

Group 0074 P2 Documentation

Design Choice

Apps:

- accounts
- restaurants
- socials
-

Anonymous users do not have the permission to edit/view user-specific information, we raise `AuthenticationFailed()` exception under this circumstance.

For pagination, the page number is set to 5.

NOTES:

Due to unknown reasons, required fields in `addRestaurantAvatar` and `addBlogImage`, `deleteRestaurantImage` do not show up in the auto-generated API documentation. However, those required fields are stated in their serializers, and we have properly handled the case where any of the required fields went missing.

Description For Apps

- accounts

This app contains all views and endpoints that are associated with user management and registration.

- Models

- User

- An extended user model to the `AbstractUser` class with additional attributes:

- birthday
 - postal code
 - avatar

- Serializers

- CreateUserSerializer

- Associated with registering a user account. The client can create a user with attributes in the user model. Only username, password, and password1 are required to create the account. Note that the client can not upload their avatar at this point yet.

- GetUserSerializer

- Returns every attribute of a user account, such as username, password, first_name, etc to the requesting client

- UpdateUserSerializer

- The user can change their information except for username and password. Note that this is the only serializer that allows the user to upload their avatar

- LoginSerializer

The serializer deserializes the username and password provided by the user and passed them into the view.
 - Views
 - CreateUserView
 - GetUserView
 - UpdateUserView
 - LoginView

This view will check the credential of the username and password inputted by the user. After validation, if the credentials are authenticated, the view will generate a token that is stored in the database and return the token to the user. If the credentials are not authenticated, the view will return a corresponding error message.
 - LogoutView

This view will delete the token provided by the user from the database if the token is valid. If not, the view will return an error message.
- Restaurant
 - Models
 - Restaurant

A restaurant model that has the following attributes

 - owner (A user model)
 - restaurant name
 - phone number
 - address
 - description
 - postal code
 - logo
 - Restaurant Image

An image object that is associated with a restaurant

 - image
 - restaurant
 - Food

A food object that has a name, price, and description. It must belong to a restaurant
 - Serializers
 - addMenu

Add a food object to this user's restaurant
 - addRestaurantAvatar

Upload a restaurant image
 - createRestaurant

Create a restaurant with the above information and automatically add the current as of the owner

- deleteRestaurantImage
Delete Restaurant Image
- editMenu
Edit a food object
- getMenu
Get a food object
- getRestaurant
Return all information of a specific restaurant
- getRestaurantAvatar
- search
Get all information of a restaurant that fitted a search string
- updateRestaurant
Update all information or partial information of the current user's restaurant
- Views
 - addMenu
If the current user has a restaurant, the view will create a food object that contains the information inputted by the user and associate the food object with his restaurant. If not, the view will reject the user's request, and return an error message
 - addRestaurantAvatar
Similar to addMenu, this view will check the state of the current user and react correspondingly.
 - createRestaurant
The view requires authentication. If the current user is authenticated, then the view will create a restaurant object with the current user as the owner and fill all the attributes with user inputs. Additionally, this view will check if the user has a restaurant or not. If so, it will return an error message without actually creating the restaurant
 - deleteRestaurantImage
This view deletes a restaurant image object with a specific image value. This view will only delete images that are associated with the current user's restaurant.
 - editMenu
Based on the current user and the food names the user inputted, the view will update the information about all the foods inputted. This view updates multiple objects. Example input for this view is:

```
{ {price : '999',
  description: 'fffff',
  food_name: 'testchange'
},
{(another food object)}}

```

- getMenu
Based on the restaurant name in the URL, the view will return all foods that are belonging to the restaurant in a paginated manner. The default number of food per page is 5 objects per page
- getRestaurant
Based on the restaurant name in the URL, the view will return all the information about the restaurant
- getRestaurantAvatar
Return all restaurant images based on the restaurant name in the URL in a paginated manner. The default number of images per page is 5.
- search
Return all the restaurants whose address or name or menu contains the word right after the “?search=” in the URL by using icontains. And the results are ordered by their followers' numbers (most popular restaurants first).
- updateRestaurant
This view will update the current user's restaurant by the information the user inputted.

