Final Project (Cloud only option)

REST API written in Ruby Sinatra DSL.

The live server is up here: http://35.196.169.169:8080/

Test results here: https://media.oregonstate.edu/media/t/0_4a5lzkiw

Test results and server logs are in the root of this repo: test_results.log and server.log

API

Authentication

Authentication is required to create/modify/delete a users posts and dogs. All requests marked with (requires auth) must provide a Github access token as part of the request body or uri
Example:

```
         DELETE /posts/<my post id>?access_token='<my github acces token>'
```

Any requests that require authentication will return a `401 Unauthorized` if no access token is provided.

Any request that attempts to modify/delete unauthorized resources will receive a `401 Unauthorized`.

Entities

Entities

Post:

```
      title: String
      post_id: Integer
      date_created: String
      content: String
      dogs_ids: Array
      owner_name: String
      owner_id: Integer
```

Dog:

```
      name: String
      dog_id: Integer
      breed: String
      breed_images: Array
      owner_name: String
      owner_id: Integer
```

Routes

`GET /posts`

Get all posts

`GET /posts/{post_id}`

get single post

`POST /posts`

Create a post (requires auth)

Request JSON:

```
    {
      "title": "My awesome dog Slippy!",
      "content": "This is where I talk about how awesome my dog slippy is",
      "dog_ids": [1234,4321],
      "access_token": "<my Github access token>"
    }
```

Response:

```
200
    {
      "title":"TEST",
      "content":"this is a test",
      "owner_name":"Stephen D. McGuckin",
      "dog_ids":[],
      "post_id":1512123217891,
      "date_created":"12/1/2017"
    }
```

`PATCH /posts/{post_id}`

Modify a post (requires auth)

`DELETE /posts/{post_id}`

Deletes a post (requires auth)

`PUT /dogs/{id}`

Replace a dog (requires auth)

`GET /dogs`

Gets all dogs

`GET /dogs/{id}`

Get a dog

`GET '/dogs/breed/{breed}`

Gets all dogs for a breed

`POST /dogs`

Create a dog (requires auth)

Request JSON:

```
      {
        "name":"Slippers",
        "breed":"american cocker spaniel"
      }
```

Response:

```
{
    "name":"Slippers",
    "breed":"american cocker spaniel",
    "owner_name":"Stephen D. McGuckin",
    "dog_id":1512115574454,
    "breed_images":
         ["https://dog.ceo/api/img/spaniel-cocker/n02102318_9307.jpg",
         "https://dog.ceo/api/img/spaniel-brittany/n02101388_4342.jpg"]
}
```

`PATCH /dogs/{id}`

Modify a dog (requires auth)

`DELETE /dogs/{id}`

Delete a dog (requires auth)

`PUT /dogs/{id}`

Replace a dog (requires auth)

Account system

This API uses Github as its 3rd party account system. That is all resources are linked to a Github user account. Authentication is required for creating resources. Access must be authorized to modifying resources, resources can only be updated or deleted with the access token of user that originally posted the resource. No authentication is required for GET request on any resource

Their are no scopes required for the access token. The API authenticates with the access token and then gets information about the users name and the users unique Github user id. On creating a resource the Github user id is persisted along with rest of the resource data and used for authorizing requests for that resource. The Github user id is not the username. The Github user id is a unique id that Github stores to uniquely identify a user account.

DOG API

On creation of Dog resource the Dog API is called to grab a random image for the dogs breed. The Dog API serves 20,000+ images of dogs for dog for all standard breeds and sub-breeds based on the Stanford Dogs data set.

Requirements

Ruby 2.4.0 or higher. You can try lower versions but I do not recommend it. If you don’t have ruby installed already
I suggest using RVM

Make sure you can run:

```
    ruby -v
```

Running the tests

These are the following steps for running the tests against the live server running on http://35.196.169.169/8080

From the root directory of this project run the following commands

Install bundler ruby gem

```
    gem install bundler
```

Install the gem dependencies

```
bundle install
```

Execute the tests

```
bundle exec rake
```