Lab 05-04 Great Job Lou! 187.5/200

1. Bugs Screen: Many of the sortBy options are not returning the correct results. -10pts
2. min/max fields allow negative numbers. – 2.5pts

horizontal line

# Overview: Issue Tracker

In this assignment we will continue to develop the front-end for our issue tracker with React.

*This assignment is worth 200 points total (or 2.0 lab grades.)*

# Step 1. Use your existing GitHub Repositories

Use your **issue-tracker-react** repository for the front-end code.

And use your **awd1111-issue-tracker** repository for the back-end code.

**Do not create a new repository!**

# Step 2. Implement features

Follow the directions below to continue building on your project.

# Tasks (200pts total)

The tasks for this lab assignment are summarized below.

* **Implement a ReportBug component to report new bugs.** And make this available via the route **"/bug/report" (20pts)**
* **Implement a search interface for bugs.** This must be available via the **"/bug/list"** route. **(20pts)**
  + See requirements for the API found in the **Lab 04-04.**
  + The search bug interface must include the fields listed below:
    - **keywords** - use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine the keywords <input> and the search <button>
    - **classification -** use a <select> field to pick a single classification. If the user does not select a classification, then all classification must be displayed in the search results.
    - **maxAge** - use an <input> field (with type=number) to enter the maximum age of bugs (measured in days.) This will remove older bugs from the search results.
    - **minAge** - use an <input> field (with type=number) to enter the minimum age of bugs (measured in days.) This will remove newer bugs from the search results.
    - **closed** - use an <input> field (with type=checkbox). This box must be **unchecked** initially. If this box is not checked, then closed bugs must be removed from the search results.
    - **sortBy** - use a <select> field to pick a sort order. The option **"newest"** must be listed first in the dropdown and selected initially. The required options are listed below:
      * value=newest, text=Newest
      * value=oldest, text=Oldest
      * value=title, text=Title
      * value=classification, text=Classification
      * value=assignedTo, text=Assigned To
      * value=createdBy, text=Reported By
* **Implement a search interface for users.** This must be available via the **"/user/list"** route. **(20pts)**
  + See requirements for the API found in the **Lab 04-04.**
  + The search user interface must include the fields listed below:
    - **keywords** - use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine the keywords <input> and the search <button> (note that the InputField component will not work here)
    - **role** - use a <select> field to pick a single role. If the user does not select a role, then all roles must be displayed in the search results.
    - **maxAge** - use an <input> field (with type=number) to enter the maximum age of users (measured in days.) This will remove older users from the search results.
    - **minAge** - use an <input> field (with type=number) to enter the minimum age of users (measured in days.) This will remove newer users from the search results.
    - **sortBy** - use a <select> field to pick a sort order. The option **"givenName"** must be listed first in the dropdown and selected initially. The required options are listed below:
      * value=givenName, text=Given Name
      * value=familyName, text=Family Name
      * value=role, text=Role
      * value=newest, text=Newest
      * value=oldest, text=Oldest
* **Display the list of comments on the bug editor page.** This must be available on the **"/bug/:bugId"** route. **(20pts)**
  + Comments must always be listed from oldest to newest. (Make sure that the backend sorts the comments in this order.)
  + Only comments for the selected bug should be displayed.
* **Add a form to post new comments on the bug editor page.** This must be available on the **"/bug/:bugId"** route. **(20pts)**
  + This form can only contain one input field. This input field is used to enter the text of the comment.
  + Use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine this <input> field with the post <button>.
* **Add a form element to classify bugs on the bug editor page.** This must be available on the **"/bug/:bugId"** route. **(20pts)**
  + Use a <select> field to select the classification. The current classification must be selected initially.
  + Use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine this <select> field with the post <button>.
* **Add a form element to assign bugs on the bug editor page.** This must be available on the **"/bug/:bugId"** route. **(20pts)**
  + Use a <select> field to select a user. The currently assigned user must be selected initially.
  + Use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine this <select> field with the post <button>.
* **Add a form element to close and open on the bug editor page.** This must be available on the **"/bug/:bugId"** route. **(20pts)**
  + Use a <select> field to select between "open" and "closed". The current status must be selected initially.
  + Use a [**Bootstrap Input Group**](https://getbootstrap.com/docs/5.1/forms/input-group/) to combine this <select> field with the post <button>.

***Continued on the next page.***

* **Use React Icons to add some Font Awesome icons to your application. (20pts)**
  + Find the documentation here: <https://react-icons.github.io/react-icons/>
  + And here: <https://react-icons.github.io/react-icons/icons?name=fa>
  + Install the package like so:

|  |
| --- |
| npm install react-icons |

* + Import the icons you want to use:

|  |
| --- |
| import { FaDoorOpen, FaDoorClosed, FaListAlt } from 'react-icons/fa'; |

* + Use an icon in your component:

|  |
| --- |
| <FaDoorOpen className="m-1" /> |