": "Y",
               "upi": "9393939393@upi",
               "bankCode": null,
               "locationCode" : null,
               "dpVerStatusFlag" : "P",
               "upiAmtBlocked" : 0.00,
               "subBrokerCode": "",
               "timestamp" : "01-12-2015 14:55:32",
               "status" : "success",
               "bids": [
                           {
                           "activityType" : "new",
                           "bidReferenceNumber": 2015120100000382,
                           "quantity": 100,
                           "atCutOff": false,
                           "price": 55.30,
```


Non-Confidential

```
                                "amount": 5530.00,
                                "status" : "success"
                                },
                                {
                                "activityType" : "modify",
                                "bidReferenceNumber": 2015120100000122,
                                "quantity": 110,
                                "atCutOff": false,
                                "price": 54.30,
                                "amount": 5863.00,
                                "status" : "success"
                            }

                    ]
                }
            ]
    }
```


Sample Response - Failed


```
    {
       "status": "failed",
       "reason": "No Records Found"
    }
```


GET /<version>/notifications/{{time}}

This API allows user to download notifications.

Method	GET
URL	https://<baseurl>/<version>/notifications/{{time}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
time	DateTime	Cutoff date time for notifications

Sample URL with Path Parameters

```
https://<baseurl>/<version>/notifications/01-12-2015%2009:30:00
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
notifications	List of Objects	List of Notification objects

Notification Object JSON Structure

Field	Type	Description
type	String(10)	alert/news/session
message	String(200)	message

Sample Response


```
    {
       "status": "success",
       "notifications": [
             {
```


Non-Confidential

```
                    "type": "alert",
                    "message": "Application entry time for symbol XYZ
    relaxed till 06:00 pm"
                },
                {
                    "type": "session",
                    "message": "Symbol PQR category QIB bid entry closed"
                }
        ]
    }
```


GET /<version>/mismatches/{{symbol}}

This API allows user to download notifications.

Method	GET
URL	https://<baseurl>/<version>/mismatches/{{symbol}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
symbol	String(12)	Issue Symbol

Sample URL with Path Parameters

https://<baseurl>/<version>/mismatches/TESTSYM01

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
mismatches	List of Objects	List of Mismatch objects

Mismatch Object JSON Structure

Field	Type	Description
symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
category	String(5)	Category code.
clientName	String(50)	Optional Client Name – Name of client
depository	String(5)	Depository Name (NSDL, CDSL)
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	String(16)	Client ID in case of NSDL Beneficiary ID in case of CDSL
nonASBA	Boolean	Flag to indicate application is Non ASBA. Applicable only for DEBT segment true – Non ASBA false – ASBA (default)
chequeNumber	Number	Cheque number Mandatory and valid if nonASBA = true
pan	String(10)	PAN No.- Permanent Account Number

Non-Confidential

bidReferenceNumber	Number	Bid Reference Number.
errorCode	String(1)	Mismatch error code
reason	String(250)	Reason text

Sample Response


```
{
   "status": "success",
   "mismatches": [
         {
               "symbol": "TEST",
               "applicationNumber": "1200299929020",
               "category": "IND",
               "clientName": "Sumit Aggarwal",
                  "chequeNumber": "",
               "depository": "NSDL",
               "dpId": "33445566",
               "clientBenId": "12345678",
               "pan": "AFAKA2323L"
                  "nonASBA": false,
               "referenceNumber": "MYREF0001",
               "errorCode" : "W",
               "reason" : "Account does not exist"
         },
         {
               "symbol": "TEST",
               "applicationNumber": "1200299929025",
               "category": "IND",
               "clientName": "Asha Aggarwal",
                  "chequeNumber": "",
               "depository": "NSDL",
               "dpId": "33445566",
               "clientBenId": "12345679",
               "pan": "AFAKA2326L",
                  "nonASBA": false,
               "referenceNumber": "MYREF0002",
               "errorCode" : "W",
               "reason" : "Account does not exist"
         }
   ]
}
```


GET /<version>/ipomaster

This API allows user to download ipo master.

Method	GET
URL	https://<baseurl>/<version>/ipomaster
Request	None
Response	JSON

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
data	List of Objects	List of IPO Master objects

IPO Master Object JSON Structure

Field	Type	Description

symbol	String(12)	Symbol
name	String(250)	Name of the issue
lotSize	Number	Bidding Lot Size
minBidQuantity	Number	Optional Minimum Quantity for Bidding
faceValue	Decimal	Face value
minPrice	Decimal	Optional Minimum Price
maxPrice	Decimal	Optional Maximum Price
tickSize	Decimal	Tick Size
cutoffPrice	Decimal	Cutoff price
biddingStartDate	Date	First Date of bidding
biddingEndDate	Date	Last Date of normal bidding
dailyStartTime	Time	Bidding start time (IPO Market Timings)
dailyEndTime	Time	Bidding end time (IPO Market Timings)
Isin	String(12)	Isin
issueSize	Number	Issue size
subType	String(19)	Type of sub type. Possible value will be SME/Call Money /SSE
registrar	String(100)	Name of registrar
t1ModStartDate	Date	First Date of T1 modification
t1ModEndDate	Date	Last Date of T1 modification
t1ModStartTime	Time	Start time for T1 modification
t1ModEndTime	Time	End time for T1 modification
issueType	String(10)	Type of Issue. Possible value will be EQUITY/ DEBT/RIGHTS/REITS/INVIT.
seriesDetails	List of Objects	List of series details. (Applicable for Debt market issues only, Ignored otherwise). See structure below
categoryDetails	List of Objects	List of Category details
subCategorySettings	List of Objects	List of Category/Subcategory Settings

Series Detail JSON

Field	Type	Description
code	String(4)	Valid series for the issue.
desc	String(200)	Series description.

Category Detail JSON

Field	Type	Description
code	String(14)	Valid category code for the issue. In case of IPO, the possible value will be RETAIL (Main Board IPO) and INDIV (SME IPO)
startTime	Time	Bidding start time. Format hh24:mm:ss. E.g. “10:00:00”
endTime	Time	Bidding end time. Format hh24:mm:ss. E.g. “17:00:00”
lastDayEndTime	Time	Last day bidding end time. Format hh24:mm:ss. E.g. “17:00:00”. Applicable only for SME IPO.

Subcategory Settings JSON

Non-Confidential

Field	Type	Description
caCode	String (14)	Valid category code for the issue In case of IPO, the possible value will be RETAIL (Main Board IPO) and INDIV (SME IPO)
subCatCode	String (5)	Valid subcategory code for the issue.
minValue	Decimal	Minimum Value
maxValue	Decimal	Maximum Value
allowCutOff	Boolean	true - if cut off is allowed false- if cut off is not allowed
discountType	String (1)	A- Absolute P- Percentage
discountPrice	Decimal	Discount Price per share in case of Absolute else Discount in percentage
maxQuantity	Number	Max Quantity
allowUpi	Boolean	true - if UPI is allowed false- if UPI is not allowed
maxUpiLimit	Decimal	Max UPI Limit

Sample Response


```
{
   "status": "success",
   "data": [
         {
               "symbol": "TESTIPO",
               "name": "Test Industries",
               "issueType" : "EQUITY",
               "lotSize": 1,
               "minBidQuantity": 100",
               "faceValue": 10.00,
               "minPrice": 12.00,
               "maxPrice": 12.50,
               "tickSize": 0.05,
               "cutoffPrice": 12.50,
               "biddingStartDate": "01-01-2019",
               "biddingEndDate": "03-01-2019",
               "dailyStartTime": "10:00:00",
               "dailyEndTime": "17:00:00",
               "isin": "INE121212121",
               "issueSize": 150000000,
               "registrar": "XYZ Registrars",
               "t1ModStartDate": "04-01-2019",
               "t1ModEndDate": "04-01-2019",
               "t1ModStartTime": "10:00:00",
               "t1ModEndTime": "13:00:00",
               "categoryDetails": [
                     {
                           "code": "RETAIL",
                           "startTime": "10:00:00",
                           "endTime": "17:00:00"
                     },
                     {
                           "code": "QIB",
                           "startTime": "10:00:00",
```


Non-Confidential

```
                                "endTime": "16:00:00"
                            }
                    ],
                       "subCategorySettings": [
                    {
                       "caCode": "QIB",
                       "subCatCode": "FI",
                       "minValue": 200001.00,
                       "maxValue": 6841800000.00,
                       "allowCutOff": false,
                       "allowUpi": false
                     },
                     {
                        "caCode": "RETAIL",
                        "subCatCode": "IND",
                        "minValue": 0.00,
                        "maxValue": 200000.00,
                        "allowCutOff": true,
                        "discountType": "A",
                        "discountPrice": 10,
                        "allowUpi": true,
                        "maxUpiLimit": 200000.00
                     }
                     ]
                },
                {
                       "symbol": "ONEMOREDEBTSSE",
                       "name": "One More Services",
                       "issueType" : "DEBT",
                       "subType" : "SSE",
                       "lotSize": 100,
                       "minBidQuantity": 100",
                       "faceValue": 110.00,
                       "minPrice": 212.00,
                       "maxPrice": 232.50,
                       "tickSize": 1.00,
                       "cutoffPrice": 232.50,
                       "biddingStartDate": "08-01-2019",
                       "biddingEndDate": "11-01-2019",
                       "dailyStartTime": "10:00:00",
                       "dailyEndTime": "17:00:00",
                       "isin": "INE121212129",
                       "issueSize": 50000000,
                       "registrar": "PQR Registrars",
                       "t1ModStartDate": "12-01-2019",
                       "t1ModEndDate": "12-01-2019",
                       "t1ModStartTime": "10:00:00",
                       "t1ModEndTime": "13:00:00",
                           "seriesDetails": [
                         {
                          "code" : "S1",
                           "desc": "Test1"
                         }
                              ]
    }
```


GET /<version>/allotment/{{fromtime}}/{{totime}}

This API allows user to download allotment data.

