Internal DownToMeet API (1.0.0)

Download OpenAPI specification: Download

Authentication

cookieSession

Session stored in a cookie.

(If you're reading this in the API documentation, ignore the "query parameter name" below. It is ignored at runtime.)

Security Scheme Type	API Key
Query parameter name:	COOKIE

Get the list of in-person meetups

If the required parameters were not specified correctly, an error is returned

QUERY PARAMETERS

```
required

number [ -90 .. 90 ]

The latitude of the center of search

number [ -180 .. 180 ]

required

The longitude of the center of search

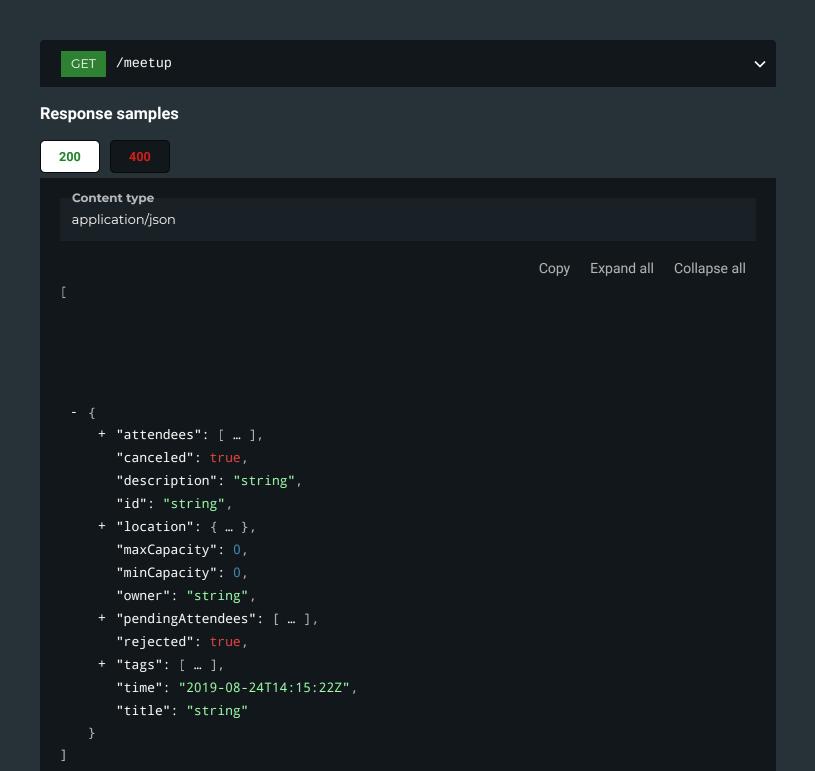
radius
required

number

Desired search radius (kilometers)
```

Responses

```
> 200 OK
> 400 Bad Request
```

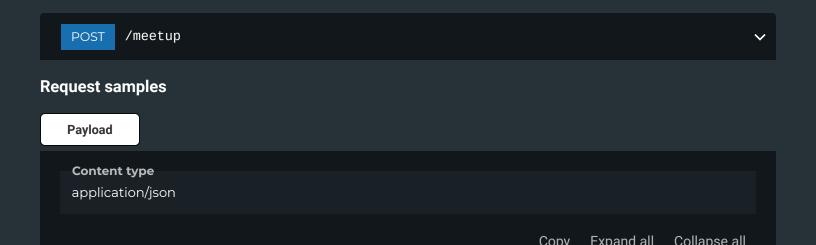


Post a new meetup

AUTHORIZATIONS:	cookieSession
REQUEST BODY SCHEMA:	application/json
The meetup to create	
→ description	string
→ location >	object (location)
→ maxCapacity	integer
→ minCapacity	integer
⊣ tags	Array of strings
⊣ time	string <date-time></date-time>
⊢ title	string

Responses

> 201 Created
> 400 Bad Request



Response samples

201

400

```
Content type
application/json
                                                               Сору
                                                                     Expand all
                                                                                 Collapse all
- "attendees": [
     "string"
  "canceled": true,
  "description": "string",
  "id": "string",
- "location": {
   + "coordinates": { ... },
      "name": "string",
     "url": "string"
  },
  "maxCapacity": 0,
  "minCapacity": 0,
  "owner": "string",
- "pendingAttendees": [
     "string"
  ],
  "rejected": true,
- "tags": [
     "string"
```

```
"time": "2019-08-24T14:15:22Z",
"title": "string"
}
```