- Socials

- Models

- Blog
A Blog model has the following attributes:
 - Restaurant (A restaurant model)
 - title
 - content
 - created_at (the time when the blog was created)
 - updated_at (the last modified time of the blog was edited)
 - BlogImage
A BlogImage model has the following attributes:
 - blog (a Blog model)
 - image (an image file, uploaded by user)
 - RestaurantLike
A RestaurantLike model has the following attributes:
 - user (a User model)
 - restaurant (a Restaurant model)
 - created_at (the time when the restaurant was liked)
 - BlogLike
A relationship entity that shows if a user liked a blog
 - user (a user model)
 - blog (A blog model)

- created_at (the time when the blog was liked)
- Follow

A relationship entity that shows if a user followed a restaurant

 - user (a user model)
 - restaurant
 - created_at (the time when the user followed this restaurant)
- Comment

An object that belongs to a restaurant written by a user

 - user (the user that writes this comment)
 - restaurant (A restaurant model, the comment belongs to this restaurant)
 - content
 - created_at (the time when the comment was created)
- Notification

A Notification model has the following attribute:

 - user (the user who takes action)
 - action (restricts to "Liked"/"Commented"/"Followed")
 - TargetUser (The owner of the blog or restaurant)
 - Target (restricts to "Blog"/"Restaurant")
 - blog_id (ONLY used when Target="Blog", IF it is not used, that will be set to -1 by default.)
 - restaurant_id (ONLY used when Target="Restaurant", IF it is not used, that will be set to -1 by default.)
 - created_at (the time when the notification happened)
- Serializers
 - addBlog

Create a blog
 - addBlogImage

Add an image to a blog
 - addCommentSerializer

Add a comment to restaurant
 - addNotification

Add a notification to a specific users about their blog/restaurant
 - deleteBlog

Delete a blog
 - feed

Get blogs from the following restaurants.
 - followRestaurantSerializer

Start following a restaurant.
 - getBlog

Get the blogs from a specific restaurant.
 - getBlogImage

Get the images from a specific blog.

- getNotification
Get the notifications of the current user.
- likeBlogSerializer
The current user likes the specific blog.
- likeRestaurantSerializer
The current user likes the specific restaurant.
- Views
 - addBlog
 - addBlogImage
 - addComment
Add comment follows the same structure as likeRestaurant and likeBlog, after the user inputs the restaurant/blog they like, a restaurantLike/blogLike object will be created and it returns the user id, restaurant/blog id, and the created time.
 - addNotification
The logged in user is able to like/comment/follow restaurants and like blogs by giving the specific action by the post method. Badrequest would be raised if the user tries to comment/follow the restaurants. addNotification views also require the user must enter the action and the target. And also, the blog_id/restaurant_id must be valid, otherwise, Badrequest would be raised.
 - deleteBlog
 - feed
Return every blog post from the restaurants this user followed, the blog posts are ordered by created time in descending order.
 - followRestaurant / unfollowRestaurant
UnfollowRestaurant has the same structure as unlikeRestaurant and unlikeBlog, after the user retracts the like/follow they have on a restaurant/blog, therestaurantLike, blogLike, restaurantFollow object will be removed and status code 204 will be returned.
 - getBlog
 - getBlogImage
 - getNotification
By filtering the TargetUser attribute in Notification model in the database and returning them to the user. The result is ordered by the data time. The newest notification first.
 - likeRestaurant / unlikeRestaurant

P2 API ^{v1}

[Base URL: localhost:8000/]
<http://localhost:8000/swagger/?format=openapi>

API Documentation for Group 0074 P2.

[Terms of service](#)

[Contact the developer](#)

BSD License

Schemes

HTTP

Django Login

Authorize



Filter by tag

accounts



POST

/accounts/api/token/

accounts_api_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
<div><div>TokenObtainPair</div><div><div>▼</div><div>{</div><div>username*</div><div>password*</div><div>}</div></div><div><div>string</div><div>title: Username</div><div>minLength: 1</div><div>string</div><div>title: Password</div><div>minLength: 1</div></div></div>	

POST

/accounts/login/

accounts_login_create

Parameters

Try it out

Name	Description
<div><div>data</div><div>★ required</div></div> <div>object</div> <div>(body)</div>	<div>Example Value Model</div> <div><div>Login</div><div><div>▼</div><div>{</div><div><div>username</div><div>*</div><div>string</div><div>title: Username</div><div>pattern: ^[\w.@+-]+\$</div><div>maxLength: 150</div><div>minLength: 1</div><div>Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.</div></div><div><div>password</div><div>*</div><div>string</div><div>title: Password</div><div>maxLength: 128</div><div>minLength: 1</div></div></div><div>}</div></div>