Method	GET

Non-Confidential

URL	https://<baseurl>/<version>/allotment/{{fromtime}}/{{totime}}
Request	Path Parameter
Response	JSON

Request Path Parameters

Field	Type	Description
fromtime	DateTime	Transactions entered after this time will be returned in the response
totime	DateTime	Transactions entered before this time will be returned in the response

Sample URL with Path Parameters

https://<<baseurl>>/<version>/allotment/01-08-2024%2009:30:00/07-08-2024%2011:30:00

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	Reason in case of failure
transactions	List of Objects	List Transaction details. See structure below

Transactions Detail JSON

Field	Type	Description
symbol	String (12)	Symbol
nameOfSecurity	String (250)	Name of the issue
applicationNumber	String (16)	Application No. (of the physical form)
panNumber	String (10)	PAN No.- Permanent Account Number
allotedQuantity	Number	Bid Allotted Quantity
price	Decimal	Bid Allotted Price.

Sample Response - Success


```
{
    "transactions": [
        {
            "symbol": "TESTIPO",
            "applicationNumber": "21454454",
            "price": 700.0,
            "nameOfSecurity": "Testing IPO",
            "allotedQuantity": 5,
            "panNumber": "KWWWP1234P"
        },
        {
            "symbol": "RESV01",
            "applicationNumber": "21454455",
            "price": 800.0,
            "nameOfSecurity": "ReserveCatLive",
            "allotedQuantity": 1000,
            "panNumber": "SIKKIMCATG"
        }
```


Non-Confidential

```
        ],
        "status": "success"
    }
```

Sample Response – Failed

```
    {
        "status": "failed",
        "reason": "No Records Found"
    }
```

POST /<version>/allotment/fetch

This API allows user to download transactions for the given filter criteria.

Method	POST
URL	https://<baseurl>/<version>/allotment/fetch
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String (12)	Mandatory Symbol
applicationNumber	String (16)	Optional Application Number
panNumber	String (10)	Optional Pan Number

Sample Request

```
    {
        "symbol": "TESTSMEIPO",
        "applicationNumber": "K02072024006",
        "panNumber": "KWWWP1234P"
    }
```

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	Reason in case of failure
transactions	List of Objects	List Transaction response. See Response JSON Structure for API "GET /<version>/allotment/{{time}}"

Sample Response - Success

```
    {
        "transactions": [
            {
                "symbol": "TESTIPO",
                "applicationNumber": "Testing IPO",
                "price": 700.0,
                "nameOfSecurity": "Testing IPO",
                "allotedQuantity": 5,
                "panNumber": "KWWWP1234P"
            },
            {
                "symbol": "RESV01",
                "applicationNumber": "Reserve Live",
                "price": 800.0,
```


Non-Confidential

```
                "nameOfSecurity": "ReserveCatLive",
                "allotedQuantity": 1000,
                "panNumber": "SIKKIMCATG"
            }
        ],
        "status": "success"
    }
```


Sample Response - Failed


```
    {
        "status": "failed",
        "reason": "No Records Found"
    }
```


Non-Confidential

GET /<version>/allotment/report/{{symbol}}

This is new symbol wise allotment API which will be available member/Bank admin and branch admin. It allows user to download transactions for the given filter criteria. API function in asynchronous mode. On the initial request for a given Symbol, if the allotment report is not yet available, the API returns an acknowledgement message indicating that the request is under processing "Your request received, file is processing". On subsequent requests, once the report is generated, the API returns the ZIP file for download. A rate limit of one request every 15 minutes has been implemented based on the combination of Symbol and User IP Address. This ensures that if multiple IPOs are active simultaneously, a user can place a multiple request for allotment file download, but different symbols. For the same symbol, only once a 15-minute window request will get accepted.

Method	GET
URL	https://<baseurl>/<version>/allotment/report
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String (12)	Mandatory Symbol

Sample URL with Path Parameters

```
https://<baseurl>/<version>/allotment/report/{{symbol}}
```

Initial Response Json

```
{
   "status":"success",
   "message":"Your request received, file is processing"
}
```

Subsequent Response

Headers

Content-Type: application/zip

Body

Zip file

POST/<version>/eforms/add

This API allows user to add an application which can be used for generation of eForms (PDF)

Method	POST
URL	https://<baseurl>/<version>/eforms/add
Request	JSON
Response	JSON

Request JSON

Non-Confidential

Field	Type	Description
symbol	String(12)	Symbol
depository	String(5)	Depository Name NSDL = NSDL CDSL = CDSL
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	String(16)	Client ID in case of NSDL Beneficiary ID in case of CDSL
pan	String(10)	PAN No.- Permanent Account Number
category	String(5)	Category code.
bankReferenceNo	String(60)	Input ASBA Blocking Ref No- Prefix D for Direct ASBA (Bank User) Prefix S for Syndicate ASBA (Bank User) Prefix T for 3IN1 account type bid (Member User. NON-MANDATORY for UPI bid.
bankAccountNo	String(45)	Optional Bank Account No
bankNameBranch	String(100)	Optional Bank Details. Bank Name, Branch Name
clientName	String(50)	Name of client
clientAddress	String(100)	Optional Client address
clientEmail	String(50)	Optional client email
clientTelephone	String(30)	Optional client telephone
formType	Number	Following values 1 = Non Repatriation 2 = Repatriation 3 = Employee / Shareholder
paymentType	String(4)	Following values FULL = Full payment PART = Partial payment
upiFlag	String(1)	Y = UPI Based Blocking N = Non UPI Based Blocking (ASBA)
Upi	String(45)	UPI of the investor client Mandatory and valid if upiFlag=Y
bankCode	String(6)	Optional Bank Code
locationCode	String(6)	Optional Location Code
chequeNumber	Number	Optional Cheque number Mandatory and valid if nonASBA = true
nonASBA	Boolean	Flag to indicate application is Non ASBA. Applicable only for DEBT segment true – Non ASBA false – ASBA (default)
syndMemCode	String(60)	Optional Syndicate Member Stamp Code
brokerCode	String(60)	Optional Broker Stamp Code
subBrokerCode	String(60)	Optional Sub Broker / Sub Agent Stamp Code

Non-Confidential

Bids	List of Objects	List of bid details. See structure below. Maximum 3 allowed

Bid Detail JSON

Field	Type	Description
quantity	Number	Optional Bid Quantity
atCutOff	Boolean	Cut off Indicator true - if opting for cut off false- if not opting for cut off
price	Decimal	Bid Price. Mandatory and valid only if "atCutOff" = false. Should be greater than 0 and maximum 2 decimal places.
amount	Decimal	Bid Amount. Should be greater than 0 and maximum 2 decimal places.

Sample Request

```
{
   "symbol": "TEST",
   "depository": "NSDL",
   "dpId": "33445566",
   "clientBenId": "12345678",
   "pan": "AFAKA2323L",
   "category": "IND",
   "bankReferenceNo": "BANKREF00001",
   "bankAccountNo": "121212121212",
   "bankNameBranch": "XYZ Bank, PQR Branch",
   "clientName": "Sumit Aggarwal",
   "clientAddress" : "Lane no 1, Building A, nth Floor, Some Locality,
Some City",
   "clientEmail" : "someemail@somedomain.com",
   "clientTelephone" : "12121212",
   "formType" : 1,
   "paymentType" : "FULL",
   "nonASBA": false,
   "upiFlag": "Y",
   "upi": "9393939393@upi",
   "bankCode": null,
   "locationCode" : null,
   "syndMemCode": "SYNDCODE001",
   "brokerCode": "BRKSTAMP001",
   "subBrokerCode": "SUBBROKERSTAMP001",
   "bids": [
         {
               "quantity": 100,
               "atCutOff": false,
               "price": 55.30,
               "amount": 5530.00
         },
         {
               "quantity": 110,
```


Non-Confidential

```
                    "atCutOff": true,
                    "amount": 5863.00
                }
            ]
        }
```


Response JSON

Field	Type	Description
symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
depository	String(5)	Depository Name NSDL = NSDL CDSL = CDSL
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	String(16)	Client ID in case of NSDL Beneficiary ID in case of CDSL
pan	String(10)	PAN No.- Permanent Account Number
category	String(5)	Category code.
bankReferenceNo	String(60)	Input ASBA Blocking Ref No- Prefix D for Direct ASBA (Bank User) Prefix S for Syndicate ASBA (Bank User) Prefix T for 3IN1 account type bid (Member User. NON-MANDATORY for UPI bid.
bankAccountNo	String(45)	Optional Bank Account No
bankNameBranch	String(100)	Optional Bank Details. Bank Name, Branch Name
clientName	String(50)	Name of client
clientAddress	String(100)	Optional Client address
clientEmail	String(50)	Optional client email
clientTelephone	String(30)	Optional client telephone
formType	Number	Following values 1 = Non Repatriation 2 = Repatriation 3 = Employee / Shareholder
paymentType	String(4)	Following values FULL = Full payment PART = Partial payment
upiFlag	String(1)	Y = UPI Based Blocking N = Non UPI Based Blocking (ASBA)
Upi	String(45)	UPI of the investor client Mandatory and valid if logged in user is a bank and upiFlag=Y
bankCode	String(6)	Optional Bank Code
locationCode	String(6)	Optional Location Code
chequeNumber	Number	Optional Cheque number
nonASBA	Boolean	Flag to indicate application is Non ASBA. Applicable only for DEBT segment true – Non ASBA

Non-Confidential

		false – ASBA (default)
syndMemCode	String(60)	Optional Syndicate Member Stamp Code
brokerCode	String(60)	Optional Broker Stamp Code
subBrokerCode	String(60)	Optional Sub Broker / Sub Agent Stamp Code
Bids	List of Objects	List of bid details. See structure below. Maximum 3 allowed

Bid Detail JSON

Field	Type	Description
quantity	Number	Optional Bid Quantity
atCutOff	Boolean	Cut off Indicator true - if opting for cut off false- if not opting for cut off
Price	Decimal	Bid Price. Mandatory and valid only if "atCutOff" = false. Should be greater than 0 and maximum 2 decimal places.
amount	Decimal	Bid Amount. Should be greater than 0 and maximum 2 decimal places.

