Database Entity-Relationship Diagram

Application

id	BigAutoField
pet	ForeignKey (id)
user	ForeignKey (user_ptr)
address	CharField
created_at	DateTimeField
digital_signature	CharField
email	EmailField
last_updated	DateTimeField
name	CharField
phone_number	CharField
qualifications	TextField
status	CharField

Session <AbstractBaseSession>

session_key	CharField
expire_date	DateTimeField
session_data	TextField

Pet

id	BigAutoField
breed	ForeignKey (id)
owner	ForeignKey (user_ptr)
pet_type	ForeignKey (id)
adoption_fee	DecimalField
adoption_location	CharField
age	FloatField
color	CharField
last_modified	DateTimeField
medical_history	TextField
name	CharField
notes	TextField
picture_1	ImageField
picture_2	ImageField
picture_3	ImageField
sex	CharField
status	CharField
weight	FloatField

AbstractBaseSession

expire_date	DateTimeField
session_data	TextField

Rating

id	BigAutoField
shelter	ForeignKey (user_ptr)
user	ForeignKey (id)
rating	PositiveIntegerField

PetSeeker

user_ptr	OneToOneField (id)

Notification

id	BigAutoField
content_type	ForeignKey (id)
user	ForeignKey (id)
created_at	DateTimeField
notification_type	CharField
object_id	PositiveIntegerField
read	BooleanField

PetShelter

user_ptr	OneToOneField (id)
mission_statement	TextField
mission_title	CharField

Breed

id	BigAutoField
pet_type	ForeignKey (id)
name	CharField

LogEntry

id	AutoField
content_type	ForeignKey (id)
user	ForeignKey (id)
action_flag	PositiveSmallIntegerField
action_time	DateTimeField
change_message	TextField
object_id	TextField
object_repr	CharField

Comment

id	BigAutoField
content_type	ForeignKey (id)
reply_to	ForeignKey (id)
user	ForeignKey (id)
created_at	DateTimeField
object_id	PositiveIntegerField
text	TextField

User <AbstractUser>

id	BigAutoField
address	CharField
banner	ImageField
date_joined	DateTimeField
description	TextField
email	EmailField
first_name	CharField
is_active	BooleanField
is_staff	BooleanField
is_superuser	BooleanField
last_login	DateTimeField
last_name	CharField
password	CharField
phone_number	CharField
profile_picture	ImageField
username	CharField
website	URLField

PetType

id	BigAutoField
name	CharField

AbstractUser <AbstractBaseUser,PermissionsMixin>

date_joined	DateTimeField
email	EmailField
first_name	CharField
is_active	BooleanField
is_staff	BooleanField
is_superuser	BooleanField
last_login	DateTimeField
last_name	CharField
password	CharField
username	CharField

Group

id	AutoField
name	CharField

PermissionsMixin

AbstractBaseUser

Permission

id	AutoField
content_type	ForeignKey (id)
codename	CharField
name	CharField

ContentType

id	AutoField
app_label	CharField
model	CharField

Relationships

- user (application)
- pet (application)
- abstract inheritance
- shelter (ratings)
- owner (pet)
- breed (pet)
- pet_type (pet)
- user (rating)
- multi-table inheritance
- reply_to (comment)
- user (logentry)
- user (comment)
- user (notification)
- multi-table inheritance
- pet_type (breed)
- abstract inheritance
- groups (user)
- abstract inheritance
- abstract inheritance
- user_permissions (user)
- permissions (group)
- content_type (logentry)
- content_type (comment)
- content_type (notification)
- content_type (permission)

PetPal API

Overview

Documentation for PetPal REST API

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/seeker/

Description

Create new pet seeker based on the given information. The required fields includes: username, password, and email Errors include: 1. Username: A user with that username already exists. 2. Email: Enter a valid email address. 3. All: This field may not be blank. A 400 Bad request would be returned. Authentication is not required for the sign up.

Additional Notes

More documentation can be found at the following URL upon run of server:
http://localhost:8000/docs/

Every Security Basic refers to JWT authorization since the swagger API document conversion does not properly identify it.

