

Smart Office API Documentation

Version 1.0.4



Apr , 05 2022

CONTENTS

1. Overview

- a. Prerequisites
- b. Support

2. Interface Specifications

- a. API Request
- b. API Response
- c. API Exception

3. Web API's

- a. Add Biometric Device
- b. Delete Biometric Device
- c. Fetch Biometric Logs
- d. Upload Users in Biometric
- e. Delete User in Biometric
- f. Fetch Live Users from Biometric
- g. Set User Expiration
- h. Fetch Biometric Commands
- i. Photo Registration
- j. Block/ unblock Users
- k. ClearAllLogsFromDeviceByTime(Only Supported in Speed Face)
- l. TriggerUserOnlineEnrollment

OVERVIEW

a) All the api is published on postman at the below link

[SmartOffice API \(getpostman.com\)](#)

b) Prerequisites

Below are the following details required to access Smart Office API's.

- Smart Office Web Application.
- Smart Office database.

This can be used in variety of Languages to access REST API's like .Net, JAVA etc.

c) Support

In case for any clarifications, Please get in touch with the API Support with Time Sheet team

SPECIFICATIONS

The API uses Http Requests to receive data and send responses in **JSON** format.

a) **API Request:**

An API request lets you contact a server with API endpoints that you want to reach and perform some action. Those actions are HTTP methods.

b) **API Response:**

An API response consists of the body, headers, and the status code.

c) **API Exception:**

if there is an exception during the processing of the request then the application will send the response in the following

WEB API'S

This Section describes various web service methods available for Biometric communication

a) Add Biometric Device

POST /api/v2/WebAPI/AddBiometric

Add new Biometric Device to Smart Office

Request JSON Object

- **APIKey (String)**: Authentication -token key.
- **DeviceName (String)**: Name of the device.
- **SerialNumber (String)**: A unique identity Id of the biometric.

Response:

- **Message**: "Device Added Successfully."

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://localhost:89/api/v2/WebAPI/AddBiometric?APIKey=420113032112&DeviceName=xyz&SerialNumber=C26044C84F352331
- Headers:** BUILD (with icons for edit and copy)
- Buttons:** Send (blue), Save (grey), Save Response (grey)
- Params:** A table showing Query Params:

KEY	VALUE	DESCRIPTION	Bulk Edit
APIKey	42011303...		
DeviceNa...	xyz		
SerialNum...	C26044C8...		
Key	Value	Description	
- Body:** A JSON response area showing the message "Device Added Successfully."
- Status:** 200 OK, 22 ms, 472 B

Other Response Messages:

- **"Device Name is already exists"** : Device with the Name given is already exists in the database
- **"Device SerialNumber is already exists"**: Device with the Serial Number given is already exists in the database

b) Delete Biometric Device

GET /api/v2/WebAPI/DeleteBiometric

To Delete Biometric Device from Smart Office

Request JSON Object

- **APIKey (String)**: Authentication-token key.
- **SerialNumber (String)**: A unique identity Id of the biometric.

Response:

- **Message**: "Biometric Deleted Successfully."

The screenshot shows a POSTMAN interface with the following details:

- Method:** GET
- URL:** http://localhost:89/api/v2/WebAPI/DeleteBiometric?APIKey=420113032112&SerialNumber=C26044C84F352331
- Params:** APIKey: 420113032112, SerialNumber: C26044C84F352331
- Body:** "Device Deleted Successfully."
- Status:** 200 OK, 354 ms, 474 B

Error Messages -

- "Device Is Not Exists." - if the mention Device Serial Number is wrong
- "Device Logs exists for this device, You can not delete" - if the Punch Records for the Device is exists

c) Fetch Biometric Logs Data

GET /api/v2/WebAPI/ GetDeviceLogs

To get Biometric log data from Smart Office database.

Request JSON Object

- **APIKey (String)**: Authentication-token key.
- **FromDate (DateTime)**: from when the logs to be send.
- **ToDate (DateTime)**: until when the logs to be send.