Sample Response


```
{
   "symbol": "TEST",
   "applicationNumber": "1231231231123",
   "depository": "NSDL",
   "dpId": "33445566",
   "clientBenId": "12345678",
   "pan": "AFAKA2323L",
   "category": "IND",
   "bankReferenceNo": "BANKREF00001",
   "bankAccountNo": "121212121212",
   "bankNameBranch": "XYZ Bank, PQR Branch",
   "clientName": "Sumit Aggarwal",
   "clientAddress" : "Lane no 1, Building A, nth Floor, Some Locality,
Some City",
   "clientEmail" : "someemail@somedomain.com",
   "clientTelephone" : "12121212",
   "formType" : 1,
      "chequeNumber": "",
   "paymentType" : "FULL",
   "nonASBA": false,
   "upiFlag": "Y",
   "upi": "9393939393@upi",
   "bankCode": null,
   "locationCode" : null,
   "syndMemCode": "SYNDCODE001",
   "brokerCode": "BRKSTAMP001",
   "subBrokerCode": "SUBBROKERSTAMP001",
   "bids": [
         {
               "quantity": 100,
               "atCutOff": false,
```


Non-Confidential

```
                    "price": 55.30,
                    "amount": 5530.00
                },
                {
                    "quantity": 110,
                    "atCutOff": true,
                    "amount": 5863.00
                }
        ]
    }
```

Sample Response - Failed

```
    {
        "status": "failed",
        "reason": "Xyz"
    }
```

Detail reason list-

Sr.no	Rejection scenarios	Status	Reason
1	Symbol length exceeds specified limit	Fail	Symbol exceeds maximum character length of 12.
2	Symbol not provided in request	Fail	Symbol is mandatory
3	Invalid user	Fail	User not found with ID:**<UserID>**
4	Symbol received in Api request is invalid	Fail	Invalid or Incorrect Symbol
5	Api request exceeds specified limit	Fail	API limit reached
6	Allotment details not available	Fail	Record not found for symbol and user
7	Initial request for allotment {<symbol>}	Success	Your request received, file is processing, please initiate download after 15 mins.
8	Subsequent request for allotment {<symbol>}	Success	Download zip file in application/zip

POST/<version>/eforms/query

This API allows user to query uploaded eForms data based on input filters

Method	POST
URL	https://<baseurl>/<version>/eforms/query
Request	JSON

Non-Confidential

Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Mandatory Symbol
fromApplicationNumber	String(16)	Mandatory. Response will include all applications from this application number onwards (inclusive)
toApplicationNumber	String(16)	Optional. Response will include all applications upto this number (inclusive)

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
eforms	List of Objects	List eForms. See Response JSON Structure for API "POST /<version>/eforms/add"

Sample Response - Success


```
{
   "status": "success",
   "eforms" : [
         {
               "symbol": "TEST",
               "applicationNumber": "1231231231120",
               "depository": "NSDL",
               "dpId": "33445566",
               "clientBenId": "12345678",
               "pan": "AFAKA2323L",
               "category": "IND",
               "bankReferenceNo": "BANKREF00001",
               "bankAccountNo": "121212121212",
               "bankNameBranch": "XYZ Bank, PQR Branch",
               "clientName": "Sumit Aggarwal",
               "clientAddress" : "Lane no 1, Building A, nth Floor, Some
Locality, Some City",
               "clientEmail" : "someemail@somedomain.com",
               "clientTelephone" : "12121212",
               "formType" : 1,
               "paymentType" : "FULL",
               "nonASBA": false,
               "upiFlag": "Y",
               "upi": "9393939393@upi",
               "bankCode": null,
               "locationCode" : null,
               "syndMemCode": "SYNDCODE001",
               "brokerCode": "BRKSTAMP001",
               "subBrokerCode": "SUBBROKERSTAMP001",
               "bids": [
                     {
                           "quantity": 100,
                           "atCutOff": false,
                           "price": 55.30,
                           "amount": 5530.00
                     },
                     {
                           "quantity": 110,
```


Non-Confidential

```
                              "atCutOff": true,
                              "amount": 5863.00
                        }
                  ]
            },
            {
                  "symbol": "TEST",
                  "applicationNumber": "1231231231123",
                  "depository": "NSDL",
                  "dpId": "33445566",
                  "clientBenId": "12345678",
                  "pan": "AFAKA2323L",
                  "category": "IND",
                  "bankReferenceNo": "BANKREF00001",
                  "bankAccountNo": "121212121212",
                  "bankNameBranch": "XYZ Bank, PQR Branch",
                  "clientName": "Sumit Aggarwal",
                  "clientAddress" : "Lane no 1, Building A, nth Floor, Some
   Locality, Some City",
                  "clientEmail" : "someemail@somedomain.com",
                  "clientTelephone" : "12121212",
                  "formType" : 1,
                  "paymentType" : "FULL",
                  "nonASBA": false,
                  "bankCode": "9999",
                  "locationCode" : "NASBAL",
                  "syndMemCode": "SYNDCODE001",
                  "brokerCode": "BRKSTAMP001",
                  "subBrokerCode": "SUBBROKERSTAMP001",
                  "bids": [
                        {
                              "quantity": 100,
                              "atCutOff": false,
                              "price": 55.30,
                              "amount": 5530.00
                        },
                        {
                              "quantity": 110,
                              "atCutOff": true,
                              "amount": 5863.00
                        }
                  ]
            }
      ]
   }
```


Sample Response - Failed


```
   {
      "status": "failed",
      "reason": "Xyz"
   }
```


GET /<version>/ncbmaster

This API allows user to download NCB master.

Method	GET
URL	https://<baseurl>/<version>/ncbmaster
Request	None
Response	JSON

Non-Confidential

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
data	List of Objects	List of NCB Master objects

NCBMaster Object JSON Structure

Field	Type	Description
symbol	String(12)	Symbol
series	String(2)	Series
name	String(250)	Name of the issue
lotSize	Number	Bidding Lot Size
faceValue	Decimal	Face value
minBidQuantity	Number	Optional Minimum Quantity for Bidding
minPrice	Decimal	Optional Minimum Price
maxPrice	Decimal	Optional Maximum Price
tickSize	Decimal	Tick Size
cutoffPrice	Decimal	Cutoff price
biddingStartDate	Date	First Date of bidding
biddingEndDate	Date	Last Date of normal bidding
dailyStartTime	Time	Bidding start time for the current day
dailyEndTime	Time	Bidding end time for the current day
t1ModStartDate	Date	First Date of T1 modification
t1ModEndDate	Date	Last Date of T1 modification
t1ModStartTime	Time	Start time for T1 modification
t1ModEndTime	Time	End time for T1 modification
isin	String(12)	Isin
issueSize	Number	Issue size
issueValueSize	Number	Issue value size
maxQuantity	Number	Max quantity
allotmentDate	DATE	Allotment date
lastDayBiddingEndTime	DATETIME	Last day bidding enddate//time

Sample Response


```
Success: {
  "data": [
    {
      "biddingStartDate": "01-11-2018",
      "symbol": "TEST",
      "minBidQuantity": 10,
      "maxQuantity": 20000000,
      "lotSize": 10,
      "t1ModEndDate": "02-11-2018",
      "dailyStartTime": "15:30:00",
      "allotmentDate": "",
      "t1ModStartTime": "20:30:00",
      "biddingEndDate": "10-12-2018",
      "t1ModEndTime": "21:00:00",
      "dailyEndTime": "11:30:00",
      "tickSize": 100,
      "cutoffPrice": "0205GS",
      "series": "GS",
      "faceValue": 1000,
      "minPrice": 0,
      "t1ModStartDate": "02-11-2018",
```


Non-Confidential

```
        "issueValueSize": 0,
        "name": "0205GS",
        "issueSize": 100000000,
        "lastDayBiddingEndTime": "",
        "maxPrice": 0,
        "isin": "IN0020150044"
      },
      {
        "biddingStartDate": "01-02-2017",
        "symbol": "TEST1",
        "minBidQuantity": 10000,
        "maxQuantity": 20000000,
        "lotSize": 10000,
        "t1ModEndDate": "03-07-2019",
        "dailyStartTime": "",
        "allotmentDate": "",
        "t1ModStartTime": "",
        "biddingEndDate": "02-07-2019",
        "t1ModEndTime": "",
        "dailyEndTime": "",
        "tickSize": 0,
        "cutoffPrice": "078GS2030",
        "series": "01",
        "faceValue": 10000,
        "minPrice": 0,
        "t1ModStartDate": "03-07-2019",
        "issueValueSize": 0,
        "name": "7.88% CG2030",
        "issueSize": 100000000,
        "lastDayBiddingEndTime": "",
        "maxPrice": 0,
        "isin": "IN0020150028"
      },
      {
        "biddingStartDate": "13-01-2017",
        "symbol": "TEST2",
        "minBidQuantity": 100,
        "maxQuantity": 100000,
        "lotSize": 100,
        "t1ModEndDate": "03-07-2019",
        "dailyStartTime": "09:00:00",
        "allotmentDate": "",
        "t1ModStartTime": "07:00:00",
        "biddingEndDate": "02-07-2019",
        "t1ModEndTime": "13:00:00",
        "dailyEndTime": "23:00:00",
        "tickSize": 10000,
        "cutoffPrice": "625CG2022",
        "series": "GS",
        "faceValue": 10000,
        "minPrice": 0,
        "t1ModStartDate": "03-07-2019",
        "issueValueSize": 0,
        "name": "6.25",
        "issueSize": 100000000,
        "lastDayBiddingEndTime": "",
        "maxPrice": 0,
        "isin": "INE00124571A"
      }
    ],
    "status": "success"
  }
```

POST/<version>/ncb/add

This API allows user to add / modify / cancel an application

Method	POST
URL	https://<baseurl>/<version>/ncb/add
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Symbol
investmentValue	Number	Invested Value
applicationNumber	String(16)	
orderNumber	Number	In case of Modification and Deletion only
price	Decimal	Bid Price.
physicalDematFlag	String(1)	D= Demat
pan	String(10)	PAN No.- Permanent Account Number
depository	String(4)	Depository Name (NSDL, CDSL)
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
clientRefNumber	String(16)	Reference Number- Internal reference number
activityType	String(1)	N=New M=Modify D=Delete

