



# P2 PetPal

## Overview

Here is all the endpoints with a short description to what they do

## Version information

*Version* : v1

## URI scheme

*Host* : 127.0.0.1:8000

*BasePath* : /

*Schemes* : HTTP

## Consumes

- `application/json`

## Produces

- `application/json`

## Security

### Basic

*Type* : basic

## Paths

### POST /accounts/api/token/

#### Description

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

#### Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">TokenObtainPair</a>

## Responses

HTTP Code	Schema
201	<a href="#">TokenObtainPair</a>

## Tags

- accounts

# POST /accounts/api/token/refresh/

## Description

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

## Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">TokenRefresh</a>

## Responses

HTTP Code	Schema
201	<a href="#">TokenRefresh</a>

## Tags

- accounts

# POST /accounts/seekers/

## Description

Creates a seeker on POST Request, does not show a list of seekers (for any user type right now)

## Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">SeekerCreate</a>

## Responses

HTTP Code	Schema
201	<a href="#">SeekerCreate</a>

## Tags

- accounts

# GET /accounts/seekers/

## Description

Creates a seeker on POST Request, does not show a list of seekers (for any user type right now)

## Responses

HTTP Code	Schema
200	< <a href="#">SeekerCreate</a> > array

## Tags

- accounts

# GET /accounts/seekers/{id}/

## Description

Endpoint to Update a seeker object on PATCH Request, delete a seeker object on DELETE Request, and retrieve a seeker object on GET Request.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">SeekerUpdate</a>

## Tags

- accounts

## PUT /accounts/seekers/{id}/

### Description

Endpoint to Update a seeker object on PATCH Request, delete a seeker object on DELETE Request, and retrieve a seeker object on GET Request.

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">SeekerUpdate</a>

### Responses

HTTP Code	Schema
200	<a href="#">SeekerUpdate</a>

## Tags

- accounts

## DELETE /accounts/seekers/{id}/

### Description

Endpoint to Update a seeker object on PATCH Request, delete a seeker object on DELETE Request, and retrieve a seeker object on GET Request.

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
204	No Content

## Tags

- accounts

# PATCH /accounts/seekers/{id}/

## Description

Endpoint to Update a seeker object on PATCH Request, delete a seeker object on DELETE Request, and retrieve a seeker object on GET Request.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">SeekerUpdate</a>

## Responses

HTTP Code	Schema
200	<a href="#">SeekerUpdate</a>

## Tags

- accounts

# POST /accounts/shelters/

## Description

Creates a shelter on POST Request, lists all shelters on GET Request

## Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">ShelterCreate</a>

## Responses

HTTP Code	Schema
201	<a href="#">ShelterCreate</a>

## Tags

- accounts

# GET /accounts/shelters/

## Description

Creates a shelter on POST Request, lists all shelters on GET Request

## Responses

HTTP Code	Schema
200	< <a href="#">Shelter</a> > array

## Tags

- accounts

# GET /accounts/shelters/{id}/

## Description

Updates a shelter's profile on PATCH Request, deletes a shelter on DELETE Request, shows a shelter profile on GET Request

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">ShelterUpdate</a>

## Tags

- accounts

## PUT /accounts/shelters/{id}/

### Description

Updates a shelter's profile on PATCH Request, deletes a shelter on DELETE Request, shows a shelter profile on GET Request

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">ShelterUpdate</a>

### Responses

HTTP Code	Schema
200	<a href="#">ShelterUpdate</a>

## Tags

- accounts

## DELETE /accounts/shelters/{id}/

### Description

Updates a shelter's profile on PATCH Request, deletes a shelter on DELETE Request, shows a shelter profile on GET Request

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

### Responses



HTTP Code	Schema
204	No Content

## Tags

- accounts

## PATCH /accounts/shelters/{id}/

### Description

Updates a shelter's profile on PATCH Request, deletes a shelter on DELETE Request, shows a shelter profile on GET Request