Response JSON data:

```
[
  {
    "EmployeeCode": "5",
    "LogDate": "2019-09-16 13:45:29",
    "SerialNumber": "",
    "PunchDirection": "in",
    "Temperature": 97.4,
    "TemperatureState": "Normal"
  }
]
```

KEY	VALUE
APIKey	420113032112
FromDate	2021-02-01
ToDate	2021-02-01
Key	Value

```

1 [
2   {
3     "EmployeeCode": "1",
4     "LogDate": "2021-02-16 12:06:52",
5     "SerialNumber": "C26044C84F352330",
6     "PunchDirection": "",
7     "Temperature": 0.0,
8     "TemperatureState": "Not Measured"
9   },
10  {
11    "EmployeeCode": "1",
12    "LogDate": "2021-02-16 12:07:30",
13    "SerialNumber": "C26044C84F352330",
14    "PunchDirection": "",
15    "Temperature": 0.0,
16    "TemperatureState": "Not Measured"
17  },
]

```

Other Biometric Response:

```
{
  "status": false,
  "message": "Serial Number does not exist."
}

{
  status: false,
  message: "Invalid API Key."
}
```

d) Upload Users to Biometric Device

POST /api/v2/WebAPI/UploadUser

To upload User into Biometric Device from Smart Office

Request JSON Object

- **APIKey (String)**: Authentication-token key.
- **EmployeeName (String)**: Name of the Employee/User.
- **EmployeeCode (String)**: Biometric Code of the Employee/User.
- **CardNumber (String)**: Card Number of user.
- **SerialNumber (String)**: To which Biometric device Serial Number User has to be uploaded, Use Comma (,) to separate multiple serial numbers.
- **VerifyMode (String)**: To Allow dual verification mode -(face+card) – This will add face & card verification mode
- **IsFaceUpload** – Pass it as true if you want to upload the face
- **IsFPUpload** – pass it true if you want to upload Fingerprint
- **IsCardUpload** – pass true if you want to upload password
- **IsBioPasswordUpload** – pass true if you want to upload password

Response:

- **Message**: "Command Send Successfully."

Untitled Request

POST http://localhost:89/api/v2/WebAPI/UploadUser?APIKey=420113032112&EmployeeCode=123&EmployeeName=xyz&CardNumber=&SerialNumbers=C26044C84F352331

Params

KEY	VALUE	Bulk Edit
APIKey	420113032112	
EmployeeCode	123	X
EmployeeName	xyz	
CardNumber		
SerialNumbers	C26044C84F352331	
Key	Value	

Body

```
1 "Command Send Successfully"
```

200 OK 25 ms 471 B Save Response

Other Response Messages

- Device is not exists.
- API key is not correct

e) Delete Users from Biometric Device

POST /api/v2/WebAPI/DeleteUser

To Delete User from Biometric Device using Smart Office

Request JSON Object

- **APIKey (String):** Authentication-token key.
- **EmployeeCode (String):** Biometric Code of the Employee/User.
- **SerialNumber (String):** To which Biometric device Serial Number User has to be deleted, Use Comma (,) to separate multiple serial numbers.

Response:

- **Message:** "Delete Command Send Successfully."

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://localhost:89/api/v2/WebAPI/DeleteUser?APIKey=420113032112&EmployeeCode=123&SerialNumbers=C26044C84F352331
- Headers:** BUILD, Send, Save
- Params:** APIKey: 420113032112, EmployeeCode: 123, SerialNumbers: C26044C84F352331
- Body:** "Command Send Successfully"
- Status:** 200 OK, 181 ms, 471 B

Other Response Messages

- Device is not exists.
- API key is not correct

f) Fetch Live Users from Biometric

GET /api/v2/WebAPI/FetchLiveUsersFromBiometric

To fetch live Users from Biometric Device using Smart Office

Request JSON Object

- **APIKey (String):** Authentication-token key.
- **SerialNumber (String):** To which Biometric device Serial Number live User has to be fetched.

Response:

```
[  
 {  
   "EmployeeCode": "01",  
   "FPCount": "1",  
   "Privilege": "USER"  
 }  
 ]
```

Other Response Messages

- Device is not exists.
- API key is not correct

g) Set User Expiration

GET /api/v2/WebAPI/SetUserExpiration

To set the expiration rule of the Biometric User in Device using Smart Office

Request JSON Object

- **APIKey (String):** Authentication-token key.
- **SerialNumber (String):** Biometric Serial Number, If Set as 0 then it will execute all on Biometric devices.
- **EmployeeCode (String):** Biometric Code of the Employee/User.
- **ExpirationDate (String):** Expiration date of the User and the Format will be **(yyyy-MM-dd)**.

