

Table of Contents

Preface	2
Purpose	2
Target Audience	2
Organization of This Document.....	2
Introduction	3
Authentication	3
Message Format.....	3
Login / Logout	5
POST /LOGIN.....	5
Request JSON values	5
Request Object Structure (JSON).....	5
Response JSON values	6
Response Object Structure (JSON)	6
POST /LOGOUT	7
Request JSON values	7
Request Object Structure (JSON).....	7
Response JSON values	7
Response Object Structure (JSON)	7
Holding Data	8
POST /UPLOAD_HOLDING	8
Request JSON values	8
Request Object Structure (JSON).....	9
Response JSON values	10
Response Object Structure (JSON)	12

Preface

Purpose

This document describes the protocol for Web API based communication and messaging between exchanges and Members. The document serves as a common guide for developing and testing interfaces between systems at Exchange and Members.

Target Audience

- Member can Upload Holding Details of their clients

Organization of This Document

This document is organized as follows:

Chapters	Description
Introduction	Overview of the document covering security features of the API and online messaging guidelines
Exchange API	This chapter gives details of the API's exposed by exchanges. These API's will be consumed by Members.

Introduction

API's will be provided as REST endpoints accessible over internet using HTTPS protocol. All messages in the online API's will be in JSON format.

Authorization

- The API consumer application is expected to send following headers (case-sensitive) in every API HTTP request.
 - **Token:** A unique identifier generated for all successful login.
 - The API provider is not required to provide any header in its response.
 - Token will have expiry of 60 minutes.

JWT mechanism can be used to generate tokens as it is widely used industry standard. (Implemented by NSE)

Message Format

- Messages will be in JSON.
- Following headers need to be provided in all API calls
 - **Content-Type:** Header value should be "application/json"
- 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)
- All request and response messages are in JSON (JavaScript Object Notation) format. (Refer <http://www.json.org/> for details).
- Some of the key specifications related to JSON and standards followed for the API's are as follows
 - JSON is built on 2 structures. Object or 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. {"ClientCode":"Test1", "PanNo":"ABCDE1234F"}
 - List contains list of Clients. E.g. ["Test1", "Test2", "Test3"]
 - A Boolean has only 2 values true or false.
 - String values are enclosed in single quote or double quotes. e.g. "ClientCode", "PanNo"
 - Numbers and decimals are represented without any thousand - separator character. Decimal indicator is dot (".")
 - All dates, times and date times 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".
 - Request Payload will be compressed by gzip.

Endpoints:

- /login
- /logout
- /tradingHoldingUpload
- /tradingNilUpload

Login / Logout**POST /LOGIN**

This API validates the user credentials. On successful login, a unique token will be provided per request and session will be established with user's system. In subsequent requests API calls this token should be passed in HTTP header "Authorization" as mentioned in general guidelines.

Note: The password need to be sent in encrypted format using AES256. The encryption private key will be generated on the user registration.

Method	POST
Request	JSON
Response	JSON

Request JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values
1	Member Code	memberCode	Char(10)	Member Code	
2	Login Id	loginId	Char(16)	Login id of member's user	
3	Password	password	Char(50)	Password of member's user	

Member Code, Login Id & Password should be same as used in C2N web portal.

Request Object Structure (JSON)

```
{
  "memberCode": "1234",
  "loginId": "1234",
  "password": "biXhp3Ha1fgxVEp48zHrvVoXmStmxPuAPHo3TVz5lHU="
}
```

Password in example is encrypted text

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values
1	Member Code	memberCode	Char(10)	Member Code	
2	Login Id	loginId	Char(16)	Login id of member's user	
3	Token	token	Char(50)	Unique Token for Session	
4	Response status	status	Char(10)	Indicating Success / Error	success/error
5	Response code	code	Array of strings. Individual element is 3 characters long.	Codes are specified in the below table.	

