

EdSoft API Documentation

API Key = [Your API Key](#)

Secret Key = [Your Secret Key](#)

1. Schedule a Call API

Endpoint: <https://backend.edsoft.us/client/schedule>

Parameters for scheduling class:

Parameter	Type	Description	Example
classTitle (String type)	Required	your class title	Trigonometry
classInfo (String type)	Required	your class info	About ALL SILVER TEA CUPS Rules
classDateTime (String type)	Required	The class's start time. Format is YYYY-MM-DD HH:MM AM/PM Note: If time is less than 10 hours, Pls prefix with "0", example: 06 PM, instead of just 6 PM	2019-10-04 06:30 AM 2019-10-04 10:30 PM
timezone (String type)	Optional	The timezone that applies to this class. The default is Etc/UTC Please select " timezone " from list of timezones	America/Chicago Click here for List of Timezones
classDuration (Integer type)	Optional	How long the class is scheduled for, in minutes. Default is 30 minutes. Max duration is 150 minutes	30
classRecording (String type)	Required	Whether to record the class or not. Possible values: yes (record) or no (don't record). Default is yes. Note: Recording doesn't start by Default, Presenter has to click recording button.	no

classAutoStart (Boolean type)	Optional	Choose true (or) false for auto starting class. Default is false	true
recordingAutoStart (Boolean type)	Optional	Choose true (or) false for auto recording class. Default is false	true
classVideoRes (Integer type)	Optional	Choose any one video resolution Note: Available resolutions are 480 (or) 640 (or) 720 (or) 1080. Default is 480	720
extendSession (Boolean type)	Optional	Choose true (or) false for session extension. Default is true	true
apiKey (String type)	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{
  "apiKey": "YOUR API KEY"
}
```

Response Body:
JSON body If success:

```
{
  "success": true,
  "classData": {
    "userName": "adam",
    "meetingId": "123456",
    "classAutoStart": false,
    "recordingAutoStart": false,
    "classVideoRes": 480,
    "extendSession": true,
    "classTitle": "dev sch check",
    "classInfo": "about dev check",
    "classDateTime": "2020-04-20 02:39 PM",
    "classTimeZoneName": "Asia/Kolkata",
    "classDuration": 10,
    "classRecording": "yes",
    "classType": "private",
  }
}
```

```
"teacherUrl": "teacher url",
"studentUrl": "student url",
"expertUrl": "expert url"
},
"message": "&#x1F603 Successfully scheduled class "
}
```

JSON Response Body if Failure:

409 STATUS CODE:

```
{
  "success" : false,
  "message" : `&#x1F610 Please Activate your account.`
}
```

500 STATUS CODE:

```
{
  "Success" : false,
  "Message" : throws the code where you went wrong
}
```

List of headers need to be added in Headers section in post request method:

Content-Type : application/json

authorization : Bearer Token

Explanation:

JWT Token Parts => 1. Header 2. Payload 3. Signature

Both JWT Header and Payload content type must be in application/json format

JWT Header = {"alg":"HS256","typ":"JWT"}

JWT Payload = {"classTitle":"your class title","classInfo":"your class

info","classDateTime":"2019-10-04 06:25

PM","timezone":"America/Chicago","classDuration":30,"classRecording":"yes","classAutoStart":true,

"recordingAutoStart":true,"classVideoRes":720,"apiKey":"YOUR API KEY HERE"}

List of Timezone:

<https://docs.google.com/document/d/1xwk3SF7ErnuwYga84yVHuQmE-EOVDMeFINxSCA4XopA/edit>



Note: Encoding JWT Header and JWT Payload must be in base64URLencode format, Not in base64encode format

Encoded JWT Header = **base64URLencode**(JWT Header);

Encoded JWT Payload = **base64URLencode**(JWT Payload);

Ref: <https://base64.guru/standards/base64url/encode>

JWT Signature = **HMAC-SHA256**(Encoded JWT Header.Encoded JWT Payload including "YOUR SECRET KEY");

Ref: <https://www.liavaag.org/English/SHA-Generator/HMAC/>

//concatenation of Header, Payload & signature with **.** or DOT, after Header & Payload

Token = Encoded JWT Header **+** **.** + Encoded JWT Payload **+** **.** + JWT Signature

Reference site to check if token is generated properly : <https://jwt.io>

Sample Node.js Example