Response:

- **Message:** "Command Send Successfully."

The screenshot shows a Postman interface with the following details:

- Method:** GET
- URL:** http://localhost:89/api/v2/WebAPI/SetUserExpiration?APIKey=420113032112&SerialNumber=C26044C84F352331&EmployeeCode=123
- Body:** JSON
 - 1 "Command Send Successfully."
- Response Status:** 200 OK (10.72 s, 472 B)

Other Response Messages

- Device is not exists.
- API key is not correct

h) Fetch Biometric Commands

GET /api/WebAPI/GetDeviceCommands

To get and check the status of the commands issued by the user in the device.

Request Query String

- **APIKey (String):** Authentication-token key.
- **FromDate (DateTime):** From when the commands need to fetched.
- **ToDate(DateTime):** Till when the commands need to be fetched.
- **SerialNumbers :** this is optional parameters if you want to get the records of the specific devices then pass the comma separated serial number for example C2688C21CB2D172711,ADZV213760072

Response:

```
{
  "status": "success",
  "message": "",
  "records": [
    {
      "Title": "Get All Users from Biometric -MUM_BKC",
      ...
    }
  ]
}
```

```

    "DeviceCode": "MUM_BKC",
    "SerialNumber": "C2689C47030A2334",
    "CreationDate": "2020-11-05 15:08",
    "ExecutionDate": "1900-01-01 00:00",
    "Status": "Pending",
    "Response": ""

},
"result": true
}

```

Untitled Request

Params		Body	
Query Params		PRETTY RAW PREVIEW VISUALIZE JSON 1 { 2 "status": "success", 3 "message": "", 4 "records": [], 5 "result": true 6 }	

Other Response Messages

- From Date cannot be greater than To Date.

```

{
  "status": "failure",
  "message": "From Date [05-Nov-2020] cannot be greater than To Date [01-Nov-2020],",
  "records": [],
  "result": false
}

```

i) Photo Registration

POST /api/WebAPI/PhotoUploadInBiometric

To register user using base64 image into the biometric.

Request JSON Object

- **APIKey (String):** Authentication-token key.
- **SerialNumber (string):** Serial Number of the Biometric.
- **EmployeeName(string):** Name of the Employee.
- **EmployeeCode(string):** Employee Biometric Code from which the user need to be get registered.
- **Base64String(string):** base64 data of the registration image.

Response:

- **Message:** "Upload Command Send Successfully."

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://localhost:89/api/v2/WebAPI/PhotoUploadInBiometric
- Status:** 200 OK (11.89 s, 479 B)
- Body:**
 - Pretty: "Upload Command Send Successfully."

Other Response Messages

- Serial Number cannot be blank.
- EmployeeCode cannot be blank.
- EmployeeName cannot be blank.
- Base64String cannot be blank.
- API key is not correct

j) Block/unblock Users

GET /api/WebAPI/BlockUserinBiometric

To register user using base64 image into the biometric.

Request JSON Object

- **APIKey (String):** Authentication-token key.
- **EmployeeCode(string):** Employee Biometric Code from which the user need to be block/unblock.
- **SerialNumber (string):** Serial Number of the Biometric.
- **BlockUser (int):** 0 – to block the user, 1 – to unblock the user.

Response:

- **Message:** “Command Send Successfully.”

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://localhost:89/api/v2/WebAPI/BlockUserinBiometric?APIKey=420113032112&EmployeeCode=123&SerialNumber=C26044C84F35
- Headers:** BUILD, Send, Save
- Params:**

KEY	VALUE
APIKey	420113032112
EmployeeCode	123
SerialNumber	C26044C84F352331
BlockUser	1
Key	Value
- Body:**
 - Pretty: 200 OK 15.00 s 472 B
 - Raw: Save Response
 - Visualize: JSON

```
1 "Command Send Successfully."
```

Other Response Messages

- Biometric device not exist with the Serial Number
- API key is not correct

Note - There are very few models which support this command, make sure to cross check whether respective models supports this command or not from the seller.

k) Add/Update Employee

GET /api/v2/WebAPI/AddEmployee

To add/update Employee details in the application .