Parameters

Type	Name	Schema
Body	**data** *required*	PetSeeker

Responses

HTTP Code	Schema
201	PetSeeker

Tags

- accounts

GET /accounts/seeker/{id}/

Description

Pet seeker profile with given ID will be retrieved. User must either be pet seeker himself or pet shelter account that has a pending application with the intended pet seeker.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer

Responses

HTTP Code	Schema
200	PetSeeker

Tags

- accounts

PUT /accounts/seeker/{id}/

Description

Update seeker profile info with given ID based on the given info User is required to be logged in as the current pet seeker account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer
Body	**data** *required*		PetSeeker

Responses

HTTP Code	Schema
200	PetSeeker

Tags

- accounts

DELETE /accounts/seeker/{id}/

Description

Delete pet seeker with the given ID. User are required to be logged in as the current pet seeker account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer

Responses

HTTP Code	Schema
204	No Content

Tags

- accounts

PATCH /accounts/seeker/{id}/

Description

Partial update seeker profile info with given ID based on the given info of pet seeker. User is

required to be logged in as the current pet seeker account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer
Body	**data** *required*		PetSeeker

Responses

HTTP Code	Schema
200	PetSeeker

Tags

- accounts

POST /accounts/shelter/

Description

Create new pet shelter based on the given information. The required fields includes: username, password, and email Errors include: 1. Username: A user with that username already exists. 2. Email: Enter a valid email address. 3. All: This field may not be blank. A 400 Bad request would be returned. Authentication is not required for the sign up.

Parameters

Type	Name	Schema
Body	**data** *required*	PetShelter

Responses

HTTP Code	Schema
201	PetShelter

Tags

- accounts

GET /accounts/shelter/

Description

Get a list of all shelter profiles with the pet shelter information. Authentication is not required.

Parameters

Type	Name	Description	Schema
Query	**page** *optional*	Page number	integer
Query	**size** *optional*	Page size	integer

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< PetShelter > array

Tags

- accounts

GET /accounts/shelter/{id}/

Description

Pet shelter profile with given ID will be retrieved. Authentication is not required.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer

Responses

HTTP Code	Schema
200	PetShelter

Tags

- accounts

PUT /accounts/shelter/{id}/

Description

Update shelter profile info with given ID based on the given info User is required to be logged in as the current pet shelter account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer
Body	**data** *required*		PetShelter

Responses

HTTP Code	Schema
200	PetShelter

Tags

- accounts

DELETE /accounts/shelter/{id}/

Description

Delete pet shelter with the given ID. User are required to be logged in as the current pet shelter account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer

Responses

HTTP Code	Schema
204	No Content

Tags

- accounts

PATCH /accounts/shelter/{id}/

Description

Partial update shelter profile info with given ID based on the given info of pet shelter. User is required to be logged in as the current pet shelter account.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this user.	integer
Body	**data** *required*		PetShelter

Responses

HTTP Code	Schema
200	PetShelter

Tags

- accounts

POST /accounts/token/

Description

Login the user with the given username and password credentials. 400 Bad request will be

returned if 1). User Does Not Exist. 2). The password does not match with the username.

A JWT corresponding to the logged-in user would be returned if authentication succeeds.

Parameters

Type	Name	Schema
Body	**data** *required*	CustomizedTokenObtain

Responses

HTTP Code	Schema
201	CustomizedTokenObtain

Tags

- accounts

POST /accounts/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	**data** *required*	TokenRefresh

Responses

HTTP Code	Schema
201	TokenRefresh

Tags

- accounts

POST /applications/

Description

Creates application based on the given details about the application. User can only create application for a pet listing that is "available". User is required to be logged in as a pet seeker.

Parameters

Type	Name	Schema
Body	**data** *required*	CreateApplication

Responses

HTTP Code	Schema
201	CreateApplication

Tags

- applications

GET /applications/

Description

If user is logged in as pet shelter, it will return a list of applications sent to the pet that user owns. If user is logged in as pet seeker, it will return all the applications that pet seeker sent. User are required to be logged in. You can provide query parameters to filter applications by status and sort by creation time or last update time. If no query parameter is provided it will not apply any filter.