Sample Request

```
{
  "symbol": "TEST",
  "investmentValue": 100,
  "applicationNumber": "1200299929020",
  "price": 55,
  "physicalDematFlag": "D",
  "pan": "AFAKA2323L",
  "depository": "NSDL",
  "dpId": "33445566",
  "clientBenId": "12345678",
  "activityType": "N",
  "clientRefNumber": "MYREF0001",
}
```

Response JSON

Field	Type	Description
symbol	String(12)	Symbol
orderNumber	Number	Order Number
series	String(2)	Series

Non-Confidential

applicationNumber	String(16)	Application No.(of the physical form)
investmentValue	Number	invested Quantity
price	Decimal	Bid Price.
totalAmountPayable	Decimal	Total Amount Payable
physicalDematFlag	String(1)	D= Demat
pan	String(10)	PAN No.- Permanent Account Number
depository	String(4)	Depository Name (NSDL, CDSL)
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
clientRefNumber	String(16)	Reference Number- Internal reference number
orderStatus	String(2)	ES = Entry success EF = Entry Fail MS = Modify Success MF = Modify Fail CS = Cancel Success CF = Cancel Fail
rejectionReason	String(100)	Rejection reason
enteredBy	String(25)	Enter By
entryTime	DATETIME	Entry time
verificationStatus	String(1)	P = Pending S = Verified F= Verification failed
verificationReason	String(100)	Verification Reason
clearingStatus	String(2)	FP = Fund Pending FS = Sent to NSCCL RJ = Rejected AL = Allotted
clearingReason	String(100)	Clearing Reason
lastActionTime	DATETIME	Last action time
status	String	success/failed
reason	String(100)	Reason text

Sample Response

```
Success: {
    "symbol": "TEST",
    "orderNumber": 2019042500000003,
    "series": "GS",
    "applicationNumber": "1200299929020",
    "investmentValue": 100,
    "price": 10500,
    "totalAmountPayable": 10500,
    "physicalDematFlag": "D",
    "pan": "AFAKA2323L",
    "depository": "NSDL",
    "dpId": "33445566",
    "clientBenId": "12345678",
    "clientRefNumber": "MYREF0001",
    "orderStatus ": "ES",
    "rejectionReason ": null,
    "enteredBy ": "samir",
```


Non-Confidential

```
        "entryTime ": "25-04-2019 12:39:01",
        "verificationStatus ": "P",
        "verificationReason ": null,
        "clearingStatus ": "FP",
        "clearingReason ": "",
        "LastActionTime ": "25-04-2019 12:39:01",
        "status" : "success"
    }
```


GET /<version>/ncb/{{time}}

This API allows user to download NCB transactions.

Method	GET
URL	https://<baseurl>/<version>/ncb/{{time}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
time	DateTime	Cutoff date time. Transactions entered after this time will be returned in the response

Sample URL with Path Parameters

https://www.nseindiaipo.com/eipo/<version>/ncb/01-12-2015%2009:30:00

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction responses. See Response JSON Structure for API "POST /<version>/ncb/add"

Sample Response - Success


```
    Success: {
      "transactions": [
        {
          "symbol": "TEST",
          "clearingStatus": "FP",
          "orderNumber": 2017011300000001,
          "depository": "NSDL",
          "clearingReason": "",
          "applicationNumber": "TMNSEIL000000001",
          "verificationStatus": "P",
          "physicalDematFlag": "D",
          "LastActionTime": "13-01-2017 14:57:16",
          "dpId": "IN301245",
          "orderStatus": "ES",
          "enteredBy": "NSEIL",
          "entryTime": "13-01-2017 14:57:16",
          "investmentValue": 100,
          "series": "GS",
          "price": 0,
          "totalAmountPayable": 0,
          "clientRefNumber": "",
```


Non-Confidential

```
        "pan": "AISPG3152H",
        "rejectionReason": "",
        "clientBenId": "31412547",
        "verificationReason": "",
        "status": "success"
      },
      {
        "symbol": "TEST2",
        "clearingStatus": "FP",
        "orderNumber": 2017011300000003,
        "depository": "CDSL",
        "clearingReason": "",
        "applicationNumber": "TMNSEIL000000003",
        "verificationStatus": "P",
        "physicalDematFlag": "D",
        "LastActionTime": "13-01-2017 14:59:24",
        "dpId": "",
        "orderStatus": "ES",
        "enteredBy": "NSEIL",
        "entryTime": "13-01-2017 14:59:24",
        "investmentValue": 1200,
        "series": "GS",
        "price": 0,
        "totalAmountPayable": 0,
        "clientRefNumber": "",
        "pan": "AISPG3152O",
        "rejectionReason": "",
        "clientBenId": "1234567898741236",
        "verificationReason": "",
        "status": "success"
       }
    ],
    "status": "success"
  }
```


Sample Response – Failed


```
  {
     "status": "failed",
     "reason": "No Records Found"
  }
```


POST /<version>/ncb/fetch

This API allows user to download NCB transactions for the given filter criteria.

Method	POST
URL	https://<baseurl>/<version>/ncb /fetch
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Mandatory Symbol
applicationNumber	String(16)	Mandatory Application No.(of the physical form)

Sample Request

Non-Confidential

```
{"symbol": "TEST","applicationNumber": "1200299929020"}
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction response. See Response JSON Structure for API "POST /<version>/ncb/add"

Sample Response - Success

```
Success: {
  "transactions": [
 {
   "symbol": "TEST",
   "clearingStatus": "FP",
   "orderNumber": 2019042500000003,
   "depository": "CDSL",
   "clearingReason": "",
   "applicationNumber": "1200299929020",
   "verificationStatus": "P",
   "physicalDematFlag": "D",
   "LastActionTime": "16-01-2017 12:17:52",
   "dpId": "",
   "orderStatus": "ES",
   "enteredBy": "NSEIL",
   "entryTime": "16-01-2017 12:17:52",
   "investmentValue": 20000,
   "series": "GS",
   "price": 10600,
   "totalAmountPayable": 2120000,
   "clientRefNumber": "",
   "pan": "ANPPM5867P",
   "rejectionReason": "",
   "clientBenId": "1201090000365411",
   "verificationReason": "",
   "status": "success"
    }
 ],
 "status": "success"
}
```

Sample Response - Failed

```
{
   "status": "failed",
   "reason": "No Records Found"
}
```

GET /<version>/ncb/mismatches/{{symbol}}

This API allows user to download mismatches.

Method	GET
URL	https://<baseurl>/<version>/ncb/mismatches/{{symbol}}
Request	Path Parameters

Non-Confidential

Response	JSON

Request Path Parameters

Field	Type	Description
Symbol	String(12)	Issue Symbol

Sample URL with Path Parameters

https://<baseurl>/eipo/<version>/ncb/mismatches/TESTSYM01

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
mismatches	List of Objects	List of Mismatch objects

Mismatch Object JSON Structure

Field	Type	Description
Symbol	String(12)	Symbol
applicationNumber	String(16)	
orderNumber	Number	In case of Modification and Deletion only
physicalDematFlag	String(1)	D= Demat
pan	String(10)	PAN No.- Permanent Account Number
depository	String(4)	NSDL CDSL
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	String(16)	Client ID in case of NSDL Beneficiary ID in case of CDSL
reason	String(250)	Reason text

Sample Response


```
{
   "status": "success",
   "notifications": [
         {
                "symbol": "TEST",
                "orderNumber": 2019042500000003,
                "applicationNumber": "1200299929020",
                "physicalDematFlag": "D"
                "depository": "NSDL",
                "dpId": "33445566",
                "clientBenId": "12345678",
                "pan": "AFAKA2323L",
                "reason" : "Account does not exist"
            },
         {
               "symbol": "TEST",
                  "orderNumber": 2019042500000004,
                  "applicationNumber": "1200299929025",
                  "physicalDematFlag": "D",
                  "depository": "NSDL",
                  "dpId": "33445566",
                  "clientBenId": "12345679",
                  "pan": "AFAKA2326L",
```


Non-Confidential

```
                    "reason" : "Account does not exist"
                }
            ]
        }
```

GET /<version>/tendermaster

This API allows user to download tender master.

Method	GET
URL	https://<baseurl>/<version>/tendermaster
Request	None
Response	JSON

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
data	List of Objects	List of tender Master objects

TenderMaster Object JSON Structure

Field	Type	Description
symbol	String(12)	Symbol
series	String(2)	Series
name	String(250)	Name of the issue
lotSize	Number	Bidding Lot Size
faceValue	Decimal	Face value
minBidQuantity	Number	Optional Minimum Quantity for Bidding
minPrice	Decimal	Optional Minimum Price
maxPrice	Decimal	Optional Maximum Price
tickSize	Decimal	Tick Size
cutoffPrice	Decimal	Cutoff price
biddingStartDate	Date	First Date of bidding
biddingEndDate	Date	Last Date of normal bidding
dailyStartTime	Time	Bidding start time for the current day
dailyEndTime	Time	Bidding end time for the current day
registrar	String(100)	Name of registrar
isin	String(12)	Isin
issueSize	Number	Issue size
issueValueSize	Number	Issue value size
maxQuantity	Number	Max quantity

Sample Response

```
{
   "status": "success",
   "data": [{
         "symbol": "TEST",
                     "series": "BB",
         "name": "Test Industries",
         "lotSize": 1,
         "faceValue": 10.00,
         "minBidQuantity": "100 ",
         "minPrice": 12.00,
```


Non-Confidential

