# Restify documentation - Introduction

CSC309 Phase 2

Winston Qiaohua Zheng

Bradley David Hebert

# Project Structure

```
.
└── P2
    ├── P2
    │   ├── __init__.py
    │   ├── __pycache__
    │   │   ├── __init__.cpython-310.pyc
    │   │   ├── __init__.cpython-38.pyc
    │   │   ├── settings.cpython-310.pyc
    │   │   ├── settings.cpython-38.pyc
    │   │   ├── urls.cpython-310.pyc
    │   │   ├── urls.cpython-38.pyc
    │   │   ├── wsgi.cpython-310.pyc
    │   │   └── wsgi.cpython-38.pyc
    │   ├── asgi.py
    │   ├── settings.py
    │   ├── urls.py
    │   └── wsgi.py
    ├── accounts
    │   ├── __init__.py
    │   ├── __pycache__
    │   │   ├── __init__.cpython-310.pyc
    │   │   ├── __init__.cpython-38.pyc
    │   │   ├── admin.cpython-310.pyc
    │   │   ├── admin.cpython-38.pyc
    │   │   ├── apps.cpython-310.pyc
    │   │   ├── apps.cpython-38.pyc
    │   │   ├── models.cpython-310.pyc
    │   │   ├── models.cpython-38.pyc
    │   │   ├── serializers.cpython-310.pyc
    │   │   ├── serializers.cpython-38.pyc
    │   │   ├── urls.cpython-310.pyc
    │   │   ├── urls.cpython-38.pyc
    │   │   ├── views.cpython-310.pyc
    │   │   └── views.cpython-38.pyc
    │   ├── admin.py
```

```
|   ├── apps.py
|   ├── avatar
|   |   └── me_noback.png
|   ├── migrations
|   |   ├── 0001_initial.py
|   |   ├── 0002_alter_user_phone_number.py
|   |   ├── __init__.py
|   |   └── __pycache__
|   |       ├── 0001_initial.cpython-310.pyc
|   |       ├── 0001_initial.cpython-38.pyc
|   |       ├── 0002_alter_user_phone_number.cpython-38.pyc
|   |       ├── __init__.cpython-310.pyc
|   |       └── __init__.cpython-38.pyc
|   ├── models.py
|   ├── serializers.py
|   ├── tests.py
|   ├── urls.py
|   └── views.py
├── blogs
|   ├── __init__.py
|   ├── __pycache__
|   |   ├── __init__.cpython-38.pyc
|   |   ├── admin.cpython-38.pyc
|   |   ├── apps.cpython-38.pyc
|   |   ├── models.cpython-38.pyc
|   |   └── urls.cpython-38.pyc
|   ├── admin.py
|   ├── apps.py
|   ├── migrations
|   |   ├── 0001_initial.py
|   |   ├── 0002_alter_blogpost_post_restaurant.py
|   |   ├── 0003_remove_blogpost_post_modify_time.py
|   |   ├── 0004_restaurantlike_blogpostlike.py
|   |   ├── 0005_restaurantfollow.py
|   |   ├── 0006_alter_restaurantfollow_restaurant_and_more.py
|   |   ├── 0007_remove_blogpost_post_shares.py
```

```
|   |   ├── 0008_notification.py
|   |   ├── __init__.py
|   |   └── __pycache__
|   |       ├── 0001_initial.cpython-38.pyc
|   |       ├── 0002_alter_blogpost_post_restaurant.cpython-38.pyc
|   |       ├── 0003_remove_blogpost_post_modify_time.cpython-38.pyc
|   |       ├── 0004_restaurantlike_blogpostlike.cpython-38.pyc
|   |       ├── 0005_restaurantfollow.cpython-38.pyc
|   |       ├── 0006_alter_restaurantfollow_restaurant_and_more.cpython-38.pyc
|   |       ├── 0007_remove_blogpost_post_shares.cpython-38.pyc
|   |       ├── 0008_notification.cpython-38.pyc
|   |       └── __init__.cpython-38.pyc
|   ├── models.py
|   ├── serializers
|   |   ├── __init__.py
|   |   ├── __pycache__
|   |   |   ├── __init__.cpython-38.pyc
|   |   |   ├── blogpostserializer.cpython-38.pyc
|   |   |   ├── notificationserializer.cpython-38.pyc
|   |   |   └── restaurantsocialserializer.cpython-38.pyc
|   |   ├── blogpostserializer.py
|   |   ├── notificationserializer.py
|   |   └── restaurantsocialserializer.py
|   ├── tests.py
|   ├── urls.py
|   └── views
|       ├── __init__.py
|       ├── __pycache__
|       |   ├── __init__.cpython-38.pyc
|       |   ├── blogpost.cpython-38.pyc
|       |   ├── feed.cpython-38.pyc
|       |   ├── notification.cpython-38.pyc
|       |   └── restaurant.cpython-38.pyc
|       ├── blogpost.py
|       ├── feed.py
|       ├── notification.py
|       └── restaurant.py
```

```
├── comments
│   ├── __init__.py
│   ├── __pycache__
│   │   ├── __init__.cpython-38.pyc
│   │   ├── admin.cpython-38.pyc
│   │   ├── apps.cpython-38.pyc
│   │   ├── models.cpython-38.pyc
│   │   ├── serializers.cpython-38.pyc
│   │   ├── urls.cpython-38.pyc
│   │   └── views.cpython-38.pyc
│   ├── admin.py
│   ├── apps.py
│   ├── migrations
│   │   ├── 0001_initial.py
│   │   ├── __init__.py
│   │   └── __pycache__
│   │       ├── 0001_initial.cpython-38.pyc
│   │       └── __init__.cpython-38.pyc
│   ├── models.py
│   ├── serializers.py
│   ├── tests.py
│   ├── urls.py
│   └── views.py
├── db.sqlite3
├── manage.py
├── requirements.txt
├── restaurants
│   ├── __init__.py
│   ├── __pycache__
│   │   ├── __init__.cpython-38.pyc
│   │   ├── admin.cpython-38.pyc
│   │   ├── apps.cpython-38.pyc
│   │   ├── models.cpython-38.pyc
│   │   ├── serializers.cpython-38.pyc
│   │   ├── urls.cpython-38.pyc
│   │   └── views.cpython-38.pyc
```

```
|   ├── admin.py
|   ├── apps.py
|   ├── migrations
|   |   ├── 0001_initial.py
|   |   ├── __init__.py
|   |   └── __pycache__
|   |       ├── 0001_initial.cpython-38.pyc
|   |       └── __init__.cpython-38.pyc
|   ├── models.py
|   ├── serializers.py
|   ├── tests.py
|   ├── urls.py
|   └── views.py
├── run.sh
└── startup.sh
```

It is separated into four apps, being accounts, blogs, comments, and restaurants. accounts handles user signup, profile viewing and token authentication (because certain actions such as liking a post or commenting on a restaurant can only be done by users that are logged in), blogs handles everything social-related except comments (so it handles blog posts, blog post likes, restaurant likes, restaurant follows, notifications, the feed, etc), comments handles comments specifically, and restaurants handles all the operations desired for a restaurant.

Various imports from Django's rest framework were made in order to enable the functionality of the website and ensure that the Restify API fits the REStful standard. For example:

- This project made use of various class-based generic Django rest framework API views such as CreateAPIView, RetrieveUpdateDestroyAPIView, ListAPIView, etc instead of using function-based views, as inheriting from these Django rest

framework API views allows for easier use when it comes to basic operations such as creating something or deleting something.

- Usage of serializers (from rest_framework import serializers) for the serialization of information obtained from the models for users, restaurants, menu items, restaurant hours, a restaurant's additional images, blog posts, comments, a restaurant's received likes, and a restaurant's followers. This allows the model information to be used and displayed properly on the Restify website
- Usage of JSON access and refresh web tokens (from rest_framework_simplejwt.views import TokenObtainPairView, TokenRefreshView) for ensuring proper user authorization when browsing Restify.
- Usage of exceptions to ensure user inputs meet specified criteria and are looking for things that exist
  - (From rest_framework.exceptions import ValidationError)

Class-based views were used over function-based views in order to enable inheritance from generic Django REST APIs and  follow the DRY principle by reducing code redundancy in favor of code reusability.

