Restaurant Social Media Networking:

Restify

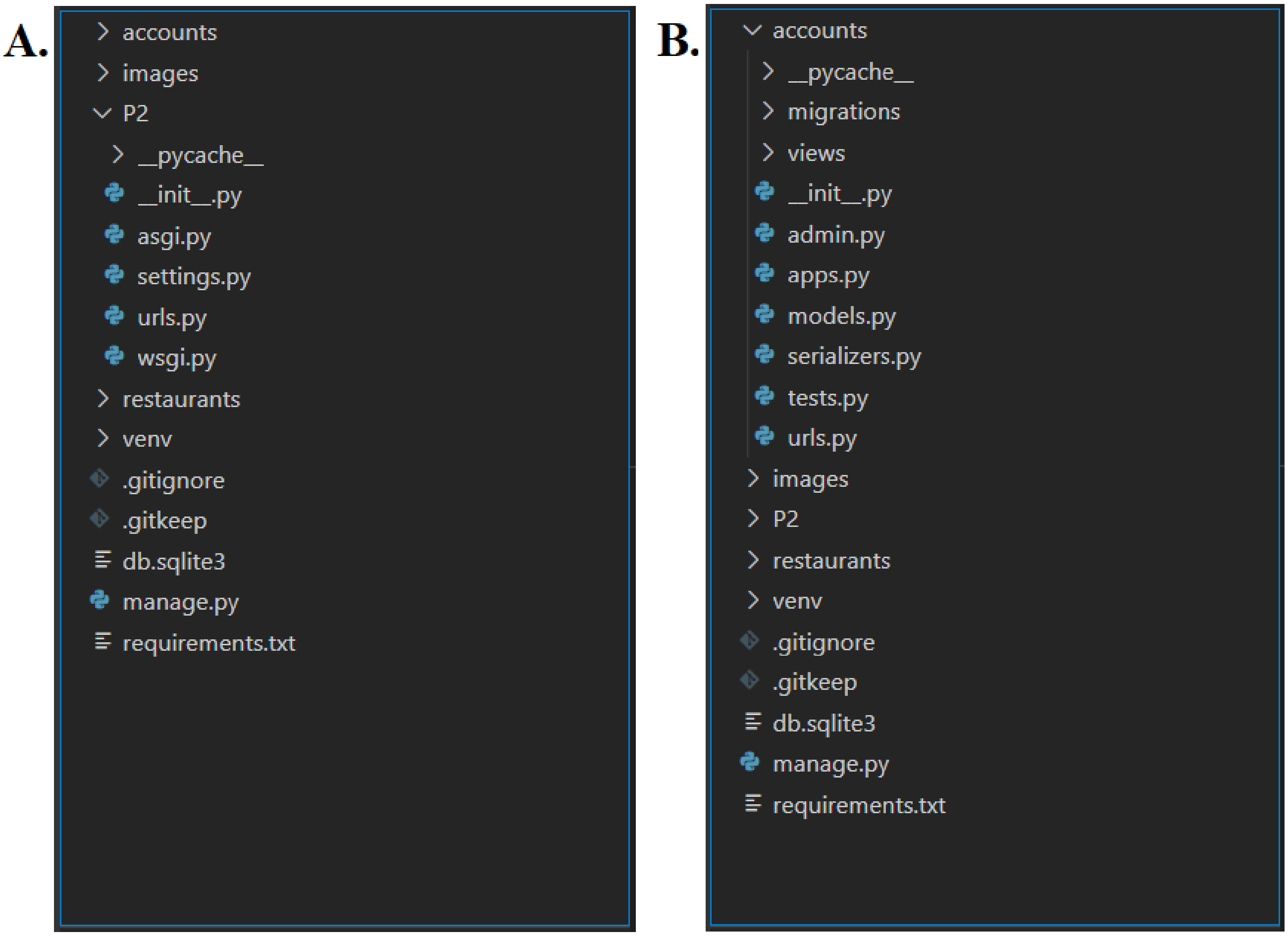
Roy Hsin Yu Lu
Riley Hannigan
Shaelynn Keiko Curran

March 15th, 2022

Table of Contents

Project Structure

Within the group__0042 repository, the entire django project, db.sqlite3 database, and virtual environment can be found after installation in the P2 directory (**Figure 1**).

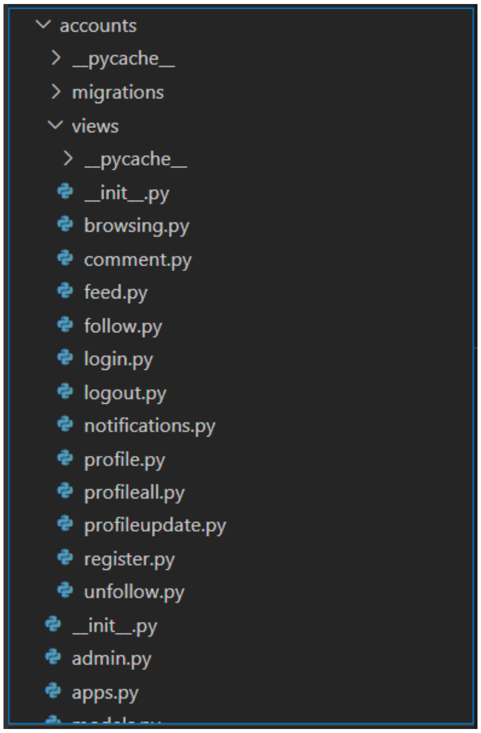


```
> accounts
> images
v P2
  > __pycache__
  __init__.py
  asgi.py
  settings.py
  urls.py
  wsgi.py
> restaurants
> venv
.gitignore
.gitkeep
db.sqlite3
manage.py
requirements.txt
```

```
v accounts
  > __pycache__
  > migrations
  > views
  __init__.py
  admin.py
  apps.py
  models.py
  serializers.py
  tests.py
  urls.py
> images
> P2
> restaurants
> venv
.gitignore
.gitkeep
db.sqlite3
manage.py
requirements.txt
```

Figure 1. A screenshot of the directory structure of the django project. A) Indicates the main project directory P2, containing settings.py. B) Indicates the accounts app directory, containing admin.py, models.py, serializers.py, urls.py, and a directory for views.

