**Module: Trainee API Upload**



**Web Server**

REST Web API

**App Server**

Authentication(User Name/PWD)

User Autherization

Token Generation with Expire Time

Token Authentication

Trainee Data

Database

To use this API, consumer need to integrate it in his system through provided URL, User can consume web API using given URL which is platform independent.

Sample code to integrate API is given at the end of the document

When user call web API first time it will ask user credentials for authentication and authorization. After successful authentication user can send trainee data with token to upload into the NCVT system.

The input parameter and response data for trainee upload are given below in this document. All the parameter should be in given data type format otherwise we will fail the request during data validation

After data sent by consumer to the API in below given format (**Input Data**), it will be validated and get response if any issue with the data and the response will be in below given format (**Response Data Format**)

Image data should be in byte format. When user will consume Web API then first need to convert image data in byte format then send trainee details data in parameters.

**Test API URL**

<http://164.100.68.244:8082/MIS/api/traineeupload/UploadTrainees/>

**Production URL**

[https://api.ncvtmis.gov.in/mis/api/traineeupload/uploadtrainees](https://apc01.safelinks.protection.outlook.com/?url=https%3A%2F%2Fapi.ncvtmis.gov.in%2Fmis%2Fapi%2Ftraineeupload%2Fuploadtrainees&data=05%7C01%7Cankesh.baloch%40wipro.com%7C668c3270d13640cb3cfa08da96362a3a%7C258ac4e4146a411e9dc879a9e12fd6da%7C1%7C0%7C637987456652800474%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C3000%7C%7C%7C&sdata=Ccf7O4BhMicVThJBm8%2BjtE5B4FfntUCHW8niS6awkPc%3D&reserved=0)/

Client Machine

Image

Byte

API Server

Byte to File

File Server

Database

1. **Input Data**

[

 {

 "StateRegNumber": "103300000820123456",

 "TraineeName": "TestData",

 "UIDNumber": "",

 "DateOfBirth": "1990-01-01",

 "Gender": "Male",

 "Category": "General",

 "FatherGuardianName": "TestFatherName",

 "MotherName": "",

 "MobileNumber": "9867545678",

 "EmailID": "test071@hgg.com",

 "Session": "82019",

 "AdmissionDate": "2019-10-08",

 "HighestQualification": "10th",

 "Trade": "Electrician",

 "Shift": "1",

 "Unit": "1",

 "IsTraineeDualMode": "No",

 "MISITICode": "GR10000033",

 "PersonwithDisability": "No",

 "PWDcategory": "",

 "EconomicWeakerSection": "No",

 "TraineeType": "Regular",

 "TraineePhoto": "[255,216,255,224]"

 },

 {

 "StateRegNumber": "103300000820123467",

 "TraineeName": "TestingData",

 "UIDNumber": "",

 "DateOfBirth": "1990-01-01",

 "Gender": "Male",

 "Category": "General",

 "FatherGuardianName": "FatherData",

 "MotherName": "",

 "MobileNumber": "9867545677",

 "EmailID": "test0671@hgg.com",

 "Session": "82019",

 "AdmissionDate": "2019-10-08",

 "HighestQualification": "10th",

 "Trade": "Electrician",

 "Shift": "1",

 "Unit": "1",

 "IsTraineeDualMode": "No",

 "MISITICode": "GR10000033",

 "PersonwithDisability": "No",

 "PWDcategory": "",

 "EconomicWeakerSection": "No",

 "TraineeType": "Regular",

 "TraineePhoto": "[255,216,255,224,0,16,74,70]"

 }

]

1. **Response Data Format**

{

 "sucessRec": 2,

    "errorRec": 0,

    "totalRec": 2,

"responseData":[

 {

 "StateRegNumber":"10330000082012345",

 "TraineeName":"TestData",

 "ErrorDescription":"Trade is not affiliated with the given ITI",

 "RecordStatus":"E",

 },

 {

 "StateRegNumber":"10330000082012346",

 "TraineeName":"TestingData",

 "ErrorDescription":" Trade is not affiliated with the given ITI ",

 "RecordStatus":"E",

 }

]