## List of Apps

**Accounts**

Allows users to

- Make an account
- Access an existing account in which they have credentials for
- Update an account
- Access an account given its id
- Get access and refresh tokens for an account

**Restaurants**

Allows users to

- View a restaurant
- Search for an existing restaurant
- Create and Update a restaurant
- Access a restaurant given its id
- Get, create, update and delete pictures from a restaurant's page
- Get, create, update and delete items from a restaurant's menu
- Get, create, update and delete times and hours from a restaurant's page

**Blogs**

Allows users to

- Create a blog post (given that the user logged in owns the restaurant they want to create a blog post for)
- View a blog post (given blog post ID)
- View a list of blog posts from a restaurant (given restaurant ID)
- Like a blog post (given blog post ID)
- Unlike a blog post (given blog post ID)
- Like a restaurant (given restaurant ID)
- Unlike a restaurant (given restaurant ID)
- Follow a restaurant (given restaurant ID)
- Unfollow a restaurant (given restaurant ID)
- View a feed of blog posts for the restaurants that the current user is following

**Comments**

Allows users to

- Create a comment on a restaurant (given restaurant ID)
- View a comment (given comment ID)
- View a list of comments for a restaurant (given restaurant ID)

# Design of Models

## User Model

The 'User' model in 'Accounts' app extends the AbstractUser model in Django. It models any user in the Resitfy network, including general Users and Restaurant Owners. 'User' Has the following fields

- id
  - A numerical ID assigned to the User to identify in the database. Assigned with a BigAutoField
- username
  - A CharField, max length 150, is unique, is required
- first_name
  - A CharField, max length 150, not need be unique, is not required
- last_name
  - A CharField, max length 150, not need be unique, is not required
- email
  - An EmailField, max length 254, is not required
- password
  - A CharField, max length 128
- groups
  - All groups a user belongs to, is a foreign key. Is a Many-to-Many relationship to 'Group'
- user_permissions
  - Is a Many-to-Many relationship to 'Group'
- is_staff
  - A Boolean value, designates whether can have admin permissions
- is_active
  - A Boolean value, designates if this user should be considered 'active'
- is_superuser
  - A Boolean value, Designates that this user has all allowed permissions
- Last_login
  - A DateTime value, designates the user's last login
- Date_joined
  - A DateTime value, designates when the user was created

## Restaurant Model

The 'Restaurant' model in 'Restaurants'  models any Restaurant in the Resitfy network. A restaurant is owned by an owner, which is a User.

It has the following fields

- name
  - A CharField, has max length of 128
- address
  - A CharField, has max length of 128
- owner
  - A One-To-One relation, references a User, On deletes it cascades when the owner is deleted
- description
  - A TextField, permitted to be blank
- primary_image
  - An ImageField,  permitted to be blank, stored in 'restaurants/primary_image/'. Is the main image for the restaurant
- banner
  - An ImageField, permitted to be blank, stored in 'restaurants/banner/'. Is the banner image for the restaurants page (goes along the top behind their name)
- image
  - An ImageField, permitted to be blank, stored in 'restaurants/image/'. Belongs to a collection of images that are displayed in a gallery on the restaurant's page
- theme
  - A CharField, has max length 128, permitted to be blank, is chosen from RestaurantThemes.choices, can include but is not limited to 'Italian', 'FastFood', or 'Buffet' (as specified in RestaurantThemes)
- website
  - A URLField, has max length 253, permitted to be blank.
- email
  - An EmailField, permitted to be blank
- phone
  - A CharField, validated with a regex validator so that it adheres to a format like '123-123-1234', has max length 12, permitted to be blank
- fax

- ○ A CharField, validated with a regex validator so that it adheres to a format like '123-123-1234', has max length 12, permitted to be blank
  - number_likes
    - ○ An IntegerField, default value is 0, permitted to be blank. Stores the number of likes a restaurant has
  - number_followers
    - ○ An IntegerField, default value is 0, permitted to be blank. Stores the number of followers a restaurant has

## Restaurant Image Model

The 'RestaurantImage' model in the 'Restaurants' app models an image for a restaurant that is displayed on the restaurants page.

- Restaurant
  - ○ A Many-To-One relation to a restaurant. On deletes it cascades when restaurant it belongs to is deleted
- Image
  - ○ An ImageField, uploads to 'restaurant/images/'
- Position
  - ○ An IntegerField, default value 0, permitted to be blank. Represents its ranking on the menu, lower values appear higher, and higher values appear lower

## Food Item Model

The 'FoodItem' model in the 'Restaurants' app models a menu or food entry on the menu for a restaurant that is displayed on the restaurants page.

- name
  - ○ A CharField, has max length 128
- restaurant
  - ○ A Many-To-One relation to a restaurant. On deletes it cascades when restaurant it belongs to is deleted
- details
  - ○ A TextField
- price
  - ○ A DecimalField, max digits is 3, and max decimals is 2, hence the maximum price representable is 999.99, permitted to be blank and null
- position

- An IntegerField, default value 0, permitted to be blank. Represents its ranking on the menu, lower values appear higher, and higher values appear lower

## Time Model

The 'Time' model in the 'Restaurants' app models an open and closing time for a day for a restaurant that is displayed on the restaurants page as its hours of service.

- Restaurant
  - A Many-To-One relation to a restaurant. On deletes it cascades when restaurant it belongs to is deleted
- Day
  - A CharField, max length 128, is chosen from Day.choices, includes each day of the week (Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday)
- Open_time
  - A TimeField, permitted to be null and blank, is the time on a particular day that the restaurant would open
- Close_time
  - A TimeField, permitted to be null and blank, is the time on a particular day that the restaurant would close

## Blog Post Model

The 'BlogPost' model in 'Blogs' app models a blog post on the Restify network. It belongs to a restaurant, and a title and description are mandatory.

- post_restaurant
  - A Many-To-One relation to a restaurant. On deletes it cascades when restaurant it belongs to is deleted
- post_title
  - A CharField, has max length 150
- post_description
  - A TextField, is the body of the blog post
- post_media
  - An ImageField, permitted to be blank
- post_likes
  - An IntegerField, defaults to 0, permitted to be blank
- post_shares
  - An IntegerField, defaults to 0, permitted to be blank
- post_create_time

- ○ A DateTimeField model, permitted to be true, sets time to current time

## Blog Post Like Model

The 'BlogPostLike' model in 'Blogs' app models a blog post like on the Restify network

- post
  - ○ A Many-To-One relation to a BlogPost. On deletes it cascades when Blog Post it belongs to is deleted
- user
  - ○ A One-To-One relation to a User. On deletes it cascades when User it belongs ot is deleted

## Restaurant Like Model

The 'RestaurantLikeModel' model in 'Blogs' app models a like for a restaurant on the Restify network

- Restaurant
  - ○ A Many-To-One relation to a Restaurant.On deletes it cascades when Restaurant it belongs to is deleted
- user
  - ○ A One-To-One relation to a User. On deletes it cascades when User it belongs ot is deleted

## Restaurant Follow Model

The 'RestaurantFollowModel' model in 'Blogs' app models a follow for a restaurant on the Restify network

- Restaurant
  - ○ A Many-To-One relation to a Restaurant. On deletes it cascades when Restaurant it belongs to is deleted
- user
  - ○ A One-To-One relation to a User.  On deletes it cascades when User it belongs ot is deleted

## Notification Model

The 'Notification' model in 'Blogs' app models a notification for a User on the Restify network. It has a receiver, a userNotifier (if the notification was due to a user action e.x a user liked your post), and a restaurantNotifier (e.x if a restaurant you follow posted a new menu item)

- Receiver
  - A One-To-One relation to User, On deletes it cascades when the User is deleted
- userNotifier
  - A One-To-One relation to User, On deletes it cascades when the User is deleted, permitted to be null and blank. Only it exclusive or restaurantNotifier should be non-null
- restaurantNotifier
  - A One-To-One relation to Restaurant, On deletes it cascades when the Restaurant is deleted, permitted to be null and blank. Only it exclusive or userNotifier should be non-null
- description
  - A TextField, permitted to be blank
- href
  - A URLField, max length of 128, permitted to be blank

## Comment Model

The 'Comment' model in 'Comments' app models a comment for a Restaurant on the Restify network