The name of the django project is P2, and it contains two apps: **accounts** and **restaurants**. Both the accounts and the restaurants apps' directory structures are similar, containing a views directory within, in addition to the default files upon creation (**Figure 2**). The views directory was created in place of the default views.py file in order to sort the endpoints within their respective .py files for organization purposes.



```
⌄ accounts
  › __pycache__
  › migrations
  ⌄ views
    › __pycache__
      __init__.py
      browsing.py
      comment.py
      feed.py
      follow.py
      login.py
      logout.py
      notifications.py
      profile.py
      profileall.py
      profileupdate.py
      register.py
      unfollow.py
    __init__.py
    admin.py
    apps.py
```

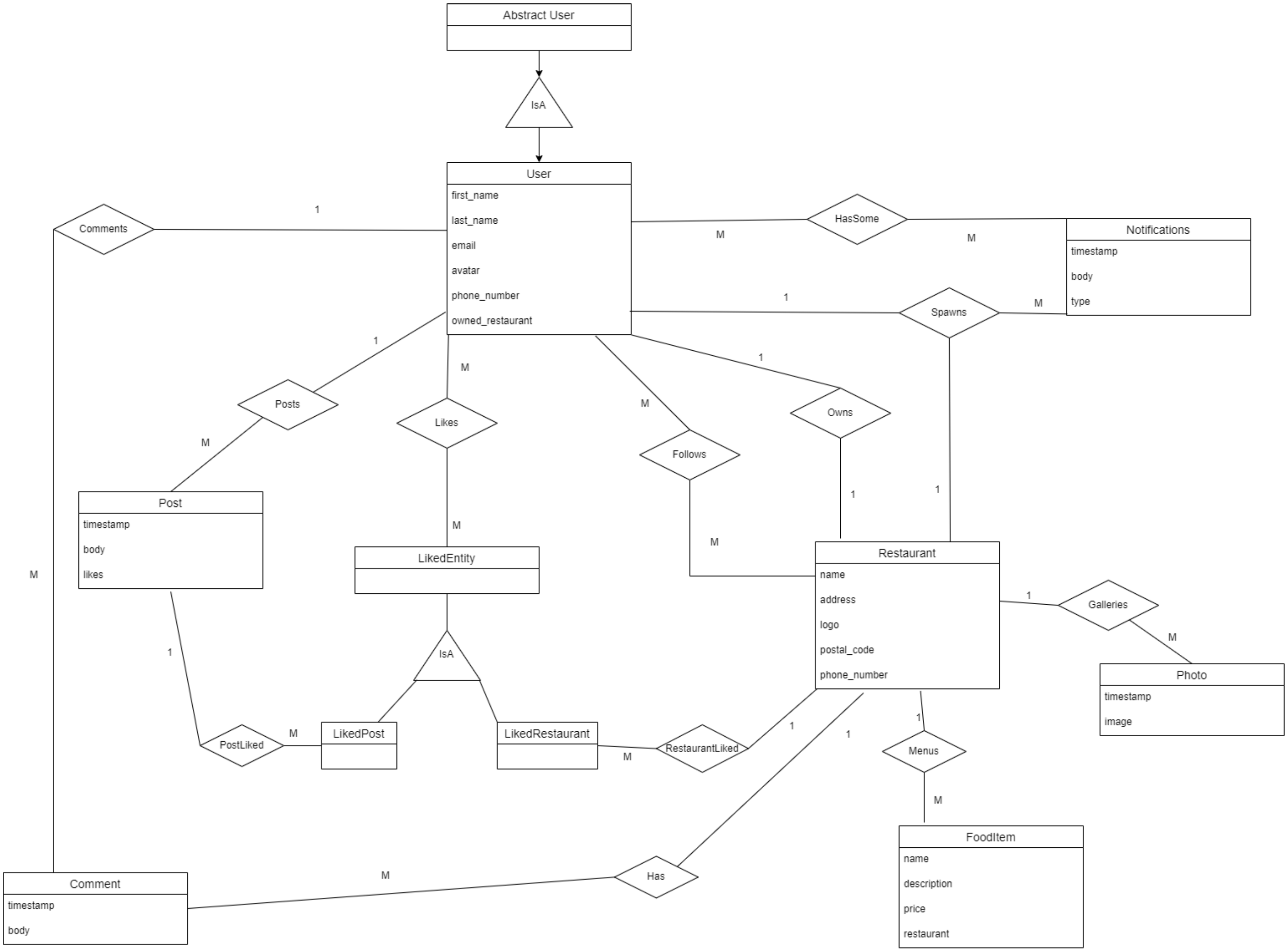
Figure 2. A screenshot of the directory structure of the accounts app, showing the layout of the views directory.

The **accounts** app handles all the models and endpoints associated with user registration, authentication, profile information, notifications, following, user's feed, and the home browsing page of popular restaurants. The user's feed retrieves all the blog posts of followed restaurants and the home browsing acts as a home menu for users and non-users alike that would like to browse the popular restaurants.

The **restaurants** app handles all the models and endpoints associated with restaurant creation and updates, menu creation and updates, blog post creation and deletion, comment creation and retrieval, and restaurant photo creation and deletion and viewing. The restaurants app also contains the functionality of liking a post/restaurant and the search function to allow the user to search restaurants by name, foods, or address.

The images directory contained within the P2 directory is the location where all images uploaded by the user upon registration (avatar), by the restaurant upon creation (logo), and by the restaurant's gallery (add photos).