Request JSON Object

```
{
    "StaffCode": "12121",
    "StaffName": "abc",
    "Gender": "Male",
    "Status": "Working",
    "CompanySName": "Default",
    "DepartmentSName": "Default",
    "Location": "Default",
    "Designation": "Default",
    "Grade": "Default",
    "Team": "Default",
    "DOJ": "2021-04-28",
    "DOC": "2021-09-15",
    "DOB": "1987-06-01",
    "DOR": "3000-01-01"
}
```

Note - All the Date format like DOJ,DOC,DOB,DOR should be in yyyy-MM-dd format

Response:

```
{
    "Message": "Employee Added Successfully",
    "Result": true
}
```

The screenshot shows the Postman interface with the following details:

- Method:** POST
- URL:** http://localhost:89/api/v2/WebAPI/AddEmployee
- Body (JSON):**

```

1 {
2     "StaffCode": "12121",
3     "StaffName": "abc",
4     "Gender": "Male",
5     "Status": "Working",
6     "CompanySName": "Default",
7     "DepartmentSName": "Default",
8     "Location": "Default",
9     "Designation": "Default"
10 }

```
- Response:**

```

1 {
2     "Message": "Employee Added Successfully",
3     "Result": true
4 }

```
- Status:** 200 OK (7.22 s, 543 B)

I) Delete Employee

GET /api/v2/WebAPI/DeleteEmployee

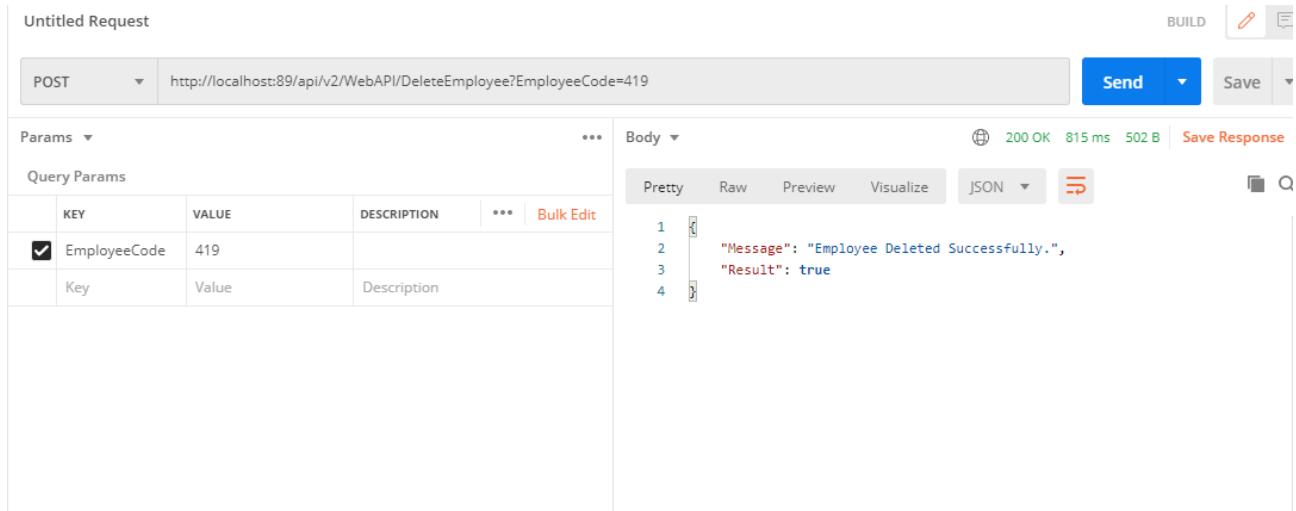
To delete the existing employee from the application.

Request JSON Object

- **EmployeeCode (string):** Employee Code to delete.