- restaurant
  - A Many-To-One relation, On deletes it cascades when the Restaurant is deleted
- user
  - A Many-To-One relation, On deletes it cascades when the User is deleted
- content
  - A TextField, stores the content of the comment
- comment_create_time
  - A DateTimeField, stores current time at when the comment is created, is permitted to be blank

# Restify API [v1]

All endpoints for phasae 2

**Schemes**

| HTTP |

Filter by tag

## accounts ⌄

| **GET** | `/accounts/` | accounts_read 🔓 |

**Parameters**                                      [Try it out]

No parameters

**Responses**          Response content type   | application/json |

| Code | Description |
|------|-------------|
| 200  |             |

Example Value   **Model**

| Code | Description |
|------|-------------|

**User** {
    id                          integer
                                *title: ID*
                                *readOnly: true*

    username              string
                    *title: Username*
                    *pattern: ^[\w.@+-]+$*
                    *readOnly: true*
                    *maxLength: 150*
                    *minLength: 1*

                    Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

    password*            string
                    *title: Password*
                    *maxLength: 128*
                    *minLength: 1*

    first_name           string
                    *title: First name*
                    *maxLength: 150*

    last_name            string
                    *title: Last name*
                    *maxLength: 150*

    email                string($email)
                    *title: Email address*
                    *maxLength: 254*

    phone_number        string
                    *title: Phone number*
                    *pattern: ^\d{3}-\d{3}-\d{4}$*
                    *maxLength: 12*

    avatar               string($uri)
                    *title: Avatar*
                    *readOnly: true*

}

---

**POST**   /accounts/                                accounts_create 🔓

### Parameters

[ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value   **Model**

**User** {
    password*            string
                    *title: Password*
                    *maxLength: 128*
                    *minLength: 1*

    first_name           string
                    *title: First name*
                    *maxLength: 150*

    last_name            string
                    *title: Last name*
                    *maxLength: 150*

    email                string($email)
                    *title: Email address*
                    *maxLength: 254*

    phone_number        string
                    *title: Phone number*
                    *pattern: ^\d{3}-\d{3}-\d{4}$*
                    *maxLength: 12*

}

### Responses

Response content type   [ application/json ]

| Code | Description |
|------|-------------|

| Code | Description |
|------|-------------|

**201**

Example Value    **Model**

**User**    {
    id                integer
                      title: ID
                      readOnly: true
    username      string
                      title: Username
                      pattern: ^[\w.@+-]+$
                      readOnly: true
                      maxLength: 150
                      minLength: 1

                      Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

    password*     string
                      title: Password
                      maxLength: 128
                      minLength: 1
    first_name    string
                      title: First name
                      maxLength: 150
    last_name     string
                      title: Last name
                      maxLength: 150
    email         string($email)
                      title: Email address
                      maxLength: 254
    phone_number   string
                      title: Phone number
                      pattern: ^\d{3}-\d{3}-\d{4}$
                      maxLength: 12
    avatar        string($uri)
                      title: Avatar
                      readOnly: true

}

---

**PUT**    /accounts/                                          accounts_update    🔓

**Parameters**                                                          [ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value    **Model**

**User**    {
    password*     string
                      title: Password
                      maxLength: 128
                      minLength: 1
    first_name    string
                      title: First name
                      maxLength: 150
    last_name     string
                      title: Last name
                      maxLength: 150
    email         string($email)
                      title: Email address
                      maxLength: 254
    phone_number   string
                      title: Phone number
                      pattern: ^\d{3}-\d{3}-\d{4}$
                      maxLength: 12

}

---

**Responses**                          Response content type    [ application/json ]

| Code | Description |
|------|-------------|
| 200 | |

Example Value    **Model**

**User**    {
    id                              integer
                            *title: ID*
                            *readOnly: true*
    username                        string
                            *title: Username*
                            *pattern: ^[\w.@+-]+$*
                            *readOnly: true*
                            *maxLength: 150*
                            *minLength: 1*

                            Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

    password*                       string
                            *title: Password*
                            *maxLength: 128*
                            *minLength: 1*
    first_name                      string
                            *title: First name*
                            *maxLength: 150*
    last_name                       string
                            *title: Last name*
                            *maxLength: 150*
    email                           string($email)
                            *title: Email address*
                            *maxLength: 254*
    phone_number                    string
                            *title: Phone number*
                            *pattern: ^\d{3}-\d{3}-\d{4}$*
                            *maxLength: 12*
    avatar                          string($uri)
                            *title: Avatar*
                            *readOnly: true*

    }

---

**PATCH**    /accounts/                                   accounts_partial_update    🔓

**Parameters**                                                              [ Try it out ]

| Name | Description |
|------|-------------|
| **data** * required<br>object<br>*(body)* | Example Value    **Model** |

**User**    {
    password*                       string
                             *title: Password*
                            *maxLength: 128*
                            *minLength: 1*
    first_name                      string
                            *title: First name*
                            *maxLength: 150*
    last_name                       string
                            *title: Last name*
                            *maxLength: 150*
    email                           string($email)
                            *title: Email address*
                            *maxLength: 254*
    phone_number                    string
                            *title: Phone number*
                            *pattern: ^\d{3}-\d{3}-\d{4}$*
                            *maxLength: 12*

    }

**Responses**                              Response content type    [ application/json ]

| Code | Description |
|------|-------------|

200

Example Value   **Model**

**User**   {
    id                          integer
                                *title: ID*
                                *readOnly: true*
    username                    string
                                *title: Username*
                                *pattern: ^[\w.@+-]+$*
                                *readOnly: true*
                                *maxLength: 150*
                                *minLength: 1*

                                Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

    password*                   string
                                *title: Password*
                                *maxLength: 128*
                                *minLength: 1*
    first_name                  string
                                *title: First name*
                                *maxLength: 150*
    last_name                   string
                                *title: Last name*
                                *maxLength: 150*
    email                       string($email)
                                *title: Email address*
                                *maxLength: 254*
    phone_number                string
                                *title: Phone number*
                                *pattern: ^\d{3}-\d{3}-\d{4}$*
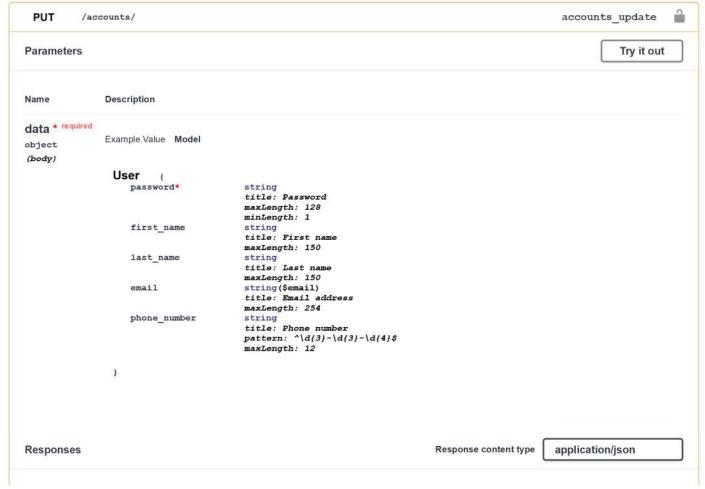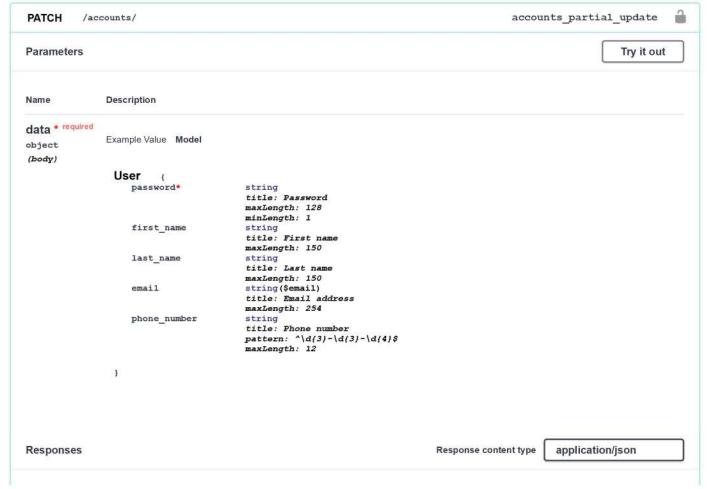                                *maxLength: 12*
    avatar                      string($uri)
                                *title: Avatar*
                                *readOnly: true*

    }

---

**POST**   `/accounts/token/`                                    accounts_token_create  🔓

Takes a set of user credentials and returns an access and refresh JSON web
token pair to prove the authentication of those credentials.

