BOSSES..

Small assignment III

Hiroshi Yamauchi has contacted us! He is working on a new **Megaman** game and wants to create a small wiki website for the game. He got stuck in the 8-bit graphic and the technology that comes with it, so he doesn't know anything about concepts such as **React**, **Redux** or even **HTTP**. That's where we come in! We are going to create this website which will be heavily depending on **Redux** along with **React**. Good luck!

Assignment description

Below is an enlisting of all the functionality within the application:

Redux

- All data coming from the server should be stored within the **Redux** store state
- The data fetched from the server should make use of **async action creators**

*The **Redux** section above is mandatory to be graded for this assignment. If the application does not use **Redux** to store data coming from the server, the assignment will not be graded.*

General

- **(10%)** There should be a navigation bar which has the following routable links:
  - ***/*** *- Should route to a welcome site which should contain some information about the website. You can choose what this text will be and how this welcome page should look like*
  - ***/bosses*** *- Should route to a site which lists all bosses*

- **(40%)** Within **/bosses** the user should be able to create a new boss providing the following information: *name, description* and *url*. The url should point to some external image. Each field is required and should be validated as well as providing useful validation message for the client. This should result in a new **Redux** store state as well as updating the server

- **(50%)** Each boss item within **/bosses** should be routable with an id, e.g. **/bosses/:bossId**. Within that site the following should be possible:
  - The boss should be editable (*for all fields*), which should result in a new **Redux** store state as well as updating the server
  - The boss should be deletable, which should result in a new **Redux** store state as well as updating the server

Server

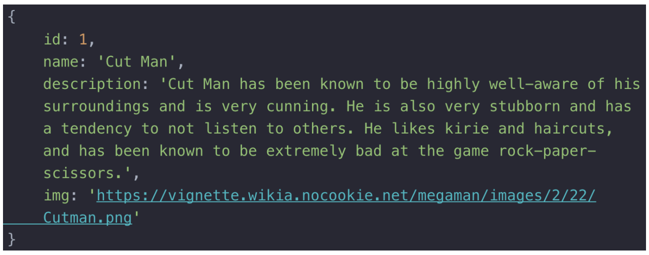
This assignment comes with a server which can be found in the **server/** folder. **This code should not be altered**. In order to run the server you can do the following:

1. In the terminal navigate to the **server/** folder
2. Run **npm install**
3. Run **npm start**
4. The server is now running on http://localhost:4500

This is an API which exposes five endpoints:

- http://localhost:4500/api/bosses **[GET]** - *Gets all bosses*
- http://localhost:4500/api/bosses/:bossId **[GET]** - *Gets a boss by id*
- http://localhost:4500/api/bosses/:bossId **[PATCH]** - *Updates a boss by id*
- http://localhost:4500/api/bosses **[POST]** - *Creates a new boss*
- http://localhost:4500/api/bosses/:bossId **[DELETE]** - *Deletes a boss by id*

The boss object looks like this:



```
{
    id: 1,
    name: 'Cut Man',
    description: 'Cut Man has been known to be highly well-aware of his
    surroundings and is very cunning. He is also very stubborn and has
    a tendency to not listen to others. He likes kirie and haircuts,
    and has been known to be extremely bad at the game rock-paper-
    scissors.',
    img: 'https://vignette.wikia.nocookie.net/megaman/images/2/22/
    Cutman.png'
}
```

Dependencies

All dependencies are allowed for this assignment. Scaffolders such as **create-react-app** are also allowed.

Submission

A single compressed file (***.zip, *.rar**) should be submitted in **Canvas**. If you are working in groups don’t forget to comment the names of each member (*excluding the one submitting*).