Responses

Response content type

application/json

▼

Code	Description
201	<div>Example Value Model</div> <div><div>Login</div><div><div>▼</div><div>{</div><div><div>username</div><div>*</div><div>string</div><div>title: Username</div><div>pattern: ^[\w.@+-]+\$</div><div>maxLength: 150</div><div>minLength: 1</div><div>Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.</div></div><div><div>password</div><div>*</div><div>string</div><div>title: Password</div><div>maxLength: 128</div><div>minLength: 1</div></div></div><div>}</div></div>

POST

/accounts/logout/

accounts_logout_create

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code

Description

201

POST

/accounts/user/register/

accounts_user_register_create

Parameters

Try it out

Name

Description

data

*

required

object

(body)

Example Value

Model

CreateUser

{

username

*

string

title: Username

pattern: ^[\w.@+-]+\$

maxLength: 150

minLength: 1

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

password

*

string

title: Password

maxLength: 128

minLength: 1

password1

*

string

title: Password1

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

birthday

string(\$date-time)

title: Birthday

x-nullable: true

phone_number

string

title: Phone number

maxLength: 11

x-nullable: true

}

Responses

Response content type

application/json

Code

Description

Code	Description
201	Example Value Model

CreateUser ▾ {

username*

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

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

password*

string
title: Password
maxLength: 128
minLength: 1

password1*

string
title: Password1
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

birthday

string(\$date-time)
title: Birthday
x-nullable: true

phone_number

string
title: Phone number
maxLength: 11
x-nullable: true

}

GET	/accounts/user/update/	accounts_user_update_read	
Parameters		<div>Try it out</div>	
No parameters			
Responses		Response content type	<div>application/json ▾</div>
Code	Description		

Code	Description
200	Example Value Model
<div><div>EditUser ▾ {</div><div><div>first_name</div><div>string</div><div>title: First name</div><div>maxLength: 150</div></div><div><div>last_name</div><div>string</div><div>title: Last name</div><div>maxLength: 150</div></div><div><div>avatar</div><div>string(\$uri)</div><div>title: Avatar</div><div>readOnly: true</div><div>x-nullable: true</div></div><div><div>email</div><div>string(\$email)</div><div>title: Email address</div><div>maxLength: 254</div></div><div><div>birthday</div><div>string(\$date-time)</div><div>title: Birthday</div><div>x-nullable: true</div></div><div><div>phone_number</div><div>string</div><div>title: Phone number</div><div>maxLength: 11</div><div>x-nullable: true</div></div></div> <div>}</div>	

PUT

/accounts/user/update/

accounts_user_update_update

Parameters

Try it out

Name

Description

data

*

required

object

(body)

Example Value

Model

EditUser

▼

{

first_name

string

title: First name

maxLength: 150

last_name

string

title: Last name

maxLength: 150

email

string(\$email)

title: Email address

maxLength: 254

birthday

string(\$date-time)

title: Birthday

x-nullable: true

phone_number

string

title: Phone number

maxLength: 11

x-nullable: true

}

Responses

Response content type

application/json

▼

Code

Description

Code	Description
200	Example Value Model
<pre> EditUser ▾ { first_name string title: First name maxLength: 150 last_name string title: Last name maxLength: 150 avatar string(\$uri) title: Avatar readOnly: true x-nullable: true email string(\$email) title: Email address maxLength: 254 birthday string(\$date-time) title: Birthday x-nullable: true phone_number string title: Phone number maxLength: 11 x-nullable: true }</pre>	

PATCH

/accounts/user/update/

accounts_user_update_partial_update

Parameters

Try it out

Name

Description

data

*

 required

object

(body)

Example Value

Model

EditUser

{

first_name

string

title: First name

maxLength: 150

last_name

string

title: Last name

maxLength: 150

email

string(\$email)

title: Email address

maxLength: 254

birthday

string(\$date-time)

title: Birthday

x-nullable: true

phone_number

string

title: Phone number

maxLength: 11

x-nullable: true

}

Responses

Response content type

application/json

Code

Description

Code	Description
200	Example Value Model
<pre> EditUser ▾ { first_name string title: First name maxLength: 150 last_name string title: Last name maxLength: 150 avatar string(\$uri) title: Avatar readOnly: true x-nullable: true email string(\$email) title: Email address maxLength: 254 birthday string(\$date-time) title: Birthday x-nullable: true phone_number string title: Phone number maxLength: 11 x-nullable: true } </pre>	