Response:

```
{  
    "Message": "Employee Deleted Successfully",  
    "Result": true  
}  
  
{  
    "Message": "Employee Not exists.",  
    "Result": false  
}
```



The screenshot shows a REST client interface with the following details:

- Untitled Request**: The title of the request.
- Method:** POST
- URL:** http://localhost:89/api/v2/WebAPI/DeleteEmployee?EmployeeCode=419
- Params:** A table showing a single query parameter: EmployeeCode with value 419.
- Body:** A JSON response object with two properties: "Message" and "Result".
- Status:** 200 OK, 815 ms, 502 B
- Buttons:** Send, Save, and other UI elements.

Query Params			
	KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/>	EmployeeCode	419	
	Key	Value	Description

Pretty	Raw	Preview	Visualize	JSON	Copy	Save
1	{					
2	"Message": "Employee Deleted Successfully.",					
3	"Result": true					
4	}					

J) Add Company

GET /api/v2/WebAPI/AddCompany

To delete the existing employee from the application.

Request JSON Object

```
{
  "BranchAddress": "648, 2nd Floor, Sri Janardhan Towers 5th Main, 1st Cross Rd, RBI Layout,
  JP Nagar 7th Phase, Naagarabhaavi, Bengaluru, Karnataka 560078",
  "BranchFullName": "SmartOffice payroll & Biometrics Solutions pvt ltd",
  "BranchShortName": "SmartOffice payroll & Biometrics Solutions pvt ltd",
  "BranchEmail": "SupportIndia@Smartofficepayroll.com",
  "BranchWebsite": "www.smartofficepayroll.com"
}
```

The screenshot shows the Postman interface with the following details:

- Method:** POST
- URL:** 45.118.183.175:86//api/v2/WebAPI/AddCompany?APIKey=521714042205
- Body:** JSON (selected)
- Request Body Content:**

```

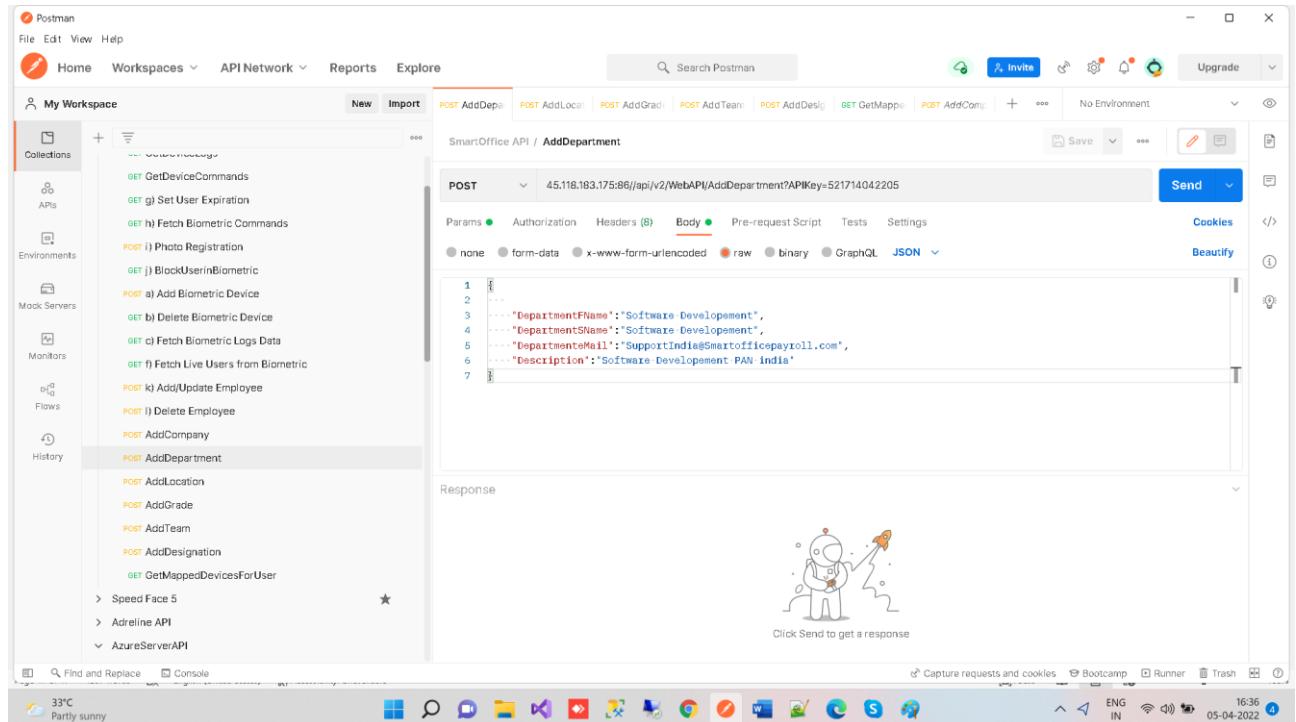
1  {
2    "BranchAddress": "648, 2nd Floor, Sri Janardhan Towers 5th Main, 1st Cross Rd, RBI Layout,
3    JP Nagar 7th Phase, Naagarabhaavi, Bengaluru, Karnataka 560078",
4    "BranchFullName": "SmartOffice payroll & Biometrics Solutions pvt ltd",
5    "BranchShortName": "SmartOffice payroll & Biometrics Solutions pvt ltd",
6    "BranchEmail": "SupportIndia@Smartofficepayroll.com",
7    "BranchWebsite": "www.smartofficepayroll.com"
  }