ER Diagram


Abstract User
IsA
User
first_name
last_name
email
avatar
phone_number
owned_restaurant
Comments
1
HasSome
M
M
Notifications
timestamp
body
type
1
Spawns
M
1
Posts
1
M
Likes
M
Follows
1
Owns
M
1
1
Post
timestamp
body
likes
M
LikedEntity
M
Restaurant
name
address
logo
postal_code
phone_number
1
Galleries
M
Photo
timestamp
image
1
IsA
PostLiked
M
LikedPost
LikedRestaurant
M
RestaurantLiked
1
1
1
Menus
M
FoodItem
name
description
price
restaurant
M
Has
Comment
timestamp
body

http://127.0.0.1:8000/ Explore

Restaurant API v1

`[ Base URL: 127.0.0.1:8000/ ]`

http://127.0.0.1:8000/doc/?format=openapi

API for our restaurant app

Schemes

HTTP

Django Login

Authorize

Filter by tag

accounts

GET `/accounts/api/browse/` `accounts_api_browse_list`

Returns all restaurants in order of popularity

Parameters

Try it out

Name	Description

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

page_size
`integer`
`(query)`
	Number of results to return per page.
	page_size - Number of results to return per p

Responses

Response content type application/json

Code **Description**

Code	Description
200	Example Value **Model**

```
 {
count*                 integer
next                   string($uri)
                       x-nullable: true
previous               string($uri)
                       x-nullable: true
results*
                           [Browsing     {
                          name*                  string
                                                 title: Name
                                                 maxLength: 200
                                                 minLength: 1
                          address*               string
                                                 title: Address
                                                 maxLength: 200
                                                 minLength: 1
                          postal_code*           string
                                                 title: Postal code
                                                 maxLength: 7
                                                 minLength: 1
                          likes                  integer
                                                 title: Likes

                        }]

}
```

GET `/accounts/api/feed/` `accounts_api_feed_list`

Returns all the posts from the restaurants the user is following in reverse chronological order

Parameters Try it out

Name	Description
page integer *(query)*	A page number within the paginated result set. page - A page number within the paginated re
page_size integer *(query)*	Number of results to return per page. page_size - Number of results to return per p

Responses **Response content type** application/json

Code	Description

Code	Description
200	Example Value **Model**

```
{
count*                  integer
next                    string($uri)
                        x-nullable: true
previous                string($uri)
                        x-nullable: true
results*
                       [Feed    {
                        timestamp              string($date-time)
                                               title: Timestamp
                                               readOnly: true
                        body*                  string
                                               title: Body
                                               minLength: 1
                        likes                  integer
                                               title: Likes
                        user*                  integer
                                               title: User
                        restaurant*            integer
                                               title: Restaurant

                      }]

}
```

POST `/accounts/api/login/` `accounts_api_login_create`

Logs a user in after being given proper authentication

Parameters Try it out

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

```
Login    {
   username*               string
                           title: Username
                           minLength: 1
   password*               string
                           title: Password
                           minLength: 1

}
```

Responses Response content type application/json

Code	Description
201	Example Value **Model**

Code **Description**

```
Login    {
   username*            string
                        title: Username
                        minLength: 1
   password*            string
                        title: Password
                        minLength: 1

}
```

POST `/accounts/api/login/refresh/` `accounts_api_login_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/api/notifications/` `accounts_api_notifications_list`

Returns list of notifications for a user in reverse chronological order

Parameters

Try it out

Name	Description
page integer *(query)*	A page number within the paginated result set. page - A page number within the paginated re
page_size integer *(query)*	Number of results to return per page. page_size - Number of results to return per p

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     {
                       source_id*          integer
                                           title: Source id
                       target_id*          integer
                                           title: Target id
                       timestamp           string($date-time)
                                           title: Timestamp
                                           readOnly: true

                                           Time of the event occurrence

                       body*               string
                                           title: Body
                                           maxLength: 200
                                           minLength: 1

                                           Description of the event

                       type*               string
                                           title: Type
                                           Enum:

                                              Array [ 5 ]

                     }]

}
```

PUT `/accounts/api/profile/edit/` `accounts_api_profile_edit_update` 🔓

Edit user's profile

Parameters Try it out

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

```
ProfileUpdate    {
    first_name           string
                         title: First name
                         maxLength: 150
    last_name            string
                         title: Last name
                         maxLength: 150
    password             string
                         title: Password
                         minLength: 1
    password2            string
                         title: Password2
                         minLength: 1
    email                string($email)
                         title: Email address
                         maxLength: 254
    phone_number         string
                         title: Phone number
                         maxLength: 31
                         x-nullable: true

                         Account phone number


}
```

Responses Response content type application/json

Code	Description
200	Example Value **Model**

Code	Description

ProfileUpdate {

```
    id                  string
                        title: Id
                        readOnly: true
                        minLength: 1
    first_name          string
                        title: First name
                        maxLength: 150
    last_name           string
                        title: Last name
                        maxLength: 150
    password            string
                        title: Password
                        minLength: 1
    password2           string
                        title: Password2
                        minLength: 1
    email               string($email)
                        title: Email address
                        maxLength: 254
    phone_number        string
                        title: Phone number
                        maxLength: 31
                        x-nullable: true

                        Account phone number

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

}
```

PATCH `/accounts/api/profile/edit/` `accounts_api_profile_edit_partial_update`

Edit user's profile

Parameters Try it out

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

Name	Description
	ProfileUpdate { ... }

```
ProfileUpdate    {
   first_name          string
                       title: First name
                       maxLength: 150
   last_name           string
                       title: Last name
                       maxLength: 150
   password            string
                       title: Password
                       minLength: 1
   password2           string
                       title: Password2
                       minLength: 1
   email               string($email)
                       title: Email address
                       maxLength: 254
   phone_number        string
                       title: Phone number
                       maxLength: 31
                       x-nullable: true

