

Additional Notes

1. General
 - a. We have a crontab used to run daily services to check if any payments are due. This is set up in cron.py.
 - b. All data is stored in our database, except the full credit card information, because that is sensitive data. We rely on Stripe to store that information and we hold a reference to the Stripe payment method.
 - c. How to use postman collection
 - i. Import the collection
 - ii. Create an environment with two variables: **url** (localhost:8000 or any base url that you are using) and **token** (set the value of this token after you register. A token will be returned in the response. Note that if you logout, you would have to login and change the token set here).
2. Admin related
 - a. An admin (superuser) is created in the run.sh script. Use "admin@mail.com" as email and "password" as password to log in to the administrative panel.
3. Card information related
 - a. We have integrated Stripe to handle payments and for this submission, we are using the test API keys from Stripe. Hence, in order for credit card number to be successfully registered, a number that is accepted by the Stripe testing environment must be used. This link (<https://stripe.com/docs/testing#cards>) provides a list of accepted card numbers.
 - b. We only allow users to have one card registered with our application at any time, so your previous card information will be replaced upon update.
4. Subscription endpoints related
 - a. We offer two types of subscriptions: monthly and yearly. For any subscription models returns, if "plan" is 0, it is a monthly subscription. If "plan" is 1, it is a yearly subscription.
5. Class endpoints related
 - a. When filtering classes, inputting a start_time means finding classes that start after start_time, not at that particular time. Similarly, if end_time is given, classes that end before that time will be filtered.
 - b. In responses of the class endpoints that return UserClassInstance models, the three possible values of the status field has these meanings:
 - i. 0: Attended (User has attended this class on that particular date)
 - ii. 1: Dropped (User has dropped this class)
 - iii. 2: Cancelled (User has cancelled this instance of the class)
 - c. List of possible error codes for class endpoints (for developers' usage):
 - /api/studios/classes/<class_id>/enroll/
 - error_code: 0
 - Class has been canceled; Unable to enroll
 - error_code: 1
 - User is already enrolled in class; Unable to enroll
 - error_code: 2
 - Class is already full, maximum capacity has been reached; Unable to enroll
 - error_code: 3
 - User does not have a subscription; Unable to enroll

- `/api/studios/classes/<class_id>/drop/`
 - `error_code: 0`
 - Class has been canceled; Unable to drop
 - `error_code: 1`
 - User has not enrolled in this class; Unable to drop
 - `error_code: 2`
 - User has already dropped this class or class instance; Unable to drop
 - `error_code: 3`
 - User does not have a subscription; Unable to drop

6. Clarification on Class Models in the Admin panel:

- a. **StudioClass** model is where we define the class and its recurring date and time. To change the recurring time for a class, admin can modify the corresponding fields in this model. To cancel the class (all future occurrences), admin will need to set it in this model as well.
- b. **CancelledClassInstance** model is where the admin can cancel a single instance of a class. Admin will have to select the class to cancel, and specify the date of the instance of the class to cancel.
- c. **TimeChangedClassInstance** model is where the admin can change the start time and end time that would be different to the recurring time of the class. Admin will have to select the class to change the time for, and specify the date of the instance of the class to change.



http://localhost:8002/

Explore

Snippets API ^{v1}

[Base URL: localhost:8002/api]

<http://localhost:8002/swagger/?format=openapi>

Test description

[Terms of service](#)

[Contact the developer](#)

BSD License

Schemes

HTTP

Django Login

Authorize



Filter by tag

accounts



GET

/accounts/cardinfo/

accounts_cardinfo_list



Retrieves the credit/debit card information of the user. Note that our application only allows every user to have one card at any time. Hence, the returned list has a length of at most one.

Parameters

Try it out

Name

Description

token * required

string

(header)

token

Responses

Response content type

application/json

Code

Description

200

Card information retrieved.

Example Value

```
[
  {
    "last4": "4242",
    "exp_month": 10,
    "exp_year": 2040
  }
]
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

PUT

/accounts/cardinfo/

accounts_cardinfo_update ^



Removes the previous the credit/debit card if it exists and adds a new one to the user. Note: We have integrated Stripe to handle payments and for this submission, we are using the test API keys from Stripe. Hence, in order for credit card number to be successfully registered, a number that is accepted by the Stripe testing environment must be used. See this link (<https://stripe.com/docs/testing#cards>) for a list of accepted card number.

Parameters

Try it out

Name

Description

Name	Description
------	-------------

data * required

object
(body)

Example Value Model

```
{
  "number": "string",
  "exp_month": 0,
  "exp_year": 0,
  "cvc": "string"
}
```

Parameter content type

application/json

token * required

string
(header)

token

Responses

Response content type

application/json

Code	Description
------	-------------

204

Card is updated successfully.

400

Invalid card details.

Example Value

```
{
  "message": "Invalid credentials"
}
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

POST

/accounts/login/

accounts_login_create ^



Authenticates user and returns a token if successful.