Get the list of remote meetups

If the required parameters were not specified correctly, an error is returned

QUERY PARAMETERS

→ tags required

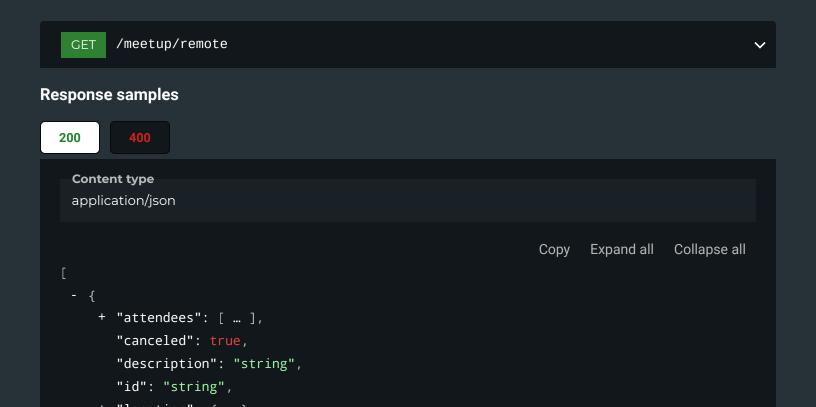
Array of strings

Interests to search for

Responses

> 200 OK

> 400 Bad Request



```
"nocation": { ... },
    "maxCapacity": 0,
    "minCapacity": 0,
    "owner": "string",
    + "pendingAttendees": [ ... ],
    "rejected": true,
    + "tags": [ ... ],
    "time": "2019-08-24T14:15:22Z",
    "title": "string"
}
```

Get the specified meetup's information

If the specified meetup does not exist, an error is returned

PATH PARAMETERS

→ id stringrequired ID of the desired meetup

Responses

> 200 OK
 > 400 Bad Request
 > 404 Not found



```
application/JSon
                                                                     Expand all
                                                                                Collapse all
                                                              Copy
- "attendees": [
      "string"
  ],
  "canceled": true,
  "description": "string",
  "id": "string",
- "location": {
  + "coordinates": { ... },
      "name": "string",
     "url": "string"
  },
  "maxCapacity": 0,
  "minCapacity": 0,
  "owner": "string",
- "pendingAttendees": [
     "string"
  ],
  "rejected": true,
- "tags": [
      "string"
  "time": "2019-08-24T14:15:22Z",
  "title": "string"
```

Delete the specified meetup

If the specified meetup does not exist, an error is returned

AUTHORIZATIONS:

cookieSession

PATH PARAMETERS

→ id

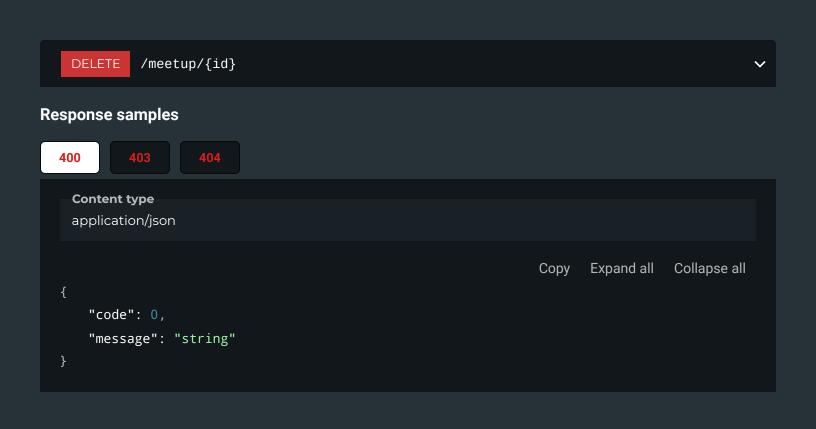
required

string

ID of the desired meetup

Responses

```
— 204 No Content
> 400 Bad Request
> 403 Forbidden
> 404 Not found
```



Patch the specified meetup

```
AUTHORIZATIONS: cookieSession

PATH PARAMETERS

id string ID of the desired meetup

REQUEST BODY SCHEMA: application/json

The updated meetup information

description string
```