Response Object Structure (JSON)

```
{
  "memberCode": "1234",
  "loginId": "1234",
  "token": "bndfg45ghjg87hjnb"
  "status": "error",
  "code": ["701"]
}
```

POST /LOGOUT

This API closes the current session and gets the user logged off.

Method	POST
Request	JSON
Response	JSON

Request JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values
1	Member Code	memberCode	Char(10)	Member Code	
2	Login Id	loginId	Char(16)	Login id of member's user	

Request Object Structure (JSON)

```
{  
  "memberCode": "1234",  
  "loginId": "1234",  
}
```

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Description	Expected Values and
1	Response status	status	Char(10)	Indicating Success / Error	
2	Response code	code	Array of strings. Individual element is 3 characters long.	Codes are specified in the below table.	

Response Object Structure (JSON)

```
{  
  "status": " error",  
  "code": [ "802"]  
}
```

Holding Data

POST /UPLOAD_HOLDING

This API will allow member to submit Holding Details of their clients to the Exchange where member can

- Add New Details
- Delete Existing Details

Method	POST
Request	JSON
Response	JSON

Request JSON values

NOTE: holdingData is Array of objects in JSON

Sr. No.	Field Description	Field Name	Data Type	Size	Description	Expected Values and Validation	Type of field	Remarks
1	Member PAN	memberPan	Characters	10	Member PAN	Basic PAN level Validations	Header field. Only once per request	
2	Last Week Date	lastWeekDay	Characters	10	Last date of the week for which holding details are being uploaded	DD-MM-YYYY	Header field. Only once per request	
3	Date	weekDay	Characters	10		DD-MM-YYYY	Header field. Only once per request	
4	Version No	versionNo	Number	5	Will be providing the version of API, initial version being 1.0	1.0	Header field. Only once per request	
5	Record Id(Unique Id)	recId	character	15	Unique id	Max length: 15	Details repeated for every row	
6	Member's Demat Account No.	dmat	Characters	16	Client Dmat details		Details. Repeated for every row	
7	Member Account Type	accountType	Characters	12	Client Account Type	POOL, CM COLL, OWN, EPI, CLNTBENEFIT, TM COLL, TM COLL-MTF, CLNT UNPAID, CLNT COLL	Details. Repeated for every row	
8	Unique Client Code (UCC)	ucc	Characters	20	Unique Client Code		Details. Repeated for	

							every row	
9	Client Name	clientName	Characters	150	Client Name		Details. Repeated for every row	
10	Client PAN	pan	Characters	10	Client PAN	Basic PAN level Validations	Details. Repeated for every row	
11	ISIN Code	isin	Characters	12	ISIN		Details. Repeated for every row	
12	Security Type	securityType	Characters	4	Security Type	EQ, MF, BOND, DEBT, PREF, IDR, WT	Details. Repeated for every row	
13	Pledged Balance (Qty.)	totalPldgQty	Number	20	Total Pledged Quantity	CHECK : UPTO 3 Decimals	Details. Repeated for every row	
14	Free Balance (Qty.)	freeBalQty	Number	20	Free Balance Quantity	CHECK : UPTO 3 Decimals	Details. Repeated for every row	
15	Total (Qty.)	totalQty	Number	20	Total Quantity	CHECK : UPTO 3 Decimals	Details. Repeated for every row	
16	Action	action	Characters	1		A - Add D – Delete	Details. Repeated for every row	

Note: Only Add and Delete functionality will be provided.