GET

/accounts/user/{username}/details/

accounts_user_details_read

🔒

Parameters

Try it out

Name	Description
<div>username * required</div> <div>string</div> <div>(path)</div>	<div>username</div>

Responses

Response content type application/json ▾

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

Code	Description
200	Example Value Model
<div><div>User ▾ {</div><div><div>username*</div><div>string</div><div>title: Username</div><div>pattern: ^[\w.@+-]+\$</div><div>maxLength: 150</div><div>minLength: 1</div><div>Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.</div></div><div><div>first_name</div><div>string</div><div>title: First name</div><div>maxLength: 150</div></div><div><div>last_name</div><div>string</div><div>title: Last name</div><div>maxLength: 150</div></div><div><div>avatar</div><div>string(\$uri)</div><div>title: Avatar</div><div>readOnly: true</div><div>x-nullable: true</div></div><div><div>email</div><div>string(\$email)</div><div>title: Email address</div><div>maxLength: 254</div></div><div><div>birthday</div><div>string(\$date-time)</div><div>title: Birthday</div><div>x-nullable: true</div></div></div> <div>}</div>	

restaurants

POST

/restaurants/add_image/

restaurants_add_image_create

Parameters

Try it out

Name

Description

data * required

object

(body)

Example Value

Model

AddRestaurantAvatar

{

}

Responses

Response content type

application/json

Code

Description

201

Example Value

Model

AddRestaurantAvatar

{

restaurant

string

title: Restaurant

readOnly: true

minLength: 1

image

string(\$uri)

title: Image

readOnly: true

}

POST

/restaurants/add_menu/

restaurants_add_menu_create



Parameters

Try it out

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

data * required**object**
(body)

Example Value | Model

```
AddMenu {
  food_name* string
    title: Food name
    maxLength: 100
    minLength: 1
  price* number
    title: Price
  description string
    title: Description
    x-nullable: true
}
```

Responses

Response content type

application/json



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

201

Example Value | Model

```
AddMenu {
  food_name* string
    title: Food name
    maxLength: 100
    minLength: 1
  price* number
    title: Price
  description string
    title: Description
    x-nullable: true
  restaurant string
    title: Restaurant
    readOnly: true
    minLength: 1
}
```

POST

/restaurants/create/


restaurants_create_create






Parameters

Try it out

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

Name	Description
data * required	
object <i>(body)</i>	Example Value Model
	<div>CreateRestaurant </div> <pre>restaurant_name* string title: Restaurant name maxLength: 150 minLength: 1 address* string title: Address maxLength: 250 minLength: 1 description* string title: Description minLength: 1 postal_code string title: Postal code maxLength: 6 x-nullable: true phone_number string title: Phone number maxLength: 20 x-nullable: true }</pre>

Responses	Response content type
	<div>application/json </div>
Code	Description
201	Example Value Model
	<div>CreateRestaurant </div> <pre>restaurant_name* string title: Restaurant name maxLength: 150 minLength: 1 logo string(\$uri) title: Logo readOnly: true x-nullable: true address* string title: Address maxLength: 250 minLength: 1 owner string title: Owner readOnly: true minLength: 1 description* string title: Description minLength: 1 postal_code string title: Postal code maxLength: 6 x-nullable: true phone_number string title: Phone number maxLength: 20 x-nullable: true }</pre>

<div>DELETE</div>	/restaurants/delete_avatar/	restaurants_delete_avatar_delete 
Parameters		<div>Try it out</div>
No parameters		

Responses

Response content type

application/json

Code	Description
204	

PUT

/restaurants/edit_menu/

restaurants_edit_menu_update

Parameters

Try it out

Name	Description
<div><div>data * required</div><div>object</div><div>(body)</div></div>	<div>Example Value Model</div> <div><div>EditMenu <div></div> {</div><div>food_name*<div>string</div><div>title: Food name</div><div>maxLength: 100</div><div>minLength: 1</div></div><div>price*<div>number</div><div>title: Price</div></div><div>description<div>string</div><div>title: Description</div><div>x-nullable: true</div></div><div>restaurant*<div>integer</div><div>title: Restaurant</div></div></div> <div>}</div>

