Abdallah Nawras
12/05/2020
Portfolio_Final_Project

UI
Home page: https://final-project-dot-hw3-an.uc.r.appspot.com
Login and get token: https://final-project-dot-hw3-an.uc.r.appspot.com/login
Create user: https://final-project-dot-hw3-an.uc.r.appspot.com/signup

NOTE: the home page will redirect to login page if user is not logged in

Data Models
Boat:
- Name: string
- Length: string
- Type: string
- Owner: string

All data fields all required (except owner, the api will handle it)
Sample json payload to create:

```
{
    "name":"test",
    "length": "20",
    "type": "sale boat"
}
```

Load:
- weight : string
- Content: string
- Delivery_date: string
- boat_id : string

All data fields are required (except boat_id)
Sample json payload to create:

```
{
    "weight": "10",
    "content": "fish",
    "delivery_date": "2020-10-5"
}
```

User:
- Username: string
- Password: string

Relationships

- Boat and Load has one-to-many relationship (a load has only one boat, boat has multiple loads)
- Boat and User has one-to-many relationship (a boat has only one user, user has multiple boats)
- Username is used as unique identifier for the user
- When making request, only supply the JWT, everything else is handled by the API

API Endpoints

Users(Public):
- **[GET] /users** => returns list of users, codes: [200, 405, 401]
  Sample response:

```
{
    "count": 3,
    "limit": 5,
    "next": "",
    "previous": "",
    "results": [
        {
            "id": "5728315943944192",
            "name": "user",
            "password":
"sha256$sI5vIUGv$56e3172453b199fa08c47a66a3af70f342caa46126f056c3f4371b92c
88aab8a"
        },
        {
            "id": "5734159817375744",
            "name": "test",
            "password":
"sha256$Nf1aOB9t$2df364ad896ed4f56cc5cb08b2c9effe8c91d6f1aa2d41af1bfe889f7
258a655"
        },
        {
            "id": "5737003958140928",
            "name": "admin",
            "password":
"sha256$LD9Y3uc9$eb1be25bf95823d5f5a1990e002a68a88d6596cc1af3b2c42cba9c9fa
5607455"
        }
```

```json
    ],
    "start": 1
}
```

Boats(Protected)
- **[POST] /boats** => creates a boat, codes: [201, 400, 401, 405, 406]
  payload :

```json
{
    "name":"test",
    "length": "20",
    "type": "sale boat"
}
```

Response:

```json
{
    "id": "5768223068782592",
    "name": "test",
    "type": "sale boat",
    "length": "20",
    "owner": "test",
    "loads": null,
    "self": "http://localhost:5000/boats/5768223068782592"
}
```

- **[GET] /boats** => get all boats for logged in user, codes: [200, 401, 403, 405]
  Response:

```json
{
    "count": 9,
    "limit": 5,
    "next": "/boats?start=6&limit=5",
    "previous": "",
    "results": [
        {
            "id": "5184813602439168",
            "name": "test",
            "type": "sale boat",
            "length": "20",
```

```json
        "owner": "test",
        "loads": null,
        "self": "http://localhost:5000/boats/5184813602439168"
    },
    {

        "id": "5203205155717120",
        "name": "test",
        "type": "sale boat",
        "length": "20",
        "owner": "test",
        "loads": null,
        "self": "http://localhost:5000/boats/5203205155717120"
    },
    {

        "id": "5674591867371520",
        "name": "test",
        "type": "sale boat",
        "length": "20",
        "owner": "test",
        "loads": null,
        "self": "http://localhost:5000/boats/5674591867371520"
    },
    {

        "id": "5694955355897856",
        "name": "test",
        "type": "sale boat",
        "length": "20",
        "owner": "test",
        "loads": null,
        "self": "http://localhost:5000/boats/5694955355897856"
    },
    {

        "id": "5726177821982720",
        "name": "test",
        "type": "sale boat",
        "length": "20",
        "owner": "test",
```

```json
            "loads": null,
            "self": "http://localhost:5000/boats/5726177821982720"
        }
    ],
    "start": 1
}
```

