

Get Token

Returns a JWT token needed to authenticate all API requests. Tokens expire after 1 hour, and should be internally stored until expiration for later use in all API requests.

- **URL**

<https://auth.recomendo.ai/v1/auth>

- **Method:**

POST

- **Header Params:**

Authorization: Basic (must add base64 encoded client ID : client secret)

- **Success Response:**

- **Code:** 200

- **Content:**

```
{"success":true,"messageCode":"CLIENT_AUTHENTICATED","payload":
{"token":"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJjbG1bnRjZCI6ImZlMGQ
xMTdmZTk2NDQ4NjI4YTAwNjJmNmNjZTU1ZWJmIiwiaWVyc2lvdCI6IjAwMDEiLCJpY
XQiOiE1MzgwNjI5NTYsImV4cCf6MTUzODA2NjU1Nn0.koqC_U20Ku4IaAKRZtmkb8AH9sY
JoFjzOvqd_0QSsPY","tokenType":"Bearer"}}
```

- **Error Response:**

- **Code:** 401 UNAUTHORIZED

- **Content:** {"message":"Unauthorized"}

Sample Call:

```
$ echo -n
"61b9a60ee5ef4efdb7048bc444faa196:NG2ey6h2C8KJ3M3zPbpvvy3KG1AjYBAwQbM5yuENy2HyDFAv
35mYXEwwnNMGAY77" | base64
NjFiOWE2MGVlNWVmNGVmZGI3MDQ4YmM0NDRmYWExOTY6TkcyZXk2aDJD0EtKM00zelBicHZ2eTNLRzFBal
1CQXdRYk01eXVFTnkySH1ERkF2MzVtWVhFd3duTk1HQXk3Nw==
$ curl -X POST -H 'Authorization: Basic
NjFiOWE2MGVlNWVmNGVmZGI3MDQ4YmM0NDRmYWExOTY6TkcyZXk2aDJD0EtKM00zelBicHZ2eTNLRzFBal
1CQXdRYk01eXVFTnkySH1ERkF2MzVtWVhFd3duTk1HQXk3Nw==' https://auth.recomendo.ai/v1/
auth
```


Sending Events

All events for creating, deleting and updating users and items follow the same pattern and only differ in the query parameters. See “Events Modeling” in the API manual.

- **URL:**

<https://events.recomendo.ai/v1/events.json>

- **Method:**

POST

- **Header Params:**

Authorization: Bearer (must add token)
Accept-Encoding: gzip (optional)
Content-Type: application/json

- **Success Response:**

- **Code: 201**

- Content:**

```
HTTP/1.1 201 Created
Server: spray-can/1.3.2
Date: Tue, 02 Jun 2018 23:13:58 GMT
Content-Type: application/json; charset=UTF-8
Content-Length: 57
```

```
{"eventId":"PVj0IP6AJ5PgSiGQW6pgswAAAUhc7EwZpCfSj5bS5yg"}
```

- **Error Response:**

- **Code: 401 UNAUTHORIZED**

- Content:** {"message":"Unauthorized"}

Sample Call:

```
$ curl -X POST -H 'Authorization: Bearer f2GcxZWef...' -H 'Content-Type: application/json' -d '{
  "event" : "$set",
  "entityType" : "user",
  "entityId" : "1",
  "properties" : {
    "user_agent" : "Mozilla/5.0 (X11; Linux x86_64) ...",
    "ip_address" : "12.34.94.107"
  }
}' https://events.recomendo.ai/v1/events.json
```

Sending a Training Request

Training is the process by which all the events are processed and analyzed and the predictive model is built. Training request should be sent whenever there is new data in the application to and we need to create a new model.

There is a 30 min limit for sending Training requests, and you should control in your application not sending training requests more than once every 30 mins.

- **URL:**

<https://client.recomendo.ai/v1/train>

- **Method:**

POST

- **Header Params:**

Authorization: Bearer (must add token)
Accept-Encoding: gzip (optional)

- **Success Response:**

- **Code:** 200
Content:

```
HTTP/1.1 200 OK
Date: Fri, 28 Sep 2018 16:08:42 GMT
Content-Type: application/json
Content-Length: 71
Connection: keep-alive
```

```
{"success":true,"messageCode":"EXECUTE_TRAIN_SCHEDULED","payload":null}
```

- **Error Response:**

- **Code:** 401 UNAUTHORIZED
Content: {"message":"Unauthorized"}

Sample Call:

```
$ curl -X POST -H 'Authorization: Bearer f2GcxZWef...' https://client.recomendo.ai/v1/train
```

Getting Predictions

Returns JSON with items and scores according to the query sent. See "Prediction Queries" in the API manual.

- **URL:**

<https://predictions.recomendo.ai/v1/queries.json>

- **Method:**

POST

- **Header Params:**

Authorization: Bearer (must add token)
Accept-Encoding: gzip
Content-Type: application/json

- **Success Response:**

- **Code:** 200
Content:

```
{ "itemScores": [ { "item": "1953", "score": 6.0684257 },  
  { "item": "625", "score": 5.41189 }, { "item": "1538", "score": 4.3050656 },  
  { "item": "1485", "score": 4.2598968 }, { "item": "1789", "score": 3.5584452 },  
  { "item": "754", "score": 3.4158173 }, { "item": "725", "score": 3.2395444 },  
  { "item": "2009", "score": 1.2909842 }, { "item": "2055", "score": 1.1727203 },  
  { "item": "1300", "score": 0.7765288 } ] }
```

- **Error Response:**

- **Code:** 401 UNAUTHORIZED
Content: { "message": "Unauthorized" }

Sample Call:

```
$ curl -H "Authorization: Bearer  
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJjbG1bnRjZCI6IjcwMTU2OTViMmQ0MTRhNjg4OTA1MzJiOTgzNWQ1NW5kIiwiaWF0IjE1MzY0MTcxMDQsImV4cCI6MTUzNjQyMDcwNH0uZjwpeGBjFRFXS2Do5gml6ItbFLO6Ab0CDPQvfti0SGY" -XPOST -d '{  
  "user" : "1"  
}' https://predictions.recomendo.ai/v1/queries.json
```