Responses

```
> 200 OK
> 400 Bad Request
> 403 Forbidden
> 404 Not found
```

```
PATCH /meetup/{id}

Request samples

Payload

Content type
application/json

Copy Expand all Collapse all

{
  "description": "string",

  "name": "string",
  "url": "string",
  "url": "string"
},
  "maxCapacity": 0,
```

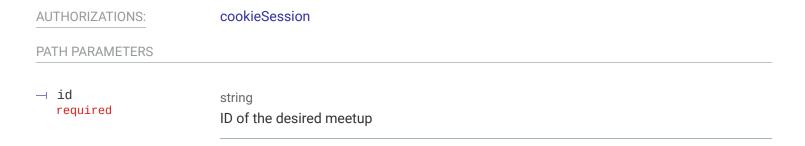
```
"minCapacity": 0,
- "tags": [
        "string"
],
      "time": "2019-08-24T14:15:22Z",
      "title": "string"
}
```

Response samples

```
200
 Content type
 application/json
                                                                Copy Expand all
                                                                                 Collapse all
 - "attendees": [
      "string"
   ],
   "canceled": true,
   "description": "string",
   "id": "string",
 - "location": {
    + "coordinates": { ... },
       "name": "string",
       "url": "string"
   },
   "maxCapacity": 0,
   "minCapacity": 0,
   "owner": "string",
 - "pendingAttendees": [
       "string"
   ],
   "rejected": true,
 - "tags": [
       "string"
    "time": "2019-08-24T14:15:22Z",
   "title": "string"
```

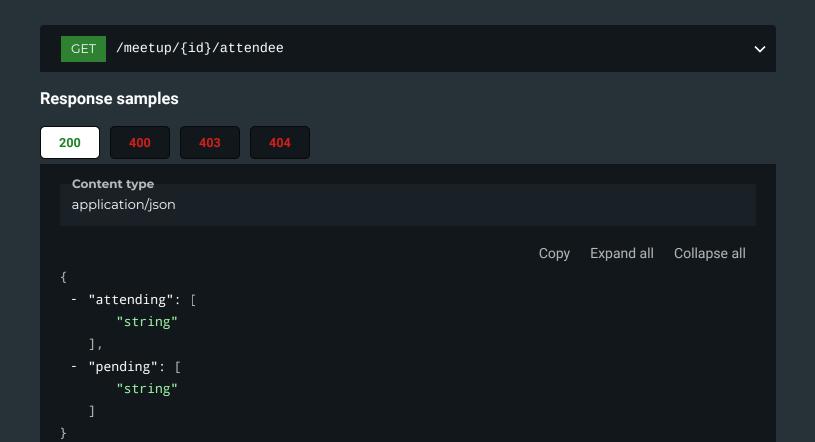
Get list of attending and pending attendees for a specified meetup

If the specified meetup does not exist, an error is returned



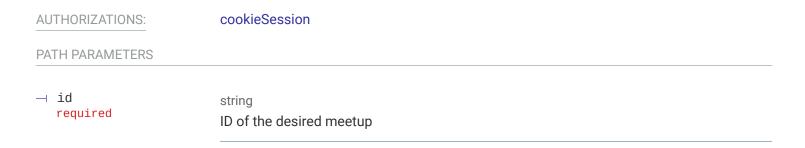
Responses

```
> 200 OK
> 400 Bad Request
> 403 Forbidden
> 404 Not found
```



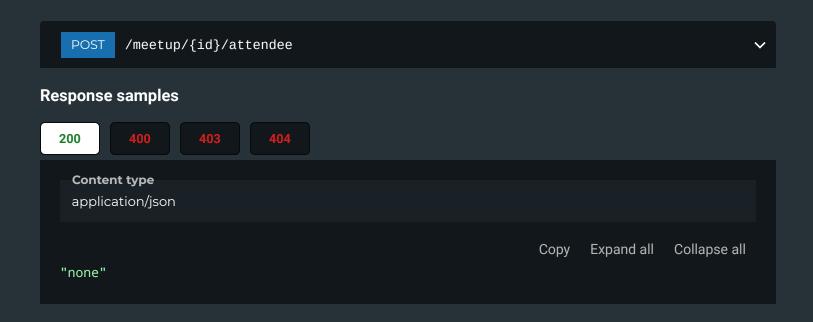
Post the current user's attendee status (to "pending") for the specified meetup