Responses

Response content type

application/json

PATCH

/restaurants/edit_menu/

restaurants_edit_menu_partial_update

Parameters

Try it out

Name	Description
data <small>* required</small>	
object <small>(body)</small>	Example Value Model
	<div>EditMenu ▾ { food_name* <small>string</small> title: Food name maxLength: 100 minLength: 1 price* <small>number</small> title: Price description <small>string</small> title: Description x-nullable: true restaurant* <small>integer</small> title: Restaurant }</div>

Responses

Response content type

application/json

Code

Description

200

Example Value | Model

EditMenu {

id

integer

title: ID

readOnly: true

food_name*

string

title: Food name

maxLength: 100

minLength: 1

price*

number

title: Price

description

string

title: Description

x-nullable: true

restaurant*

integer

title: Restaurant

}

GET

/restaurants/get/{restaurant_name}/

restaurants_get_read

Parameters

Try it out

Name

Description

restaurant_name

* required

string

(path)

restaurant_name

Responses

Response content type

application/json

Code

Description

Code	Description
200	Example Value Model
<div><div>GetRestaurant ▾ {</div><div><div>restaurant_name*</div><div>string</div><div>title: Restaurant name</div><div>maxLength: 150</div><div>minLength: 1</div></div><div><div>address*</div><div>string</div><div>title: Address</div><div>maxLength: 250</div><div>minLength: 1</div></div><div><div>owner*</div><div>string</div><div>title: Owner</div><div>minLength: 1</div></div><div><div>description*</div><div>string</div><div>title: Description</div><div>minLength: 1</div></div><div><div>postal_code</div><div>string</div><div>title: Postal code</div><div>maxLength: 6</div><div>x-nullable: true</div></div></div> <div>}</div>	

GET /restaurants/get_image/{restaurant_name}/		restaurants_get_image_read
Parameters		Try it out
Name	Description	
page integer (query)	A page number within the paginated result set.	
	page - A page number within the paginated result set.	
page_size integer (query)	Number of results to return per page.	
	page_size - Number of results to return per page.	
restaurant_name * required string (path)	restaurant_name	
Responses		Response content type application/json ▾
Code	Description	

Code	Description
200	Example Value Model
<pre>▼ { count* integer next string(\$uri) x-nullable: true previous string(\$uri) x-nullable: true results* ▼ [GetRestaurantImage ▼ { restaurant string title: Restaurant readOnly: true minLength: 1 image string(\$uri) title: Image readOnly: true }] }</pre>	

GET

/restaurants/get_menu/{restaurant_name}/

restaurants_get_menu_read

Parameters

Try it out

Name	Description
page integer (query)	A page number within the paginated result set. <div>page - A page number within the paginated result set</div>
page_size integer (query)	Number of results to return per page. <div>page_size - Number of results to return per page</div>
restaurant_name * required string (path)	restaurant_name

Responses

Response content type application/json

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

Code	Description
200	Example Value Model
<pre> { count* integer next string(\$uri) x-nullable: true previous string(\$uri) x-nullable: true results* { GetMenu { food_name* string title: Food name maxLength: 100 minLength: 1 price* number title: Price restaurant string title: Restaurant readOnly: true minLength: 1 description string title: Description x-nullable: true } } }</pre>	

GET

/restaurants/search/

restaurants_search_list

Parameters

Try it out

Name	Description
page integer (query)	A page number within the paginated result set. <div>page - A page number within the paginated result set</div>
page_size integer (query)	Number of results to return per page. <div>page_size - Number of results to return per page</div>

Responses

Response content type application/json

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

Code	Description
200	Example Value Model
<pre>▼ { count* next previous results* ▼ {Search ▼ { id owner* restaurant_name* phone_number address* description* postal_code logo } } }</pre>	<pre>integer string(\$uri) x-nullable: true string(\$uri) x-nullable: true ▼ {Search ▼ { id title: ID readOnly: true owner* title: Owner minLength: 1 restaurant_name* title: Restaurant name maxLength: 150 minLength: 1 phone_number title: Phone number maxLength: 20 x-nullable: true address* title: Address maxLength: 250 minLength: 1 description* title: Description minLength: 1 postal_code title: Postal code maxLength: 6 x-nullable: true logo string(\$uri) title: Logo readOnly: true x-nullable: true }} }</pre>

