

Student: Joaquin Saldana
Assignment: Final Project

This is the REST API documentation for my final project. Below is a description of how the API is implemented and the purpose of the application.

Main Link: <https://final-project-saldanaj.appspot.com>

Video Link: <https://youtu.be/KWquFTuGgKU>

Description

Personally, there's nothing greater than the unconditional love of my dog. If you carry the same opinion, then you will agree there's nothing that makes me happier than keeping that awesome smile, and that means doing all that things my dog loves. This includes long walks, visits to the dog park and the beach. Above all, my dog loves meeting other dogs he considers as friends, but it can be hard keeping track of all of your dogs friends and their owners.

This API and mobile application makes it easier for you to enter your dogs canine friend information such as name and breed (along with other data) and information on their owner's (such as name, age, and preferred park to visit).

Below is the description of the entities and their properties:

1. Dog
 - a. Name – string
 - b. Breed – string
 - c. Age – integer
 - d. Owner – string
 - e. Alive – boolean
2. Human
 - a. Name – string
 - b. Age – integer
 - c. Favorite Dog Park – string
 - d. Dogs Owned – structured property

Below are the URL's and the verbs (HTTP actions) for these URL's.

Dogs

Create Dog

POST /dogs

Notes

- Dog entity has a property titled “alive” and it’s automatically set to true when the dog is created.

Name	Type	Description
name (required)	String	Name of the dog
breed (required)	String	The dog breed
age (required)	Integer	The age of the dogs

List All Dogs

GET /dogs

List Specific Dog

GET /dogs/{dog id}

Modify A Dog

PATCH /dogs/{dog id}

Notes

- If modification is successful, the return a 201 response code if not a 505 error code

Name	Type	Description
name (optional)	String	Name of the dog
breed (optional)	String	The dog breed
age (optional)	Integer	The age of the dogs

Change Dog Alive Status

PUT /dogs/{dog id}

Name	Type	Description
alive (required)	Boolean	Dog alive status

Delete A Dog

DELETE /dogs/{dog id}

Notes

- If a dog is deleted, it's removed from the owner's list of owned dogs

Humans

Create Human

POST /humans

Name	Type	Description
name (required)	String	Name of the human
age	Integer	Age of the human
favorite_park	String	Favorite park they take their dog to

List All Humans

GET /humans

List Specific Human

GET /humans/{human id}

Modify A Human

PATCH /humans/{human id}

Notes

- If modification is successful, the return a 201 response code if not a 505 error code

Name	Type	Description
name (optional)	String	Name of the human
age (optional)	Integer	Age of the human
favorite_park (optional)	String	Favorite park they take their dog to

Delete A Human

DELETE /humans/{human id}

Notes

- If a human is deleted, the dog(s) in his list of owned dogs are set to no owner (dog owner property is set to none/null)

Link Dog To Owner

Much of the method for this request handler follows the same methodology that is found for GitHub Gists

Link Dog to Owner/Human

PUT /dogs/{dog id}/owner/{human id}

Notes:

- If a human is associated w/ a dog, the dog is also added to the humans list of dogs owned
- If you attempt to add a dog that is already linked to this human you will receive a 408 error code w/ response writing "Dog Already In List"
- If you attempt to add an owner to a dog that already has an owner you will receive a 403 response code w/ response writing "Dog Already Has Owner. Must Remove Current Owner Before Proceeding", forcing the user to perform an HTTP DELETE request to the URL /dogs/{dog id}/owner/{human id} to keep data integrity

If successful, the response is as follows:

```
{
  "self": "/humans/aghkZXZ-Tm9uZXISCxIFSHVtYW4YgICAgICAkAoM",
  "favDogPark": "Brickell Park",
  "name": "Christian",
  "age": 32,
  "human_id": "aghkZXZ-Tm9uZXISCxIFSHVtYW4YgICAgICAkAoM",
  "dogsOwned": [
    {
      "dog_link": "/dogs/aghkZXZ-Tm9uZXIQCxIDRG9nGICAgICAgNAJDA",
      "dog_id": "aghkZXZ-Tm9uZXIQCxIDRG9nGICAgICAgNAJDA"
    }
  ]
}
```

Unlink Human/Owner From Dog

DELETE /dogs/{dog id}/owner/{human id}

Notes:

- If a human is dis-associated w/ a dog, the dog is removed from the human's list of dogs owned
- If the dog does not have an owner, you will receive a 403 error w/ the response writing "Error: Dog Does Not Have Owner"
- If the dog is not found in the humans list of owned dogs, you will receive a 408 response code w/ response writing "Error: Dog Is Not Present In the Humans List of Dogs Owned"

If successful, the response is as follows:

```
[
  {
    "dogsOwned": [
      {
        "dog_link": "/dogs/aghkZXZ-Tm9uZXlQCxIDRG9nGICAgICAgNAJDA",
        "dog_id": "aghkZXZ-Tm9uZXlQCxIDRG9nGICAgICAgNAJDA"
      }
    ],
    "age": 32,
    "human_id": "aghkZXZ-Tm9uZXlSCxIFSHVtYW4YgICAgICAgAaM",
    "self": "/humans/aghkZXZ-Tm9uZXlSCxIFSHVtYW4YgICAgICAgAaM",
    "favDogPark": "Brickell Park",
    "name": "Christian"
  },
  {
    "name": "Tigger",
    "age": 18,
    "dog_id": "aghkZXZ-Tm9uZXlQCxIDRG9nGICAgICAgNALDA",
    "owner": null,
    "self": "/dogs/aghkZXZ-Tm9uZXlQCxIDRG9nGICAgICAgNALDA",
    "alive": true,
    "breed": "Mixed Terrier"
  }
]
```