Parameters

Type	Name	Description	Schema
Query	**order_by** *required*	Options for sorting. Negative sign (-) indicates descending order.	enum (created_at, -created_at, last_updated, -last_updated)
Query	**page** *optional*	Page number	integer
Query	**size** *optional*	Page size	integer
Query	**status** *optional*	Status to be filtered.	enum (accepted, denied, pending, withdrawn)

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< CreateApplication > array

Tags

- applications

GET /applications/{id}/

Description

Retrieves application with the given ID. User is required to be logged in either as pet seeker or pet shelter. Pet seeker can only retrieve application that they sent and pet shelter can only retrieve application that they received.

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Schema
200	UpdateApplication

Tags

- applications

PUT /applications/{id}/

Description

Updates application with the given ID. User is required to be logged in either as pet seeker or pet shelter. - Shelter can only update the status of an application from pending to accepted or denied. - Pet seeker can only update the status of an application from pending or accepted to withdrawn.

Pet seeker can only update application that they sent and pet shelter can only update application that they received.

Note: there are no difference between put and patch because we only have 1 required field.

Parameters

Type	Name	Schema
Path	**id** *required*	string
Body	**data** *required*	UpdateApplication

Responses

HTTP Code	Schema
200	UpdateApplication

Tags

- applications

PATCH /applications/{id}/

Description

Do partial update on the application with the given ID. User is required to be logged in either as pet seeker or pet shelter. - Shelter can only update the status of an application from pending to accepted or denied. - Pet seeker can only update the status of an application from pending or accepted to withdrawn.

Pet seeker can only update application that they sent and pet shelter can only update application that they received.

Note: there are no difference between put and patch because we only have 1 required field.

Parameters

Type	Name	Schema
Path	**id** *required*	string
Body	**data** *required*	UpdateApplication

Responses

HTTP Code	Schema
200	UpdateApplication

Tags

- applications

POST /comments/application/

Description

Create comment on an Application. Ensures that the User who wishes to comment is part of the Application, whether that be a Seeker or Shelter of the Application. Verifies that comment replies cannot be directed across models (e.g. Cannot reply to Application when on PetShelter, Cannot reply to Application from a different Application).

Parameters

Type	Name	Schema
Body	**data** *required*	Comment

Responses

HTTP Code	Schema
201	Comment

Tags

- comments

GET /comments/application/

Description

Returns paginated list of comments for an Application given the object_id (primary key) of the Application. Orders by creation date.

Parameters

Type	Name	Description	Schema
Query	**page** *optional*	Page number	integer
Query	**size** *optional*	Page size	integer

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< Comment > array

Tags

- comments

POST /comments/rating/

Description

Create rating for a PetShelter from a User.

Parameters

Type	Name	Schema	
Body	**data** *required*	[Rating](#)	

Responses

HTTP Code	Schema
201	[Rating](#)

Tags

- comments

GET /comments/rating/

Description

Returns all the ratings for a given shelter.

Parameters

Type	Name	Description	Schema
Query	**page** *optional*	Page number	integer
Query	**size** *optional*	Page size	integer

Responses

HTTP Code	Schema
200	[Response 200](#)

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< [Rating](#) > array

- comments

POST /comments/shelter/

Parameters

Type	Name	Schema
Body	**data** *required*	Comment

Responses

HTTP Code	Schema
201	Comment

Tags

- comments

GET /comments/shelter/

Description

Returns paginated list of comments for a PetShelter given the object_id (primary key) of the PetShelter. Orders by creation date.

Parameters

Type	Name	Description	Schema
Query	**page** *optional*	Page number	integer
Query	**size** *optional*	Page size	integer

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< Comment > array

Tags

- comments

GET /comments/{id}/

Description

Retrieves a single comment given the primary key in URL.

Parameters

Type	Name	Description	Schema
Path	**id** *required*	A unique integer value identifying this comment.	integer

Responses

HTTP Code	Schema
200	Comment

Tags

- comments

POST /notifications/

Description

Create a new notification based on the provided data. The notification type and user association are determined by the type of user and the type of content object involved. - If the user is a PetSeeker and the content type represents an Application, it creates a status update notification. - If the user is a PetShelter and the content type represents an Application, it creates an application creation notification. - If the user is a PetShelter and the content type represents a Comment, and the

comment is made on Shelter, it creates a new review notification. - If the content type represents a Comment and comment was made on an Application, it creates a new message notification. - If the content type represents a Pet, it creates a new pet listing notification.