Parameters

[Try it out](#)

Name

Description

data * required

object
(body)

Example Value Model

```
{
  "email": "string",
  "password": "string"
}
```

Parameter content type

application/json

Responses

Response content type

application/json

Code

Description

200

Successfully logged in.

Example Value

```
{
  "token": "e7c48016072c8abf89b52996409bbb568b6e110a"
}
```

401

Invalid credentials.

Example Value

```
{
  "message": "Invalid credentials"
}
```

POST

/accounts/logout/

accounts_logout_create ^



Logs user out and deletes token auth.

Parameters

Try it out

Name	Description
------	-------------

token * required string (header)	<input type="text" value="token"/>
---	------------------------------------

Responses

Response content type

application/json

Code	Description
------	-------------

200	Successfully logged out.
-----	--------------------------

401	Token not provided, or token is invalid.
-----	--

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

GET

/accounts/paymenthistory/

accounts_paymenthistory_list



Get payments (both past and future), starting from the requested <start_datetime> param, with a page limit.

To get more payments information, send another request with a later <start_datetime>.

Parameters

Try it out

Name	Description
------	-------------

Name	Description
------	-------------

token * required	
-------------------------	--

string (header)	
--------------------	--

	token
--	-------

start_datetime	
----------------	--

string (query)	
-------------------	--

	start_datetime
--	----------------

Responses	Response content type
-----------	-----------------------

--	--

	application/json
--	------------------

Code	Description
------	-------------

--	--

200	
-----	--

	Payments information returned.
--	--------------------------------

	Example Value
--	---------------

	<pre>{ "data": { "history": [{ "datetime": "2022-11-16T11:37:03.615750Z", "amount": "10.00", "last4": "5556", "exp_month": 10, "exp_year": 2023 }], "future": [{ "amount": 10, "last4": "4242", "exp_month": 10, "exp_year": 2040, "datetime": "2022-12-16T00:00:00" }] } }</pre>
--	---

400	
-----	--

	Invalid datetime given.
--	-------------------------

	Example Value
--	---------------

	<pre>{ "start_datetime": "time data '2022-11-200:00:00' does not match format '%Y-%m-%dT%H:%M:%S'" }</pre>
--	--

Code**Description**

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

GET**/accounts/profile/****accounts_profile_list** ^**Parameters****Try it out****Name****Description****token** * required

string

(header)

token

Responses**Response content type****application/json****Code****Description**

200

Successfully fetched profile.

Example Value

```
{
  "first_name": "Johnny",
  "last_name": "Smith",
  "email": "john.smith@gmail.com",
  "avatar": "http://127.0.0.1:8000/media/user_avatars/forest_wHQ6Bmp.jpeg",
  "phone_number": "123456789"
}
```

Code**Description**

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

PATCH**/accounts/profile/****accounts_profile_partial_update** ^**Parameters****Try it out****Name****Description****first_name**

string

*(formData)**minLength: 1*

first_name

last_name

string

*(formData)**minLength: 1*

last_name

email

string(\$email)

*(formData)**minLength: 1*

email

avatar

file

(formData)

Choose File No...n

phone_number

string

*(formData)**maxLength: 15**x-nullable: true*

phone_number

token * required

string

(header)

token

Responses

Response content type

application/json

Code

Description

200

Successfully updated user profile.

Example Value

```
{
  "first_name": "Johnny",
  "last_name": "Smith",
  "email": "john.smith@gmail.com",
  "avatar": "http://127.0.0.1:8000/media/user_avatars/forest_wHQ6Bmp.jpeg",
  "phone_number": "123456789"
}
```

400

Bad request: Email has been taken or is invalid, password is too short, or first name/last name/password is empty.

