

 README.md

BrewDogChallenge - Dermot Cooney



Required NodeJS version 8.2.1. [NodeJS](#)

This project was generated with [Angular CLI](#) version 6.1.5.

Installing Dependencies

Run `npm install` to install all main and dev dependencies required for running the application and unit tests.

Additionally run `npm install -g @angular/cli` (This is needed to build the project if you wish to use ng instead of npm).

Build

Run `npm run build` to build the project. The build artifacts will be stored in the `dist/` directory.

Development server

Run `npm start` for a dev server. Navigate to `http://localhost:4200/`. The app will automatically reload if you change any of the source files.

Running unit tests

Run `npm test` to execute the unit tests.

Architectural Decisions

- Choice of Angular:** I selected Angular over the use of ReactJS as I have significantly more Angular experience and I didn't believe I'd build up enough React knowledge to complete this in under 4 days.
- Sorting of Beer Lister data by name:** This is due to ease of use for the end user.
- Use of flattened data structures:** I found it simpler to maintain and manipulate list item data (beer, ingredients and method) using an array of flattened objects. This decision was made when building the ingredients list. I opted to sort this list such that unsorted items that did not depend on other ingredients to be added at an earlier stage came first (i.e. malt, then hops with `add=start`, `add=middle` and finally `add=end` ordering). The purpose of this ordering was to provide the user with a sorted list to avoid any confusion in the process.
- Encapsulated functionality:** This allowed for simplifying the html of the two main pages by creating separate components for specific elements (e.g. beer-details contains beer-detail-card, ingredient-detail-card and method-detail components).

Known Issues

- Styling needs improvement:** As I come from an experienced backend background my CSS knowledge is not advanced (I focused more on the javascript end), there may be some issues with the responsiveness of the UI on smaller displays (<720p, mobile devices more than 4 years old).

2. **Beer Lister:** Currently, the landing page displays all beers returned from the GET beers API. This will become a performance issue when the list numbers in hundreds. Due to available time, I did not prioritise the creation of an infinity scroll or a "more" button.
3. **Error handling:** I did not have time to complete an error handling implementation to inform the user when an API call fails. If there was this would take the form of a dialog box on the beer lister and beer details page instead of what is rendered on success.

Dependencies

All project dependencies are listed in the package.json. No additional dependencies outside Angular and Rxjs were used here.

Other considered features

1. A dropdown to enable the user to change the sort of the beer lister.

Notes

1. I set the interval multiplier on the countdown timer to count once per second for illustration purposes. This can be changed by editing the seconds input in step-complete.component e.g a value of 60 will count down once per minute, etc.

Assumptions

1. As no units of measurement were specified on duration for mashing, I assume these are in minutes.