**Parameters**                                                                   [ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value   **Model**

**TokenObtainPair**   {
    username*                   string
                                *title: Username*
                                *minLength: 1*
    password*                   string
                                *title: Password*
                                *minLength: 1*

    }

**Responses**                        Response content type   [ application/json ]

| Code | Description |
|------|-------------|

| Code | Description |
|------|-------------|

**201**

Example Value    **Model**

**TokenObtainPair**    {
  username*              string
                        *title: Username*
                        *minLength: 1*
  password*              string
                        *title: Password*
                        *minLength: 1*

}

---

**POST**    /accounts/token/refresh/                     accounts_token_refresh_create 🔓

Takes a refresh type JSON web token and returns an access type JSON web token if the refresh token is valid.

**Parameters**                                                        [ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value    **Model**

**TokenRefresh**    {
  refresh*              string
                        *title: Refresh*
                        *minLength: 1*

}

**Responses**                      Response content type    [ application/json ]

| Code | Description |
|------|-------------|

**201**

Example Value    **Model**

**TokenRefresh**    {
  refresh*              string
                        *title: Refresh*
                        *minLength: 1*
  access                string
                        *title: Access*
                        *readOnly: true*
                        *minLength: 1*

}

---

**GET**    /accounts/{account_id}/                          accounts_read 🔓

**Parameters**                                                        [ Try it out ]

| Name | Description |
|------|-------------|

**account_id** * required
string
*(path)*

```
account_id
```

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|

200
**blogs**          Example Value    **Model**                                                    ⌄

**User**
**POST**    /blogs/c    .tepost/                                          blogs_createpost_create    🔓

**Parameters**                                                                **Try it out**

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value    **Model**

**BlogPost**        {
    post_restaurant*        integer
                            *title: Post restaurant*
    post_title*             string
                            *title: Post title*
                            *maxLength: 150*
                            *minLength: 1*
    post_description*       string
                            *title: Post description*
                            *minLength: 1*

    }

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|

201
        Example Value    **Model**

Code        Description

**BlogPost**      {
    post_restaurant*        integer
                    *title: Post restaurant*
    post_title*             string
                    *title: Post title*
                    *maxLength: 150*
                    *minLength: 1*
    post_description*       string
                    *title: Post description*
                    *minLength: 1*
    post_media              string($uri)
                    *title: Post media*
                    *readOnly: true*
    post_likes              integer
                    *title: Post likes*
                    *readOnly: true*
    post_create_time        string($date-time)
                    *title: Post create time*
                    *readOnly: true*

    }

---

**GET**     `/blogs/feed/`                                          blogs_feed_list   🔓

**Parameters**                                                              [ Try it out ]

Name        Description

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

**Responses**                               Response content type   [ application/json ]

Code        Description

200

---

**POST**    `/blogs/notifications/`                          blogs_notifications_create   🔓

**Parameters**                                                              [ Try it out ]

Name        Description

**data** * required
object      Example Value    **Model**
(body)

---

| Name | Description |
| --- | --- |

**Notification**        {
    userNotifier       integer
                          *title: UserNotifier*
                          *x-nullable: true*
    restaurantNotifier   integer
                          *title: RestaurantNotifier*
                          *x-nullable: true*
    description        string
                          *title: Description*
    href               string($uri)
                          *title: Href*
                          *maxLength: 128*

       }

## Responses

Response content type      **application/json**

| Code | Description |
| --- | --- |
| 201 | |

Example Value   **Model**

**Notification**        {
    receiver           integer
                          *title: Receiver*
                          *readOnly: true*
    userNotifier       integer
                          *title: UserNotifier*
                          *x-nullable: true*
    restaurantNotifier   integer
                          *title: RestaurantNotifier*
                          *x-nullable: true*
    description        string
                          *title: Description*
    href               string($uri)
                          *title: Href*
                          *maxLength: 128*

       }

---

**GET**     /blogs/notifications/list/{notification_id}/      blogs_notifications_list_read 🔓

## Parameters

Try it out

| Name | Description |
| --- | --- |
| page<br>integer<br>*(query)* | A page number within the paginated result set.<br><br>page - A page number within the paginated re |
| notification_id * required<br>string<br>*(path)* | notification_id |

## Responses

Response content type      **application/json**

| Code | Description |
|------|-------------|

**200**

Example Value     **Model**

```
{
    count*              integer
    next                string($uri)
                        x-nullable: true
    previous            string($uri)
                        x-nullable: true
    results*            [Notification  {
                        receiver            integer
                                            title: Receiver
                                            readOnly: true
                        userNotifier        integer
                                            title: UserNotifier
                                            x-nullable: true
                        restaurantNotifier  integer
                                            title: RestaurantNotifier
                                            x-nullable: true
                        description         string
                                            title: Description
                        href                string($uri)
                                            title: Href
                                            maxLength: 128

                        }]

}
```

---

**POST**     `/blogs/restaurants/follow/{id}`                          blogs_restaurants_follow_create 🔓

### Parameters                                                                    [ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value     **Model**

**RestaurantFollow**     {

}

**id** * required
string
*(path)*

[ id ]

### Responses                               Response content type  [ application/json ]

| Code | Description |
|------|-------------|

**201**

Example Value     **Model**

**RestaurantFollow**     {

}

---

**POST**     `/blogs/restaurants/like/{id}`                            blogs_restaurants_like_create 🔓

## Parameters

<div style="text-align: right">**Try it out**</div>

| Name | Description |
|------|-------------|
| **data** * required <br> object <br> *(body)* | Example Value   **Model** <br><br>     **RestaurantLike**    { <br><br>     } |
| **id** * required <br> string <br> *(path)* | [ id ] |

## Responses

Response content type    application/json

| Code | Description |
|------|-------------|
| 201 | Example Value   **Model** <br><br>     **RestaurantLike**    { <br><br>     } |

---

**GET**    `/blogs/restaurants/unfollow/{id}`      blogs_restaurants_unfollow_read 🔓

### Parameters

<div style="text-align: right">**Try it out**</div>

| Name | Description |
|------|-------------|
| **id** * required <br> string <br> *(path)* | [ id ] |

### Responses

Response content type    application/json

| Code | Description |
|------|-------------|
| 200 | |

---

**PUT**    `/blogs/restaurants/unfollow/{id}`      blogs_restaurants_unfollow_update 🔓

### Parameters

<div style="text-align: right">**Try it out**</div>

| Name | Description |
|------|-------------|

| Name | Description |
|------|-------------|
| id * required<br>string<br>*(path)* | id |

**Responses**                                                  Response content type    | application/json |

| Code | Description |
|------|-------------|
| 200 | |

---

**PATCH**   `/blogs/restaurants/unfollow/{id}`              `blogs_restaurants_unfollow_partial_update` 🔓

**Parameters**                                                                          Try it out

| Name | Description |
|------|-------------|
| id * required<br>string<br>*(path)* | id |

**Responses**                                                  Response content type    | application/json |

| Code | Description |
|------|-------------|
| 200 | |

---

**DELETE**   `/blogs/restaurants/unfollow/{id}`                `blogs_restaurants_unfollow_delete` 🔓

**Parameters**                                                                          Try it out

| Name | Description |
|------|-------------|
| id * required<br>string<br>*(path)* | id |

**Responses**                                                  Response content type    | application/json |

| Code | Description |
|------|-------------|
| 204  |             |

---

**GET**    /blogs/restaurants/unlike/{id}     blogs_restaurants_unlike_read 🔓

**Parameters**

Try it out

| Name | Description |
|------|-------------|
| **id** * required<br>string<br>*(path)* | id |

**Responses**

Response content type   application/json

| Code | Description |
|------|-------------|
| 200  |             |

---

**PUT**    /blogs/restaurants/unlike/{id}     blogs_restaurants_unlike_update 🔓

**Parameters**

Try it out

| Name | Description |
|------|-------------|
| **id** * required<br>string<br>*(path)* | id |