                       Account phone number

}
```

Responses   Response content type **application/json**

Code	Description
200	Example Value **Model**

```
ProfileUpdate    {
   id                  string
                       title: Id
                       readOnly: true
                       minLength: 1
   first_name          string
                       title: First name
                       maxLength: 150
   last_name           string
                       title: Last name
                       maxLength: 150
   password            string
                       title: Password
                       minLength: 1
   password2           string
                       title: Password2
                       minLength: 1
   email               string($email)
                       title: Email address
                       maxLength: 254
   phone_number        string
                       title: Phone number
                       maxLength: 31
                       x-nullable: true

                       Account phone number

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

}
```

GET `/accounts/api/profile/{id}/` accounts_api_profile_read 🔒

Returns the profile of a user with id of id
To create please provide:

- Path Param - **id**: the id of the user who's profile we are viewing

Parameters

Try it out

Name	Description
id * required string *(path)*	id

Responses

Response content type: application/json

Code	Description
200	Example Value **Model**

```
Profile  {
    id                    integer
                          title: ID
                          readOnly: true
    first_name            string
                          title: First name
                          maxLength: 150
    last_name             string
                          title: Last name
                          maxLength: 150
    username*             string
                          title: Username
                          pattern: ^[\w.@+-]+$
                          maxLength: 150
                          minLength: 1

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

    email                 string($email)
                          title: Email address
                          maxLength: 254
    phone_number          string
                          title: Phone number
                          maxLength: 31
                          x-nullable: true

                          Account phone number

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

                          Account profile icon

}
```

POST `/accounts/api/register/` accounts_api_register_create 🔒

Register a new user with the given information

Parameters Try it out

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

```
Register  {
   first_name           string
                        title: First name
                        maxLength: 150
   last_name            string
                        title: Last name
                        maxLength: 150
   password*            string
                        title: Password
                        minLength: 1
   password2*           string
                        title: Password2
                        minLength: 1
   username*            string
                        title: Username
                        minLength: 1
   email*               string
                        title: Email
                        minLength: 1
   phone_number         string
                        title: Phone number
                        maxLength: 31
                        x-nullable: true

                        Account phone number

}
```

Responses Response content type application/json

Code	Description
201	Example Value **Model**

Code **Description**

```
Register    {
    first_name           string
                         title: First name
                         maxLength: 150
    last_name            string
                         title: Last name
                         maxLength: 150
    password*            string
                         title: Password
                         minLength: 1
    password2*           string
                         title: Password2
                         minLength: 1
    username*            string
                         title: Username
                         minLength: 1
    email*               string
                         title: Email
                         minLength: 1
    phone_number         string
                         title: Phone number
                         maxLength: 31
                         x-nullable: true

                         Account phone number

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

                         Account profile icon


}
```

PUT `/accounts/api/restaurant/{id}/follow/` `accounts_api_restaurant_follow_update` 🔒

Adds restaurant to user's follower list, adds user to restaurant's following list
To create please provide:

- Path Param - **id**: the id of the restaurant that the user wants to follow

Parameters Try it out

Name	Description
data * required `object` *(body)*	Example Value **Model**

```
FollowedRestaurant    {
    followers
                          [
                      uniqueItems: true
                      integer]

}
```

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

Responses Response content type application/json

Code	Description
200	Example Value **Model**

```
FollowedRestaurant  {
   followers                  [
                          uniqueItems: true
                          integer]

}
```

PATCH `/accounts/api/restaurant/{id}/follow/` `accounts_api_restaurant_follow_partial_update`

Adds restaurant to user's follower list, adds user to restaurant's following list
To create please provide:

- Path Param - **id**: the id of the restaurant that the user wants to follow

Parameters Try it out

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

```
FollowedRestaurant  {
   followers                  [
                          uniqueItems: true
                          integer]

}
```

id * required string *(path)*	id

Responses Response content type application/json

Code	Description
200	Example Value **Model**

Code **Description**

FollowedRestaurant

```
{
followers    [
             uniqueItems: true
             integer]
}
```

PUT `/accounts/api/restaurant/{id}/unfollow/` `accounts_api_restaurant_unfollow_update` 🔒

Removes restaurant from user's follower list, removes user from restaurant's following list
To create please provide:

- Path Param - **id**: the id of the restaurant that the user wants to unfollow

Parameters Try it out

Name	Description
data * required object *(body)*	Example Value **Model** **UnfollowedRestaurant**

```
{
followers    [
             uniqueItems: true
             integer]
}
```

restaurants

GET required `/restaurants/api/restaurant/comment/{id}/` Adds a comment to a restaraunt To create please provide: `restaurants_api_restaurant_comment_read` 🔒

- Path Param - **id**: the id of the comment being retrieved

Parameters Try it out

Name	Description
id * required string *(path)*	id Example Value **UnfollowedRestaurant**

Responses Response content type application/json

Code **Description**

🔒

Code	Description
200	Example Value **Model**

```
Comment    {
   timestamp         string($date-time)
                     title: Timestamp
                     readOnly: true
   body*             string
                     title: Body
                     minLength: 1
   user              integer
                     title: User
                     readOnly: true
   restaurant        integer
                     title: Restaurant
                     readOnly: true
}
```

required

Example Value

UnfollowedRestaurant

uniqueItems: true
integer]

POST `/restaurants/api/restaurant/comment/{id}/post/` Adds a comment to a restaraunt To create please provide: `restaurants_api_restaurant_comment_post_create`

required

- Path Param - **id**: the id of the restaurant being commented on

Parameters Try it out

Name	Description
id * required string *(path)*	id

Example Value

UnfollowedRestaurant

Responses Response content type application/json

Code	Description
201	

POST `/restaurants/api/restaurant/create/` `restaurants_api_restaurant_create_create`

Creates a restaurant for the current user
The user must not already have a restaurant

Parameters Try it out

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

