⚾

(Senior) Frontend Engineering Challenge

Context

Our user is an industrial engineer (IE) at an electronic manufacturing service (EMS). She wants to view an overview of all of her open projects. Since there are 100 available she wants a way to sort them so the most relevant ones are displayed first.

One Project consists of:

- Project Name
- Creation Date
- Status: This is either inProgress, won or lost.
- id

User Story Acceptance Criteria:

1. As a User Given that I have 100 Projects then I can see 100 Project cards with data on the name, creation date and status.
2. Each project must be displayed as a card (e.g. https://material-ui.com/components/cards/) and not in a table.
3. As a User, Given that I am on the Page, Then I can see 2 buttons to sort by creation date.
4. As a User, Given that I have selected sort by earliest then I can see the earliest first
5. As a User, Given that I have selected sort by latest then I can see the latest first

The implementation of this user story is attached to this document.

The basic setup is:

1. run `yarn` to install the packages.
2. run `yarn run-server` to run the mock backend. The API route for projects is a GET request: `http://localhost:3004/projects`
3. run `yarn start` to run the frontend.
4. run `yarn test` to run the tests (these should break for now)

API Design

- The call `/projects` returns an object with a `data` property which contains an array of objects of the following type:

```
{
'projectName': string // name of the project
'creationDate': string // date in the form of ISO8601 https://opensource.zalando.com/restful-api-guidelines/#126
'status': string // can either be inProgress, won, lost
'id': string //uuid of the project
  }
```

Submission Details

This challenge consists of three parts:

1. Finding and fixing errors in the code

The first part involves fixing errors in the code when the backend is running from `yarn run-server`. Normally this would be done as a code review, but in the context of this challenge, please fix these errors and make a brief list of what you have changed. You are encouraged to be creative in making the application fit the above acceptance criteria.

> 💡 There are some errors in the backend response. Instead of fixing the errors in the response itself please handle these errors in the frontend.

2. Extending the sorting and filtering functionality

We have received some feedback from the user to extend the functionality to include filtering and searching

1. As a User, Given that I am on the Page, Then I can search by project name
2. As a User, Given that I am on the Page, then I can filter by status
3. As a User, Given that I am on the Page, then I can filter projects between two creation dates.

Please can you add these features as extra commits to your newly fixed application.

3. Identifying and fixing performance problems

1. Cancel `yarn run-server` and instead run `yarn run-server-big` . You will notice of performance problems in the frontend as the API returns thousands of results.
2. Your final task is to ensure that the application is able to handle this large amount of incoming data so it does no have any performance problems.

Good Luck and let us know if you have any questions! 😌