Example Value

```
{
  "email": "User with this email already exists.",
  "password": "Password must be at least 8 characters.",
  "first_name": "First names can not be empty",
  "last_name": "Last names can not be empty"
}
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

POST

/accounts/register/

accounts_register_create ^



Registers a new user.

Parameters

Try it out

Name

Description

NameDescription

data * required

object
(body)

Example ValueModel

```
{
  "first_name": "string",
  "last_name": "string",
  "email": "user@example.com",
  "phone_number": "string",
  "password": "string"
}
```

Parameter content type

application/json

Responses

Response content type

application/json

CodeDescription

201

User created.

Example Value

```
{
  "first_name": "John",
  "last_name": "Smith",
  "email": "john.smith@gmail.com",
  "phone_number": "123456789"
}
```

studios

400

GET

/studios/classes/history/

studios_classes_history_list



Example Value

Retrieves clas

```
{
  "email": "User with this email already exists.",
  "password": "Password must be at least 8 characters."
}
```

Parameters

Try it out

NameDescription

Name	Description
page integer (query)	A page number within the paginated result set. <div>page</div>
Responses	Response content type <div>application/json</div>

Code	Description
200	<div><div>Example Value</div><div>Model</div><pre>{ "count": 0, "next": "string", "previous": "string", "results": [{ "id": 0, "class_name": "string", "start_time": "string", "end_time": "string", "date": "2022-11-17", "status": 0, "studio": 0, "studio_class": 0 }] }</pre></div>
401	<div>Token not provided, or token is invalid.</div> <div><div>Example Value</div><pre>{ "detail": "Invalid token header. No credentials provided." }</pre></div>

GET	/studios/classes/schedule/	studios_classes_schedule_list ^	
Retrieves class schedule for a user.			
Parameters	<div>Try it out</div>		

Name	Description
------	-------------

page integer (query)	A page number within the paginated result set.
----------------------------	--

page

token * required string (header)
--

token

Responses

Response content type

application/json

Code	Description
------	-------------

200

Example Value Model

```
{
  "count": 0,
  "next": "string",
  "previous": "string",
  "results": [
    {
      "id": 0,
      "class_name": "string",
      "start_time": "string",
      "end_time": "string",
      "date": "2022-11-17",
      "status": 0,
      "studio": 0,
      "studio_class": 0
    }
  ]
}
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

POST

/studios/classes/{class_id}/drop/

studios_classes_drop_create ^



Allows user to drop an instance of a class.

Parameters

Try it out

Name

Description

data * required

object
(body)

Example Value Model

```
{  
  "date": "string"  
}
```

Parameter content type

application/json

token * required

string
(header)

token

class_id * required

string
(path)

class_id

Responses

Response content type

application/json

Code

Description

200

Class instance dropped successfully.

400

Bad request: Class has already been cancelled. or user is not enrolled in this class, or class has already been dropped. See [Additional Notes](#) section at the end of the documentation for the list of error_code.

Example Value

```
{  
  "error_code": 0,  
  "error_msg": "Class has been cancelled"  
}
```

PATCH

/studios/classes/{class_id}/drop/

studios_classes_drop_partial_update ^



Allows user to drop a class.

Parameters

[Try it out](#)

Name

Description

token * requiredstring
(header)

token

class_id * requiredstring
(path)

class_id

Responses

Response content type

[application/json](#)

Code

Description

200

Class dropped successfully.

400

Bad request: Class has already been cancelled. or user is not enrolled in this class, or class has already been dropped. See [Additional Notes](#) section at the end of the documentation for the list of error_code.

Example Value

```
{
  "error_code": 0,
  "error_msg": "Class has been cancelled"
}
```

POST

/studios/classes/{class_id}/enrol/

studios_classes_enrol_create ^



Allows user to enroll in a class.

Parameters

[Try it out](#)

Name	Description
------	-------------

token * required

string
(header)

token

class_id * required

string
(path)

class_id

Responses

Response content type

application/json

Code	Description
------	-------------

201

User enrolled successfully.

400

Bad request: Class has already been cancelled, or user is already enrolled in this class. or class has reached capacity, or user does not have a valid subscription. See [Additional Notes](#) section at the end of the documentation for the list of error_code.

Example Value

```
{
  "error_code": 2,
  "error_msg": "Class is full"
}
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

GET

/studios/list/{location}/ GET /api/studios/list/<location>

studios_list_list



Studios must be listed based on geographic proximity from location parameter.
Location is a latitude,longitude pair (e.g. 43.656997372,-79.390331772)

Studios can be filtered by name, amenities, class names, and coaches.

Parameters

[Try it out](#)

Name	Description
------	-------------

page
integer
(query)

A page number within the paginated result set.

coach
string
(query)

Coach Name

studio_name
string
(query)

amenity
string
(query)

Name of the amenity

class_name
string
(query)

location * required
string
(path)

Responses

Response content type

application/json

Code	Description
------	-------------

Code

Description

200

Example Value Model