```
Restaurant   {
    name*            string
                     title: Name
                     maxLength: 200
                     minLength: 1
    address*         string
                     title: Address
                     maxLength: 200
                     minLength: 1
    postal_code*     string
                     title: Postal code
                     maxLength: 7
                     minLength: 1
    phone_number     string
                     title: Phone number
                     maxLength: 31
                     x-nullable: true

                     Account phone number


}
```

Responses Response content type application/json

Code	Description
201	Example Value **Model**

Code **Description**

```
Restaurant  {
   name*                 string
                         title: Name
                         maxLength: 200
                         minLength: 1
   address*              string
                         title: Address
                         maxLength: 200
                         minLength: 1
   logo                  string($uri)
                         title: Logo
                         readOnly: true
                         x-nullable: true
   postal_code*          string
                         title: Postal code
                         maxLength: 7
                         minLength: 1
   phone_number          string
                         title: Phone number
                         maxLength: 31
                         x-nullable: true

                         Account phone number

   owner                 string
                         title: Owner
                         readOnly: true
                         minLength: 1
   followers
                             [
                         readOnly: true
                         uniqueItems: true
                         integer]
   likes                 integer
                         title: Likes
                         readOnly: true

}
```

PUT `/restaurants/api/restaurant/edit/` `restaurants_api_restaurant_edit_update`

Update the details for a given restaurant
User must own a restaurant

Parameters Try it out

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

Name **Description**

```
Restaurant  {
   name*              string
                      title: Name
                      maxLength: 200
                      minLength: 1
   address*           string
                      title: Address
                      maxLength: 200
                      minLength: 1
   postal_code*       string
                      title: Postal code
                      maxLength: 7
                      minLength: 1
   phone_number       string
                      title: Phone number
                      maxLength: 31
                      x-nullable: true

                      Account phone number


}
```

Responses

Response content type **application/json**

Code **Description**

200

Example Value **Model**

```
Restaurant  {
   name*              string
                      title: Name
                      maxLength: 200
                      minLength: 1
   address*           string
                      title: Address
                      maxLength: 200
                      minLength: 1
   logo               string($uri)
                      title: Logo
                      readOnly: true
                      x-nullable: true
   postal_code*       string
                      title: Postal code
                      maxLength: 7
                      minLength: 1
   phone_number       string
                      title: Phone number
                      maxLength: 31
                      x-nullable: true

                      Account phone number

   owner              string
                      title: Owner
                      readOnly: true
                      minLength: 1
   followers
                         [
                      readOnly: true
                      uniqueItems: true
                      integer]
   likes              integer
                      title: Likes
                      readOnly: true

}
```

PATCH `/restaurants/api/restaurant/edit/` `restaurants_api_restaurant_edit_partial_update` 🔓

Update the details for a given restaurant
User must own a restaurant

Parameters Try it out

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

```
Restaurant   {
    name*               string
                        title: Name
                        maxLength: 200
                        minLength: 1
    address*            string
                        title: Address
                        maxLength: 200
                        minLength: 1
    postal_code*        string
                        title: Postal code
                        maxLength: 7
                        minLength: 1
    phone_number        string
                        title: Phone number
                        maxLength: 31
                        x-nullable: true

                        Account phone number

}
```

Responses Response content type application/json

Code	Description
200	Example Value **Model**

Code	Description

```
Restaurant    {
    name*                 string
                          title: Name
                          maxLength: 200
                          minLength: 1
    address*              string
                          title: Address
                          maxLength: 200
                          minLength: 1
    logo                  string($uri)
                          title: Logo
                          readOnly: true
                          x-nullable: true
    postal_code*          string
                          title: Postal code
                          maxLength: 7
                          minLength: 1
    phone_number          string
                          title: Phone number
                          maxLength: 31
                          x-nullable: true

                          Account phone number

    owner                 string
                          title: Owner
                          readOnly: true
                          minLength: 1
    followers
                              [
                          readOnly: true
                          uniqueItems: true
                          integer]
    likes                 integer
                          title: Likes
                          readOnly: true

}
```

POST `/restaurants/api/restaurant/item/add/` `restaurants_api_restaurant_item_add_create`

Creates a new food item and adds it to the restaurant of the user
The user must own a restaurant

Parameters

Try it out

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

Name **Description**

```
FoodItem    {
   name*                string
                        title: Name
                        maxLength: 200
                        minLength: 1

                        Name of the menu item

   description*         string
                        title: Description
                        maxLength: 200
                        minLength: 1

                        Description of the menu item

   price*               number
                        title: <django.core.validators.MinValueValidator object at
                        0x00000219448533A0>

                        Price of the menu item

}
```

Responses **Response content type** application/json

Code	Description
201	Example Value **Model**

```
FoodItem    {
   name*                string
                        title: Name
                        maxLength: 200
                        minLength: 1

                        Name of the menu item

   description*         string
                        title: Description
                        maxLength: 200
                        minLength: 1

                        Description of the menu item

   price*               number
                        title: <django.core.validators.MinValueValidator object at
                        0x00000219448533A0>

                        Price of the menu item

}
```

PUT `/restaurants/api/restaurant/item/{id}/edit/` Update the details for a given food item User must own the restaurant that the food item belongs in the menu of `restaurants_api_restaurant_item_edit_update`

- Path Param - **id**: id of the food item to update

Parameters

Try it out

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

```
FoodItem    {
   name*                 string
                         title: Name
                         maxLength: 200
                         minLength: 1

                         Name of the menu item

   description*          string
                         title: Description
                         maxLength: 200
                         minLength: 1

                         Description of the menu item

   price*                number
                         title: <django.core.validators.MinValueValidator object at
                         0x00000219448533A0>

                         Price of the menu item

}
```

Name	Description
id * required string *(path)*	id

Responses

Response content type application/json

Code	Description
200	Example Value **Model**