```
            "maxPrice": 12.50,
            "tickSize": 0.05,
            "cutoffPrice": 12.50,
            "biddingStartDate": "01-01-2019",
            "biddingEndDate": "03-01-2019",
            "dailyStartTime": "10:00:00",
            "dailyEndTime": "17:00:00",
            "isin": "INE121212121",
            "issueSize": 150000000,
            "issueValueSize": 0.00,
            "registrar": "XYZ Registrars",
            "maxQuantity": 4563,
        }]
    }
```


POST/<version>/tender/add

This API allows user to add / modify / cancel an application in tender.

Method	POST
URL	https://<baseurl>/<version>/tender/add
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Symbol
series	String(2)	Series
applicationNumber	String(20)	Application No.(of the physical form)
category	String(5)	Category code.
bidQuantity	Number	Bid Quantity
price	Decimal	Bid Price. Mandatory and valid only if "atCutOff" = false. Should be greater than 0 and maximum 2 decimal places.
clientName	String(50)	Optional Client Name – Name of client
clientCode	String(10)	Client Code
cpCode	String(12)	Cp Code
physicalDematFlag	String(1)	P= physical D= Demat
depository	String(5)	Depository Name (NSDL, CDSL)
pan	String(10)	PAN No.- Permanent Account Number
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
folioNo	String(16)	Folio No
certificate1	String(16)	Certificate 1
distinctiveFrom1	String(16)	Distinctive From 1
distinctiveTo1	String(16)	Distinctive To 1
certificate2	String(16)	Certificate 2
distinctiveFrom2	String(16)	Distinctive From 2
distinctiveTo2	String(16)	Distinctive To 2

Non-Confidential

certificate3	String(16)	Certificate 3
distinctiveFrom3	String(16)	Distinctive From 3
distinctiveTo3	String(16)	Distinctive To 3
reference	String(16)	Reference Number- Internal reference number
orderNumber	Number	Order number
activityType	String(2)	ER = Entry request MR = Modify Request CR = Cancel Request

Sample Request

```
{
      "symbol": "TEST",
      " series ": "BB",
      "applicationNumber": "1200299929020",
      "category": "IND",
   "bidQuantity": 100,
   "price": 55.30,
   "clientName": "Sumit Aggarwal",
   "clientCode": "326Q2",
   "cpCode": "CITI12345",
   "physicalDematFlag": "P",
   "depository": null,
   "pan": "AFAKA2323L",
   "dpId": null,
   "clientBenId": null,
   "folioNo":"ABF001",
   "certificate1": "21035",
   "distinctiveFrom1": "10015",
   "distinctiveTo1": "100125",
   "referenceNumber": "MYREF0001",
   "orderNumber" : "20190406",
   "activityType" : "ER"
}
```

IN CASE PHYSICAL/DEMATFLAG IS DEMAT

```
{
   "symbol": "TEST",
   "series": "BB",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "bidQuantity": 100,
   "price": 55.30,
   "clientName": "Sumit Aggarwal",
   "clientCode": "326Q2",
   "cpCode": "CITI12345",
   "physicalDematFlag": "D",
   "depository": "NSDL",
   "pan": "AFAKA2323L",
   "dpId": "33445566",
   "clientBenId": "12345678",
   "folioNo":null,
   "certificate1": null,
   "distinctiveFrom1": null,
   "distinctiveTo1": null,
   "referenceNumber": "MYREF0001",
   "orderNumber" : "20190406",
   "activityType" : "ER"
}
```


Non-Confidential

Response JSON

Field	Type	Description
symbol	String(12)	Symbol
series	String(2)	Series
applicationNumber	String(20)	Application No.(of the physical form)
category	String(5)	Category code.
bidQuantity	Number	Bid Quantity
price	Decimal	Bid Price. Mandatory and valid only if "atCutOff" = false. Should be greater than 0 and maximum 2 decimal places.
clientName	String(50)	Optional Client Name – Name of client
clientCode	String(10)	Client Code
cpCode	String(12)	Cp Code
physical/Demat Flag	String(1)	P= physical D= Demat
depository	String(5)	Depository Name (NSDL, CDSL)
pan	String(10)	PAN No.- Permanent Account Number
dpId	String(8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
folioNo	String(16)	Folio No
certificate1	String(16)	Certificate 1
distinctiveFrom1	String(16)	Distinctive From 1
distinctiveTo1	String(16)	Distinctive To 1
certificate2	String(16)	Certificate 2
distinctiveFrom2	String(16)	Distinctive From 2
distinctiveTo2	String(16)	Distinctive To 2
certificate3	String(16)	Certificate 3
distinctiveFrom3	String(16)	Distinctive From 3
distinctiveTo3	String(16)	Distinctive To 3
orderStatus	String (2)	ES = Entry success EF = Entry Fail MS = Modify Success MF = Modify Fail CS = Cancel Success CF = Cancel Fail
reference	String(16)	Reference Number- Internal reference number
status	String	success/failed
reasonCode	Number	Fail reason code. Valid only if "status" = "failed"
reason	String(100)	Reason text

Sample Response

Non-Confidential

```
{
                "symbol": "TEST",
                " series ": "BB",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "bidQuantity": 100,
   "price": 55.30,
   "reason": "Orders not allowed during this time of day. ",
   "clientName": "Sumit Aggarwal",
   "clientCode": "326Q2",
   "cpCode": "CITI12345",
   "physical/DematFlag": "P",
   "depository": null,
   "pan": "AFAKA2323L",
   "dpId": null,
   "clientBenId": null,
   " folioNo":"ABF001",
   " certificate1": "21035",
   " distinctiveFrom1": "10015",
   " distinctiveTo1": "100125",
   "referenceNumber": "MYREF0001",
   "orderNumber" : "20190406",
   "rejectionReason": "Orders not allowed during this time of day. ",
   "entryTime": "2026-09-23T13:20:07.163+00:00",
   "activityType" : "ER",
   "status" : "success",
   "rtaStatus": "NA",
   "enteredBy": "",
   "rtaRejectionReason": null,
   "amount": "100000.00",
   "custodian": null,
   "custodianStatus": "NA"

}
```


IN CASE PHYSICAL/DEMATFLAG IS DEMAT


```
{
                "symbol": "TEST",
               " series ": "BB",
   "applicationNumber": "1200299929020",
   "category": "IND",
   "bidQuantity": 100,
   "price": 55.30,
   "reason": "Orders not allowed during this time of day. ",
   "clientName": "Sumit Aggarwal",
   "clientCode": "326Q2",
   "cpCode": "CITI12345",
   " physical/DematFlag": "D",
   "depository": "NSDL",
   "pan": "AFAKA2323L",
   "dpId": "33445566",
   "clientBenId": "12345678",
   " folioNo":null,
   " certificate1": null,
   " distinctiveFrom1": null,
   " distinctiveTo1": null,
   "referenceNumber": "MYREF0001",
   "orderNumber" : "20190406",
   "rejectionReason": "Orders not allowed during this time of day. ",
   "entryTime": "2026-09-23T13:20:07.163+00:00",
   "activityType": "ER",
```


Non-Confidential

```
        "status" : "success",
        "rtaStatus": "NA",
        "enteredBy": "",
        "rtaRejectionReason": null,
        "amount": "100000.00",
        "custodian": null,
        "custodianStatus": "NA"

    }
```


GET /<version>/tender/{{time}}

This API allows user to download transactions entered after a particular time.

Method	GET
URL	https://<baseurl>/<version>/tender/{{time}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
time	DateTime	Cutoff date time. Transactions entered after this time will be returned in the response

Sample URL with Path Parameters

```
https://<baseurl>/<version>/tender/01-12-2015%2009:30:00
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction responses. See Response JSON Structure for API "POST /<version>/tender/add"

Sample Response - Success


```
    {
        "symbol": "TEST",
        "series ": "BB",
        "applicationNumber": "1200299929020",
        "category": "IND",
        "bidQuantity": 100,
        "price": 55.30,
        "clientName": "Sumit Aggarwal",
        "clientCode": "326Q2",
        "cpCode": "CITI12345",
        "physicalDematFlag": "P",
        "depository": null,
        "pan": "AFAKA2323L",
        "dpId": null,
        "clientBenId": null,
        "folioNo":"ABF001",
        "certificate1": "21035",
        "distinctiveFrom1": "10015",
        "distinctiveTo1": "100125",
```


Non-Confidential

```
    "referenceNumber": "MYREF0001",
    "orderNumber" : "20190406",
    "rtaStatus": "NA",
    "enteredBy": "NSEIL",
    "rtaRejectionReason": "",
    "amount": 47500.0,
    "custodian": "",
    "custodianStatus": "NA",
    "entryTime": "24-09-2026 10:35:27",
    "series": "BB",
    "Issuetype": 7,
    "category": "IND",
    "rejectionReason": "",
    "activityType" : "ER",
    "status" : "success"
}
```


IN CASE PHYSICAL/DEMATFLAG IS DEMAT


```
{
    "symbol": "TEST",
    "series ": "BB",
    "applicationNumber": "1200299929020",
    "category": "IND",
    "bidQuantity": 100,
    "price": 55.30,
    "clientName": "Sumit Aggarwal",
    "clientCode": "326Q2",
    "cpCode": "CITI12345",
    "physicalDematFlag": "D",
    "depository": "NSDL",
    "pan": "AFAKA2323L",
    "dpId": "33445566",
    "clientBenId": "12345678",
    "folioNo":null,
    "certificate1": null,
    "distinctiveFrom1": null,
    "distinctiveTo1": null,
    "referenceNumber": "MYREF0001",
    "orderNumber" : "20190406",
    "rtaStatus": "NA",
    "enteredBy": "NSEIL",
    "rtaRejectionReason": "",
    "amount": 47500.0,
    "custodian": "",
    "custodianStatus": "NA",
    "entryTime": "24-09-2026 10:35:27",
    "series": "BB",
    "Issuetype": 7,
    "category": "IND",
    "rejectionReason": "",
    "activityType" : "ER",
    "status" : "success"
}
```


Sample Response – Failed


```
{
    "status": "failed",
    "reason": "No Records Found"
}
```


Non-Confidential

POST /<version>/tender/fetch

This API allows user to download tender transactions for the given filter criteria.