```
var base64url = require('base64url'); // used for base64URI encoding from base64
```

```
var formData = require('form-data'); // for request body
```

```
var request = require('request'); // for sending request
```

```
Var base64_Header = Buffer.from(`{"alg":"HS256","typ":"JWT"}`).toString('base64');
```

```
Var encodedbase64Url_Header = base64url.fromBase64(base64_Header );
```

```
Var base64_Payload = Buffer.from(`{"classTitle":"s1","classInfo":"about  
s1","classDateTime":"2019-10-04 5:00
```

```
PM","classDuration":30,"classRecording":"yes","classAutoStart":true,"recordingAutoStart":true,"clas  
sVideoRes":720,"apiKey":"YOUR API KEY"}`).toString('base64');
```

```
Var encodedbase64Url_Payload = base64url.fromBase64(base64_Payload );
```

```
var base64_Signature = crypto.createHmac('sha256','YOUR SECRET  
KEY').update(`${encodedbase64Url_Header }${encodedbase64Url_Payload }`).digest('base64');
```

```
Var encodedbase64Url_Signature = base64url .fromBase64(base64_Signature);
```

```
Var token =
```

```
`${encodedbase64Url_Header}.${encodedbase64Url_Payload}.${encodedbase64Url_Signature}`
```

```
Var formData = { "apiKey":"YOUR API KEY" };
```

```
request.post({  
  headers: {  
    'Content-Type': 'application/json',  
    'authorization': `Bearer ${token}`,  
  },  
  url: 'https://backend.edsoft.us/client/schedule',  
  formData: formData  
},  
(err, httpResponse, body)=>{  
  
  }  
})
```

NOTE: The process of sending the request is similar to SCHEDULE A CLASS. Just add “parameters that needs to be updated” in the payload.

2. Add Attendees API

Description: Creating new learners under Admin Account, Learners can get into their dashboard by passing their fullName as their password with provided email

Endpoint: <https://backend.edsoft.us/client/addAttendees>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response.	771231



attendeeList	Optional	List of attendees in JSON format	List of attendees Key value pairs
apiKey	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{  
    "apiKey": "YOUR API KEY"  
}
```

Note: Please look into the below example of json payload which needs to be converted into base64url according to the requirements.

JSON Payload Example:

```
{  
  "meetingId": 123456,  
  "attendeeList": [  
    {  
      "fullName": "attendee10",  
      "email": "attendee10@gmail.com",  
      "countryCode": 91,  
      "phoneNumber": 9454545456  
    },  
    {  
      "fullName": "attendee11",  
      "email": "attendee11@gmail.com",  
      "countryCode": 91,  
      "phoneNumber": 9454545457  
    },  
    {  
      "fullName": "attendee12",  
      "email": "attendee12@gmail.com",  
      "countryCode": 91,  
      "phoneNumber": 9454545458  
    }  
  ]  
}
```

Response Body:**IF SUCCESS:**

```
{
  "success": true,
  "attendeesAdded": [
    {
      "attendeeEmail": "attendee10@gmail.com",
      "attendeeld": "0ASGP1830645"
    },
    {
      "attendeeEmail": "attendee11@gmail.com",
      "attendeeld": "7CLPGK633273"
    },
    {
      "attendeeEmail": "attendee12@gmail.com",
      "attendeeld": "4W62HD507889"
    }
  ],
  "attendeesRejected": [],
  "message": "rejected Attendees are those who are Registered Earlier in EdSoft VC Platform but not verified"
}
```

3. Get Single VC Url API

Description: vc url for a viewer or expert or presenter

Endpoint: <https://backend.edsoft.us/client/getInUrl>

Method: POST method

Parameter	Type	Condition	Description	Example
presenterName	Optional	If role is presenter	Join url for presenter	John
expertName	Optional	If role is expert	Join url for expert	Shiva



attendeeEmail	Optional	If role is viewer	EdSoft registered email of this user	attendee@gmail.com
attendeeld	Optional	If role is viewer	EdSoft attendee Id of this user	xxxxxxxxxx
meetingId	Required		meetingId received from SCHEDULE A CLASS API response	789133
role	Required		presenter (or) viewer (or) expert	presenter
apiKey	Required		EdSoft provided API Key	YOUR API KEY

