Comments and Replies Model (comments app)

Comment

Rating, author, text, created_at, (content_type, content_object, object_id)
Fields in () for associating a comment with an Application/Shelter.

Reply

Same as comment fields.
Extra field: comment (for recording the comment that this reply is made for)

Endpoints for Comments and Replies (comments app)

Details page for single application comment, add an application reply
Endpoint:/{user_id}/applications/{application_id}/comments/{comment_id}/
Methods: GET, POST
Success Response: (GET) Details of the specific application comment and all of its replies, including comment_id, text, user_id, created_at, rating. (POST) New reply saved to db and the list of replies refreshes.
Validation Errors:
  ● User not authorized
  ● User/Application/Comment not found
  ● Reply content is required
Additional Notes: Only accessible for logged-in user (shelter who received the application or pet seeker who submitted the application).

List of all application comments for single application, add an application comment
Endpoint: /{user_id}/applications/{application_id}/comments/
Methods: GET, POST
Success Response: (GET) List of all comments and their replies associated with an application, including comment_id, text, user_id, created_at, rating, sorted by creation time in descending order. (POST) New comment saved to db and the list of comments refreshes.
Validation Errors:
  ● User not authorized
  ● User/Application not found
  ● Comment content/rating is required
Additional Notes: Only accessible for logged-in user (shelter who received the application or pet seeker who submitted the application).

Details page for single shelter comment, add a shelter reply
Endpoint: /{user_id}/shelters/{shelter_id}/comments/{comment_id}/
Methods: GET, POST
Success Response: (GET) Details of the specific shelter comment and all of its replies, including comment_id, text, user_id, created_at, rating. (POST) New reply saved to db and the list of replies refreshes.
Validation Errors:
- User/Shelter/Comment not found
- Reply content is required
Additional Notes: Only accessible for logged-in user (shelter or pet seeker).

List of all shelter comments for single shelter, add a shelter comment
Endpoint: /{user_id}/shelters/{shelter_id}/comments
Methods: GET, POST
Success Response: (GET) List of all comments and their replies for a shelter, including comment_id, text, user_id, created_at, rating, sorted by creation time in descending order. This should look like a page full of shelter reviews (layout would be similar to youtube reviews under a video). (POST) New comment saved to db and the list of comments refreshes.
Validation Errors:
- User/Shelter not found
- Comment content/rating is required
Additional Notes: Only accessible for logged-in user (shelter or pet seeker).

Notifications Model (notifications app)

Notification

Recipient, created_at, is_read, (content_type, content_object, object_id)
Fields in () for associating a comment with an Application/Shelter.

Endpoints for notifications

List Notifications, Delete all notifications for a user
Endpoint: /{user_id}/notifications/
Method: GET. DELETE
Success Response: (GET) Retrieves a list of notifications for the logged-in user. Supports filtering by read/unread status and is sorted by creation time. (DELETE) Deletes all notifications for a user from the db.

Retrieve a Single Notification (auto-update from unread to read), Delete a notification
Endpoint: /{user_id}/notifications/{notification_id}/
Method: GET, PATCH, DELETE
Success Response: (GET) Retrieves the details of a specific notification. (PATCH) Accessing this endpoint can automatically mark the notification as read. (DELETE) Deletes the specified notification from the db.

Pet

Model:
- Pet
- Attributes: id, name, gallery, specie, breed, age, size, color, gender, location, health, characteristics, story, status, shelter

Endpoint: `/pet`:
- The api endpoint that handles the creation of a new pet object.
- method: `POST`
- Fields/Payload: `id, name, gallery, specie, breed, age, size, color, gender, location, health, characteristics, story, status, shelter`
- Response: Newly created pet object
- Validation Errors: GET not allowed, not authenticated, not shelter, missing field

Endpoint: `/pets/<pet_id>`:
- The api endpoint that handles the retrieve, update, and delete of a specific pet listing
- method: `GET PUT DELETE`
- Fields/Payload: `id, name, gallery, specie, breed, age, size, color, gender, location, health, characteristics, story, status, shelter`
- Response: Pet Detail if GET, Updated Pet Detail/403 if PUT, 204/403 if DELETE
- Validation Errors: not authenticated, not allowed(not a shelter, not owned by the shelter)
- Additional Note: Can edit not all the fields

Endpoint: `/pets`:
- The api endpoint that retrieves all the pets filtered as a list
- method: `GET`
- Fields/Payload: `id, name, gallery, specie, breed, age, size, color, gender, location, health, characteristics, story, status, shelter`
- Response: List of Pet with details
- Additional Note: uses ',' to separate values of a single filter, for example status=Available,Adopted

Accounts

Models

CustomUser

- Extends: AbstractBaseUser
- Attributes: email, password, avatar
- Username field (primary key): email
- Email and password are required.

