Development Scenario – Product Warehouse

Technologies:

Type Script/JavaScript

NodeJS

ReactJS

GraphQL – Apollo server

PostgreSQL

Rest API

Abstract

A customer has a number of warehouses. They wish to track the movement of stock in and out of each warehouse. A warehouse has a particular size, this represents the maximum stock amount allowed in this warehouse. Each warehouse is stocked with products. Products have a size per unit amount. Products can be imported, or exported from a warehouse at which point the product, amount and date are recorded. The customer has hazardous and non-hazardous products. It is very important that hazardous products are not kept in the same warehouse as non-hazardous products. Imports can be in the future or the past. The customer will want to see what the current stock level is in a warehouse and what stock space is remaining.

Requirement

Build a small stock management application, consisting of two pages. It must be written in ReactJS consuming GraphQL Api (Apollo Server). The GraphQL Api should call another REST Api only for calculation operations (you can find one open source or create one by yourself, your choice).The data should be stored in PostgreSQL. The applications should be written in TypeScript or JavaScript.

Screen 1;

Product entry screen. Allows the user to quickly add products to the master products list and see a full list of the product list.

Screen 2;

Warehouse stock movement screen. A user can switch between warehouses within the page. The page must show the current stock amount, free stock space remaining. The user can see a historic list of Imports and exports as well as add a new import or export.

Note

This is an open scenario designed to test your knowledge and use of both front end and back end technologies. The requirements are simplistic in order for you to use some imagination. Assume that you already have a data structure containing your list of warehouses. Nice things to see within the code are as follows; Design Patterns, Validation, Responsive layout, Architectural design, Relational Database structure. Front end styling libraries are your choice although standard bootstrap is perfectly fine.