**Responses**

Response content type   application/json

| Code | Description |
|------|-------------|
| 200  |             |

---

**PATCH**    /blogs/restaurants/unlike/{id}     blogs_restaurants_unlike_partial_update 🔓

**Parameters**

Try it out

| Name | Description |
|------|-------------|
| **id** * required<br>string | id |

| Name | Description |
|------|-------------|
| *(path)* | |

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|
| 200 | |

---

**DELETE**    /blogs/restaurants/unlike/{id}              blogs_restaurants_unlike_delete    🔓

**Parameters**                                                            **Try it out**

| Name | Description |
|------|-------------|
| id * required<br>string<br>*(path)* | id |

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|
| 204 | |

---

**GET**    /blogs/viewlist/{id}                                  blogs_viewlist_read    🔓

**Parameters**                                                            **Try it out**

| Name | Description |
|------|-------------|
| page<br>integer<br>*(query)* | A page number within the paginated result set.<br><br>page - A page number within the paginated re |
| id * required<br>string<br>*(path)* | id |

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|

**200**

Example Value   **Model**

```
{
    count*              integer
    next                string($uri)
                        x-nullable: true
    previous            string($uri)
                        x-nullable: true
    results*            [BlogPost {
                        post_restaurant*    integer
                                            title: Post restaurant
                        post_title*         string
                                            title: Post title
                                            maxLength: 150
                                            minLength: 1
                        post_description*   string
                                            title: Post description
                                            minLength: 1
                        post_media          string($uri)
                                            title: Post media
                                            readOnly: true
                        post_likes          integer
                                            title: Post likes
                                            readOnly: true
                        post_create_time    string($date-time)
                                            title: Post create time
                                            readOnly: true
                        }]
}
```
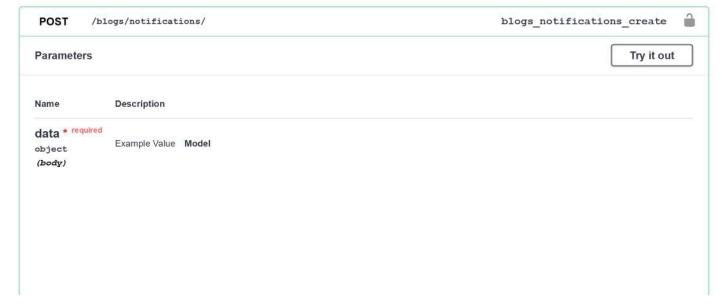
---

**POST**   /blogs/viewpost/like/{id}                                 blogs_viewpost_like_create   🔓

### Parameters

[ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
(body)

Example Value   **Model**

**BlogPostLike**   {
}

**id** * required
string
(path)

[ id ]

### Responses

Response content type   [ application/json ]

| Code | Description |
|------|-------------|

**201**

Example Value   **Model**

**BlogPostLike**   {
}

---

**GET** `/blogs/viewpost/unlike/{id}`                           blogs_viewpost_unlike_read 🔓

**Parameters**                                                                    [ Try it out ]

| Name | Description |
| --- | --- |
| id * required<br>string<br>*(path)* | id |

**Responses**                                    Response content type    [ application/json ]

| Code | Description |
| --- | --- |
| 200 | |

---

**PUT** `/blogs/viewpost/unlike/{id}`                         blogs_viewpost_unlike_update 🔓

**Parameters**                                                                    [ Try it out ]

| Name | Description |
| --- | --- |
| id * required<br>string<br>*(path)* | id |

**Responses**                                    Response content type    [ application/json ]

| Code | Description |
| --- | --- |
| 200 | |

---

**PATCH** `/blogs/viewpost/unlike/{id}`                blogs_viewpost_unlike_partial_update 🔓

**Parameters**                                                                    [ Try it out ]

| Name | Description |
| --- | --- |
| id * required<br>string<br>*(path)* | id |

**Responses**                                    Response content type    [ application/json ]

| Code | Description |
| --- | --- |
| 200 | |

### DELETE /blogs/viewpost/unlike/{id}    blogs_viewpost_unlike_delete 🔓

**Parameters**                                                          [ Try it out ]

| Name | Description |
| --- | --- |
| id * required<br>string<br>(path) | [ id ] |

**Responses**                          Response content type    [ application/json ]

| Code | Description |
| --- | --- |
| 204 | |

### GET /blogs/viewpost/{id}    blogs_viewpost_read 🔓

**Parameters**                                                          [ Try it out ]

| Name | Description |
| --- | --- |
| id * required<br>integer<br>(path) | A unique integer value identifying this blog post.<br><br>[ id - A unique integer value identifying this blo ] |

**Responses**                          Response content type    [ application/json ]

| Code | Description |
| --- | --- |
| 200 | Example Value   **Model** |

| Code | Description |
|------|-------------|

**BlogPost** {
    post_restaurant*     integer
                    *title: Post restaurant*
    post_title*     string
                    *title: Post title*
                    *maxLength: 150*
                    *minLength: 1*
    post_description*     string
                    *title: Post description*
                    *minLength: 1*
    post_media     string($uri)
                    *title: Post media*
                    *readOnly: true*
    post_likes     integer
                    *title: Post likes*
                    *readOnly: true*
    post_create_time     string($date-time)
                    *title: Post create time*
                    *readOnly: true*

}

---

**PUT**    **/blogs/viewpost/{id}**          `blogs_viewpost_update`  🔓

### Parameters

[ Try it out ]

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value   **Model**

**BlogPost** {
    post_restaurant*     integer
                    *title: Post restaurant*
    post_title*     string
                    *title: Post title*
                    *maxLength: 150*
                    *minLength: 1*
    post_description*     string
                    *title: Post description*
                    *minLength: 1*

}

**id** * required
integer
*(path)*

A unique integer value identifying this blog post.

[ id - A unique integer value identifying this blo ]

### Responses

Response content type   [ application/json ]

| Code | Description |
|------|-------------|
| 200 | |

Example Value   **Model**

| Code | Description |
|------|-------------|

**BlogPost** {
    post_restaurant*    integer
         *title: Post restaurant*
    post_title*    string
         *title: Post title*
         *maxLength: 150*
         *minLength: 1*
    post_description*    string
         *title: Post description*
         *minLength: 1*
    post_media    string($uri)
         *title: Post media*
         *readOnly: true*
    post_likes    integer
         *title: Post likes*
         *readOnly: true*
    post_create_time    string($date-time)
         *title: Post create time*
         *readOnly: true*

}

---

**PATCH**   /blogs/viewpost/{id}         blogs_viewpost_partial_update 🔓

**Parameters**                           **Try it out**

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value   **Model**

**BlogPost** {
    post_restaurant*    integer
         *title: Post restaurant*
    post_title*    string
         *title: Post title*
         *maxLength: 150*
         *minLength: 1*
    post_description*    string
         *title: Post description*
         *minLength: 1*

# comments                ⌄

---

**POST**   /comments/createcomment/{id}       comments_createcomment_create 🔓

required
**Parameters**                           **Try it out**

| Name | Description |
|------|-------------|

**data** * required
object
*(body)*

Example Value   **Model**

**Comment** {
    content*    string
         *title: Content*
         *minLength: 1*

}

**id** * required
string
*(path)*

Example Value
id

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

| Code | Description |
|------|-------------|
| 201 | **BlogPost** |

Example Value    **Model**

```
Comment      {
    restaurant_id        string
                         title: Restaurant id
                         readOnly: true
    content*             string
                         title: Content
                         minLength: 1
    comment_create_time  string($date-time)
                         title: Comment create time
                         readOnly: true

}
```

}

---

**GET**    `/comments/restaurant/{id}`                          comments_restaurant_read  🔓

### Parameters                                                                    🔒    [ Try it out ]

| Name | Description |
|------|-------------|
| **page**<br>integer<br>*(query)* required | A page number within the paginated result set.<br><br>[ page - A page number within the paginated r ] |
| **id** * required<br>string<br>*(path)* | [ id ] |

