

Requirements for Frontend Developer

Important notes: The project has already set up Redux with redux-saga and axios. FEEL FREE to change those dependencies for something that you are more comfortable with.

New Features:

FET-001 “Details page for a single beer”:

Description:

As a user I want to see the details of a single beer in a different page of the list.

This detail page should show at least the information of:

- Name
- Description
- First Brewed
- Image
- Foods that pair with that beer
- Abv and Ibu
- Contributor/s

It also has to show the difference between Original Gravity and Final gravity (target_og and target_fg params).

Requirements:

When you click on an item on the list, you’ll be sent to a new page with the details specified.

You should be able to go back from that page to the list.

You should be able to refresh the page and still see the details.

Details page will have a button. When you click on it the difference between the two gravities will be displayed.

FET-002 “Search by name”

Description:

As a user I want to search by name of the beer. In order to achieve that there will be an input to do that search on top of the list.

Requirements:

When I type something on the input, I'll get the beers that contains that string. If I delete the input, it will fetch all beers again.

FET-003 “Pagination”

Description:

As a user I want to be able to see more beers than the initials ones. Therefore, there should be some method to paginate the list.

Requirements:

If I see 0..n items, in the next page I want to see the n+1..2n items.

**Number of items per page is up to the developer.*

**The method of pagination will be on the developer. It can be with scroll detection or other ways. Infinite scroll will be a plus.*

Requirements for Fullstack Developer

If you are applying for a Fullstack position you will have to provide an ExpressJs app that accomplish next requirements:

Task 2 will be not necessary for fullstack test

Frontend will communicate only with your Express app and then you will do everything there. Searching, managing pagination, etc.

- You can't call to Beers API but on initializing to fetch all beers and keep them on a mongoDB (you could have a json preloaded to avoid this call).
- You'll have to provide an endpoint for pagination with “no params”. Everything should be managed inside the backend. Something like “/beers/nextPage” and the response has everything that the frontend might need. (Naming is up to candidate).