missBug
Part 3

missBug is a bug management system, it allows users to add / remove and update bugs, lets continue.

Lets Implement login

Model

The bug should have the following properties:

```json
{
    "_id": "abc123",
    "title": "Cannot save a Car",
    "description": "problem when clicking Save",
    "severity": 3,
    "createdAt": 1542107359454,
    "creator": {
      "_id": "u101",
      "fullname": "Puki Ka"
    }
}
```

Add User Support

Backend

- Add user.json – that holds all the users and a userService
- Add userRoute
    - /api/auth/signup – adds a new user to the file
    - /api/auth/login – check if username and password are correct - generate a loginToken and return a mini-user to the frontend
        - When New Bug – add the creator from the loginToken
        - Only the bug's creator can DELETE/UPDATE a bug
          Update only updatable fields
    - /api/auth/logout – clear the cookie
- Test your API from POSTMAN

Frontend

- Use or Create the component: <login-signup>
- Add a userService
  - Implement the functions: `login`, `signup`, `logout`, `getLoggedinUser`
  - Use the sessionStorage to hold the loggedinUser and survive browser refresh
- Implement two forms for login and signup

- Add a user-details page
  - This is a user profile page
  - Show the user's **bugs** (bugs that he has created)
    - Can you use your bug-list component?
  - At the header, add a Profile link that route to user-details page of the logged-in user.

Implement ownership

  - When adding a new Bug – add the creator from the cookie
  - Only the bug's creator can DELETE/UPDATE a bug
  - Use **postman** to test the APIs

Add Admin Support

- Add isAdmin to the user entity
- Hard-coded mark a user (username: admin, pass: admin) as admin in your user.json file
- Admin can delete / edit all bugs
  - Admin has a link to user-list page where he can view and delete users Prevent deletion of users that own bugs

Deploy to Render

Follow the needed steps to upload your project to Render.com

Set up a SECRET1 environment variable holding the encryption key