Request Body:

```
{
  "apiKey": "YOUR API KEY"
}
```

Json payload example for viewer:

```
{
  "attendeeEmail": "attendee1@gmail.com",
  "attendeeld": "O3VXYM084605",
  "meetingId": "1592790415",
  "role": "viewer",
  "apiKey": "Your api key"
}
```

Response Body:

IF SUCCESS:

```
{
  "success": true,
  "getInUrl": "YOUR url for vc platform"
}
```

IF FAILS:

```
{
  success: false,
```



```
message: `Meeting Not found`,  
err  
}
```

4. Get Bulk VC Urls API

Description: getting vc urls for given requirements

Endpoint: <https://backend.edsoft.us/client/getBulkUrls>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response.	771231
presenterName	Optional	Name of the presenter who takes a class.	John
attendeeList	Optional	List of attendees in JSON format	List of attendees Key value pairs
expertList	Optional	List of experts. By default Max 2 experts are allowed.	List of expert key value pairs

Request Body:

```
{  
  "apiKey": "YOUR API KEY"  
}
```

Note: Please look into the below example of json payload which needs to be converted into base64url according to the requirements.

JSON Payload Example:

```
{  
  "meetingId": "xxxxxxxxxx",
```

```
"presenterName": "thiru",
"attendeeList": [
  {
    "attendeeEmail": "attendee2@gmail.com",
    "attendeeId": "QEA073509457"
  },
  {
    "attendeeEmail": "attendee1@gmail.com",
    "attendeeId": "O3VXYM084605"
  },
  {
    "attendeeEmail": "attendee3@gmail.com",
    "attendeeId": "C2DGF521956"
  }
],
"expertList": [
  {
    "name": "expert1"
  },
  {
    "name": "expert2"
  }
]
}
```

Response Body

IF SUCCESS:

```
{
  "success": true,
  "attendeeUrls": [
    {
      "attendee": "attendee1",
      "join_url": "You will get JOIN URL for this attendee"
    },
    {
      "attendee": "attendee2",
      "join_url": "You will get JOIN URL for this attendee"
    }
  ]
}
```

```
],  
"expertUrls": [  
  {  
    "expert": "expert_name1",  
    "join_url": "You will get JOIN URL for this expert"  
  },  
  {  
    "expert": "expert_name2",  
    "join_url": "You will get JOIN URL for this exert"  
  }  
],  
"presenterUrl": "You will get presenter URL HERE"  
}
```

5. Recording API

Endpoint: <https://backend.edsoft.us/client/getRecordingUrls>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response	789133
apiKey	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{  
  "apiKey": "YOUR API KEY"  
}
```

Response Body:

IF SUCCESS:

```
{  
  "success": true,  
  "recordingUrls": [  
    {  
      "Watch": "1st recording watch url",  
      "Download": "1st recording download url"  
    },  
  ],  
}
```

```

{
  "Watch": "2nd recording watch url",
  "Download": "2nd recording download url"
},
.
.
]
}

```

IF FAILURE:

```

{
  "success": false,
  "message": "Recordings Not Available."
}

```

6. Update/Edit Class API

Endpoint: <https://backend.edsoft.us/client/updateSchedule>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response	789133
classTitle	Optional	Title for the class to be updated	Your title
classInfo	Optional	Info for the class to be updated	Your info
classDateTime	Optional	The class's start time. Format is YYYY-MM-DD HH:MM AM/PM Note: If time is less than 10 hours, Pls prefix with "0", example: 06 PM, instead of just 6 PM	2019-11-05 03:50 PM
timezone	Optional	Select the timezone from the link that you want to apply to update this scheduled class.	America/Chicago Click here for List of Timezones
classDuration	Optional	How long the class is scheduled for, in minutes.	45

		Default is 30 minutes. Max duration is 150 minutes	
classRecording	Optional	Whether to record the class or not. Possible values: yes (record) or no (don't record). Default is yes. Note: Recording doesn't start by Default, Presenter has to click recording button.	yes

Request Body:

```
{
  "apiKey": "YOUR API KEY"
}
```

Response Body:
IF SUCCESS:

```
{
  "success": true,
  "classData": {
    "meetingId": "YOUR Meeting id",
    "classTitle": "Your class title",
    "classInfo": "Your class info",
    "classDateTime": "Your class datetime",
    "timezone": "Your class timezone",
    "classDuration": "Your class duration",
    "classRecording": "Your class recording"
  },
  "message": "&#x1F60A; Scheduled class updated successfully"
}
```

7. Scheduled Info API

Endpoint: <https://backend.edsoft.us/client/getScheduledInfo>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response	789133



apiKey	Required	EdSoft provided API Key	YOUR API KEY
--------	----------	-------------------------	--------------

Request Body:

```
{
  "apiKey": "YOUR API KEY"
}
```

Response Body:

```
{
  "success": true,
  "scheduleData": {
    "classTitle": "dev sch check",
    "classInfo": "about dev check",
    "classDateTime": "2020-04-20 02:39 PM",
    "timezone": "Asia/Kolkata",
    "classDuration": 10,
    "classRecording": "yes",
    "classType": "private",
    "teacherUrl": "teacher url",
    "studentUrl": "student url",
    "expertUrl": "expert url"
  }
}
```

8. Delete/Cancel a Class API

Endpoint: <https://backend.edsoft.us/client/deleteSchedule>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response	789133
apiKey	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{  
  "apiKey": "YOUR API KEY"  
}
```

Response Body:

Status code: 200

```
{  
  "success": true,  
  "message": "Class with meetingId XXXX has been deleted successfully."  
}
```

Status code: 409

```
{  
  "success": false,  
  "message": "meetingId Not found."  
}
```

9. Deny Student API

Description: denying student accessing VC Platform

Endpoint: <https://backend.edsoft.us/client/denyStudentAccess>

Method: POST method

Parameter	Type	Description	Example
attendeeEmail	Required	EdSoft registered email of this user	attendee@gmail.com
attendeeld	Required	EdSoft attendee Id of this user	xxxxxxxxxxx
role	Required	presenter (or) viewer (or) expert	presenter
apiKey	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{  
  "apiKey": "YOUR API KEY"  
}
```

Json payload example for viewer:

```
{
  "attendeeEmail": "attendee1@gmail.com",
  "attendeeId": "O3VXYM084605",
  "meetingId": "1592790415",
  "apiKey": "Your api key"
}
```

Response Body:

IF SUCCESS:

```
{
  "success": true,
  "message": "&#x1F60A; denied student access successfully"
}
```

IF FAILS:

```
{
  success: false,
  message: `student with provided info not found`
}
```

10. Attendees Report API

Endpoint: <https://backend.edsoft.us/client/getAttendeesReport>

Method: POST method

Parameter	Type	Description	Example
meetingId	Required	meetingId received from SCHEDULE A CLASS API response	789133
apiKey	Required	EdSoft provided API Key	YOUR API KEY

Request Body:

```
{
  "apiKey": "YOUR API KEY"
}
```


Response Body:**Status code: 200**

```
{
  "success": true,
  "attendeesReport": [
    {
      "clientIps": [
        "Client ip1"
      ],
      "startTimes": [
        "2020-06-19 10:44 AM"
      ],
      "endTimes": [
        "2020-06-19 11:22 AM"
      ],
      "fullName": "ex",
      "role": "expert"
    },
    {
      "clientIps": [
        "Client ip2"
      ],
      "startTimes": [
        "2020-06-19 10:47 AM"
      ],
      "endTimes": [
        "2020-06-19 10:54 AM"
      ],
      "fullName": "stu",
      "role": "viewer"
    },
    ,,,,,etc
  ],
  "presenterStartTime": "2020-06-19T04:09:35.242Z",
  "presenterEndTime": "2020-06-19T10:06:34.852Z"
}
```

Status code: 409

```
{
  "success": false,
```



```
"message": "Class with meetingId xxxxxxx not found"  
}
```