GET

/restaurants/update/

restaurants_update_read

🔒

Parameters

Try it out

No parameters

Responses

Response content type

application/json ▼

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

Code	Description
200	Example Value Model
<pre>UpdateRestaurant { restaurant_name* string title: Restaurant name maxLength: 150 minLength: 1 address* string title: Address maxLength: 250 minLength: 1 description* string title: Description minLength: 1 postal_code string title: Postal code maxLength: 6 x-nullable: true phone_number string title: Phone number maxLength: 20 x-nullable: true }</pre>	

PUT

/restaurants/update/

restaurants_update_update

Parameters

Try it out

Name	Description
<div><div>data</div><div><div>*</div> required</div></div>	
<div><div>object</div><div>(body)</div></div>	Example Value Model
<div><div>UpdateRestaurant</div><div><div><div>restaurant_name*</div><div>string</div><div>title: Restaurant name</div><div>maxLength: 150</div><div>minLength: 1</div></div><div><div>address*</div><div>string</div><div>title: Address</div><div>maxLength: 250</div><div>minLength: 1</div></div><div><div>description*</div><div>string</div><div>title: Description</div><div>minLength: 1</div></div><div><div>postal_code</div><div>string</div><div>title: Postal code</div><div>maxLength: 6</div><div>x-nullable: true</div></div><div><div>phone_number</div><div>string</div><div>title: Phone number</div><div>maxLength: 20</div><div>x-nullable: true</div></div></div></div>	

 |Responses

Response content type

application/json

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

Code	Description
200	Example Value Model
<div><div>UpdateRestaurant</div><div><div><div>restaurant_name*</div><div>string</div><div>title: Restaurant name</div><div>maxLength: 150</div><div>minLength: 1</div></div><div><div>address*</div><div>string</div><div>title: Address</div><div>maxLength: 250</div><div>minLength: 1</div></div><div><div>description*</div><div>string</div><div>title: Description</div><div>minLength: 1</div></div><div><div>postal_code</div><div>string</div><div>title: Postal code</div><div>maxLength: 6</div><div>x-nullable: true</div></div><div><div>phone_number</div><div>string</div><div>title: Phone number</div><div>maxLength: 20</div><div>x-nullable: true</div></div></div></div>	

PATCH

/restaurants/update/

restaurants_update_partial_update

Parameters

Try it out

Name	Description
<div><div>data</div><div><div>*</div> required</div></div>	
<div><div>object</div><div>(body)</div></div>	Example Value Model
<div><div>UpdateRestaurant</div><div><div><div>restaurant_name*</div><div>string</div><div>title: Restaurant name</div><div>maxLength: 150</div><div>minLength: 1</div></div><div><div>address*</div><div>string</div><div>title: Address</div><div>maxLength: 250</div><div>minLength: 1</div></div><div><div>description*</div><div>string</div><div>title: Description</div><div>minLength: 1</div></div><div><div>postal_code</div><div>string</div><div>title: Postal code</div><div>maxLength: 6</div><div>x-nullable: true</div></div><div><div>phone_number</div><div>string</div><div>title: Phone number</div><div>maxLength: 20</div><div>x-nullable: true</div></div></div></div>	

Responses

Response content type

application/json

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

Code	Description
200	Example Value Model
<div><div>UpdateRestaurant</div><div><div><div>restaurant_name*</div><div>string</div><div>title: Restaurant name</div><div>maxLength: 150</div><div>minLength: 1</div></div><div><div>address*</div><div>string</div><div>title: Address</div><div>maxLength: 250</div><div>minLength: 1</div></div><div><div>description*</div><div>string</div><div>title: Description</div><div>minLength: 1</div></div><div><div>postal_code</div><div>string</div><div>title: Postal code</div><div>maxLength: 6</div><div>x-nullable: true</div></div><div><div>phone_number</div><div>string</div><div>title: Phone number</div><div>maxLength: 20</div><div>x-nullable: true</div></div></div></div>	

socials

POST

/socials/add_blog/

socials_add_blog_create

Parameters

Try it out

Name

Description

data

*

 required

object

Example Value | Model

(body)

AddBlog

title*

string

title: Title

maxLength: 100

minLength: 1

content*

string

title: Content

minLength: 1

Responses

Response content type

application/json

Code

Description

Code	Description
201	Example Value Model
<div><div>AddBlog ▾ {</div><div><div>title*</div><div>string</div><div>title: Title</div><div>maxLength: 100</div><div>minLength: 1</div></div><div><div>content*</div><div>string</div><div>title: Content</div><div>minLength: 1</div></div></div> <div>}</div>	

