Beast test assignment

Since the project shold contain both frontend and backend part I decided to use monorepo approach which eases development, local startup and (theoretically) deployment.

Along with that docker compose configuration was created to allow startup with a single command and ensure stability across any host.

Backend

Backend part is a simple NestJS application using postgreSQL as a relational database with a few endpoints serving described business logic.

Only one table is used to represent boats that are present in the system. This is minimalistic representation which can be extended according to the needs. For example statuses can be represented as a different relation (if more data about them is needed) and connected with boats utilizing one-to-one relation.

Frontend

Styling was not highlighted as an important part of the assignment so it was decided to go with a material-ui component's library which might not provide enough flexibility in styling and creating custom components therefore provides ready-to-use blocks with extensive APIs which are good looking and easy to use.

Context API is used for handling global notifications which are used both for error handling and for notifying user about successful events.

Though inline styling is not the most concise way of styling React components it was used in this case due to reasons stated in the first paragraph of the section. If styling had to be extended to the degree where it becomes necesssary to get rid of inline definitions i would switch to styled-components of TailwindCSS.