Warehouse Management

About:

Warehouse management is application use to create, store, and search products with product location(box).

Technology Stack:

1. Front End (warehouse-management)
    - Angular CLI Angular cli version 8.3.19

2. Back End (warehouse-service)
    - Java 8
    - Spring Boot 2.5.8
    - Apache Maven 3.3.9
    - H2 Database
    - Swagger - 2.9.2
    - Junit
    - Docker

Technology Justification:

***Angular Framework**:*

Angular is an open-source front-end framework developed by Google for creating dynamic modern web apps.

Using angular can achieve the high levels of consistency of the application.

It's also easier for the new developer to pick and understanding the structure of the app.

It is easy to maintain a large code base, thereby it reduces cost and improve the efficiency.

Spring Boot:

Spring Boot framework can quickly build application without worrying about the configurations.

Easy to autoconfigure all components for a production-grade Spring app.

Avoid all the manual work of writing boilerplate code, annotations, and complex XML configurations.

Integration of Spring Boot with the Spring ecosystem which includes Spring Data, Spring Security, Spring ORM, and Spring JDBC is quick.

Eases the dependency and comes with Embedded Servlet Container.

Java:

Java is an Object-Oriented and a general-purpose programming language that helps to create programs and applications on any platform. Java comes up with a bundle of advantages that lets you stick with it like (Platform independent, Supports multithreading etc.)

Swagger:

Swagger provides a platform to document, test and even write the structure of the API in a very convenient way.

Architecture Approach:

Backend for Front end architecture approach used to implement the warehouse management application.

Multiple frontend application interfaces can call their respective BFF backends in parallel, and dedicated backend services can respond faster.

Following BFF architecture reduces the time to make modifications and enhancements in backend systems with dedicated teams working on the upgrades.

The BFF layer in the overall system architecture can benefit from hiding sensitive or unnecessary data before transferring it to the frontend application interface, which helps simplify the system.

BFF backend systems can use any protocol like FTP, SOAP, Rest etc.

Application Functional flow:

Create Box Endpoint:

- User can create the box by providing Box name and capacity of the box.
- Creation of duplication of box not permitted.

Get Available box Endpoint:

- User can get the available box's only when available capacity of the greater that zero.

Add Product Endpoint:

- User can be able to add the product only to the available capacity boxes.
- User can add same product to the multiple boxes.
- User cannot add the product to the box if the available capacity is zero.

Search Product Endpoint:

- User can search the product, by product name by complete name or few characters or by free text.
- Applications shows all the location of the product

Endpoint Response status:

- ok – Success status code
- ko – failure status code

Note:

To view the application request UUID and input request update the log level to debug in the log-back.xml (Backend Application)