Method	POST
URL	https://<baseurl>/<version>/tender/fetch
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String(12)	Mandatory Symbol
applicationNumber	String(20)	Mandatory Application No.(of the physical form)

Sample Request

```
{"symbol": "TEST","applicationNumber": "1200299929020"}
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	Reason in case of failure
transactions	List of Objects	List Transaction response. See Response JSON Structure for API "POST /<version>/tender/add"

Sample Response - Success


```
{
   "status": "success",
   "transactions" : [
         {
               "symbol": "TEST",
               "series ": "BB",
               "applicationNumber": "1200299929020",
               "category": "IND",
               "bidQuantity": 100,
               "price": 55.30,
               "clientName": "Sumit Aggarwal",
               "clientCode": "326Q2",
               "cpCode": "CITI12345",
               "physicalDematFlag": "P",
               "depository": null,
               "pan": "AFAKA2323L",
               "dpId": null,
               "clientBenId": null,
               "folioNo":"ABF001",
               "certificate1": "21035",
               "distinctiveFrom1": "10015",
               "distinctiveTo1": "100125",
               "referenceNumber": "MYREF0001",
               "orderNumber" : "20190406",
               "rtaStatus": "NA",
               "enteredBy": "NSEIL",
               "rtaRejectionReason": "",
               "amount": 9500.0,
               "custodian": "",
               "custodianStatus": "NA",
```


Non-Confidential

```
                "entryTime": "16-09-2026 13:01:03",
                "Issuetype": 7,
                "category": "IND",
                "rejectionReason": "",
                "activityType" : "ER",
                "status" : "success"
            }
    ]
}
```


IN CASE PHYSICAL/DEMATFLAG IS DEMAT


```
{
    "status": "success",
    "transactions" : [{
                "symbol": "TEST",
                " series ": "BB",
    "applicationNumber": "1200299929020",
    "category": "IND",
    "bidQuantity": 100,
    "price": 55.30,
    "clientName": "Sumit Aggarwal",
    "clientCode": "326Q2",
    "cpCode": "CITI12345",
    "physicalDematFlag": "D",
    "depository": "NSDL",
    "pan": "AFAKA2323L",
    "dpId": "33445566",
    "clientBenId": "12345678",
    "folioNo":null,
    "certificate1": null,
    "distinctiveFrom1": null,
    "distinctiveTo1": null,
    "referenceNumber": "MYREF0001",
    "orderNumber" : "20190406",
    "rtaStatus": "NA",
    "enteredBy": "NSEIL",
    "rtaRejectionReason": "",
    "amount": 9500.0,
    "custodian": "",
    "custodianStatus": "NA",
    "entryTime": "16-09-2026 13:01:03",
    "Issuetype": 7,
    "category": "IND",
    "rejectionReason": "",
    "activityType" : "ER",
    "status" : "success"
            }
    ]
}
```


Sample Response - Failed


```
 {
    "status": "failed",
    "reason": "No Records Found"
 }
```


GET /<version>/sgbmaster

This API allows user to download SGB master.

Method	GET

Non-Confidential

URL	https://<baseurl>/<version>/sgbmaster
Request	NONE
Response	JSON

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	reason
data	List of Objects	List of SGB Master objects

SGBMaster Object JSON Structure

Field	Type	Description
symbol	String (12)	Symbol
series	String (2)	Series
name	String (250)	Name of the issue
issueType	String (10)	Type of Issue.
lotSize	Number	Bidding Lot Size
faceValue	Decimal	Face value
minBidQuantity	Number	Optional Minimum Quantity for Bidding
minPrice	Decimal	Minimum Price will be considered as Online Price
maxPrice	Decimal	Maximum Price will be considered as Offline Price
tickSize	Decimal	Tick Size
biddingStartDate	Date	First Date of bidding
biddingEndDate	Date	Last Date of normal bidding
dailyStartTime	Time	Bidding start time for the current day
dailyEndTime	Time	Bidding end time for the current day
t1ModStartDate	Date	First Date of T1 modification
t1ModEndDate	Date	Last Date of T1 modification
t1ModStartTime	Time	Start time for T1 modification
t1ModEndTime	Time	End time for T1 modification
Isin	String (12)	Isin
issueSize	Number	Issue size
issueValueSize	Number	Issue value size
maxQuantity	Number	Max quantity
allotmentDate	DATE	Allotment date
incompleteModEndDate	DATE	Incomplete Modification end date

Sample Response


```
Success: {
  "data": [
    {
      "biddingStartDate": "01-11-2018",
      "symbol": "TEST",
      "issueType": "GB",
      "minBidQuantity": 10,
      "maxQuantity": 20000000,
      "lotSize": 10,
      "t1ModEndDate": "02-11-2018",
      "dailyStartTime": "15:30:00",
```


Non-Confidential

```
        "allotmentDate": "",
        "t1ModStartTime": "20:30:00",
        "biddingEndDate": "10-12-2018",
        "t1ModEndTime": "21:00:00",
        "dailyEndTime": "11:30:00",
        "tickSize": 100,
        "series": "GB",
        "faceValue": 1000,
        "minPrice": 0,
        "t1ModStartDate": "02-11-2018",
        "issueValueSize": 0,
        "name": "Sovereign Gold Bond",
        "issueSize": 100000000,
        "incompleteModEndDate ": "",
        "maxPrice": 0,
        "isin": "IN0020150044"
      },
      {
        "biddingStartDate": "01-02-2017",
        "symbol": "TEST1",
        "issueType": "GB",
        "minBidQuantity": 10000,
        "maxQuantity": 20000000,
        "lotSize": 10000,
        "t1ModEndDate": "03-07-2019",
        "dailyStartTime": "",
        "allotmentDate": "",
        "t1ModStartTime": "",
        "biddingEndDate": "02-07-2019",
        "t1ModEndTime": "",
        "dailyEndTime": "",
        "tickSize": 0,
        "series": "GB",
        "faceValue": 10000,
        "minPrice": 0,
        "t1ModStartDate": "03-07-2019",
        "issueValueSize": 0,
        "name": "Sovereign Gold Bond",
        "issueSize": 100000000,
        "incompleteModEndDate": "",
        "maxPrice": 0,
        "isin": "IN0020150028"
      },
      {
        "biddingStartDate": "13-01-2017",
        "symbol": "TEST2",
        "minBidQuantity": 100,
        "maxQuantity": 100000,
        "lotSize": 100,
        "t1ModEndDate": "03-07-2019",
        "dailyStartTime": "09:00:00",
        "allotmentDate": "",
        "t1ModStartTime": "07:00:00",
        "biddingEndDate": "02-07-2019",
        "t1ModEndTime": "13:00:00",
        "dailyEndTime": "23:00:00",
        "tickSize": 10000,
        "series": "GB",
        "faceValue": 10000,
        "minPrice": 0,
        "onlinePrice": 3000,
        "offlinePrice": 3500,
```


Non-Confidential

```
        "t1ModStartDate": "03-07-2019",
        "issueValueSize": 0,
        "name": "Sovereign Gold Bond",
        "issueSize": 100000000,
        "incompleteModEndDate": "",
        "maxPrice": 0,
        "isin": "INE00124571A"
      }
    ],
    "status": "success"
  }
```


POST/<version>/sgb/add

This API allows user to add / modify / cancel an application

Method	POST
URL	https://<baseurl>/<version>/sgb/add
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String (12)	Symbol
orderNumber	Number	Mandatory In case of Modification and Cancelation else ignored
price	Decimal	Online Price: - Min Issue Price Offline Price: - Max Issue Price
quantity	Number	Bid Quantity
physicalDematFlag	String (1)	D = Demat P = Physical
clientCode	String (10)	Valid client code for Physical order else ignored
pan	String (10)	PAN No.- Permanent Account Number
depository	String (4)	Depository Name (NSDL, CDSL)
dpId	String (8)	DP ID - Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
clientRefNumber	String (16)	Reference Number- Internal reference number
activityType	String (2)	ER = Entry request MR = Modify Request CR = Cancel Request

Sample Request


```
  [
    {
        "symbol": "TEST",
        "quantity": 100,
        "price": 55,
        "physicalDematFlag": "D",
        "pan": "AFAKA2323L",
        "depository": "NSDL",
        "dpId": "33445566",
        "clientBenId": "12345678",
```


Non-Confidential

```
            "activityType": "ER",
            "clientRefNumber": "MYREF0001"
        }
    ]
```

Response JSON

Field	Type	Description
symbol	String (12)	Symbol
orderNumber	Number	Order Number
series	String (2)	Series
applicationNumber	String (16)	Application Number
quantity	Number	invested Quantity
price	Decimal	Bid Price.
physicalDematFlag	String (1)	D = Demat P = Physical
clientCode	String (10)	Valid client code for Physical order else ignored
pan	String (10)	PAN No.- Permanent Account Number
depository	String (4)	Depository Name (NSDL, CDSL)
dpId	String (8)	DP ID Ignored in case of CDSL.
clientBenId	Number	Client ID in case of NSDL(8) Beneficiary ID in case of CDSL(16)
clientRefNumber	String (16)	Reference Number- Internal reference number
orderStatus	String (2)	ES = Entry success EF = Entry Fail MS = Modify Success MF = Modify Fail CS = Cancel Success CF = Cancel Fail
rejectionReason	String (100)	Rejection reason
enteredBy	String (25)	Enter By
entryTime	DATETIME	Entry time
verificationStatus	String (1)	P = Pending S = Verified F= Verification failed
verificationReason	String (100)	Verification Reason
clearingStatus	String (2)	FP = Fund Pending FS = Sent to NSCCL RJ = Rejected FR = Funds Received RS = RBI Submitted RA = RBI Allotted
clearingReason	String (100)	Clearing Reason
lastActionTime	DATETIME	Last action time
status	String	success/failed
reason	String (100)	Reason text

Sample Response

```
    Success:
```


Non-Confidential

```
[
   {
         "symbol": "TEST",
         "orderNumber": 2019042500000003,
         "quantity": 100,
         "applicationNumber": "1200299929020",
         "price": 55,
         "physicalDematFlag": "D",
         "pan": "AFAKA2323L",
         "depository": "NSDL",
         "dpId": "33445566",
         "clientBenId": "12345678",
         "orderStatus": "ES",
         "clientRefNumber": "MYREF0001,
         "enteredBy ": "samir",
         "entryTime ": "25-04-2019 12:39:01",
         "verificationStatus ": "P",
         "verificationReason ": null,
         "clearingStatus ": "FP",
         "clearingReason ": "",
         "lastActionTime ": "25-04-2019 12:39:01",
         "status": "success"
   }
]
```


GET /<version>/sgb/{{time}}

This API allows user to download SGB transactions.

Method	GET
URL	https://<baseurl>/<version>/sgb/{{time}}
Request	Path Parameter
Response	JSON

Request Path Parameters

Field	Type	Description
time	DateTime	Cutoff date time. Transactions entered after this time will be returned in the response

Sample URL with Path Parameters