Request Object Structure (JSON)

```
{
  "memberPan": "ABCDE1234F",
  "lastWeekDay": "07-06-2019",
  "weekDay": "03-06-2019",
  "versionNo": 1.0,
  "holdingData": [
    {
      "recId": "111916112102351",
      "dmat": "1200010156568800",
      "accountType": "POOL",
      "ucc": "dummy",
      "clientName": "client name 1",
      "pan": "ABCFD1234G",
      "isin": "INE000AB1000",
      "securityType": "EQ",
      "totalPldgQty": 0,
      "freeBalQty": 10,
      "totalQty": 10,
      "action": "A"
    },
    {
      "recId": "299352523429591",
      "dmat": "1200110156568800",
      "accountType": "EPI",
      "ucc": "dummy1",
      "clientName": "client name 2",
      "pan": "ABCFD7894G",
      "isin": "INE001VN1000",
      "securityType": "BOND",
      "totalPldgQty": 0,
      "freeBalQty": 10,
      "totalQty": 10,
      "action": "D"
    }
  ]
}
```

Note: Details records count is restricted to 5,00,000 records (This will reduce number of request)

The Record ID(Unique Id) field (recId) is generated at the API consumer application. The server will not validate this field for uniqueness or any other criteria. The server will validate only max length of this field.

The Version Number(versionNo) is used for identifying the request. On future changes in api the version number will be modified accordingly.

Response JSON values

Sr. No.	Field Description	Field Name	Data Type	Size	Description	Expected Values and Validation	Type of field	Remarks
1	Member PAN	memberPan	Characters	10	Member PAN		Header field. Only once per request	
2	Last Week Date	lastWeekDay	Characters	10	Last date of the week for which holding details are being uploaded		Header field. Only once per request	
3	Date	weekDay	Characters	10			Header field. Only once per request	
4	Version No	versionNo	Number	5	Will be providing the version of API, initial version being 1.0	1.0	Header field. Only once per request	
5	Record Id(Unique Id)	recId	character	15	Unique id	Max length: 15	Details repeated for every row	
6	Success or error status	status	Characters	10	Status of request acceptance	"success" or "error"	Details repeated for every row	
7	Response code for the record	code	Array of strings	3 characters for individual member of array	Response code for the record	Possible response codes are listed in the table below. The value is an array.	Details repeated for every row	
8	Header status	hdrStatus	Characters	10		"success" or "error" or "partial"	Header field. Only once per response.	
9	Header code	hdrCode	Array of strings	10		Possible error codes are specified in the table below	Header field. Only once per response.	
10	Acknowledgment ID	ackId	characters	15		Unique request ID generated by the server	Header field only. Once per response	

Response Object Structure (JSON)

```
{
  "memberPan": "ABCDE1234F",
  "lastWeekDay": "07-06-2019",
  "weekDay": "03-06-2019",
  "versionNo": 1.0,
  "holdingData": [
    {
      "recId": "111916112102351",
      "status": "success",
      "code": ["601"]
    },
    {
      "recId": "299352523429591",
      "status": "error",
      "code": ["902", "909"]
    }
  ],
  "hdrStatus": "partial",
  "hdrCode": ["603"],
  "ackId": "1956403061901"
}
```

The server will generate a unique acknowledgment id (ackId) against each request and send it back in the response.

POST /UPLOAD_NIL_Sub

This API will allow member to submit NIL submission for Trading Members.

- Add New Details

Method	POST
Request	JSON
Response	JSON

Request JSON Structure

Sr. No.	Field Description	Field Name	Data Type	Size	Description	Expected Values and Validation	Type of field	Remarks
1	Member PAN	memberPan	Characters	10		Basic PAN level Validations	Header field. Only once per request	
2	Last Week Date	lastWeekDay	Characters	10	Last date of the week for which holding details are being uploaded	DD-MM-YYYY	Header field. Only once per request	
3	Date	weekDay	Characters	10		DD-MM-YYYY	Header field. Only once per request	
4	Version No	versionNo	Number	5	Will be providing the version of API, initial version being 1.0	1.0	Header field. Only once per request	
5	Record Id(Unique Id)	recId	character	15	Unique id	Max length: 15	Details repeated for every row	
6	Unique Client Code (UCC)	ucc	Characters	20				
7	Client PAN	cpan	Characters	10		Basic PAN level validations		