- **[GET] /boats/{id}** => get one boat for logged in user, codes:[200, 401, 403, 404, 405]
Response:

```json
{
    "id": "5695708787113984",
    "name": "test",
    "type": "sale boat",
    "length": "20",
    "owner": "test",
    "loads": [],
    "self": "http://localhost:5000/boats/5695708787113984"
}
```

- **[PATCH] /boats/{id}** => patch boat, codes: [200, 401, 403, 404, 405, 406]
Payload:

```json
{
    "name":"patched",
    "length": 20,
    "type": "sale boat"
}
```

Response:

```json
{
    "id": "5695708787113984",
    "name": "patched",
    "type": "sale boat",
    "length": "20",
    "owner": "test",
    "loads": [],
    "self": "http://localhost:5000/boats/5695708787113984"
}
```

- **[DELETE] /boats/{id}** => deleted a boat, codes: [204, 403, 404, 401, 405]

Response: None

Loads(protected)
- **[POST] /loads** => creates a load, codes: [201, 400, 401, 405, 406]
  Payload:
```json
{
    "weight": "10",
    "content": "fish",
    "delivery_date": "2020-10-5"
}
```

  Response:
```json
{
    "id": "5756247156457472",
    "weight": 10,
    "content": "fish",
    "delivery_date": "2020-10-5",
    "boat_id": null,
    "self": "http://localhost:5000/loads/5756247156457472"
}
```

- **[GET] /loads/{id}** => gets one load, codes [200, 401, 403, 404, 405]
  Response: same as above

- **[GET] /loads** => get list of loads, codes[200, 401, 405]
  Response:
```json
{
    "count": 2,
    "limit": 5,
    "next": "",
    "previous": "",
    "results": [
        {
            "id": "5651072089587712",
            "weight": 10,
```

```
                "content": "fish",
                "delivery_date": "2020-10-5",
                "boat_id": null,
                "self": "http://localhost:5000/loads/5651072089587712"
            },
            {
                "id": "5661864209940480",
                "weight": 10,
                "content": "fish",
                "delivery_date": "2020-10-5",
                "boat_id": null,
                "self": "http://localhost:5000/loads/5661864209940480"
            }
        ],
        "start": 1
    }
```

- **[PATCH] /loads/{id}** => updates a load, codes: [200, 401, 403, 405, 404, 406]
    Payload:

```
{
    "weight": "10",
    "content": "patched",
    "delivery_date": "2020-10-5"
}
```

    Response:
```
{
    "id": "5723707007827968",
    "weight": 10,
    "content": "patched",
    "delivery_date": "2020-10-5",
    "boat_id": null,
    "self": "http://localhost:5000/loads/5723707007827968"
}
```

- **[DELETE] /loads/{id}** => deletes a load, codes: [204, 401, 403, 404, 405]
    Response: none

Relationships(protected)

- **[PUT] /loads/{id}/boats/{id}** => assigns a load to a boat, codes: [200, 401, 403, 404, 405]
    Response:

```json
{
   "id": "5723502418067456",
   "weight": 10,
   "content": "fish",
   "delivery_date": "2020-10-5",
   "boat_id": "5757464578359296",
   "self": "http://localhost:5000/loads/5723502418067456"
}
```

NOTE: when getting the boat now, loads are associated with it

```json
{
   "id": "5757464578359296",
   "name": "test",
   "type": "sale boat",
   "length": "20",
   "owner": "test",
   "loads": [
       "5723502418067456"
   ],
   "self": "http://localhost:5000/boats/5757464578359296"
}
```

- **[DELETE] /loads/{id}/boats/{id}** => unassign load from boat, codes: [200, 401, 403, 404, 405]

      Response:

```json
{
    "id": "5723502418067456",
    "weight": 10,
    "content": "fish",
    "delivery_date": "2020-10-5",
    "boat_id": null,
    "self": "http://localhost:5000/loads/5723502418067456"
}
```