### Responses                          Response content type    [ application/json ]

| Code | Description |
|------|-------------|
| 200 | |

Example Value    **Model**

```
{
count*                integer
next                  string($uri)
                      x-nullable: true
previous              string($uri)
                      x-nullable: true
results*
                         [Comment      {
                         restaurant_id        string
                                              title: Restaurant id
                                              readOnly: true
                         content*             string
                                              title: Content
                                              minLength: 1
                         comment_create_time  string($date-time)
                                              title: Comment create time
                                              readOnly: true

                         }]

}
```

**GET**    `/comments/viewcomment/{id}`        `comments_viewcomment_read` 🔓

**Parameters**

<div align="right">

Try it out

</div>

| Name | Description |
|------|-------------|
| id * required<br>string<br>*(path)* | id |

**Responses**      Response content type    `application/json`

| Code | Description |
|------|-------------|
| 200 | Example Value   **Model** |

**Comment**    {
```
    restaurant_id        string
                         title: Restaurant id
                         readOnly: true
    content*             string
                         title: Content
                         minLength: 1
    comment_create_time  string($date-time)
                         title: Comment create time
                         readOnly: true

}
```

## restaurants      ⌄

**GET**    `/restaurants/`        `restaurants_read` 🔓

**Parameters**

<div align="right">

Try it out

</div>

No parameters

**Responses**      Response content type    `application/json`

| Code | Description |
|------|-------------|
| 200 | Example Value   **Model** |

| Code | Description |
|------|-------------|

**Restaurant** {

| id | integer |
| | *title: ID* |
| | *readOnly: true* |
| owner_id | string |
| | *title: Owner id* |
| | *readOnly: true* |
| name* | string |
| | *title: Name* |
| | *maxLength: 128* |
| | *minLength: 1* |
| address* | string |
| | *title: Address* |
| | *maxLength: 128* |
| | *minLength: 1* |
| description | string |
| | *title: Description* |
| theme | string |
| | *title: Theme* |
| | Enum: |
| | Array [ 7 ] |
| primary_image | string($uri) |
| | *title: Primary image* |
| | *readOnly: true* |
| image | string($uri) |
| | *title: Image* |
| | *readOnly: true* |
| banner | string($uri) |
| | *title: Banner* |
| | *readOnly: true* |
| phone | string |
| | *title: Phone* |
| | *pattern: ^\d{3}-\d{3}-\d{4}$* |
| | *maxLength: 12* |
| fax | string |
| | *title: Fax* |
| | *pattern: ^\d{3}-\d{3}-\d{4}$* |
| | *maxLength: 12* |
| website | string($uri) |
| | *title: Website* |
| | *maxLength: 253* |
| email | string($email) |
| | *title: Email* |
| | *maxLength: 254* |
| number_likes | integer |
| | *title: Number likes* |
| number_follows | integer |
| | *title: Number follows* |
| liked | string |
| | *title: Liked* |
| | *readOnly: true* |
| followed | string |
| | *title: Followed* |
| | *readOnly: true* |

}

---

**POST**  /restaurants/                                    restaurants_create  🔓

| Parameters | Try it out |
|------------|------------|

| Name | Description |
|------|-------------|
| **data** * required<br>object<br>*(body)* | Example Value  **Model** |

| Name | Description |
|------|-------------|

**Restaurant** {

| | |
|---|---|
| name* | string<br>*title: Name*<br>*maxLength: 128*<br>*minLength: 1* |
| address* | string<br>*title: Address*<br>*maxLength: 128*<br>*minLength: 1* |
| description | string<br>*title: Description* |
| theme | string<br>*title: Theme*<br>Enum:<br>　　Array [ 7 ] |
| phone | string<br>*title: Phone*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| fax | string<br>*title: Fax*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| website | string($uri)<br>*title: Website*<br>*maxLength: 253* |
| email | string($email)<br>*title: Email*<br>*maxLength: 254* |
| number_likes | integer<br>*title: Number likes* |
| number_follows | integer<br>*title: Number follows* |

}

## Responses

Response content type　　application/json

| Code | Description |
|------|-------------|
| 201 | Example Value　**Model** |

| Code | Description |
|---|---|

**Restaurant** {
| | |
|---|---|
| id | integer<br>*title: ID*<br>*readOnly: true* |
| owner_id | string<br>*title: Owner id*<br>*readOnly: true* |
| name* | string<br>*title: Name*<br>*maxLength: 128*<br>*minLength: 1* |
| address* | string<br>*title: Address*<br>*maxLength: 128*<br>*minLength: 1* |
| description | string<br>*title: Description* |
| theme | string<br>*title: Theme*<br>Enum:<br><br>   Array [ 7 ] |
| primary_image | string($uri)<br>*title: Primary image*<br>*readOnly: true* |
| image | string($uri)<br>*title: Image*<br>*readOnly: true* |
| banner | string($uri)<br>*title: Banner*<br>*readOnly: true* |
| phone | string<br>*title: Phone*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| fax | string<br>*title: Fax*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| website | string($uri)<br>*title: Website*<br>*maxLength: 253* |
| email | string($email)<br>*title: Email*<br>*maxLength: 254* |
| number_likes | integer<br>*title: Number likes* |
| number_follows | integer<br>*title: Number follows* |
| liked | string<br>*title: Liked*<br>*readOnly: true* |
| followed | string<br>*title: Followed*<br>*readOnly: true* |

}

---

**PUT** /restaurants/ restaurants_update 🔓

### Parameters

[ Try it out ]

| Name | Description |
|---|---|
| **data** * required<br>object<br>*(body)* | Example Value **Model** |

| Name | Description |
| --- | --- |

**Restaurant** {

| | |
| --- | --- |
| name* | string |
| | title: Name |
| | maxLength: 128 |
| | minLength: 1 |
| address* | string |
| | title: Address |
| | maxLength: 128 |
| | minLength: 1 |
| description | string |
| | title: Description |
| theme | string |
| | title: Theme |
| | Enum: |
| | Array [ 7 ] |
| phone | string |
| | title: Phone |
| | pattern: ^\d{3}-\d{3}-\d{4}$ |
| | maxLength: 12 |
| fax | string |
| | title: Fax |
| | pattern: ^\d{3}-\d{3}-\d{4}$ |
| | maxLength: 12 |
| website | string($uri) |
| | title: Website |
| | maxLength: 253 |
| email | string($email) |
| | title: Email |
| | maxLength: 254 |
| number_likes | integer |
| | title: Number likes |
| number_follows | integer |
| | title: Number follows |

}

## Responses

Response content type    application/json

| Code | Description |
| --- | --- |
| 200 | |
| | Example Value   **Model** |

| Code | Description |
|------|-------------|

**Restaurant** {

| | |
|---|---|
| id | integer |
| | *title: ID* |
| | *readOnly: true* |
| owner_id | string |
| | *title: Owner id* |
| | *readOnly: true* |
| name* | string |
| | *title: Name* |
| | *maxLength: 128* |
| | *minLength: 1* |
| address* | string |
| | *title: Address* |
| | *maxLength: 128* |
| | *minLength: 1* |
| description | string |
| | *title: Description* |
| theme | string |
| | *title: Theme* |
| | Enum: |
| | Array [ 7 ] |
| primary_image | string($uri) |
| | *title: Primary image* |
| | *readOnly: true* |
| image | string($uri) |
| | *title: Image* |
| | *readOnly: true* |
| banner | string($uri) |
| | *title: Banner* |
| | *readOnly: true* |
| phone | string |
| | *title: Phone* |
| | *pattern: ^\d{3}-\d{3}-\d{4}$* |
| | *maxLength: 12* |
| fax | string |
| | *title: Fax* |
| | *pattern: ^\d{3}-\d{3}-\d{4}$* |
| | *maxLength: 12* |
| website | string($uri) |
| | *title: Website* |
| | *maxLength: 253* |
| email | string($email) |
| | *title: Email* |
| | *maxLength: 254* |
| number_likes | integer |
| | *title: Number likes* |
| number_follows | integer |
| | *title: Number follows* |
| liked | string |
| | *title: Liked* |
| | *readOnly: true* |
| followed | string |
| | *title: Followed* |
| | *readOnly: true* |