```

K) Add Department

Request Json

```
{
  "DepartmentFName": "Software Developement",
  "DepartmentSName": "Software Developement",
  "DepartmenteMail": "SupportIndia@Smartofficepayroll.com",
  "Description": "Software Developement PAN india"
}
```



L) Add Location

URL - 45.118.183.175:86//api/v2/WebAPI/AddLocation?APIKey=521714042205

Request Json

```
{
  "LocationName": "Bangalore1",
  "LocationCode": "Bangalore1",
  "eMail": "SupportIndia@Smartofficepayroll.com",
  "LocationLatitude": "12.9337623",
  "LocationLongitude": "77.5828713",
  "Radius": "500",
  "LocationFullAddress": "482, JPNagar - 560078"
}
```

POST ▼ 45.118.183.175:86//api/v2/WebAPI/AddLocation?APIKey=521714042205

Params ● Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ▾

```

1 {
2   ...
3   "LocationName": "Bangalore1",
4   "LocationCode": "Bangalore1",
5   "eMail": "SupportIndia@Smartofficepayroll.com",
6   "LocationLatitude": "12.9337623",
7   "LocationLongitude": "77.5828713",
8   "Radius": "500",
9   "LocationFullAddress": "482, JPNagar - 560078"
10 }

```

M) Add Designation

Request JSON –

```
{
  "DesignationsName": "Senior Software Developer",
  "DesignationCode": "Senior Software Developer"
}
```

URL - 45.118.183.175:86//api/v2/WebAPI/AddDesignation?APIKey=521714042205

POST ▼ 45.118.183.175:86//api/v2/WebAPI/AddDesignation?APIKey=521714042205

Params ● Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ▾

```

1 {
2   ...
3   "DesignationsName": "Senior Software Developer",
4   "DesignationCode": "Senior Software Developer"
5 }

```

N) Add Grade

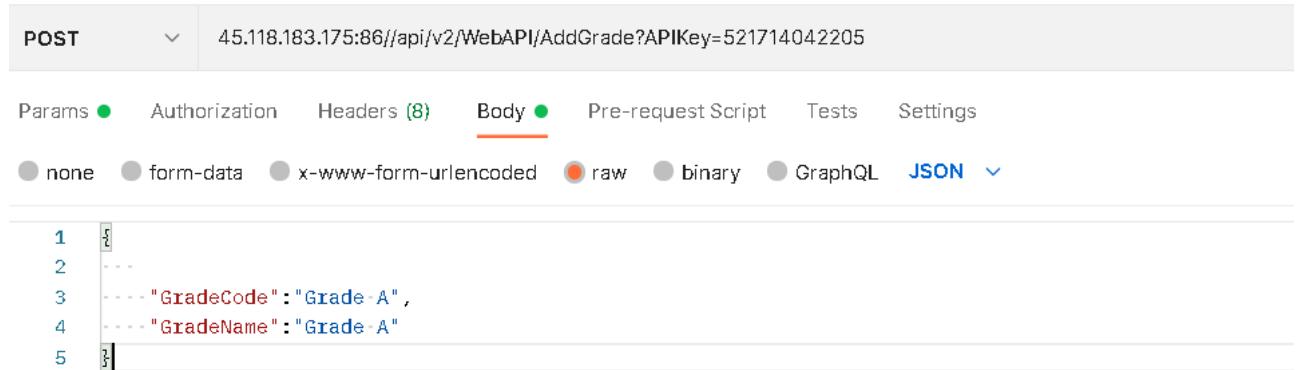
Request JSON –

```
{  
  
    "GradeCode": "Grade A",  
    "GradeName": "Grade A"  
}
```