Shelter

- Extends: CustomUser
- Attributes: organization, phone number, address, country, state, city, zip, mission statement
- All attributes are required.

Seeker

- Extends: CustomUser
- Attributes: first name, last name, phone number, address, country, state, city, zip
- All attributes are optional.

Preference

- Extends: Model
- Attributes: preference (choices), owner (foreign key: Seeker)
- Preference is required.
- One-to-many relationship with Seeker (a seeker can have many preferences or no preferences).

Endpoints

/api/token

For token-based authentication (login).

Methods:
- POST

- Payload: email, password
- Success response: Refresh token + access token.
- Errors:
  - This field is required.
  - No active account found with the given credentials

/api/token/refresher

For token-based authentication (login).

Methods:
- POST
  - Payload: access token
  - Success response: new access token

/shelters

For creating a shelter account (POST) or viewing a list of shelters (GET).

Methods:
- POST
  - Payload: email, password, organization, phone number, address, country, state, city, zip, mission statement
  - Success response: Newly created shelter object.
  - Errors:
    - This field is required.
    - Enter a valid email address.
    - Enter a valid phone number.
    - custom user with this email already exists.
- GET
  - Payload: none
  - Success response: List of all shelters.

/shelters/<int:pk>

For retrieving (GET) or updating (PUT, PATCH) a shelter account.

Methods:
- GET
  - Payload: none
  - Success response: Shelter details of requested shelter.
  - Errors:

- ■ Not found.
- ● PUT, PATCH
  - ○ Payload: email, password, organization, phone number, address, country, state, city, zip, mission statement
  - ○ Success response: Updated shelter details
  - ○ Errors:
    - ■ You do not have permission to perform this action.
    - ■ Enter a valid email address.
    - ■ Enter a valid phone number.
    - ■ custom user with this email already exists.

/shelters/<int:pk>/pets

For deleting all the pet listings of a shelter.

Methods:
- ● DELETE
  - ○ Payload: none
  - ○ Success response: Successfully deleted seeker's pet listings.
  - ○ Errors:
    - ■ You do not have permission to perform this action.

/seekers

For creating a seeker account.

Methods:
- ● POST
  - ○ Payload: email, password, first name, last name, phone number, address, country, state, city, zip
  - ○ Success response: Newly created seeker object.
  - ○ Errors:
    - ■ This field is required.
    - ■ Enter a valid email address.
    - ■ Enter a valid phone number.
    - ■ custom user with this email already exists.

/seekers/<int:pk>

For retrieving (GET) or updating (PUT, PATCH) a seeker account.

Methods:

- GET
  - Payload: none
  - Success response: Seeker details of requested seeker.
  - Errors:
    - You do not have permission to perform this action.
- PUT, PATCH
  - Payload: email, password, first name, last name, phone number, address, country, state, city, zip
  - Success response: Updated seeker details.
  - Errors:
    - You do not have permission to perform this action.
    - Enter a valid email address.
    - Enter a valid phone number.
    - custom user with this email already exists.

/seekers/<int:pk>/applications

For deleting all the applications of a seeker.

Methods:
- DELETE
  - Payload: none
  - Success response: Successfully deleted seeker's applications.
  - Errors:
    - You do not have permission to perform this action.

Applications

Models

Application

- Extends: models.Model
- Attributes: seeker (foreign key), shelter (foreign key), pet (foreign key), status (pending, accepted, denied, withdrawn), message, created_at, updated_at
- Ordering: -created_at, -updated_at

Endpoints

/pets/<int:pk>/applications

For creating an application.

Methods:
- POST
  - Payload: message (optional)
  - Success response: A newly created application by the requested user (seeker) for the pet with ID "pk".
  - Errors:
    - You do not have permission to perform this action.
    - Not found.
    - Duplicate application.

/applications/<int:pk>

For retrieving an application (GET) or updating an application (PUT, PATCH).

Methods:
- GET
  - Payload: none
  - Success response: Details of requested application.
  - Errors:
    - You do not have permission to perform this action.
    - Not found.
- PUT, PATCH
  - Payload: status

- ○ Success response: Application with updated status. Notification created for the shelter/seeker if the current user is seeker/shelter.
- ○ Errors:
  - ■ You do not have permission to perform this action.
  - ■ Not found.

/applications

For a shelter to retrieve a list of applications for their pets.

Methods:
- ● GET
  - ○ Payload: none
  - ○ Success response: A list of applications for the shelter making the request.
  - ○ Errors:
    - ■ You do not have permission to perform this action.