Parameters

Type	Name	Schema
Body	**data** *required*	Notification

Responses

HTTP Code	Schema
201	Notification

Tags

- notifications

GET /notifications/

Description

Retrieve a list of notifications for the authenticated user with filtering. - Users (shelter and seeker) can only view their own notifications. - User can filter notifications by read/unread to get all unread notifications. - It is sorted by creation time in descending order (latest first).

Parameters

Type	Name	Description	Schema
Query	**page** *optional*	Page number	integer
Query	**read** *optional*	Read attribute used to filter.	boolean
Query	**size** *optional*	Page size	integer

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< Notification > array

Tags

- notifications

GET /notifications/{id}/

Description

Retrieve details of a user's notification and provide links to associated models. - If the notification refers to a comment, link to the new comment added. - If the notification refers to an application, link to application creation and status update. - If the application refers to a pet, link to a new pet listing. It also updates the state of a notification from "unread" to "read".

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Schema
200	Notification

Tags

- notifications

DELETE /notifications/{id}/

Description

Delete the notification with the given id. The notification has to be associated with the user.

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Schema
204	No Content

Tags

- notifications

POST /pets/

Description

Create new pet based on the given details of the pet. User are required to be logged in as pet shelter.

Parameters

Type	Name	Schema
Body	**data** *required*	Pet

Responses

HTTP Code	Schema
201	Pet

Tags

- pets

GET /pets/

Description

Searches pets based on various filters and sorting options. Authentication is not required. 400 Bad Request will be returned when filter options for the query parameters are invalid. When filter options are not given, it will return all pets.

Parameters

Type	Name	Description	Schema
Query	**adoption_fee** *optional*	Fee associated with adopting the pet.	string (decimal)
Query	**adoption_fee***gt* optional__	Upper bound (exclusive) of adoption_fee for filtering.	string (decimal)
Query	**adoption_fee***gte* optional__	Upper bound (inclusive) of adoption_fee for filtering.	string (decimal)
Query	**adoption_fee***lt* optional__	Lower bound (exclusive) of adoption_fee for filtering.	string (decimal)
Query	**adoption_fee***lte* optional__	Lower bound (inclusive) of adoption_fee for filtering.	string (decimal)
Query	**age** *optional*	Age of the pet.	number
Query	**age***gt* optional__	Upper bound (exclusive) of age for filtering.	number
Query	**age***gte* optional__	Upper bound (inclusive) of age for filtering.	number
Query	**age***lt* optional__	Lower bound (exclusive) of age for filtering.	number
Query	**age***lte* optional__	Lower bound (inclusive) of age for filtering.	number
Query	**breed** *optional*	Id of the Breed model, representing the breed of the pet (e.g. American Bobtail).	integer
Query	**color** *optional*	Color of the pet. null represents unknown/others.	enum (brown, gray, white, black, orange)
Query	**order_by** *optional*	Options for sorting. Negative sign (-) indicates descending order.	enum (name, -name, age, -age, weight, -weightadoption_fee, -adoption_fee)
Query	**owner** *optional*	Pet shelter that owns the pet.	string
Query	**page** *optional*	Page number	integer
Query	**pet_type** *optional*	Id of the PetType model, representing the type (or category) of the pet such as dog, cat, birds, etc. null represents unknown/others.	integer

Type	Name	Description	Schema
Query	**sex** *optional*	Sex of the pet. null represents unknown/others.	enum (male, female)
Query	**size** *optional*	Page size	integer
Query	**status** *optional*	Pet adoption status.	enum (withdrawn, pending, adopted, available)
Query	**weight** *optional*	Weight of the pet.	number
Query	**weight*gt*** optional__	Upper bound (exclusive) of weight for filtering.	number
Query	**weight*gte*** optional__	Upper bound (inclusive) of weight for filtering.	number
Query	**weight*lt*** optional__	Lower bound (exclusive) of weight for filtering.	number
Query	**weight*lte*** optional__	Lower bound (inclusive) of weight for filtering.	number

Responses

HTTP Code	Schema
200	Response 200

Response 200

Name	Description	Schema
count *optional*	Total number of items returned.	integer
next *optional*	Link for next page.	string (uri)
previous *optional*	Link for previous page.	string (uri)
results *required*		< Pet > array

Tags

- pets

GET /pets/{id}/

Description

Pet with given ID will be retrieved. Authentication is not required.