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">ShelterUpdate</a>

### Responses

HTTP Code	Schema
200	<a href="#">ShelterUpdate</a>

## Tags

- accounts

## POST /api/token/

### Description

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

### Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">TokenObtainPair</a>

## Responses

HTTP Code	Schema
201	<a href="#">TokenObtainPair</a>

## Tags

- api

# POST /api/token/refresh/

## Description

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

## Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">TokenRefresh</a>

## Responses

HTTP Code	Schema
201	<a href="#">TokenRefresh</a>

## Tags

- api

# POST /comments/application/{application\_id}/

## Description

Creates an Application Comment on POST Request, given application\_id

## Parameters

Type	Name	Schema
Path	<b>application_id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">ApplicationComment</a>

## Responses

HTTP Code	Schema
201	<a href="#">ApplicationComment</a>

## Tags

- comments

# GET /comments/application/{application\_id}/

## Description

Creates an Application Comment on POST Request, given application\_id

## Parameters

Type	Name	Description	Schema
Path	<b>application_id</b> <i>required</i>		string
Query	<b>page</b> <i>optional</i>	A page number within the paginated result set.	integer

## Responses

HTTP Code	Schema
200	<a href="#">Response 200</a>

### Response 200

Name	Schema
<b>count</b> <i>required</i>	integer
<b>next</b> <i>optional</i>	string (uri)
<b>previous</b> <i>optional</i>	string (uri)
<b>results</b> <i>required</i>	< <a href="#">ApplicationComment</a> > array

## Tags

- comments

## POST /comments/review/{shelter\_id}/

### Description

This endpoint creates a shelter comment (or review) on POST Request given a shelter\_id

### Parameters

Type	Name	Schema
Path	<b>shelter_id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">Comment</a>

### Responses

HTTP Code	Schema
201	<a href="#">Comment</a>

### Tags

- comments

## GET /comments/review/{shelter\_id}/

### Description

This endpoint creates a shelter comment (or review) on POST Request given a shelter\_id

### Parameters

Type	Name	Description	Schema
Path	<b>shelter_id</b> <i>required</i>		string
Query	<b>page</b> <i>optional</i>	A page number within the paginated result set.	integer

### Responses

HTTP Code	Schema
200	<a href="#">Response 200</a>

### Response 200

Name	Schema
<b>count</b> <i>required</i>	integer
<b>next</b> <i>optional</i>	string (uri)
<b>previous</b> <i>optional</i>	string (uri)
<b>results</b> <i>required</i>	< <a href="#">Comment</a> > array

## Tags

- comments

# GET /comments/specific\_application/{comment\_id}/

## Description

This Endpoint shows a Comment from an application on a GET Request given comment\_id

## Parameters

Type	Name	Schema
<b>Path</b>	<b>comment_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">ApplicationComment</a>

## Tags

- comments

# GET /comments/specific\_review/{comment\_id}/

## Description

This Endpoint shows a Comment from a shelter on a GET Request given comment\_id

## Parameters

Type	Name	Schema
Path	<b>comment_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">Comment</a>

## Tags

- comments

# POST /notifications/application/{application\_id}/

## Description

This endpoint creates a notification for a submitted application when a POST request is sent

## Parameters

Type	Name	Schema
Path	<b>application_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
201	No Content

## Tags

- notifications

# POST /notifications/application\_comment/{comment\_id}/

## Description

This Endpoint creates a notification when a POST Request is sent given comment\_id signifying that a new application comment was created

## Parameters

Type	Name	Schema
Path	<b>comment_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
201	No Content

## Tags

- notifications

# POST /notifications/comment/{comment\_id}/

## Description

This endpoint creates a notification when a POST Request is sent, given a comment\_id

## Parameters

Type	Name	Schema
Path	<b>comment_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
201	No Content

## Tags

- notifications

# GET /notifications/notification/{id}/

## Description

This endpoint shows a notification on GET Request and when you make a request with GET, it marks the notification as read. It also deletes a notification on Delete Request

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	No Content

## Tags

- notifications

# DELETE /notifications/notification/{id}/

## Description

This endpoint shows a notification on GET Request and when you make a request with GET, it marks the notification as read. It also deletes a notification on Delete Request

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
204	No Content

## Tags

- notifications

# GET /notifications/notifications/

## Description

This endpoint lists all notifications on GET Request



## Parameters

Type	Name	Description	Schema
Query	<b>ordering</b> <i>optional</i>	Which field to use when ordering the results.	string
Query	<b>page</b> <i>optional</i>	A page number within the paginated result set.	integer