POST

/socials/add_blog_image/

socials_add_blog_image_create

Parameters

Try it out

Name

Description

data

*

 required

object

(body)

Example Value

Model

AddBlogImage

{

blog

*

integer

title: Blog

}

Responses

Response content type

application/json

Code

Description

201

Example Value

Model

AddBlogImage

{

blog

*

integer

title: Blog

image

string(\$uri)

title: Image

readOnly: true

x-nullable: true

}

POST

/socials/add_notification/{user_name}/

socials_add_notification_create

Parameters

Try it out

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

Name	Description
data * required object <i>(body)</i>	Example Value Model <div><div>AddNotification ▾ { action* string title: Action Enum: > Array [3] Target* string title: Target Enum: > Array [2] blog_id integer title: Blog id restaurant_id integer title: Restaurant id }</div></div>
user_name * required string <i>(path)</i>	<div>user_name</div>

Responses	Response content type
	<div>application/json ▾</div>
Code	Description
201	Example Value Model <div><div>AddNotification ▾ { id integer title: ID readOnly: true user string title: User readOnly: true minLength: 1 TargetUser string title: Targetuser readOnly: true minLength: 1 action* string title: Action Enum: > Array [3] Target* string title: Target Enum: > Array [2] blog_id integer title: Blog id restaurant_id integer title: Restaurant id created_at string(\$date-time) title: Created at readOnly: true }</div></div>

POST

/socials/comment/{restaurant_name}/

socials_comment_create

Parameters

Try it out

Name	Description
data * required object (body)	Example Value Model <div>AddComment ▼ { content * string title: Content maxLength: 255 minLength: 1 }</div>
restaurant_name * required string (path)	<input type="text" value="restaurant_name"/>

Responses

Response content type application/json ▼

Code	Description
201	Example Value Model <div>AddComment ▼ { id integer title: ID readOnly: true user string title: User readOnly: true minLength: 1 restaurant string title: Restaurant readOnly: true minLength: 1 content * string title: Content maxLength: 255 minLength: 1 }</div>

DELETE /socials/delete_blog/{blog_id}/		socials_delete_blog_delete	
Parameters		Try it out	
Name	Description		
blog_id * required string (path)	<input type="text" value="blog_id"/>		
Responses		Response content type application/json ▼	
Code	Description		
204			

GET

/socials/feed/

socials_feed_list

Parameters

Try it out

Name	Description
page integer (query)	A page number within the paginated result set. <div>page - A page number within the paginated result set</div>
page_size integer (query)	Number of results to return per page. <div>page_size - Number of results to return per page</div>

Responses

Response content type

application/json

Code	Description
200	<div>Example Value Model</div> <div><div><div>▼ {</div><div>count*integer</div><div>nextstring(\$uri) x-nullable: true</div><div>previousstring(\$uri) x-nullable: true</div><div>results*<div>▼ [Feed ▼ {<div>idinteger title: ID readOnly: true</div><div>restaurantstring title: Restaurant readOnly: true minLength: 1</div><div>title*string title: Title maxLength: 100 minLength: 1</div><div>content*string title: Content minLength: 1</div><div>created_atstring(\$date-time) title: Created at readOnly: true</div><div>updated_atstring(\$date-time) title: Updated at readOnly: true</div></div>}]</div></div></div>

POST

/socials/follow/{restaurant_name}/

socials_follow_create

Parameters

Try it out

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

Name	Description
data * required object <i>(body)</i>	Example Value Model <div>FollowRestaurant ▼ { user* integer title: User restaurant* integer title: Restaurant }</div>
restaurant_name * required string <i>(path)</i>	<div>restaurant_name</div>

Responses

Response content type

application/json

Code

Description

201

Example Value | Model

FollowRestaurant

user*

integer

title: User

restaurant*

integer

title: Restaurant

created_at

string(\$date-time)

title: Created at

readOnly: true

<div>GET /socials/get_blog/{restaurant_name}/</div>		socials_get_blog_read	
Parameters		<div>Try it out</div>	
Name	Description		
page integer <i>(query)</i>	A page number within the paginated result set.		
	<div>page - A page number within the paginated r</div>		
page_size integer <i>(query)</i>	Number of results to return per page.		
	<div>page_size - Number of results to return per p</div>		
restaurant_name * required string <i>(path)</i>	<div>restaurant_name</div>		
Responses		Response content type <div>application/json ▼</div>	