URL – 45.118.183.175:86//api/v2/WebAPI/AddGrade?APIKey=521714042205

SmartOffice API / **AddGrade**

 Save



The screenshot shows the Postman application interface. A POST request is being made to the URL `45.118.183.175:86//api/v2/WebAPI/AddGrade?APIKey=521714042205`. The 'Body' tab is selected, showing a JSON payload:

```
1 {  
2     "GradeCode": "Grade A",  
3     "GradeName": "Grade A"  
4 }  
5 }
```

O)Add Team

Requested JSON –

```
{  
  
    "TeamCode": "Developement Team",  
    "TeamName": "Developement Team"  
}
```

URL - 45.118.183.175:86//api/v2/WebAPI/AddTeam?APIKey=521714042205

SmartOffice API / AddTeam

POST ▼ 45.118.183.175:86//api/v2/WebAPI/AddTeam?APIKey=521714042205

Params ● Authorization Headers (8) ● Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON ▼

```

1  {
2    ...
3    "TeamCode": "Development-Team",
4    "TeamName": "Development-Team"
5  }

```

P) ClearAllLogsFromDevice – This api will clear all the Attendance Log Records from Device

URL - 45.118.183.175:86//api/v2/WebAPI/ ClearAllLogsFromDevice?APIKey=521714042205& SerialNumber= C261185157361E2C

SmartOffice API / p) ClearAllLogsFromDevice

GET ▼ 45.118.183.175:86//api/v2/WebAPI/ClearAllLogsFromDevice?APIKey=521714042205& SerialNumber= C261185157361E2C Send ▼

Params ● Authorization Headers (6) ● Body ● Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> APIKey	521714042205			
<input checked="" type="checkbox"/> SerialNumber	C261185157361E2C			
Key	Value	Description		

Q) ClearLogsFromDeviceByTime - This api will clear all the Attendance Log Records from Device by specific time. This api only supported in Speed Face Model

URL - 45.118.183.175:86//api/v2/WebAPI/ClearLogsFromDeviceByTime?APIKey=521714042205& SerialNumber=C261185157361E2C&StartTime=2022-05-09 09:00&EndTime=2022-05-09 07:00

The screenshot shows the Postman interface with a GET request to the URL: 45.118.183.175:86/api/v2/WebAPI/ClearLogsFromDeviceByTime?APIKey=521714042205&SerialNumber=C261185157361E2... . The 'Params' tab is selected, displaying the following query parameters:

KEY	VALUE	DESCRIPTION
APIKey	521714042205	
SerialNumber	C261185157361E2C	
StartTime	2022-05-09 09:00	
EndTime	2022-05-09 07:00	
Key	Value	Description

R) TriggerUserOnlineEnrollment- This api will Trigger the Online Enrollment command

URL -

207.148.121.63:86/api/v2/WebAPI/TriggerUserOnlineEnrollment?APIKey=593910012317&SerialNumber=0426134400988&EmployeeCode=447789&EmployeeName=ritesh singh&backup_number=1

The screenshot shows the Postman interface with a GET request to the URL: 207.148.121.63:86/api/v2/WebAPI/TriggerUserOnlineEnrollment?APIKey=593910012317&SerialNumber=0426134400988&EmployeeCode=447789&EmployeeName=ritesh singh&backup_number=1 . The 'Params' tab is selected, displaying the following query parameters:

KEY	VALUE	DESCRIPTION	... Bulk Edit
APIKey	593910012317		
SerialNumber	0426134400988		
EmployeeCode	447789		
EmployeeName	ritesh singh		
backup_number	1		
Key	Value	Description	