## Responses

HTTP Code	Schema
200	<a href="#">Response 200</a>

### Response 200

Name	Schema
<b>count</b> <i>required</i>	integer
<b>next</b> <i>optional</i>	string (uri)
<b>previous</b> <i>optional</i>	string (uri)
<b>results</b> <i>required</i>	< <a href="#">Notification</a> > array

## Tags

- notifications

# POST /notifications/pet/{pet\_id}/

## Description

This endpoint creates a notification for a submitted application when a POST request is sent

## Parameters

Type	Name	Schema
Path	<b>pet_id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
201	No Content

## Tags

- notifications

## Retrieves and updates an application

```
GET /pets/application/{id}/
```

## Description

retrieve (GET) gets the application (pk) for a specific pet for this user (usable by seeker or shelter involved in this application)

update (PUT/PATCH) can only update the status of an application.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">Application</a>

## Tags

- pets

## Retrieves and updates an application

```
PUT /pets/application/{id}/
```

## Description

retrieve (GET) gets the application (pk) for a specific pet for this user (usable by seeker or shelter involved in this application)

update (PUT/PATCH) can only update the status of an application.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">Application</a>

## Responses

HTTP Code	Schema
200	<a href="#">Application</a>

## Tags

- pets

# Retrieves and updates an application

```
PATCH /pets/application/{id}/
```

## Description

retrieve (GET) gets the application (pk) for a specific pet for this user (usable by seeker or shelter involved in this application)

update (PUT/PATCH) can only update the status of an application.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">Application</a>

## Responses

HTTP Code	Schema
200	<a href="#">Application</a>

## Tags

- pets

## List all applications for current user (shelter)

GET /pets/applications/

### Description

Sort can be done on "created\_date", "modified\_date" e.g. ordering=created\_date

Filter can be done on status e.g. status=accepted

Pagination of 5 pets per page, can be paged using page=1, page=2 ...

### Parameters

Type	Name	Description	Schema
Query	<b>page</b> <i>optional</i>	A page number within the paginated result set.	integer

### Responses

HTTP Code	Schema
200	<a href="#">Response 200</a>

#### Response 200

Name	Schema
<b>count</b> <i>required</i>	integer
<b>next</b> <i>optional</i>	string (uri)
<b>previous</b> <i>optional</i>	string (uri)
<b>results</b> <i>required</i>	< <a href="#">Application</a> > array

## Tags

- pets

# POST /pets/pet/

## Description

Takes information on a new pet to be listed as "available" under the current user who must be a shelter. Creates and returns the listing that was created for this pet.

## Parameters

Type	Name	Schema
Body	<b>data</b> <i>required</i>	<a href="#">Pet</a>

## Responses

HTTP Code	Schema
201	<a href="#">Pet</a>

## Tags

- pets

## Updates (PUT/PATCH), deletes (DELETE), and retrieves (GET) a pet listing.

```
GET /pets/pet/{id}/
```

## Description

Update takes in a new pet to overwrite the corresponding fields of pet that the current user (a shelter) owns.

Delete removes the pet listing, must be the owner of this pet (a shelter).

Get retrieves the listing and returns the pet's information.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

## Responses

HTTP Code	Schema
200	<a href="#">Pet</a>

## Tags

- [pets](#)

## Updates (PUT/PATCH), deletes (DELETE), and retrieves (GET) a pet listing.

```
PUT /pets/pet/{id}/
```

## Description

Update takes in a new pet to overwrite the corresponding fields of pet that the current user (a shelter) owns.

Delete removes the pet listing, must be the owner of this pet (a shelter).

Get retrieves the listing and returns the pet's information.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">Pet</a>

## Responses

HTTP Code	Schema
200	<a href="#">Pet</a>

## Tags

- [pets](#)

## Updates (PUT/PATCH), deletes (DELETE), and retrieves (GET) a pet listing.

```
DELETE /pets/pet/{id}/
```

### Description

Update takes in a new pet to overwrite the corresponding fields of pet that the current user (a shelter) owns.

Delete removes the pet listing, must be the owner of this pet (a shelter).

Get retrieves the listing and returns the pet's information.

### Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string

### Responses

HTTP Code	Schema
204	No Content

### Tags

- pets

## Updates (PUT/PATCH), deletes (DELETE), and retrieves (GET) a pet listing.

```
PATCH /pets/pet/{id}/
```

### Description

Update takes in a new pet to overwrite the corresponding fields of pet that the current user (a shelter) owns.

Delete removes the pet listing, must be the owner of this pet (a shelter).

Get retrieves the listing and returns the pet's information.

## Parameters

Type	Name	Schema
Path	<b>id</b> <i>required</i>	string
Body	<b>data</b> <i>required</i>	<a href="#">Pet</a>

## Responses

HTTP Code	Schema
200	<a href="#">Pet</a>

## Tags

- [pets](#)

## Search for all pets.

GET /pets/pets/

## Description

By default shows only "available" pets if a "status" filter is not provided.

Search parameter searches "name" and "animal" fields. e.g. search=charlie

Ordering can be done on "size", "listed", or "name" e.g. ordering=size

Filtering can be done on "shelter", "seeker", "breed", "size", "color", or "gender". e.g. color=red

Pagination of 10 pets per page, can be paged using page=1, page=2 ...

## Parameters

Type	Name	Description	Schema
Query	<b>ordering</b> <i>optional</i>	Which field to use when ordering the results.	string
Query	<b>page</b> <i>optional</i>	A page number within the paginated result set.	integer
Query	<b>search</b> <i>optional</i>	A search term.	string



## Responses

HTTP Code	Schema
200	<a href="#">Response 200</a>

### Response 200

Name	Schema
<b>count</b> <i>required</i>	integer
<b>next</b> <i>optional</i>	string (uri)
<b>previous</b> <i>optional</i>	string (uri)
<b>results</b> <i>required</i>	< <a href="#">Pet</a> > array

## Tags

- [pets](#)

## Creates an application for a given pet (pet\_id) (must be a seeker)

```
POST /pets/{pet_id}/applications/
```

## Description

Send preferred\_contact (email or phone) and description fields

## Parameters

Type	Name	Schema
<b>Path</b>	<b>pet_id</b> <i>required</i>	string
<b>Body</b>	<b>data</b> <i>required</i>	<a href="#">Application</a>

## Responses

HTTP Code	Schema
201	<a href="#">Application</a>

## Tags

- pets

# Definitions

## Application

Name	Description	Schema
<b>created_date</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>description</b> <i>required</i>	Length : 1 - 600	string
<b>id</b> <i>optional</i> <i>read-only</i>		integer
<b>modified_date</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>pet</b> <i>optional</i> <i>read-only</i>		integer
<b>preferred_contact</b> <i>required</i>		enum (email, phone)
<b>seeker</b> <i>optional</i> <i>read-only</i>		integer
<b>status</b> <i>optional</i>		enum (accepted, pending, denied, withdrawn)

## ApplicationComment

Name	Description	Schema
<b>application</b> <i>optional</i> <i>read-only</i>		integer
<b>content</b> <i>optional</i>	Minimum length : 1	string

Name	Description	Schema
<b>date</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>id</b> <i>optional</i> <i>read-only</i>		integer
<b>sender</b> <i>optional</i> <i>read-only</i>		integer

## Comment

Name	Description	Schema
<b>commenter_id</b> <i>optional</i> <i>read-only</i>		integer
<b>content</b> <i>optional</i>	<b>Minimum length : 1</b>	string
<b>date</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>id</b> <i>optional</i> <i>read-only</i>		integer
<b>rating</b> <i>optional</i>	<b>Minimum value : 1</b> <b>Maximum value : 5</b>	integer
<b>shelter_id</b> <i>optional</i> <i>read-only</i>		integer

## Notification

Name	Description	Schema
<b>content</b> <i>required</i>	<b>Length : 1 - 500</b>	string
<b>id</b> <i>optional</i> <i>read-only</i>		integer
<b>read_status</b> <i>optional</i>		boolean

Name	Description	Schema
<b>timestamp</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>type</b> <i>optional</i>		enum (comment, application, application_comment)
<b>user</b> <i>optional</i> <i>read-only</i>		integer