If the specified meetup does not exist, an error is returned



Responses

> 200 OK
> 400 Bad Request
> 403 Forbidden
> 404 Not found



Patch the current user's attendee status for the specified meetup

If the specified meetup does not exist, an error is returned

AUTHORIZATIONS: cookieSession

PATH PARAMETERS

Id string ID of the desired meetup

REQUEST BODY SCHEMA: application/json

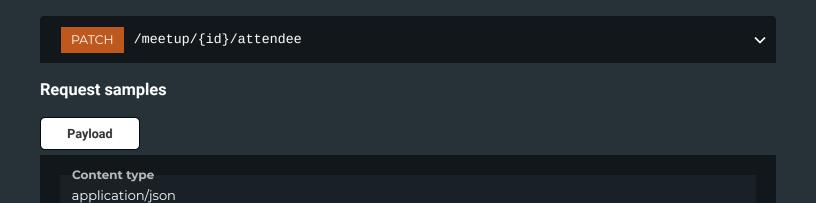
The id of the user being patched, plus their attendee status. Let attendee be empty if patching current user attendee string

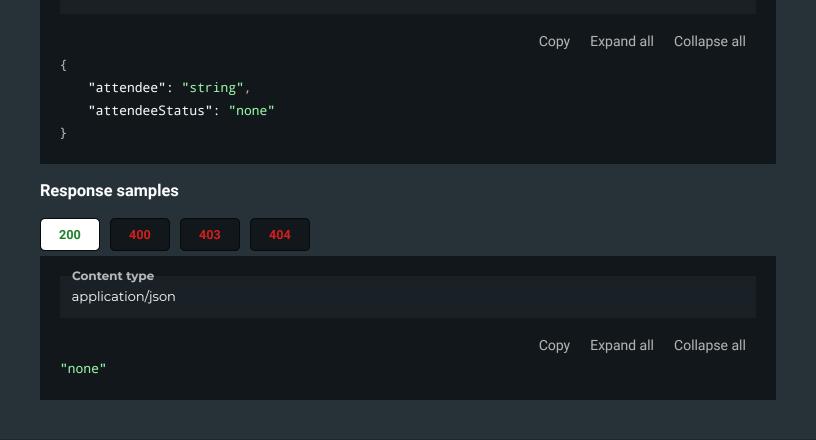
attendee string

attendeeStatus string (attendeeStatus)
Default: "none"
Enum: "attending" "pending" "rejected" "none"

Responses

> 200 OK
> 400 Bad Request
> 403 Forbidden
> 404 Not found



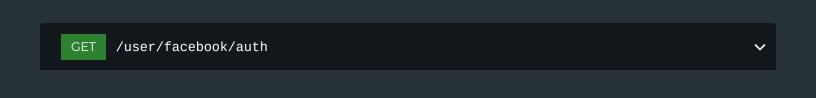


Facebook OAuth login

Start a Facebook OAuth login flow here.

Responses

> 303 Redirect to Facebook login prompt.



Facebook OAuth redirect

If authentication fails, the user is not logged in.



Responses

> 200 Simple HTML file that will redirect user to the home page.

> 303 Redirect to home page.

GET /user/facebook/redirect

Google OAuth login

Start a Google OAuth login flow here.

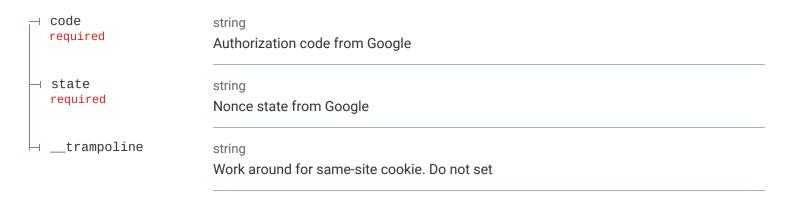
Responses

> 303 Redirect to Google login prompt.

Google OAuth redirect

If authentication fails, the user is not logged in.

QUERY PARAMETERS



Responses

- **> 200** Simple HTML file that will redirect user to the home page.
- > 303 Redirect to home page.

GET /user/google/redirect

Log out the user

Responses

— 204 Remove session cookie and redirect to home page.

~

Get the current user's information

If user is not logged in, an error response is returned.

