

## **Model Availability**

**Endpoint:** availability

**Method:** get, post

**Payloads:** 'availability\_level', 'date', 'start\_time', 'end\_time'

### **Description:**

This endpoint is using to access all the available times identified by the user currently logged in. (i.e preferred to meet from 09:00-10:00 on 2024-03-01).

- **GET** method retrieve all the available time blocks created by the user currently logged-in. If the request is unauthenticated, a 401 UNAUTHORIZED response will be returned.
- **POST** method is used to update the list of availabilities when user add a new available time block. If succeed, returned 201 CREATED response. User must be logged-in, if the request is unauthenticated, a 401 UNAUTHORIZED response will be returned.

**Endpoint:** availability/<int:id>

**Method:** get, put, delete

**Payloads:** 'availability\_level', 'date', 'start\_time', 'end\_time'

### **Description:**

This endpoint is using for access a specific time block identified as 'preferred' or 'lesspreferred' by its id.

- **GET** method retrieve the time block by its id and list the detailed information of this time block. If the request is unauthenticated, a 401 UNAUTHORIZED response will be returned.
- **PUT** method retrieve the time block by its id and update the information accordingly. If no change made. If update succeed, a 200 OK response will be returned. User must be logged-in, if the request is unauthenticated, a 401 UNAUTHORIZED response will be returned. User who updated must be the user who created the availability-block, otherwise returned 403 FORBIDDEN
- **DELETE** method delete the time block by id and remove it from the list of availabilities. User must be logged-in, if the request is unauthenticated, a 401 UNAUTHORIZED response will be returned. User who delete the block must be the user who created it, if no returned 403 FORBIDDEN

**Endpoint:** availability/<int:meeting\_id>/contact\_availability

**Method:** post, get

**Payloads:** 'availability\_level', 'date', 'start\_time', 'end\_time',

### **Description:**

- **POST** method is used to update the list of availabilities when contacts add a new available time block. If succeed, returned 201 CREATED response. User must be logged-in, if the request is unauthenticated, a 401 UNAUTHORIZED response will be returned.

- **GET** method retrieve all the available time blocks created by the contact currently logged-in. If the request is unauthenticated, a 401 UNAUTHORIZED response will be returned.

### **Model Contact**

**Endpoint:** contacts/

**Methods:** get, post

**Payloads:** 'id', 'first\_name', 'last\_name', 'email', 'user'

**Description:**

- **Get** method retrieve all the contact that belongs to this user and return their id, first name, last name, email. Post method adds a contact to the database with their id, first name, last name, email.

**Endpoint:** contacts/<int:contact\_id>/

**Methods:** get, delete, put

**Payloads:** 'id', 'first\_name', 'last\_name', 'email', 'user'

**Description:**

- **Get** retrieve the contact information for the contact using contact id.
- **Put** modify the contact's first name, last name and email using contact id.
- **Delete** will delete this contact from the database using the contact id.

**Endpoint:** contacts/<int:contact\_id>/meetings/<str:meeting\_status>/

**Methods:** get

**Payloads:** ['id', 'scheduled\_time', 'duration', 'deadline', 'title', 'location', 'meeting\_status', 'content', 'suggested\_time1', 'suggested\_time2', 'suggested\_time3']

**Description:**

- **Get** retrieve the meeting information that their meeting status belongs to either Unconfirmed, Unscheduling and Confirmed for this contact.

**Endpoint:** contacts/<int:contact\_id>/meetings/detail/<int:meeting\_id>/

**Methods:** get, delete, patch

**Payloads:** ['id', 'scheduled\_time', 'duration', 'deadline', 'title', 'location', 'meeting\_status', 'content', 'suggested\_time1', 'suggested\_time2', 'suggested\_time3']

**Description:**

- **Get** retrieve certain information of meeting using meeting id.
- **Delete** will delete the meeting information using the meeting id.
- **Patch** modify the meeting status from Scheduling to Confirmed and add the selected time retrieved from the front end to the scheduled\_time field.

### **Model: Accounts**

**End point:** accounts/

**Methods:** post, get, patch

**Payloads:** 'username', 'password', 'password2', 'email', 'first\_name', 'last\_name'

**Description:**

- **POST:** Registering a new user - Users can sign up by providing their username, password (with confirmation), email, and name. The password is validated using Django's password validation system, and a check is performed to ensure the password and confirmation match.
- **GET:** Retrieving the current user's profile - Authenticated users can fetch their profile information, including username, email, first name and last name.
- **PATCH:** Updating the current user's profile - Allows authenticated users to update their profile information: email, first name, last name, and password. If the password is updated, a confirmation password (password2) is also required. The username is read-only and cannot be changed.

**End point:** accounts/password/

**Methods:** put

**Payloads:** 'username' OR 'uid', 'token', 'new\_password', 'confirmed\_password'

**Description:**

- **PUT:** When 'username' is in payloads, an email will be sent to the user with the desired username. The user's ID and a unique token will also be included in a link in the body of the email. When 'uid' and 'token' are in payloads, the user's password will be updated

**Model: Meeting**

**End point:** meeting/

**Methods:** post, get

**Payloads:** duration, deadline, title, 'email', location, content, 'suggested\_time1', 'suggested\_time2', 'suggested\_time3', 'scheduled\_time', 'user', 'contact', 'preferred\_user', 'preferred\_contact', 'meeting\_status'

**Description:**

- **POST:** Creating a New Meeting - Users can create a new meeting by providing necessary details such as the meeting's duration, deadline, title, location, content, the contact involved, and the availability preference level. The meeting status is defaults to 'Unconfirmed'. The authenticated user creating the meeting is automatically associated as the meeting's user.
- **GET:** When accessed with a GET request by an authenticated user, this endpoint lists all the meetings where the authenticated user is involved and the meeting\_status is set to 'Confirmed'.

**End point:** meeting/<int:meeting\_id>/

**Methods:** get, patch

**Payloads:** 'scheduled\_time' 'meeting\_status'

**Description:**

- **PATCH:** Updating Meeting Details - This method allows for the dynamic updating of a meeting's scheduled time or status. It is particularly useful for transitioning a meeting from 'Scheduling' to 'Confirmed' once a suitable time

has been found, or to update the meeting's scheduled time as plans change. The endpoint ensures that only relevant and allowed fields can be updated, maintaining data integrity and consistency.

- **GET:** Retrieving Availability Overlaps - This method is designed to facilitate the scheduling process by identifying overlapping available times between the meeting's user and contact. It takes into consideration the preferences set by both parties to prioritize certain times.

**End point:** meeting/<int:meeting\_id>/user\_availability/

**Methods:** get

**Payloads:** None

**Description:**

- **GET:** Retrieving User Availability - When accessed, this method retrieves the availability blocks for the user who is part of the meeting specified by meeting\_id. It differentiates between preferred and lesspreferred availability times, based on the user's settings.

