The sign *XXXXX* means that this url can only be accessed with token provided

accounts:

# get token for user to login

Endpoint: /api/token/

Methods: POST

Fields/payload: username, password

Success: Give back the dictionary of "access token" and "refresh token"

# get the new access token and refresh token based on valid refresh token input.

Endpoint: /api/token/refresh/

Methods POST

Fields/payload: refresh,

Success: Give back the new dictionary of "access token" and "refresh token"

# create a new account

Endpoint: /accounts/create/

Methods: POST

Fields/payload: username, password1, password2, first_name, last_name, email, phone_num, avatar

Success: Give back the information of newly account just created.

# edit an existing account with user_id *XXXXX*

Endpoint: /accounts/edit/<int:user_id>/

Methods: POST

Fields/payload: username, password1, password2, first_name, last_name, email, phone_num, avatar

Success: Give back the information of newly account just created.

# Create new reservations:

Endpoint: accounts/reservation/create/<int:property_id>/

Method: POST

#host approve order with order_num *XXXXX*

Endpoint: accounts/reservation/approve/<int:order_num>/

Method: PUT, PATCH

#host/client cancel order order_num *XXXXX*

Endpoint: accounts/reservation/host-cancel/<int:order_num>/

Method: PUT, PATCH

Endpoint: accounts/reservation/client-cancel/<int:order_num>/

Method: PUT, PATCH

#host terminate order order_num *XXXXX*

Endpoint: accounts/'reservation/terminate/<int:order_num>/

Method: PUT, PATCH

#client denied order order_num *XXXXX*

Endpoint: accounts/'reservation/denied/<int:order_num>/

Method: PUT, PATCH

# list all reservation

Endpoint: accounts/reservation/list/

Methods: GET

# list reservation with a specific user type *host* or *client*

Endpoint :accounts/reservation/list/<str:type_user>/

Methods: GET

# list reservation with a specific user_type *host* or *client* and state 'pending', 'approve', 'cancel_request', 'complete', 'terminate' , 'denied', 'cancel', 'complete'

Endpoint: accounts/reservation/list/<str:type_user>/<str:state>/

Method: GET

property:

# create a new property *XXXXX*

Endpoint: property/create/

Method: POST

Payload: 'property_name', 'owner_username', 'city', 'country', 'address', 'num_of_guests', 'num_of_beds', 'property_type'

# edit an existing property with property_id *XXXXX*

Endpoint: property/edit/<int:property_id>/

Method: POST

Payload: 'property_name', 'owner_username', 'city', 'country', 'address', 'num_of_guests', 'num_of_beds', 'property_type'

# delete an existing property with property_id *XXXXX*

Endpoint: property/delete/<int:property_id>/

Method: DELETE

# check the detail of a property with property_id

Endpoint: property/detail/<int:property_id>/

Method: GET

# show the list of all properties

Endpoint: property/list/

Method: GET

# add an available time slot for the property with property_id *XXXXX*

Endpoint: property/<int:property_id>/avail/add/

Method: POST

Payload: 'start_date', 'end_date', 'price'

# edit an existing available time slot *XXXXX*

Endpoint: property/avail/<int:avail_id>/edit/

Method: POST

```
        Payload: 'start_date', 'end_date', 'price'


social:
# write a comment for a user with target_id *XXXXX*
        Endpoint: social/comment/user/<int:target_id>/create/
        Method: POST
        Payload: 'content', 'rate'
# view all comments for a user with target_id
        Endpoint: social/comment/user/<int:target_id>/view/
        Method: GET
# not required by the rubric, untested
         Endpoint: social/comment/user/<int:target_id>/delete/<int:comment_id>/
# write a comment for a property with target_id *XXXXX*
        Endpoint: social/comment/property/<int:target_id>/create/
        Method: POST
        Payload: 'content', 'rate'
# view all comments for a property with target_id
         social/comment/property/<int:target_id>/view/
# not required by the rubric, untested
         social/comment/property/<int:target_id>/delete/<int:comment_id>/
# write a reply to an existing comment with target_id *XXXXX*
         social/reply/<int:target_id>/create/
# view all the notifications for the current logged in user *XXXXX*
        Endpoint: social/notification/view/
        Method: GET
# view the detail information of a comment with comment_id
        Endpoint: social/comment/detail/<int:comment_id>/
        Method: GET
```