```
https://<<baseurl>>/<version>/sgb/01-12-2015%2009:30:00
```

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	Reason in case of failure
transactions	List of Objects	List Transaction responses. See Response JSON Structure for API "POST /<version>/sgb/add"

Sample Response - Success


```
Success: {
  "transactions": [{
         "symbol": "TEST",
```


Non-Confidential

```
            "orderNumber": 2019042500000003,
            "quantity": 100,
            "applicationNumber": "1200299929020",
            "price": 55,
            "physicalDematFlag": "D",
            "pan": "AFAKA2323L",
            "depository": "NSDL",
            "dpId": "33445566",
            "clientBenId": "12345678",
            "orderStatus": "ES",
            "clientRefNumber": "MYREF0001",
            "rejectionReason ": null,
            "enteredBy ": "samir",
            "entryTime ": "25-04-2019 12:39:01",
            "verificationStatus ": "P",
            "verificationReason ": null,
            "clearingStatus ": "FP",
            "clearingReason ": "",
            "lastActionTime ": "25-04-2019 12:39:01",
            "status": "success"

      },
      {
            "symbol": "TEST",
            "orderNumber": 2019042500000004,
            "quantity": 100,
            "applicationNumber": "1200299929020",
            "price": 55,
            "physicalDematFlag": "D",
            "pan": "AFAKA2323L",
            "depository": "NSDL",
            "dpId": "33445566",
            "clientBenId": "12345678",
            "orderStatus": "ES",
            "clientRefNumber": "MYREF0001",
            "rejectionReason ": null,
            "enteredBy": "samir",
            "entryTime": "25-04-2019 12:39:01",
            "verificationStatus ": "P",
            "verificationReason ": null,
            "clearingStatus ": "FP",
            "clearingReason ": "",
            "lastActionTime ": "25-04-2019 12:39:01",
            "status": "success"
      }
   ]
```


Sample Response – Failed


```
   {
      "status": "failed",
      "reason": "No Records Found"
   }
```


POST /<version>/sgb/fetch

This API allows user to download SGB transactions for the given filter criteria.

Method	POST
URL	https://<baseurl>/<version>/sgb/fetch
Request	JSON

Non-Confidential

Response	JSON

Request JSON

Field	Type	Description
symbol	String (12)	Mandatory Symbol
orderNumber	String (16)	Mandatory Order No

Sample Request

```
{"symbol": "TEST","orderNumber": "1200299929020"}
```

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	Reason in case of failure
transactions	List of Objects	List Transaction response. See Response JSON Structure for API "POST /<version>/sgb/add"

Sample Response - Success

```
Success: {
  "transactions": [
  [{
          "symbol": "TEST",
          "orderNumber": 2019042500000003,
          "quantity": 100,
          "applicationNumber": "1200299929020",
          "price": 55,
          "physicalDematFlag": "D",
          "pan": "AFAKA2323L",
          "depository": "NSDL",
          "dpId": "33445566",
          "clientBenId": "12345678",
          "orderStatus": "ES",
          "clientRefNumber": "MYREF0001",
          "rejectionReason": null,
          "enteredBy": "samir",
          "entryTime": "25-04-2019 12:39:01",
          "verificationStatus ": "P",
          "verificationReason ": null,
          "clearingStatus": "FP",
          "clearingReason": "",
          "lastActionTime": "25-04-2019 12:39:01",
          "status": "success"
    }
]
```

Sample Response - Failed

```
{
   "status": "failed",
   "reason": "No Records Found"
}
```

GET /<version>/sgb/mismatches/{{symbol}}

This API allows user to download mismatches.

Non-Confidential

Method	GET
URL	https://<baseurl>/<version>/sgb/mismatches/{{symbol}}
Request	Path Parameter
Response	JSON

Request Path Parameters

Field	Type	Description
symbol	String (12)	Issue Symbol

Sample URL with Path Parameters

https://<<baseurl>>/eipo/<version>/sgb/mismatches/TESTSYM01

Response JSON

Field	Type	Description
status	String (10)	success / failed
reason	String (100)	reason
mismatches	List of Objects	List of Mismatch objects

Mismatch Object JSON Structure

Field	Type	Description
symbol	String (12)	Symbol
applicationNumber	String (16)	
orderNumber	Number	In case of Modification and Deletion only
pan	String (10)	PAN No.- Permanent Account Number
depository	String (4)	NSDL CDSL
dpId	String (8)	DP ID - Ignored in case of CDSL.
clientBenId	String (16)	Client ID in case of NSDL / Beneficiary ID in case of CDSL
reason	String (250)	Reason text

Sample Response


```
{
   "status": "success",
   "mismatches": [
         {
                  "symbol": "TEST",
                  "orderNumber": 2019042500000003,
                  "applicationNumber": "1200299929020",
                  "depository": "NSDL",
                  "dpId": "33445566",
                  "clientBenId": "12345678",
                  "pan": "AFAKA2323L",
                  "reason" : "Account does not exist"
         },
         {
               "symbol": "TEST",
                  "orderNumber": 2019042500000004,
                  "applicationNumber": "1200299929025",
                  "depository": "NSDL",
                  "dpId": "33445566",
```


Non-Confidential

```
                    "clientBenId": "12345679",
                    "pan": "AFAKA2326L",
                    "reason" : "Account does not exist"
                }
            ]
    }
```

GET /<version>/sgb/incomplete/{{symbol}}

This API allows user to download incomplete details.

Method	GET
URL	https://<baseurl>/<version>/sgb/incomplete/{{symbol}}
Request	Path Parameter
Response	JSON

Request Path Parameters

Field	Type	Description
symbol	String (12)	Issue Symbol

Sample URL with Path Parameters

```
https://<<baseurl>>/eipo/<version>/sgb/incomplete/TESTSYM01
```

Response JSON

Field	Type	Description
status	String(10)	success / failed
reason	String(100)	reason
incomplete	List of Objects	List of Incomplete objects

Incomplete Object JSON Structure

Field	Type	Description
symbol	String (12)	Issue Symbol
orderNumber	Number	Order Number
incompleteCodesList	List of Number	List of Incomplete Error Codes Please refer List of Incomplete Codes section under Appendix A

Sample Response

```
    {
       "status": "success",
       "incomplete": [
                  {
                     "symbol": "TEST",
                     "orderNumber": 2022042500000005,
                     "incompleteCodesList": [ 301, 308]
                  },
                  {
                     "symbol": "TEST",
                     "orderNumber": 2022042500000003,
                     "incompleteCodesList": [ 301, 304, 306]
                  }
            ]
    }
```

POST/<version>/sgb/incomplete/update

Non-Confidential

This API allows user to modify incomplete investor details for an application

Method	POST
URL	https://<baseurl>/<version>/sgb/incomplete/update
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
symbol	String (12)	Mandatory
orderNumber	Number	Mandatory
firstHolderDOB	DATE	First Holder Date of Birth
firstHolderState	String (50)	State
firstHolderBankIfsc	String (11)	Bank Account IFSC
firstHolderBankAccNo	String (16)	Bank Account Number
secondHolderPAN	String (10)	Second Holder PAN
secondHolderDOB	DATE	Second Holder Date of Birth
thirdHolderPAN	String (10)	Third Holder PAN NO
thirdHolderDOB	Number	Third Holder Date of Birth
guardianFirstName	String (250)	Guardian First Name
guardianRelation	String (10)	Guardian Relation
guardianDOB	DATE	Guardian Date of Birth
guardianPAN	String (10)	Guardian PAN

Sample Request

```
[
   {
         "symbol": "TEST",
         "orderNumber": 2019042500000005,
         "firstHolderDOB": "25-10-1986",
         "firstHolderBankAccNo": "911010056464605"
   }

]
```

Response JSON

Field	Type	Description
status	String	success/failed
reason	String (100)	Reason text

Sample Response

```
[
   {
         "status": "success"
   }
]
```

GET /<version>/holidaymaster/{{fromdate}}/{{todate}}

This API allows user to download holiday master.

Non-Confidential

Method	GET
URL	https://<baseurl>/<version>/holidaymaster/{{fromdate}}/{{todate}}
Request	Path Parameters
Response	JSON

Request Path Parameters

Field	Type	Description
fromdate	Date	Holiday declared after this date will be returned in the response
todate	Date	Holiday declared before this date will be returned in the response

Sample URL with Path Parameters

https://<baseurl>/<version>/ holidaymaster /21-12-2025/01-04-2026

Response JSON

Field	Type	Description
status	String (10)	success / failed
data	List of Objects	List of Holiday Master objects

Holiday Master Object JSON Structure

Field	Type	Description
date	Date	Holiday Date
desc	String (250)	Reason for holiday
segmentEligibility	List of String	List of applicable segments Possible values are below

Type	List of applicable segments
IPO	Equity, SME, FPO
RIGHTS	RIGHTS, Call Money
DEBT	DEBT IPO, SSE
REIT	REIT
INVIT	INVIT
TENDER	Takeover, Buyback, Delisting
GB	Gold Bond
NCB	G-Sec, SDL, T-Bills
ETF	ETF
NFO	NFO

Sample Response


```
{
    "data": [
        {
            "date": "31-12-2025",
            "desc": "NEW YEAR",
            "segmentEligibility": [
                "IPO",
                "DEBT",
                "RIGHTS",
                "TENDER"
            ]
        },
        {
```


Non-Confidential

