Endpoints

Signup

Endpoint: 'accounts/api/signup/'

Methods: POST

Fields/payload: username, email, password, password2, profile: {phone_num, address, postal_code, province}

Description: Sign the user up with the extended user model.

Login

Endpoint: 'accounts/api/login/'

Methods: POST

Fields/payload: username, password

Description: Log the user in with the username and password used to register. A Token will be generated which will allow the user to be authenticated.

Profile View

Endpoint: 'accounts/api/<profile_id>/profile/view/'

Methods: GET

Authorization: Bearer Token

Description: View profile of the user with id <profile_id>. Users can only access their own profiles.

Profile Edit

Endpoint: 'accounts/api/<profile_id>/profile/edit/'

Methods: PATCH

Fields/payload: username, email, password, password2, profile: {phone_num, address, postal_code, province}

Authorization: Bearer Token

Description: Edit profile of the user with id <profile_id>. Users can only edit their own profiles.

Note: All fields are required.

Edit Payment

Endpoint: 'accounts/api/<profile_id>/profile/edit-payment/'

Methods: GET, POST

Fields/payload: user, credit_card_num, card_code, card_expr

Authorization: Bearer Token

Description: Edit payment card, code, and card expiry of the user with id <profile_id>. Users can only edit their own payment information.

Change Subscription

Endpoint: 'accounts/api/<profile_id>/subscription/change/<subscription_id>'

Methods: POST

Fields/payload: user, to_subscription_plan, status

Authorization: Bearer Token

Description: Change subscription of the user with id <profile_id> to subscription with id <subscription_id>. Users can start a new subscription with status “S” and cancel subscription with status “C”. Users can only edit their own payment information.

Note: Users can only delete or subscribe to a new subscription plan once a day.

Class Enrollment

Endpoint: 'accounts/api/<profile_id>/enroll/'

Methods: POST, DELETE

Fields/payload: user, cls_instance, recurring

Authorization: Bearer Token

Description: Enroll user with id <profile_id> in class instance. If recurring is set to true, then user will be enrolled in all available classes occurring after cls_instance.

Note: A class is available if the capacity hasn’t been reached and it has not been cancelled.

Payment History

Endpoint: 'accounts/<profile_id>/payment-history/'

Methods: GET

Authorization: Bearer Token

Description: Retrieve payment history of user with id <profile_user>

Future Payments

Endpoint: 'accounts/<profile_id>/future-payment/'

Methods: GET

Authorization: Bearer Token

Description: Retrieve future payments of user with id <profile_user>.

User Schedule

Endpoint: 'accounts/<profile_id>/classes/schedule/'

Methods: GET

Authorization: Bearer Token

Description: Retrieve class schedule of user with id <profile_user>. Requires schedules to be created for schedules to show.

Studio Information

Endpoint: ‘studios/<studio_id>/info/’

Authorization: Bearer Token

Methods: GET

Description: Retrieve information of a studio, such as name, address, postal code, and phone number with id <studio_id>. If the studio does not exist, the response will return 404. Users need to be logged in.

Closest Studio

Endpoint: ‘studios/api/closest_studio/?longitude=<longitude>&latitude=<latitude>&postal_code=<postal_code>’ ’

Methods: GET

Params: longitude, latitude, postal_code

Authorization: Bearer Token

Description: Returns a list of studios from closest to further away from the location provided by the longitude and latitude. If these parameters are not provided, the location is set to a University of Toronto IP address because theim requesting user's location is set to localhost if a request is made from the dev server. Users can only see this if logged in.

Studio Search

Endpoint:'
/studios/search?studio_name=<studio_name>&amenity=<amenity>&class_name=<class_name>&coach=<coach>'

Methods: GET

Fields/payload: page

Params: studio_name, amenity, class_name, coach

Authorization: Bearer Token

Description: returns a list of all studios matching the required fields. At least 0 query parameters should be given. At most all query parameters should be given. All fields in payload are required.

Class Search

Endpoint:'/studios/<studio_id>/search?class_name=<class_name>&coach=<coach>&date=<date>&start_time=<start_time>&end_time=<end_time>'

Methods: GET

Fields/payload: page

Params: studio_id, class_name, coach, date, start_time, end_time

Authorization: Bearer Token

Description: Returns a list of class instances matching the required fields. All fields in payload are required. Class instances are sorted by start time.

Class Schedule

Endpoint: 'studios/<studio_id>/classes/schedule/'

Methods: GET

Fields/payload: page

Authorization: Bearer Token

Description: Returns the class schedule of the studio with id <studio_id>. Users need to be logged in. The studio needs classes with class instances occurring. A class provides a “host” for class instances. Class instances contain information including a start time, end time, a foreign key to a Class, and number of people enrolled.

Studio Amenities

Endpoint: ‘studios/<studio_id>/amenities/’

Methods: GET

Fields/payload: None

Authorization: Bearer Token

Description: Returns a paginated view of the amenities for a studio with id studio_id. Amenities need to exist for this functionality to show. Returns 404 if the studio doesn’t exist.

Studio Images

Endpoint: ‘studios/<studio_id>/images/’

Methods: GET

Fields/payload: None

Authorization: Bearer Token

Description: Returns a paginated view of the images for a studio with id studio_id. Images are encoded to base64 for serialization. Includes the main image and any extra images in with foreign keys to the given studio. Returns 404 if the studio doesn’t exist.

Models

Accounts (App): (/accounts/)

- UserProfile(model)
  - Attributes:
    - First Name
    - Last name
    - Email
    - Phone number
    - avatar

- Website admin: have access to create a studio
- Address
- Postal code
- Province

SubscriptionPlan(model)

- Price
- Interval
- Deleted (boolean)

Enrollment (Model):

Attributes:

- Foreign key to user
- Foreign key to class instance

SubscriptionPlanChange(Model):

Attributes:

- Foreign key to user
- Date_of_change
- Foreign key to SubscriptionPlan
- Status (‘S’ or ‘C’)

CreditCardChange(Model):

Attributes:

- Foreign key to User
- Credit_card_num
- Date of Change
- Card Code
- Card Expr

Studios (App): (/studios/)

Image(Model):

Attributes:

- Image
- Studio Foreign Key

Base64Image(Model):

Attributes:

- Image_b64, base64 encoding of the image
- Foreign key to studio

Amenity(Model):

Attributes:

- Type
- Numbers
- Access Level (optional)
- Foreign key to studio

Studio(Model):

Attributes:

- Name

- Address
- Geographical Location
- Postal Code
- Phone Number
- Main image
- Foreign key to ImageSet

Class(Model):

Attributes:

- Name
- Description
- Coach
- List of keywords (ClassKeywords model)
- Capacity
- Times (ClassTimes model)

ClassKeywords(Model)

Attributes:

- keyword
- Foreign key to Class

ClassInstance(Model)

Attributes:

- Start_datetime
- End_datetime
- Canceled
- Foreign key to class
- Num enrolled