

Brent Doil

CS 493 FINAL

Documentation:

GitHub:

<https://github.com/bmdoil/bdoil-cs493-grading-final>

App: <http://app.brentdoil.com>

This is a coding question and answer database, inspired by leetcode.com.

Users must be authenticated to view any routes

Users must register to answer questions.

Only admins can create questions at this time

All questions have an `author_id`, the id of the user who submitted the question.

Each question has a question body, the test code, and the solution, as well as the difficulty of the question (Easy, Moderate, Hard)

Scores track each user's answers to questions. This is a many-to-many relationship that tracks whether the user answered the question correctly, how many points they received, and how long their code took to run.

Users:

- username
- email
- password
- active
- admin

Questions:

- `author_id`
- `body`
- `test_code`

- test_solution
- difficulty

Scores:

- user_id
- question_id
- correct
- points
- runtime

The front end is a work in progress. The server-side endpoints are fully functioning and **full API documentation is being served live with Swagger:**

<http://app.brentdoil.com/swagger>

Each endpoint is documented fully. This includes parameters, example values, and response codes.

POST	/login/login	Logs a user in
GET	/login/me	Returns the logged in user's profile
POST	/login/register	Creates a new user. Copy the token in t
GET	/login/signout	Sigs a user out. This only works client s
DELETE	/question/{question_id}/user/{user_id}	
GET	/question/{question_id}/user/{user_id}	f
PUT	/question/{question_id}/user/{user_id}	u

Most of the endpoints require authentication. These are designated with the lock symbol

Log in using the POST /login/login route with:

email: admin@admin.com

password: admin

Copy the response token and click the green Authorize button. Paste the token and click authorize.

You are now authorized as an admin user. Admin users have full access to all the routes.

To test normal user routes, please register a new user with the POST /login/register endpoint. New users are by default not admins.

Pagination: Each root entity GET can take a `page` parameter.

Responses will also provide next_page and prev_page links for pagination.

```
{
  "data": {
    "num_users": 101,
    "users": [
      {
        "id": 1,
        "username": "admin",
        "email": "admin@admin.com",
        "active": true,
        "admin": true,
        "self": "http://localhost/users/1"
      },
      {
        "id": 2,
        "username": "zpoadajfsw",
        "email": "oketsqtzpl@example.com",
        "active": true,
        "admin": false,
        "self": "http://localhost/users/2"
      },
      {
        "id": 3,
        "username": "mwtsigfqsp",
        "email": "gjvhprdyu@example.com",
```

```
    "active": true,
    "admin": false,
    "self": "http://localhost/users/3"
  },
  {
    "id": 4,
    "username": "vtbvxpoxwi",
    "email": "onyfgbqtsf@example.com",
    "active": true,
    "admin": false,
    "self": "http://localhost/users/4"
  },
  {
    "id": 5,
    "username": "lnbazzusoj",
    "email": "gtkstwdsog@example.com",
    "active": true,
    "admin": false,
    "self": "http://localhost/users/5"
  }
]
},
"status": "success",
"next_page": "/users?page=2",
"prev_page": null
}
```

Running the tests

Grading Branch for CS493 Final Project

Setup:

- Clone it to your machine: `git clone`
<https://github.com/bmdoil/bdoil-cs493-grading-final.git>

OR extract `bdoil-cs493-grading-final.zip` from the assignment submission

- `cd bdoil-cs493-grading-final`

To test the live routes

- [Navigate to my API documentation] (<http://app.brentdoil.com/swagger>)

To test locally:

Note: Locally the app uses ports 5005, 8888, and 3010. If you have conflicts with these ports, they can be changed in `docker-compose.yml`

- `docker` & `docker-compose` must be installed

- `docker -v` && `docker-compose -v`

- `export BASE_URL=http://localhost`

- `docker-compose up -d --build`

- `docker-compose exec server python manage.py recreate_db`

- `docker-compose exec server python manage.py add_test_data`

- `docker-compose exec server python manage.py test`

- <http://localhost/swagger>

##To test with Travis CI

- After forking the repository, sign up on travis-ci.org if you don't already have an account

- Navigate to the repo on travis:

[https://travis-ci.org/\[yourusername\]/bdoil-cs493-grading](https://travis-ci.org/[yourusername]/bdoil-cs493-grading)

- Click 'Activate Repository'

- Push the repo to initiate the tests