```
                "date": "21-02-2026",
                "desc": "TEST",
                "segmentEligibility": [
                    "NCB",
                    "GB"
                ]
            }
        ],
        "status": "success"
    }
```


Callback APIs

Summary

API	Description
POST /<version>/appdpstatus	Call back API for updating application DP Verification status
POST /<version>/apppaystatus	Call back API for updating application UPI payment status
POST /<version>/notification	Call back API for pushing host notifications to member/bank system

General Instructions

Following headers need to be provided in all API calls

Authorization: Value should contain the base64 encoded text of SHA256 (SHA1 ({{password}})). For example, for user password “Pass@123” the header value should be “MTdiOTNmNWFiNTZhZjYxNGUwZDg5OGVkNDcxYTZhMjlkZjNmYTJhYWQ1YjI3M2ZiZDlhOWVmYjhhMWMxYWNmMg==”

API Reference

POST /<version>/appdpstatus

This API exposed by the member/bank system will allow the host system to inform the member/bank system about the DP verification status updates.

Method	POST
URL	https://<memberurl>/<version>/appdpstatus
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
Symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
dpVerStatusFlag	String(1)	DP Verification Status for PAN DP combination check P = Pending S = Success F = Failed
dpVerFailCode	String(10)	Latest DP Verification failure code given by depository
dpVerReason	String(250)	DP Verification failure reason.

Non-Confidential

Sample Request

```
{
   "symbol": "TEST",
   "applicationNumber": "1200299929020",
   "dpVerStatusFlag": "S",
   "dpVerReason": null,
   "dpVerFailCode": null
}
```

Response JSON

Field	Type	Description
Status	String(30)	success / failed
Reason	String(100)	Failure reason text. Only if “status” = “failed”

Sample Response – Success

```
{
   "status": "success"
}
```

Sample Response – Failed

```
{
   "status": "failed",
   "reason": "Application no does not exist"
}
```

POST /<version>/apppaystatus

This API exposed by the member/bank system will allow the host system to inform the member/bank system about the UPI Payment status updates.

Method	POST
URL	https://<memberurl>/<version>/apppaystatus
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
Symbol	String(12)	Symbol
applicationNumber	String(16)	Application No.(of the physical form)
upiPaymentStatusFlag	Number	Last UPI Payment status received from sponsor bank for the given application 0=Request Sent 1=Request Failed 10=Request Accepted By Sponsor Bank 11=Rejected due to invalid UPI 12=Rejected by Sponsor Bank 13=Block/Release Request Rejected Due To UPI 2.0 Not Being Supported By Investor Bank 21=Rejected by Investor Bank 22=Block/Release Request Rejected By Client Bank Due To Technical Reason 31=Rejected by Investor

Non-Confidential

		100=Accepted by Investor 110=Block Released (Due to cancellation of order)
upiAmtBlocked	Decimal	Amount blocked in case of applications with upiFlag=Y
upiPayReason	String(100)	UPI Payment failure reason.

Sample Request

```
{
   "symbol": "TEST",
   "applicationNumber": "1200299929020",
   "upiPaymentStatusFlag": 100,
   "upiAmtBlocked": 125000.00,
   "upiPayReason": null
}
```

Response JSON

Field	Type	Description
Status	String(30)	success / failed
Reason	String(100)	Failure reason text. Only if “status” = “failed”

Sample Response – Success

```
{
   "status": "success"
}
```

Sample Response – Failed

```
{
   "status": "failed",
   "reason": "Application no does not exist"
}
```

POST /<version>/notification

This API exposed by the member/bank system will allow the host system to inform the member/bank system about any notification.

Method	POST
URL	https://<memberurl>/<version>/notification
Request	JSON
Response	JSON

Request JSON

Field	Type	Description
type	Number	Notification type 1 = IPO detail modified 2 = Category bidding start 3 = Category bidding end
symbol	String(12)	Symbol
data	Structure	Notification type wise structure 1 = null 2 = Category Start Stop JSON 3 = Category Start Stop JSON

Non-Confidential

timestamp	Timestamp	Notification timestamp. Format : dd-MM-yyyy hh24:mm:ss

Category Start Stop JSON

Field	Type	Description
category	String(14)	Category Code

Sample Request

```
{
   "type": 1,
   "symbol": "TESTIPO",
   "timestamp": "15-05-2021 09:34:15"
}
```

```
{
   "type": 2,
   "symbol": "TESTIPO",
   "data": {
         "category": "RETAIL"
   },
   "timestamp": "15-05-2021 10:00:00"
}
```

Response JSON

Field	Type	Description
status	String(30)	success / failed
reason	String(100)	Failure reason text. Only if “status” = “failed”

Sample Response – Success

```
{
   "status": "success"
}
```

Sample Response – Failed

```
{
   "status": "failed",
   "reason": "Application no does not exist"
}
```


Non-Confidential

Appendix A -Error Messages

Error Code	Message
1	Order status changed
2	Invalid Symbol
3	Invalid Member
4	Invalid Dealer
5	Issue is not open
6	Access denied for issue
7	Market is not open
8	Invalid Depository Name
9	Invalid DP id
10	Invalid Ben. Id
11	Missing field: <Field Name>
12	Invalid field: <Field Name>
13	Bidding not allowed for <issue type>
14	Invalid bid amount for category <Category Name>
15	Inactive category <Category Name>
16	Cutoff not allowed for category <Category Name>
17	Only <Max Count> transactions per application are allowed
18	Order value entered exceeds the user limit value
19	Bid amount exceeds the total value size of issue
20	Bid entry not started for category <Category Name>
21	Bid entry closed for category <Category Name>
22	Issue level setting not found/inactive for issue <Issue Symbol>
23	Non-ASBA bidding not allowed for <Issue Symbol or Member Type >
24	Physical allotment not allowed for the issue <Issue Symbol >
25	Modification of bid not allowed for issue <Issue Symbol >
26	Cancellation of bid not allowed for issue <Issue Symbol >
27	Modification of quantity not allowed for issue <Issue Symbol >
28	Modification of price not allowed for issue <Issue Symbol >
29	Modification of bid amount not allowed for issue <Issue Symbol >
30	Downward modification of quantity not allowed for issue <Issue Symbol >
31	Downward modification of price not allowed for issue <Issue Symbol >
32	Downward modification of bid amount not allowed for issue <Issue Symbol >
33	Upward modification of quantity not allowed for issue <Issue Symbol >
34	Upward modification of price not allowed for issue <Issue Symbol >
35	Upward modification of bid amount not allowed for issue <Issue Symbol >
36	Invalid length of field <Field Name>
37	Mismatch field <Field Name>
38	Invalid IFSC
39	Duplicate order number in request

Non-Confidential

40	Only 1 record allowed for series <Series Name>
41	Invalid application number
42	Category setting not found.
43	Application number used by other Bank/Member/Web Dealer
44	Access denied on the application
45	Invalid bid amount
46	Offline not allowed
47	Offline not allowed for category <Code>
48	Bid modification/Partial bid cancellation is not allowed for ASBA Pay applications
49	ASBA Pay orders are not allowed
50	Bid entry closed for asbapay orders
51	Partial cancellation of bid not allowed for issue <Issue Symbol>
200	Invalid Cut-Off Indicator
201	Invalid bid Price
202	Price is greater than max. price range
203	Price is less than min. price range
204	Bid Price should be in multiple of tick size
205	Bid quantity is less than min market lot
206	Bid quantity should be multiple of market lot
207	Invalid Activity type
208	Missing Bid reference number
209	Record not exist.
501	Bid level error
1501	Invalid message code
1502	Incorrect message size
1503	Invalid sequence of message
1504	System Error.OMS not connected
9999	System error
9998	System Error.Order processing failed due to concurrent request

SGB API - List of Incomplete Codes

Code	Reason	Field ID
301	First Holder DOB is missing	firstHolderDOB
302	First Holder State is missing	firstHolderState
303	First Holder Bank Ifsc is missing	
304	First Holder Bank Ifsc should be 11 characters code	
305	First Holder Bank Ifsc: First 4 characters should be alphabet. Remaining 7 should be Numeric	firstHolderBankIfsc
306	First Holder Bank Acc is missing	
307	First Holder Bank Acc shall have a maximum length of 16 digits	
308	First Holder Bank Acc should be Numeric	firstHolderBankAccNo

Non-Confidential

309	Second Holder PAN is missing	secondHolderPAN
310	Second Holder DOB is missing	secondHolderDOB
311	Third Holder PAN is missing	thirdHolderPAN
312	Third Holder DOB is missing	thirdHolderDOB
313	Guardian First Name is missing	guardianFirstName
314	Guardian Relation is missing	guardianRelation
315	Guardian DOB is missing	guardianDOB
316	Guardian PAN is missing	guardianPAN

Appendix B – Base URL

The <baseurl> in the urls might be changed and shall be communicated separately in case of any changes.

Base URL	New
Production	bidcollection.nseindia.com/eipo
UAT	eipouat.nseindia.com/eipo

Appendix C – EIPO- API rate limits

To prevent system abuse, misuse or overloading of the API infrastructure, Exchange has applied API limits for logins and various transactions request. Market participants using the Exchange EIPO platforms are requested to take note of the same and ensure to maintain their activity is within the limits to avoid disconnection/blocking/rejection.

The limits are as mentioned below:

API Name	Purpose of the API	Rate limit	Remarks
V3/login	Login to the host system	2 requests per user per sec	
transactions/add	For single request	1000 requests per user every sec	
transactions/addbulk	For placing 1 or more bids	100 requests (max records up to 100 per request) per user every sec	
transactions/fetch	For fetching transaction details	25 requests per user per sec	Symbol & single application number will be the input
transactions/<time>	For fetching transaction details using date & time	1 request (max records up to 25000) per user every 15 mins	Date & time will be the input

Non-Confidential

transactions/slice/<fromtime>/<totime>	For fetching transaction details using date & time	1 request (max records up to 25000) per user every 15 mins	Date & time will be the input
allotment/fetch	For fetching allotment details	10 requests per user per sec	symbol is mandatory along with either application number or pan number
allotment/<fromtime>/<totime>	For fetching allotment details using date & time	1 request (max records up to 25000) per user every 15 mins	Date & time will be the input
allotment/report/<symbol>	For downloading allotment details using symbol	1 request per user ,per symbol every 15 mins	Symbol input

Non-Confidential