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Schema
200	Pet

Tags

- pets

PUT /pets/{id}/

Description

Update pet with given ID based on the given details of the pet. User are required to be logged in as pet shelter. User are required to be the owner of the pet.

Parameters

Type	Name	Schema
Path	**id** *required*	string
Body	**data** *required*	Pet

Responses

HTTP Code	Schema
200	Pet

Tags

- pets

DELETE /pets/{id}/

Description

Delete pet with the given ID. User are required to be logged in as pet shelter. User are required to be the owner of the pet.

Parameters

Type	Name	Schema
Path	**id** *required*	string

Responses

HTTP Code	Schema
204	No Content

Tags

- pets

PATCH /pets/{id}/

Description

Partial update pet with given ID based on the given details of the pet. User are required to be logged in as pet shelter. User are required to be the owner of the pet.

Parameters

Type	Name	Schema
Path	**id** *required*	string
Body	**data** *required*	Pet

Responses

HTTP Code	Schema
200	Pet

- pets

Definitions

Comment

Name	Description	Schema
content_type *optional* *read-only*	Model type that is being commented on	integer
created_at *optional* *read-only*	Date when the comment was created	string (date-time)
id *optional* *read-only*		integer
object_id *required*	Primary Key of the model being commented on	integer
reply_to *optional*	Who the comment is referring to	integer
text *required*	Text of the comment **Length** : 1 - 500	string
user *optional* *read-only*	User who wrote the comment	integer

CreateApplication

Name	Description	Schema
address *required*	Sender's address. **Length** : 1 - 128	string
created_at *optional* *read-only*	Timestamp when the application was created.	string (date-time)
digital_signature *required*	Digital signature of the sender. **Length** : 1 - 128	string
email *required*	Sender's email address. **Length** : 1 - 254	string (email)

Name	Description	Schema
id *optional* *read-only*	Identifier of the application.	integer
last_updated *optional* *read-only*	Timestamp when the most recent update was made on the application. Adding comment for this application will also update this value.	string (date-time)
name *required*	Name of the sender who sent the application (can be different from the username). **Length** : 1 - 128	string
pet *required*	Pet associated with the application.	integer
phone_number *required*	Sender's phone number. **Length** : 1 - 12	string
qualifications *required*	Sender's qualifications for adopting the pet. **Minimum length** : 1	string
status *optional*	Status of the application indicating whether application	enum (accepted, denied, pending, withdrawn)
user *optional* *read-only*	Pet seeker who sent the application.	string

CustomizedTokenObtain

Name	Description	Schema
password *required*	**Minimum length** : 1	string
username *required*	**Minimum length** : 1	string

Notification

Name	Description	Schema
content_type *required*	The content type of the referenced object (used to find the notification type).	integer
created_at *optional* *read-only*	The timestamp when the notification was created.	string (date-time)

Name	Description	Schema
id *optional* *read-only*	Identifier of the notifications.	integer
model_link *optional* *read-only*	Link to its associated model. **Minimum length** : 1	string (uri)
notification_t ype *optional*	The type of notification. **Maximal length** : 50	string
object_id *required*	The ID of the referenced object.	integer
read *optional* *read-only*	Indicates whether the notification has been read (True) or not (False).	boolean
user *optional* *read-only*	The user associated with this notification.	integer

Pet

Name	Description	Schema
adoption_fee *required*	Fee associated with adopting the pet.	string (decimal)
adoption_loca tion *required*	Location where this pet is available for adoption. **Length** : 1 - 128	string
age *required*	Age of the pet. **Minimum value** : 0	number
breed *optional*	Id of the Breed model, representing the breed of the pet (e.g. American Bobtail).	integer
color *optional*	Color of the pet. null represents unknown/others.	enum (brown, gray, white, black, orange)
id *optional* *read-only*	Id of the pet.	integer
last_modified *optional* *read-only*	Time when the most recent update was made on pet.	string (date-time)