```
FoodItem    {
   name*                 string
                         title: Name
                         maxLength: 200
                         minLength: 1

                         Name of the menu item

   description*          string
                         title: Description
                         maxLength: 200
                         minLength: 1

                         Description of the menu item

   price*                number
                         title: <django.core.validators.MinValueValidator object at
                         0x00000219448533A0>

                         Price of the menu item

}
```

PATCH `/restaurants/api/restaurant/item/{id}/edit/` Update the details for a given food item User must own the restaurant that the food item belongs in the menu of `restaurants_api_restaurant_item_edit_partial_update`

- Path Param - **id**: id of the food item to update

Parameters

Try it out

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

```
FoodItem    {
   name*                string
                        title: Name
                        maxLength: 200
                        minLength: 1

                        Name of the menu item

   description*         string
                        title: Description
                        maxLength: 200
                        minLength: 1

                        Description of the menu item

   price*               number
                        title: <django.core.validators.MinValueValidator object at
                        0x00000219448533A0>

                        Price of the menu item

}
```

Name	Description
id * required string *(path)*	id

Responses

Response content type application/json

Code	Description
200	Example Value **Model**

Code	Description

```
FoodItem    {
   name*                 string
                         title: Name
                         maxLength: 200
                         minLength: 1

                         Name of the menu item

   description*          string
                         title: Description
                         maxLength: 200
                         minLength: 1

                         Description of the menu item

   price*                number
                         title: <django.core.validators.MinValueValidator object at
                         0x00000219448533A0>

                         Price of the menu item

}
```

POST `/restaurants/api/restaurant/photo/add/` `restaurants_api_restaurant_photo_add_create`

Adds a photo to the gallery of the users restaurant
User must own a restaurant

Parameters

Try it out

Name	Description

data * required
object
(body)
	Example Value **Model**
	Photo {
	}

Responses

Response content type application/json

Code	Description
201	Example Value **Model**

Code **Description**

```
Photo    {
    restaurant            string
                          title: Restaurant
                          readOnly: true
                          minLength: 1
    image                 string($uri)
                          title: Image
                          readOnly: true

}
```

DELETE `/restaurants/api/restaurant/photo/{id}/delete/` `restaurants_api_restaurant_photo_delete_delete`

"

Removes a single photo from a restaurants gallery
The user must own the restaurant that the photo belongs to, to delete it

- Path Param - **id**: id of the photo to delete

Parameters Try it out

Name	Description
id * required string *(path)*	id

Responses Response content type application/json

Code	Description
204	

POST `/restaurants/api/restaurant/post/` `restaurants_api_restaurant_post_create`

Creates a post for the restaurant
The user must not already have a restaurant

Parameters Try it out

Name **Description**

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

```
Post   {
   body*                string
                        title: Body
                        minLength: 1
   likes                integer
                        title: Likes

}
```

Responses Response content type application/json

Code	Description
201	Example Value **Model**

```
Post   {
   timestamp            string($date-time)
                        title: Timestamp
                        readOnly: true
   body*                string
                        title: Body
                        minLength: 1
   likes                integer
                        title: Likes
   user                 string
                        title: User
                        readOnly: true
                        minLength: 1
   restaurant           string
                        title: Restaurant
                        readOnly: true
                        minLength: 1

}
```

GET `/restaurants/api/restaurant/post/{id}/` Returns all posts for a given restaurant `restaurants_api_restaurant_post_read`

- Path Param - **id**: id of the restaurant to get posts from

Parameters Try it out

Name	Description
id * required string *(path)*	the id of the restaurant that the posts belong to id - the id of the restaurant that the posts belc

Name	Description
page integer *(query)*	A page number within the paginated result set. page - A page number within the paginated re
page_size integer *(query)*	Number of results to return per page. page_size - Number of results to return per p

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*                 [Post    {
                         timestamp          string($date-time)
                                            title: Timestamp
                                            readOnly: true
                         body*              string
                                            title: Body
                                            minLength: 1
                         likes              integer
                                            title: Likes
                         user               string
                                            title: User
                                            readOnly: true
                                            minLength: 1
                         restaurant         string
                                            title: Restaurant
                                            readOnly: true
                                            minLength: 1

                      }]

}
```

PUT `/restaurants/api/restaurant/post/{id}/like/` `restaurants_api_restaurant_post_like_update`

At the click of a like button, increment the number of likes by 1 for a post and create a LikedPost object that tracks the user information.

- Path Param - **id**: id of the post to like

Parameters Try it out

Name	Description

data * required
object
(body)
	Example Value **Model**
	LikedPost {
	likes integer *title: Likes*
	}

id * required string *(path)*	id

Responses Response content type **application/json**

Code	Description

200	Example Value **Model**
	LikedPost {
	likes integer *title: Likes*
	}

PATCH `/restaurants/api/restaurant/post/{id}/like/` `restaurants_api_restaurant_post_like_partial_update`

At the click of a like button, increment the number of likes by 1 for a post and create a LikedPost object that tracks the user information.

- Path Param - **id**: id of the post to like

Parameters Try it out

Name	Description

data * required
`object`
(body)
	Example Value **Model**
	LikedPost {
	likes integer
	title: Likes
	}

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

Responses Response content type **application/json**

Code	Description

200	Example Value **Model**
	LikedPost {
	likes integer
	title: Likes
	}

PUT `/restaurants/api/restaurant/post/{id}/unlike/` `restaurants_api_restaurant_post_unlike_update`

At the click of a like button, decrement the number of likes by 1 for a post and remove the user from a LikedPost object that tracks the user information.

Assume that a user cannot unlike a post without previously liking it.