```
{
  "count": 0,
  "next": "string",
  "previous": "string",
  "results": [
    {
      "id": 0,
      "name": "string",
      "address": "string",
      "lat": 90,
      "lon": 180,
      "postal_code": "string",
      "phone_number": "string",
      "images": [
        {
          "id": 0,
          "studio_image": "string"
        }
      ],
      "amenities": [
        {
          "id": 0,
          "amenity_type": "string",
          "quantity": 0
        }
      ],
      "directions_url": "string"
    }
  ]
}
```

404

Studio with the given id does not exist.

Example Value

```
{
  "detail": "Not found."
}
```

GET

/studios/{studio_id}/

studios_read



Parameters

Try it out

Name

Description

studio_id * required

string

(path)

studio_id

Responses

Response content type

application/json

Code

Description

200

Example Value Model

```
{
  "id": 0,
  "name": "string",
  "address": "string",
  "lat": 90,
  "lon": 180,
  "postal_code": "string",
  "phone_number": "string",
  "images": [
    {
      "id": 0,
      "studio_image": "string"
    }
  ],
  "amenities": [
    {
      "id": 0,
      "amenity_type": "string",
      "quantity": 0
    }
  ],
  "directions_url": "string"
}
```

404

Studio with the given id does not exist.

Example Value

```
{
  "detail": "Not found."
}
```

GET

/studios/{studio_id}/classes/

GET

/api/studios/<studio_id>/classes

studios_classes_1
ist



Returns a list of classes for studio id, ordered by their start time from now.

Classes can be filtered by class name, coach name, date, and time range.

Parameters

Try it out

Name

Description

Name	Description
page integer (query)	A page number within the paginated result set. <input type="text" value="page"/>
studio_id integer (query)	<input type="text" value="studio_id"/>
class_name string (query)	<input type="text" value="class_name"/>
coach string (query)	Coach Name <input type="text" value="coach"/>
class_day string (query)	The day of the class - 0 for Monday, 1 for Tuesday, etc. <input type="text" value="class_day"/>
start_time string (query)	The start time of the time range to filter - must be in HH:MM format <input type="text" value="start_time"/>
end_time string (query)	The end of the time range to filter - must be in HH:MM format <input type="text" value="end_time"/>
weeks integer (query)	The number of weeks from today to get classes for (Default 1) <input type="text" value="weeks"/>

Responses

Response content type

application/json

Code**Description**

200

Example Value Model

```
{
  "count": 0,
  "next": "string",
  "previous": "string",
  "results": [
    {
      "date": "2022-11-17",
      "start_time": "string",
      "end_time": "string",
      "studio_class": {
        "id": 0,
        "studio": 0,
        "description": "string",
        "class_name": "string",
        "coach": "string",
        "keywords": [
          {
            "id": 0,
            "keyword": "string"
          }
        ],
        "capacity": 0,
        "day": 0
      }
    }
  ]
}
```

404

Studio with the given id does not exist.

Example Value

```
{
  "detail": "Not found."
}
```

subscriptions

**GET****/subscriptions/****subscriptions_list**

Get all subscriptions.

Parameters**Try it out**

Name

Description

page
integer
(query)

A page number within the paginated result set.

page

Responses

Response content type

application/json

Code

Description

200

Example Value Model

```
{
  "count": 0,
  "next": "string",
  "previous": "string",
  "results": [
    {
      "id": 0,
      "name": "string",
      "price": "string",
      "plan": 0
    }
  ]
}
```

POST

/subscriptions/cancel/

subscriptions_cancel_create



Cancel the subscription of user. If user does not have one, the request is ignored.

Parameters

Try it out

Name

Description

token * required
string
(header)

token

Responses

Response content type

application/json

Code

Description

200

Successfully cancelled.

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

PUT

/subscriptions/{subscription_id}/subscribe

subscriptions_subscribe_update



Subscribe the user to the subscription given by subscription_id.

Parameters

Try it out

Name

Description

token * required

string
(header)

token

subscription_id * required

string
(path)

subscription_id

Responses

Response content type

application/json

Code

Description

200

Successfully subscribed.

Code**Description**

400

Bad request: Subscription does not exist or user does not have a card set up.

Example Value

```
{
  "subscription_id": "subscription with id 1 does not exist",
  "user": "User does not have a payment method set up"
}
```

401

Token not provided, or token is invalid.

Example Value

```
{
  "detail": "Invalid token header. No credentials provided."
}
```

Models**UserRegister****UserClassInstance****StudioImage****Amenity****Studio**

Keyword

StudioFilter

ClassInstance

Subscription