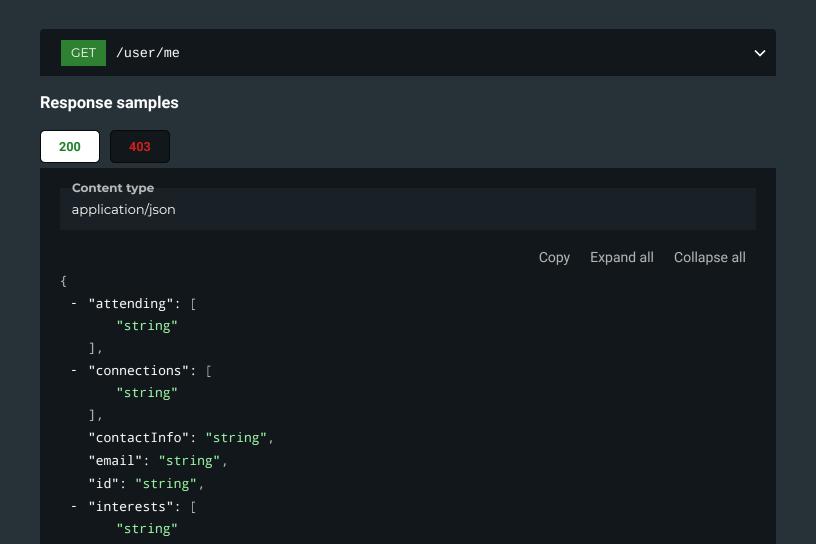
AUTHORIZATIONS:

cookieSession

Responses

> 200 OK

> 403 Forbidden



```
],
    "joinDate": "2019-08-24T14:15:22Z",
- "location": {
        "lat": -90,
        "lon": -180
    },
    "name": "string",
- "ownedMeetups": [
        "string"
    ],
- "pendingApproval": [
        "string"
    ],
    "profilePic": "string"
}
```

Get the specified user's information

If specified user does not exist, an error is returned.

PATH PARAMETERS

→ id string
required ID of the desired user

Responses

> 200 OK
> 404 Not found

GET /user/{id}

Response samples

```
Content type
application/json
                                                              Copy
                                                                     Expand all
                                                                                Collapse all
- "attending": [
      "string"
  ],
- "connections": [
      "string"
  ],
  "contactInfo": "string",
  "email": "string",
  "id": "string",
- "interests": [
      "string"
  "joinDate": "2019-08-24T14:15:22Z",
- "location": {
     "lat": -90,
      "lon": -180
  },
  "name": "string",
- "ownedMeetups": [
      "string"
  ],
- "pendingApproval": [
     "string"
  "profilePic": "string"
```

Patch the specified user

If specified user does not exist or current user is not the specified user, an error is returned.

→ id

required

string

ID of the desired user

REQUEST BODY SCHEMA: application/json

Updated user information

→ attending Array of strings (meetupID)

→ connections Array of strings

→ contactInfo string

⊣ email string

→ id string (userID)

required

string (userID)

required

output

Description:

| Comparison of the comparison of

→ interests Array of strings

→ joinDate string <date-time>

→ location > object (coordinates)

→ name string

→ ownedMeetups Array of strings (meetupID)

→ pendingApproval Array of strings (meetupID)

□ profilePic string

Responses

> 200 OK

> 403 Forbidden

> 404 Not found

PATCH /user/{id}

v

Request samples

Payload

```
Content type
application/json
                                                                                 Collapse all
                                                                      Expand all
                                                               Copy
- "attending": [
      "string"
  ],
- "connections": [
      "string"
  ],
  "contactInfo": "string",
  "email": "string",
  "id": "string",
- "interests": [
      "string"
  ],
  "joinDate": "2019-08-24T14:15:22Z",
- "location": {
      "lat": -90,
      "lon": -180
  },
  "name": "string",
- "ownedMeetups": [
      "string"
  ],
- "pendingApproval": [
      "string"
  ],
  "profilePic": "string"
```

Response samples

Content type
application/json

Copy Expand all Collapse all

```
- "attending": [
     "string"
- "connections": [
    "string"
  ],
  "contactInfo": "string",
  "email": "string",
  "id": "string",
- "interests": [
    "string"
  ],
  "joinDate": "2019-08-24T14:15:22Z",
- "location": {
     "lat": -90,
     "lon": -180
  },
  "name": "string",
- "ownedMeetups": [
    "string"
  ],
- "pendingApproval": [
    "string"
  ],
  "profilePic": "string"
```