}

1. totalRec – Total trainee record submitted by consumer
2. errorRec – Count of records which failed in validation
3. sucessRec - Count of records which successfully uploaded in NCVT system
4. ErrorDescription – Reason why record get failed in validation
5. RecordStatus – E => Error occurred for record, S => When all record got uploaded successfully
6. Field details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Data Size** | **Mandatory** |
| Session\* | Session for which sheet is being updated | 5 digit | Yes |
| StateRegNumber | Unique Number to represent each individual trainees | 18 character | Yes |
| MISITICode\* | ITI Code of MIS system, in which trainee will get enrolled | 20 character | Yes |
| AdmissionDate\* | Admission date of trainee in ITI | valid date | Yes |
| TraineeName\* | Trainee Name (Special character, and numerical values wouldn’t be allowed) | 60 character | Yes |
| MobileNumber\* | Mobile Number of Trainee  | 10 digit | yes |
| EmailID\* | Email ID of Trainee | 100 character | Yes |
| DateOfBirth\* | Date of Birth of Trainee | valid date | Yes |
| Gender\* | Gender of Trainee | 12 character | Yes |
| Category\* | Category of Trainee | 50 character | Yes |
| PersonwithDisability\* | Person with Disability of Trainee | 3 character | Yes |
| PWDcategory\* | Type of PWD | 50 character | yes |
| EconomicWeakerSection\* | Whether they belong to EWS  | 3 character | yes |
| FatherGuardianName\* | Father/Guardian Name of Trainee (Special character, and numerical values wouldn’t be allowed) | 50 character | Yes |
| Mother\_Name | Mother name of Trainee (Optional) - (Special character, and numerical values wouldn’t be allowed) | 50 character | No |
| Trade\* | CTS Trade of Trainee | 150 character | Yes |
| Shift\* | Shift of Trainee | 1 digit | Yes |
| Unit\* | Unit of Trainee | 1 digit | Yes |
| UIDNumber | (Optional) UID or Aadhaar Number of Trainee | 20 digit | No |
| HighestQualification\* | Name of highest qualification of the Trainee | 50 character | Yes |
| IsTraineeDualMode\* | Trainee is in dual trade or not | 3 character | Yes |
| TraineePhoto\* | Photograph of trainee | byte | Yes |
| TraineeType\* | Trainee type private/regular | 10 character | Yes |

**Master Data**



**Sample code to integrate API (C# Dot Net):-**

using (var client = new HttpClient())

{

string strFiledata = System.IO.File.ReadAllText(@"D:\TraineeData.txt");



var httpContent = new StringContent(strFiledata, Encoding.UTF8, "application/json");

client.DefaultRequestHeaders.Add("username", "testuser");

client.DefaultRequestHeaders.Add("password", "testpassword");

client.BaseAddress = new Uri("http://164.100.68.244:8082/APP/api/traineeupload/");

var response = client.PostAsync("UploadTrainees", httpContent).Result;

}

**Format and logic to generate State Registration Number:-**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| SCVT ITI CODE | SCVT TRADE CODE | ADMISSION TYPE (NORMAL/ IMC-MOU) |  | MONTH | YEAR | SERIAL NUMBER |

**UI Design:-**

Scenario 1: When consumer have data in excel or any file format.



Design the UI to select the file and submit the button, on button submit consumer have to convert file data to json format and pass into the API body.

After getting response consumer required to handle response data and can display on UI as above format.

Scenario 2: When consumer have data in their system inside any database.



Design the UI to click submit the button, on button submit consumer have to get data from his database and convert it to json format and pass into the API body.

After getting response consumer required to handle response data and can display on UI as above format.

**Dependencies**

1. We need NIC help to open the required port
2. We need NIC help to map the URL and publically access the API

**Risk and Challenges**

1. User can make duplicate entries as we are considering unique user on the basis of name, email and mobile number. Users can have multiple emails and mobile numbers.
2. If user will share their credentials then anyone can make entries.
3. If numbers of concurrent API consumer increase then we have performance issues.

**Testing Of API –**

* **Install postman on local machine or add it in your chrome browser as extension.**
* **Open postman and follow the below steps**
1. **Create new request and type API URL and select HTTP verbs as POST**

****

1. **Select Header Tab and provide required parameters Content-Type as application/json,**

**Usename and password**

****

1. **Select body tab, select raw radio button option, and select JSON as data format and past your data in proper format in body area.**



1. **Click on send button, response will be display in response area**

****