## Pet

Name	Description	Schema
<b>allow_notifs</b> <i>optional</i>		boolean
<b>animal</b> <i>required</i>	Length : 1 - 200	string
<b>birthday</b> <i>optional</i>		string (date)
<b>breed</b> <i>optional</i>	Maximal length : 200	string
<b>caretaker</b> <i>optional</i>	Maximal length : 200	string
<b>color</b> <i>required</i>	Length : 1 - 50	string
<b>description</b> <i>required</i>	Length : 1 - 600	string
<b>gender</b> <i>required</i>		enum (male, female, other)
<b>id</b> <i>optional</i> <i>read-only</i>		integer
<b>listed</b> <i>optional</i> <i>read-only</i>		string (date-time)
<b>name</b> <i>required</i>	Length : 1 - 200	string

Name	Description	Schema
<b>profile_image</b> <i>optional</i> <i>read-only</i>		string (uri)
<b>shelter</b> <i>optional</i> <i>read-only</i>		integer
<b>size</b> <i>required</i>		enum (small, medium, large)
<b>status</b> <i>optional</i>		enum (available, withdrawn, adopted)
<b>vaccine_statuses</b> <i>optional</i>	Maximal length : 200	string

## SeekerCreate

Name	Description	Schema
<b>address</b> <i>optional</i>	Maximal length : 200	string
<b>confirm_password</b> <i>required</i>	Minimum length : 1	string
<b>email</b> <i>required</i>	Length : 1 - 254	string (email)
<b>password</b> <i>required</i>	Minimum length : 1	string
<b>phone_number</b> <i>optional</i>	Maximal length : 10	string
<b>username</b> <i>required</i>	Minimum length : 1	string

## SeekerUpdate

Name	Description	Schema
<b>address</b> <i>optional</i>	Maximal length : 200	string
<b>age</b> <i>optional</i>		integer

Name	Description	Schema
<b>description</b> <i>optional</i>	Maximal length : 600	string
<b>email</b> <i>required</i>	Length : 1 - 254	string (email)
<b>first_name</b> <i>optional</i>	Maximal length : 200	string
<b>last_name</b> <i>optional</i>	Maximal length : 200	string
<b>phone_number</b> <i>optional</i>	Maximal length : 10	string
<b>profile_image</b> <i>optional</i> <i>read-only</i>		string (uri)

## Shelter

Name	Description	Schema
<b>address</b> <i>required</i>	Length : 1 - 200	string
<b>description</b> <i>required</i>	Length : 1 - 600	string
<b>email</b> <i>required</i>	Length : 1 - 254	string (email)
<b>phone_number</b> <i>optional</i>	Maximal length : 10	string
<b>shelter_image</b> <i>optional</i> <i>read-only</i>		string (uri)
<b>shelter_name</b> <i>required</i>	Length : 1 - 128	string
<b>website</b> <i>optional</i>	Maximal length : 128	string (uri)

## ShelterCreate

Name	Description	Schema
<b>address</b> <i>required</i>	Length : 1 - 200	string
<b>confirm_password</b> <i>required</i>	Minimum length : 1	string
<b>description</b> <i>required</i>	Length : 1 - 600	string
<b>email</b> <i>required</i>	Length : 1 - 254	string (email)
<b>password</b> <i>required</i>	Minimum length : 1	string
<b>phone_number</b> <i>optional</i>	Maximal length : 10	string
<b>shelter_name</b> <i>required</i>	Length : 1 - 128	string
<b>username</b> <i>required</i>	Minimum length : 1	string
<b>website</b> <i>optional</i>	Maximal length : 128	string (uri)

## ShelterUpdate

Name	Description	Schema
<b>address</b> <i>required</i>	Length : 1 - 200	string
<b>description</b> <i>required</i>	Length : 1 - 600	string
<b>email</b> <i>required</i>	Length : 1 - 254	string (email)
<b>phone_number</b> <i>optional</i>	Maximal length : 10	string
<b>shelter_image</b> <i>optional</i> <i>read-only</i>		string (uri)
<b>website</b> <i>optional</i>	Maximal length : 128	string (uri)

## TokenObtainPair

Name	Description	Schema
<b>password</b> <i>required</i>	Minimum length : 1	string
<b>username</b> <i>required</i>	Minimum length : 1	string

## TokenRefresh

Name	Description	Schema
<b>access</b> <i>optional</i> <i>read-only</i>	Minimum length : 1	string
<b>refresh</b> <i>required</i>	Minimum length : 1	string