Name	Description	Schema
medical_history *required*	Medical history of the pet. **Minimum length** : 1	string
name *required*	Name of the pet. **Length** : 1 - 128	string
notes *required*	Additional notes that Pet Shelter can provide about the pet. **Minimum length** : 1	string
owner *optional* *read-only*	Pet shelter that owns the pet.	string
pet_type *optional*	Id of the PetType model, representing the type (or category) of the pet such as dog, cat, birds, etc. null represents unknown/others.	integer
picture_1 *optional* *read-only*	Picture of the pet.	string (uri)
picture_2 *optional* *read-only*	Picture of the pet.	string (uri)
picture_3 *optional* *read-only*	Picture of the pet.	string (uri)
sex *optional*	Sex of the pet. null represents unknown/others.	enum (male, female)
status *optional*	Pet adoption status.	enum (withdrawn, pending, adopted, available)
weight *required*	Weight of the pet. **Minimum value** : 0	number

PetSeeker

Name	Description	Schema
address *optional*	address of the user **Maximal length** : 128	string
banner *optional* *read-only*	user profile banner	string (uri)
description *optional*	user description	string

Name	Description	Schema
email *required*	email of the user **Length** : 1 - 254	string (email)
first_name *optional*	**Maximal length** : 150	string
id *optional* *read-only*		integer
last_name *optional*	**Maximal length** : 150	string
password *required*	**Minimum length** : 1	string
phone_number *optional*	phone number of the user **Maximal length** : 12	string
profile_picture *optional* *read-only*	user profile picture	string (uri)
username *required*	Required. 150 characters or fewer. Letters, digits and @/.//-/_ only. + **Length** : `1 - 150` + **Pattern** : `"^[\\w.@-]+$"`	string

PetShelter

Name	Description	Schema
address *optional*	address of the user **Maximal length** : 128	string
banner *optional* *read-only*	user profile banner	string (uri)
description *optional*	user description	string
email *required*	email of the user **Length** : 1 - 254	string (email)
first_name *optional*	**Maximal length** : 150	string
id *optional* *read-only*		integer

Name	Description	Schema
last_name *optional*	**Maximal length** : 150	string
mission_state ment *optional*	mission_statement of the pet shelter	string
mission_title *optional*	mission_title of the pet shelter **Maximal length** : 128	string
password *required*	**Minimum length** : 1	string
phone_numbe r *optional*	phone number of the user **Maximal length** : 12	string
profile_pictur e *optional* *read-only*	user profile picture	string (uri)
username *required*	Required. 150 characters or fewer. Letters, digits and @/.//-/_ only. + **Length** : `1 - 150` + **Pattern** : `"^[\\w.@-]+$"`	string

Rating

Name	Description	Schema
rating *optional*	Integer value between 0-5 that depicts the rating of the shelter by the user **Maximum value** : 5	integer
shelter *required*	Shelter that the rating is for	string
user *optional* *read-only*	User who gave the rating	integer

TokenRefresh

Name	Description	Schema
access *optional* *read-only*	**Minimum length** : 1	string
refresh *required*	**Minimum length** : 1	string

UpdateApplication

Name	Description	Schema
address *optional* *read-only*	Sender's address. **Minimum length** : 1	string
created_at *optional* *read-only*	Timestamp when the application was created.	string (date-time)
digital_signature *optional* *read-only*	Digital signature of the sender. **Minimum length** : 1	string
email *optional* *read-only*	Sender's email address. **Minimum length** : 1	string (email)
id *optional* *read-only*		integer
last_updated *optional* *read-only*	Timestamp when the most recent update was made on the application. Adding comment for this application will also update this value.	string (date-time)
name *optional* *read-only*	Name of the sender who sent the application (can be different from the username). **Minimum length** : 1	string
pet *optional* *read-only*	Pet associated with the application.	integer
phone_number *optional* *read-only*	Sender's phone number. **Minimum length** : 1	string
qualifications *optional* *read-only*	Sender's qualifications for adopting the pet. **Minimum length** : 1	string
status *required*	Status of the application indicating whether application	enum (accepted, denied, pending, withdrawn)
user *optional* *read-only*	Pet seeker who sent the application.	string