- Path Param - **id**: id of the post to like

Parameters Try it out

Name	Description

data * required
`object`
(body)
	Example Value **Model**
	UnlikedPost {
	likes integer
	title: Likes
	}

Name	Description
id * required `string` *(path)*	id

Responses Response content type application/json

Code	Description
200	Example Value **Model**

```
UnlikedPost    {
   likes              integer
                      title: Likes

}
```

PATCH `/restaurants/api/restaurant/post/{id}/unlike/` `restaurants_api_restaurant_post_unlike_partial_update`

At the click of a like button, decrement the number of likes by 1 for a post and remove the user from a LikedPost object that tracks the user information.

Assume that a user cannot unlike a post without previously liking it.

- Path Param - **id**: id of the post to like

Parameters Try it out

Name	Description
data * required `object` *(body)*	Example Value **Model**

```
UnlikedPost    {
   likes              integer
                      title: Likes

}
```

Name	Description
id * required `string` *(path)*	id

Responses Response content type application/json

Code	Description
200	Example Value **Model**

```
UnlikedPost    {
    likes              integer
                       title: Likes

}
```

GET /restaurants/api/restaurant/search/ restaurants_api_restaurant_search_list

Queries all restaurants by a string and returns results sorted by popularity(likes)
The string can contain multiple terms (space delimited). The search will filter restaurants
that contain the terms be either name, address, or an item in their menu

- Query Param - **query**: a string of one or more space delimited search terms

Parameters Try it out

Name	Description
query * required string *(query)*	one or more space delimited search terms query - one or more space delimited search t
page integer *(query)*	A page number within the paginated result set. page - A page number within the paginated re
page_size integer *(query)*	Number of results to return per page. page_size - Number of results to return per p

Responses **Response content type** application/json

Code	Description
200	Example Value **Model**

Code **Description**

```
    {
    count*                 integer
    next                   string($uri)
                           x-nullable: true
    previous               string($uri)
                           x-nullable: true
    results*                 [Restaurant    {
                             name*                  string
                                                    title: Name
                                                    maxLength: 200
                                                    minLength: 1
                             address*               string
                                                    title: Address
                                                    maxLength: 200
                                                    minLength: 1
                             logo                   string($uri)
                                                    title: Logo
                                                    readOnly: true
                                                    x-nullable: true
                             postal_code*           string
                                                    title: Postal code
                                                    maxLength: 7
                                                    minLength: 1
                             phone_number           string
                                                    title: Phone number
                                                    maxLength: 31
                                                    x-nullable: true

                                                    Account phone number

                             owner                  string
                                                    title: Owner
                                                    readOnly: true
                                                    minLength: 1
                             followers
                                                        [...]
                             likes                  integer
                                                    title: Likes
                                                    readOnly: true

                           }]

}
```

GET `/restaurants/api/restaurant/{id}/details/` Returns all details for a given restaurant `restaurants_api_restaurant_details_read` 🔒

- Path Param - **id**: id of the restaurant to get details from

Parameters Try it out

Name	Description
id * required `string` *(path)*	id

Responses Response content type application/json

Code **Description**

Code	Description
200	Example Value **Model**

```
Restaurant  {
    name*                    string
                             title: Name
                             maxLength: 200
                             minLength: 1
    address*                 string
                             title: Address
                             maxLength: 200
                             minLength: 1
    logo                     string($uri)
                             title: Logo
                             readOnly: true
                             x-nullable: true
    postal_code*             string
                             title: Postal code
                             maxLength: 7
                             minLength: 1
    phone_number             string
                             title: Phone number
                             maxLength: 31
                             x-nullable: true

                             Account phone number

    owner                    string
                             title: Owner
                             readOnly: true
                             minLength: 1
    followers
                                 [
                             readOnly: true
                             uniqueItems: true
                             integer]
    likes                    integer
                             title: Likes
                             readOnly: true

}
```

PUT `/restaurants/api/restaurant/{id}/like/` `restaurants_api_restaurant_like_update`

At the click of a like button, increment the number of likes by 1 for a restaurant and create a LikedRestaurant object that tracks the user information.

- Path Param - **id**: id of the restaurant to like

Parameters [Try it out]

Name	Description
data * required object *(body)*	Example Value **Model**
id * required string	id

```
LikedRestaurant  {
    likes                    integer
                             title: Likes

}
```

Name	Description
(path)	

Responses Response content type application/json

Code	Description
200	Example Value **Model**

```
LikedRestaurant    {
    likes              integer
                       title: Likes

}
```

PATCH `/restaurants/api/restaurant/{id}/like/` `restaurants_api_restaurant_like_partial_update`

At the click of a like button, increment the number of likes by 1 for a restaurant and create a LikedRestaurant object that tracks the user information.

- Path Param - **id**: id of the restaurant to like

Parameters Try it out

Name	Description
data * required `object` *(body)*	Example Value **Model** LikedRestaurant { likes integer title: Likes }
id * required `string` *(path)*	id

```
LikedRestaurant    {
    likes              integer
                       title: Likes

}
```

Responses Response content type application/json

Code	Description
200	Example Value **Model**

Code	Description

```
LikedRestaurant    {
   likes              integer
                      title: Likes

}
```

GET `/restaurants/api/restaurant/{id}/menu/` Returns menu for a given restaurant `restaurants_api_restaurant_menu_list`

- Path Param - **id**: id of the restaurant to get menu from

Parameters

Try it out

Name	Description
page `integer` *(query)*	A page number within the paginated result set. page - A page number within the paginated re
page_size `integer` *(query)*	Number of results to return per page. page_size - Number of results to return per p
id * required `string` *(path)*	id

Responses

Response content type application/json

Code	Description
200	Example Value **Model**

Code **Description**

```
{
count*                  integer
next                    string($uri)
                        x-nullable: true
previous                string($uri)
                        x-nullable: true
results*                    [FoodItem    {
                            name*                 string
                                                  title: Name
                                                  maxLength: 200
                                                  minLength: 1

                                                  Name of the menu item

                            description*          string
                                                  title: Description
                                                  maxLength: 200
                                                  minLength: 1

                                                  Description of the menu item

                            price*                number
                                                  title:
                                                  <django.core.validators.MinValueValidator
                                                  object at 0x00000219448533A0>
```