}

---

**PATCH** /restaurants/      restaurants_partial_update 🔓

**Parameters**     [ Try it out ]

| Name | Description |
|------|-------------|
| **data** * required<br>object<br>*(body)* | Example Value   **Model** |

| Name | Description |
| --- | --- |

**Restaurant**    {

    name*             string
                       *title: Name*
                       *maxLength: 128*
                       *minLength: 1*

    address*        string
                       *title: Address*
                       *maxLength: 128*
                       *minLength: 1*

    description    string
                       *title: Description*

    theme           string
                       *title: Theme*
                       Enum:
                           Array [ 7 ]

    phone           string
                       *title: Phone*
                       *pattern: ^\d{3}-\d{3}-\d{4}$*
                       *maxLength: 12*

    fax             string
                       *title: Fax*
                       *pattern: ^\d{3}-\d{3}-\d{4}$*
                       *maxLength: 12*

    website       string($uri)
                       *title: Website*
                       *maxLength: 253*

    email           string($email)
                       *title: Email*
                       *maxLength: 254*

    number_likes    integer
                       *title: Number likes*

    number_follows   integer
                       *title: Number follows*

}

## Responses

Response content type    application/json

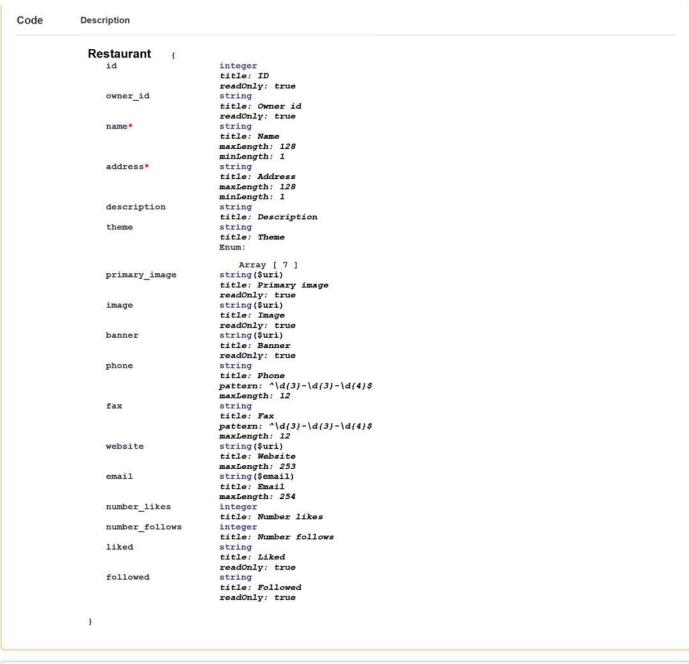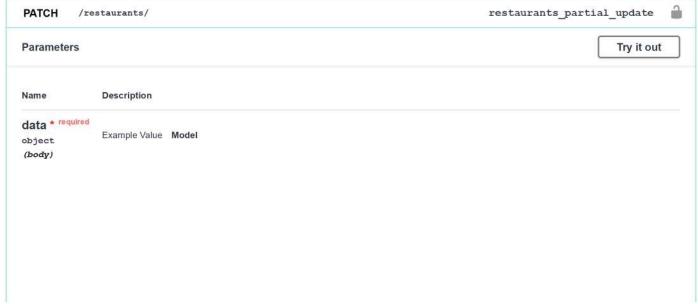| Code | Description |
| --- | --- |
| 200 | Example Value    **Model** |

| Code | Description |
|------|-------------|

**Restaurant** {

| | |
|--|--|
| id | integer<br>*title: ID*<br>*readOnly: true* |
| owner_id | string<br>*title: Owner id*<br>*readOnly: true* |
| name* | string<br>*title: Name*<br>*maxLength: 128*<br>*minLength: 1* |
| address* | string<br>*title: Address*<br>*maxLength: 128*<br>*minLength: 1* |
| description | string<br>*title: Description* |
| theme | string<br>*title: Theme*<br>Enum:<br><br>Array [ 7 ] |
| primary_image | string($uri)<br>*title: Primary image*<br>*readOnly: true* |
| image | string($uri)<br>*title: Image*<br>*readOnly: true* |
| banner | string($uri)<br>*title: Banner*<br>*readOnly: true* |
| phone | string<br>*title: Phone*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| fax | string<br>*title: Fax*<br>*pattern: ^\d{3}-\d{3}-\d{4}$*<br>*maxLength: 12* |
| website | string($uri)<br>*title: Website*<br>*maxLength: 253* |
| email | string($email)<br>*title: Email*<br>*maxLength: 254* |
| number_likes | integer<br>*title: Number likes* |
| number_follows | integer<br>*title: Number follows* |
| liked | string<br>*title: Liked*<br>*readOnly: true* |
| followed | string<br>*title: Followed*<br>*readOnly: true* |

}

---

**GET** /restaurants/hours/day/{day}/                    restaurants_hours_day_read 🔓

**Parameters**                                                          [ Try it out ]

| Name | Description |
|------|-------------|
| **day** * required<br>string<br>*(path)* | [ day ] |

**Responses**            Response content type   [ application/json ]

| Code | Description |
|------|-------------|

| Code | Description |
|------|-------------|
| 200  |             |

---

**PUT**      /restaurants/hours/day/{day}/          restaurants_hours_day_update  🔓

**Parameters**                                                    Try it out

| Name | Description |
|------|-------------|

**day** * required
string
*(path)*

[ day ]

**Responses**                          Response content type   | application/json |

| Code | Description |
|------|-------------|
| 200  |             |

---

**PATCH**    /restaurants/hours/day/{day}/       restaurants_hours_day_partial_update  🔓

**Parameters**                                                    Try it out

| Name | Description |
|------|-------------|

**day** * required
string
*(path)*

[ day ]

**Responses**                          Response content type   | application/json |

| Code | Description |
|------|-------------|
| 200  |             |

---

**GET**      /restaurants/hours/list/{restaurant_id}/      restaurants_hours_list_read  🔓

**Parameters**                                                    Try it out

| Name | Description |
|------|-------------|
| **restaurant_id** * required<br>`string`<br>*(path)* | restaurant_id |

**Responses**                                   Response content type   [ application/json ]

| Code | Description |
|------|-------------|
| 200 | Example Value   **Model** |

```
[Time      {
day            string
               title: Day
               readOnly: true
               Enum:
                   Array [ 7 ]
open_time      string
               title: Open time
               x-nullable: true
close_time     string
               title: Close time
               x-nullable: true

}]
```

---

**POST**   `/restaurants/image/`                    restaurants_image_create 🔓

**Parameters**                                                          [ Try it out ]

No parameters

**Responses**                                   Response content type   [ application/json ]

| Code | Description |
|------|-------------|
| 201 | |

---

**GET**   `/restaurants/image/list/{restaurant_id}/`      restaurants_image_list_read 🔓

**Parameters**                                                          [ Try it out ]

| Name | Description |
|------|-------------|
| **page**<br>`integer`<br>*(query)* | A page number within the paginated result set.<br>page - A page number within the paginated re |

| Name | Description |
|------|-------------|
| **restaurant_id** * required<br>string<br>*(path)* | restaurant_id |

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|
| 200 | |

Example Value    **Model**

```
{
    count*              integer
    next                string($uri)
                        x-nullable: true
    previous            string($uri)
                        x-nullable: true
    results*            [RestaurantImage    {
                        id                  integer
                                            title: ID
                                            readOnly: true
                        restaurant_id       string
                                            title: Restaurant id
                                            readOnly: true
                        image               string($uri)
                                            title: Image
                                            readOnly: true
                        position            integer
                                            title: Position

                        }]

}
```

---

**GET**    `/restaurants/image/{image_id}/`                    restaurants_image_read   🔓

**Parameters**                                                              Try it out

| Name | Description |
|------|-------------|
| **image_id** * required<br>string<br>*(path)* | image_id |

**Responses**                                          Response content type    application/json

| Code | Description |
|------|-------------|
| 200 | |

---

**DELETE**    `/restaurants/image/{image_id}/`                 restaurants_image_delete   🔓

## Parameters

Try it out

| Name | Description |
|---|---|
| **image_id** * required<br>string<br>*(path)* | image_id |