Code	Description
200	Example Value Model
<pre> { count* next previous results* { id restaurant title* content* created_at updated_at } } </pre>	

GET

/socials/get_blog_image/{blog_id}/

socials_get_blog_image_read

Parameters

Try it out

Name	Description
page integer (query)	A page number within the paginated result set. <div>page - A page number within the paginated result set</div>
page_size integer (query)	Number of results to return per page. <div>page_size - Number of results to return per page</div>
blog_id * required string (path)	blog_id

Responses

Response content type application/json

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

Code	Description
200	Example Value Model
<pre> { count* integer next string(\$uri) x-nullable: true previous string(\$uri) x-nullable: true results* { GetBlogImage { id integer title: ID readOnly: true restaurant string title: Restaurant readOnly: true minLength: 1 image string(\$uri) title: Image readOnly: true x-nullable: true blog* integer title: Blog } } }</pre>	

GET

/socials/get_notification/

socials_get_notification_list

Parameters

Try it out

Name	Description
page integer (query)	A page number within the paginated result set. <div>page - A page number within the paginated result set</div>
page_size integer (query)	Number of results to return per page. <div>page_size - Number of results to return per page</div>

Responses

Response content type

application/json

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

Code	Description
200	Example Value Model
<pre> { count* integer next string(\$uri) previous string(\$uri) results* {GetNotification} { id integer user string restaurant string TargetUser string action* string Target* {Array [3]} blog_id integer restaurant_id integer created_at string(\$date-time) } }</pre>	

POST

/socials/like_blog/{blog_id}/

socials_like_blog_create

Parameters

Try it out

Name	Description
data * required	
object	Example Value Model
(body)	<div>LikeBlog {</div>
blog_id * required	
string	blog_id
(path)	

Responses

Response content type application/json

Code	Description
201	Example Value Model <div><div>LikeBlog</div><div><div><div>id</div><div>integer</div><div>title: ID</div><div>readOnly: true</div></div><div><div>user</div><div>string</div><div>title: User</div><div>readOnly: true</div><div>minLength: 1</div></div><div><div>blog</div><div>string</div><div>title: Blog</div><div>readOnly: true</div><div>minLength: 1</div></div><div><div>created_at</div><div>string(\$date-time)</div><div>title: Created at</div><div>readOnly: true</div></div></div></div>

POST

/socials/like_restaurant/{restaurant_name}/

socials_like_restaurant_create

Parameters

Try it out

Name

Description

data

★ required

object

(body)

Example Value

Model

LikeRestaurant

▼ {

}

restaurant_name

★ required

string

(path)

restaurant_name

Responses

Response content type

application/json

▼

Code

Description

201

Example Value

Model

LikeRestaurant

▼ {

id

integer

title: ID

readOnly: true

user

string

title: User

readOnly: true

minLength: 1

restaurant

string

title: Restaurant

readOnly: true

minLength: 1

created_at

string(\$date-time)

title: Created at

readOnly: true

}

DELETE

/socials/unfollow/{restaurant_name}/

socials_unfollow_delete

Parameters

Try it out

Name	Description
restaurant_name * required string (path)	<input type="text" value="restaurant_name"/>

Responses

Response content type

application/json

Code	Description
204	

DELETE

/socials/unlike_blog/{blog_id}/

socials_unlike_blog_delete

Parameters

Try it out

Name	Description
blog_id * required string (path)	<input type="text" value="blog_id"/>

Responses

Response content type

application/json

Code	Description
204	

DELETE

/socials/unlike_restaurant/{restaurant_name}/

socials_unlike_restaurant_delete

Parameters

Try it out

Name	Description
restaurant_name * required string (path)	<input type="text" value="restaurant_name"/>

Responses

Response content type

application/json

Code	Description
204	

Models



TokenObtainPair >

Login >

CreateUser >

EditUser >

User >

AddRestaurantAvatar >

AddMenu >

CreateRestaurant >

EditMenu >

GetRestaurant >

GetRestaurantImage >

GetMenu >

Search >

UpdateRestaurant >

AddBlog >

AddBlogImage >

AddNotification >

AddComment >

Feed >

FollowRestaurant >

GetBlog >

GetBlogImage >

GetNotification >

LikeBlog >

LikeRestaurant >