Models

```
Browsing    {
   name*                string
                        title: Name
                        maxLength: 200
                        minLength: 1
   address*             string
                        title: Address
                        maxLength: 200
                        minLength: 1
   postal_code*         string
                        title: Postal code
                        maxLength: 7
                        minLength: 1
   likes                integer
                        title: Likes

}
```

Parameters Try it out

```
Feed    {
   timestamp            string($date-time)
                        title: Timestamp
                        readOnly: true
   body*                string
                        title: Body
                        minLength: 1
   likes                integer
                        title: Likes
   user*                integer
                        title: User
   restaurant*          integer
                        title: Restaurant

}
```

page - A page number within the paginated re

```
Login    {
   username*            string
                        title: Username
                        minLength: 1
   password*            string
                        title: Password
                        minLength: 1

}
```

TokenRefresh

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

}
```

```
{
```

Notification

```
{
    source_id*            integer
                          title: Source id
    target_id*            integer
                          title: Target id
    timestamp             string($date-time)
                          title: Timestamp
                          readOnly: true

                          Time of the event occurrence

    body*                 string
                          title: Body
                          maxLength: 200
                          minLength: 1

                          Description of the event

    type*                 string
                          title: Type
                          Enum:

                              Array [ 5 ]

}
```

ProfileUpdate

```
{
    id                    string
                          title: Id
                          readOnly: true
                          minLength: 1
    first_name            string
                          title: First name
                          maxLength: 150
    last_name             string
                          title: Last name
                          maxLength: 150
    password              string
                          title: Password
                          minLength: 1
    password2             string
                          title: Password2
                          minLength: 1
    email                 string($email)
                          title: Email address
                          maxLength: 254
    phone_number          string
                          title: Phone number
                          maxLength: 31
                          x-nullable: true

                          Account phone number

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

}
```

Profile

```
Profile    {
    id                  integer
                        title: ID
                        readOnly: true
    first_name          string
                        title: First name
                        maxLength: 150
    last_name           string
                        title: Last name
                        maxLength: 150
    username*           string
                        title: Username
                        pattern: ^[\w.@+-]+$
                        maxLength: 150
                        minLength: 1

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

    email               string($email)
                        title: Email address
                        maxLength: 254
    phone_number        string
                        title: Phone number
                        maxLength: 31
                        x-nullable: true

                        Account phone number

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

                        Account profile icon

}
```

Parameters

Try it out

Register

```
Register    {
    first_name          string
                        title: First name
                        maxLength: 150
    last_name           string
                        title: Last name
                        maxLength: 150
    password*           string
                        title: Password
                        minLength: 1
    password2*          string
                        title: Password2
                        minLength: 1
    username*           string
                        title: Username
                        minLength: 1
    email*              string
                        title: Email
                        minLength: 1
    phone_number        string
                        title: Phone number
                        maxLength: 31
                        x-nullable: true

                        Account phone number

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

                        Account profile icon

}
```

Example Value Model

```
FollowedRestaurant  {
   followers                 [
                         uniqueItems: true
                         integer]

}
```

```
UnfollowedRestaurant  {
   followers                 [
                         uniqueItems: true
                         integer]

}
```

```
Comment  {
   timestamp             string($date-time)
                         title: Timestamp
                         readOnly: true
   body*                 string
                         title: Body
                         minLength: 1
   user                  integer
                         title: User
                         readOnly: true
   restaurant            integer
                         title: Restaurant
                         readOnly: true

}
```

```
Restaurant  {
   name*                 string
                         title: Name
                         maxLength: 200
                         minLength: 1
   address*              string
                         title: Address
                         maxLength: 200
                         minLength: 1
   logo                  string($uri)
                         title: Logo
                         readOnly: true
                         x-nullable: true
   postal_code*          string
                         title: Postal code
                         maxLength: 7
                         minLength: 1
   phone_number          string
                         title: Phone number
                         maxLength: 31
                         x-nullable: true

                         Account phone number

   owner                 string
                         title: Owner
                         readOnly: true
                         minLength: 1
   followers                 [
                         readOnly: true
                         uniqueItems: true
                         integer]
   likes                 integer
                         title: Likes
                         readOnly: true

}
```

```
FoodItem    {
   name*                string
                        title: Name
                        maxLength: 200
                        minLength: 1

                        Name of the menu item

   description*         string
                        title: Description
                        maxLength: 200
                        minLength: 1

                        Description of the menu item

   price*               number
                        title: <django.core.validators.MinValueValidator object at 0x00000219448533A0>

                        Price of the menu item


}
```

```
Photo    {
   restaurant           string
                        title: Restaurant
                        readOnly: true
                        minLength: 1
   image                string($uri)
                        title: Image
                        readOnly: true

}
```

```
Post    {
   timestamp            string($date-time)
                        title: Timestamp
                        readOnly: true
   body*                string
                        title: Body
                        minLength: 1
   likes                integer
                        title: Likes
   user                 string
                        title: User
                        readOnly: true
                        minLength: 1
   restaurant           string
                        title: Restaurant
                        readOnly: true
                        minLength: 1

}
```

```
LikedPost    {
   likes                integer
                        title: Likes

}
```

```
UnlikedPost    {
   likes                integer
                        title: Likes

}
```

```
LikedRestaurant    {
   likes                integer
                        title: Likes

}
```

```
Gallery   {
   restaurant*          integer
                        title: Restaurant
   image                string($uri)
                        title: Image
                        readOnly: true

                        Photo for gallery

   timestamp            string($date-time)
                        title: Timestamp
                        readOnly: true

}
```

```
UnlikedRestaurant    {
   likes                integer
                        title: Likes

}
```