## Responses

Response content type: application/json

| Code | Description |
|---|---|
| 204 | |

---

**POST** `/restaurants/menu/`                    restaurants_menu_create 🔓

## Parameters

Try it out

No parameters

## Responses

Response content type: application/json

| Code | Description |
|---|---|
| 201 | |

---

**GET** `/restaurants/menu/list/{restaurant_id}/`        restaurants_menu_list_read 🔓

## Parameters

Try it out

| Name | Description |
|---|---|
| **page**<br>integer<br>*(query)* | A page number within the paginated result set.<br><br>page - A page number within the paginated re |
| **restaurant_id** * required<br>string<br>*(path)* | restaurant_id |

## Responses

Response content type: application/json

Code          Description

200

Example Value    **Model**

```
{
count*              integer
next                string($uri)
                    x-nullable: true
previous            string($uri)
                    x-nullable: true
results*
        [FoodItem    {
        id              integer
                        title: ID
                        readOnly: true
        restaurant_id   string
                        title: Restaurant id
                        readOnly: true
        name*           string
                        title: Name
                        maxLength: 128
                        minLength: 1
        details*        string
                        title: Details
                        minLength: 1
        price           number($decimal)
                        title: Price
                        x-nullable: true
        position        integer
                        title: Position

        }]

}
```

---

**GET**      /restaurants/menu/{menuitem_id}/                            restaurants_menu_read  🔓

Parameters                                                                 Try it out

| Name | Description |
|------|-------------|
| menuitem_id * required<br>string<br>(path) | menuitem_id |

Responses                                           Response content type    application/json

Code          Description

200

---

**PUT**      /restaurants/menu/{menuitem_id}/                            restaurants_menu_update  🔓

Parameters                                                                 Try it out

| Name | Description |
|------|-------------|

**menuitem_id** * required
string
*(path)*

[ menuitem_id ]

**Responses**                          Response content type    [ application/json ]

| Code | Description |
|------|-------------|
| 200 | |

---

**PATCH**   /restaurants/menu/{menuitem_id}/                    restaurants_menu_partial_update  🔓

**Parameters**                                              [ Try it out ]

| Name | Description |
|------|-------------|

**menuitem_id** * required
string
*(path)*

[ menuitem_id ]

**Responses**                          Response content type    [ application/json ]

| Code | Description |
|------|-------------|
| 200 | |

---

**DELETE**   /restaurants/menu/{menuitem_id}/                    restaurants_menu_delete  🔓

**Parameters**                                              [ Try it out ]

| Name | Description |
|------|-------------|

**menuitem_id** * required
string
*(path)*

[ menuitem_id ]

**Models**                                                              ⌄

**User**

**Responses**                          Response content type  |  application/json

**TokenObtainPair**

**TokenRefresh**

204

**BlogPost**

GET /restaurants/search/ restaurants_search_list

**Notification**

**RestaurantFollow**

page

A page number within the paginated result set

**RestaurantLike**

**BlogPostLike**

**Comment**

Code Description

**Restaurant**

**Time**

**RestaurantImage**

**FoodItem**

```
                {
count*               integer
next                 string($uri)
                     x-nullable: true
previous             string($uri)
                     x-nullable: true
results*
            [Restaurant        {
id                   integer
                     title: ID
                     readOnly: true
owner_id             string
                     title: Owner id
                     readOnly: true
name*                string
                     title: Name
                     maxLength: 128
                     minLength: 1
address*             string
                     title: Address
                     maxLength: 128
                     minLength: 1
description          string
                     title: Description
theme                string
                     title: Theme
                     Enum:
                         Array [ 7 ]
primary_image        string($uri)
                     title: Primary image
                     readOnly: true
image                string($uri)
                     title: Image
                     readOnly: true
banner               string($uri)
                     title: Banner
                     readOnly: true
phone                string
                     title: Phone
                     pattern: ^\d{3}-\d{3}-\d{4}$
                     maxLength: 12
fax                  string
                     title: Fax
                     pattern: ^\d{3}-\d{3}-\d{4}$
                     maxLength: 12
website              string($uri)
                     title: Website
                     maxLength: 253
email                string($email)
                     title: Email
                     maxLength: 254
number_likes         integer
                     title: Number likes
number_follows       integer
                     title: Number follows
liked                string
                     title: Liked
                     readOnly: true
followed             string
                     title: Followed
                     readOnly: true
            }]

        }
```

---

**GET**  /restaurants/{restaurant_id}/                        restaurants_read  🔓

### Parameters                                                        [ Try it out ]

| Name | Description |
| --- | --- |
| restaurant_id * required<br>string<br>(path) | restaurant_id |

**Responses**                 Response content type    `application/json`

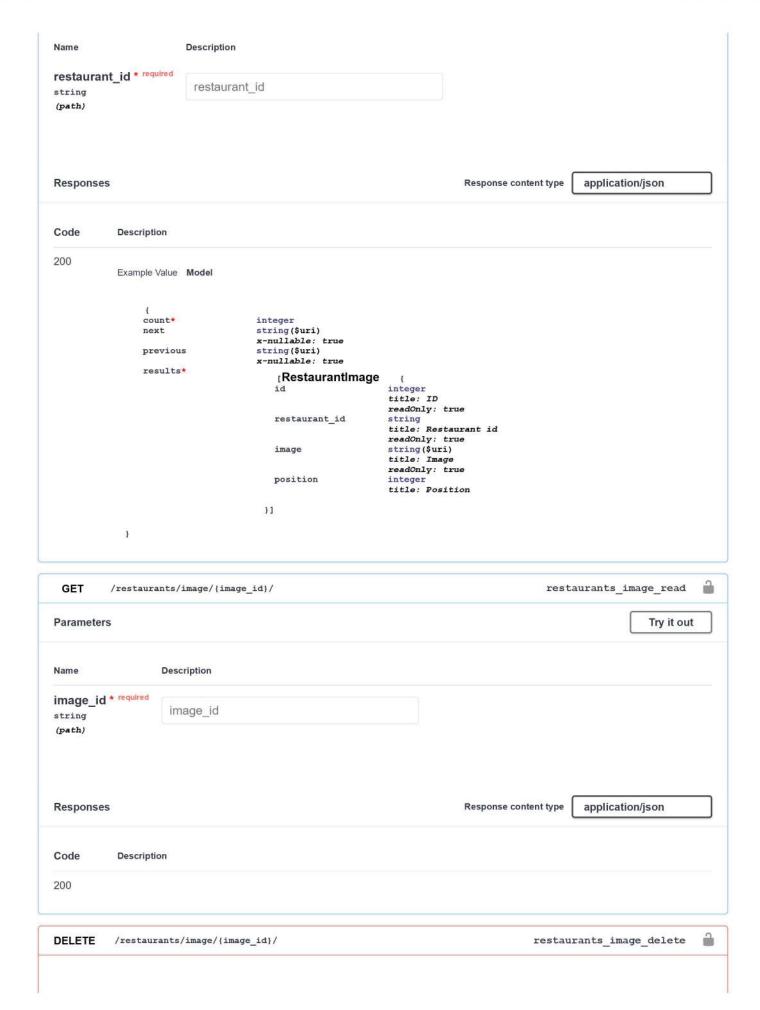| Code | Description |
| --- | --- |
| 200 | |

Example Value   **Model**

**Restaurant**   {

```
id                integer
                  title: ID
                  readOnly: true
owner_id          string
                  title: Owner id
                  readOnly: true
name*             string
                  title: Name
                  maxLength: 128
                  minLength: 1
address*          string
                  title: Address
                  maxLength: 128
                  minLength: 1
description       string
                  title: Description
theme             string
                  title: Theme
                  Enum:

                    Array [ 7 ]
primary_image     string($uri)
                  title: Primary image
                  readOnly: true
image             string($uri)
                  title: Image
                  readOnly: true
banner            string($uri)
                  title: Banner
                  readOnly: true
phone             string
                  title: Phone
                  pattern: ^\d{3}-\d{3}-\d{4}$
                  maxLength: 12
fax               string
                  title: Fax
                  pattern: ^\d{3}-\d{3}-\d{4}$
                  maxLength: 12
website           string($uri)
                  title: Website
                  maxLength: 253
email             string($email)
                  title: Email
                  maxLength: 254
number_likes      integer
                  title: Number likes
number_follows    integer
                  title: Number follows
liked             string
                  title: Liked
                  readOnly: true
followed          string
                  title: Followed
                  readOnly: true

}
```