Request Object Structure (JSON)

```
{
  "memberPan": "ABCDE1234F",
  "lastWeekDay": "07-06-2019",
  "weekDay": "03-06-2019",
  "versionNo": 1.0,

  "holdingData": [
    {
      "recId": "111916112102351",
      "ucc": "1200010156568800",
      "cpan": "ABCDE1234S"
    },
    {
      "recId": "299352523429591",
      "ucc": "1200010156568800",
      "cpan": "VBCDE1234S"
    }
  ]
}
```

Response JSON Structure

Sr. No.	Field Description	Field Name	Data Type	Size	Description	Expected Values and Validation	Type of field	Remarks
1	Member PAN	memberPan	Character s	10			Header field. Only once per request	
2	Last Week Date	lastWeekDay	Character s	10	Last date of the week for which holding details are being uploaded		Header field. Only once per request	
3	Date	weekDay	Character s	10			Header field. Only once per request	
4	Version No	versionNo	Number	5	Will be providing the version of API, initial version being 1.0	1.0	Header field. Only once per request	
5	Record Id(Unique Id)	recId	character	15	Unique id	Max length: 15	Details repeated for every row	
6	Success or error status	status	Characters	10	Status of request acceptance	"success" or "error"	Details repeated for every row	
7	Response code for the record	code	Array of strings	3 characters for individual member of array	Response code for the record	Possible response codes are listed in the table below. The value is an array.	Details repeated for every row	
8	Header status	hdrStatus	Character s	10		"success" or "error" or "partial"	Header field. Only once per response.	
9	Header code	hdrCode	Array of	10		Possible error	Header field.	

			strings			codes are specified in the table above	Only once per response.	
10	Acknowledgment ID	ackId	characters	15		Unique request ID generated by the server	Header field only. Once per response	

Response Object Structure (JSON)

```
{
  "memberPan": "ABCDE1234F",
  "lastWeekDay": "07-06-2019",
  "weekDay": "06-03-2019",
  "versionNo" : 1.0,

  "holdingData": [
    {
      "recId": "111916112102351",
      "status": "error",
      "code": ["905", "903"]
    },
    {
      "recId": "299352523429591",
      "status": "success",
      "code": ["601"]
    }
  ],
  "hdrStatus": "partial",
  "hdrCode": ["603"]
  "ackId": "1956403061901"
}
```

Master Code List that will be sent in the response are as following:

Code	Description
601	Success
602	General Error
603	Partial Submission
604	Entire holdingData is incorrect
605	Something went wrong
606	Method unkown
701	Invalid member code or login id or password
702	User is disabled
703	Invalid member code or login id
802	Invalid token
803	Auth token expired
804	"Request exceeds max number of records" when number of records in the request exceed max number of records allowed. Currently, it is 500000.
805	"Kindly first submit data for week ended <earliest lastWeekDay Date>" for the requests where any previous submission is missing. This message specifies which week's submission is pending.
806	Member PAN no not proper
807	weekDay Date does not belong to selected holding period
808	Holding weekDay date format not proper
809	lastWeekDay isn't a saturday
810	Holding lastWeekDay date format not proper
901	Demat account no not proper
902	Invalid member account type
903	Ucc not proper
904	Client name not proper
905	Client pan no not proper
906	ISIN code not proper
907	Invalid security type
908	Pledge Balance not proper
909	Pledge Balance not proper, Check if this split character is required here should contain max 3 digit after decimal
910	Free Balance not proper

911	Free Balance not proper, Check if this split character is required here should contain max 3 digit after decimal
912	Total not proper
913	Total not proper, Check if this split character is required here should contain max 3 digit after decimal
914	Total should be sum of pledged and free balance
915	Action not proper, Check if this split character is required here First request should not contain any delete records
916	Action not proper
917	Not valid record, blank/NULL values not allowed
918	Duplicate Row
919	Invalid request id
920	Invalid version No

This page is intentionally left blank