Specification Document for The Project

“SportyShoes eCommerce Portal”

Phase3: PG FSD Implement Frameworks t

Simplilearn / CalTech

MARCH 2022 COHORT

Prepared By

Bakau Onafuwa

August 27th, 2022

|  |  |
| --- | --- |
| Name | Bakau Onafuwa |
| Email | Bakau.onafuwa@softgineer.com |
| GitHub Repository | [git@github.com:homozapien/sportyshoes.git](mailto:git@github.com:homozapien/sportyshoes.git)  <https://github.com/homozapien/sportyshoes.git> |
| Project Management | Agile |
| Agile Methodologies | SCRUM |

**Objective**

The main objective of this assessment project is to develop a prototype e-Commerce web application to avail customers the opportunity to buy sporting shoes.

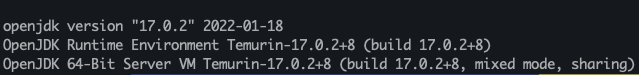
For the prospective customer to be able to achieve this, an administrative account will provide the necessary backend data entities to provide the necessary domain models for the customer interactions.

Notably, this web application was developed to fulfil the core business requirements with additional features (e.g. reporting dashboard, filterable reports etc) to make the usage more intuitive and realistic.

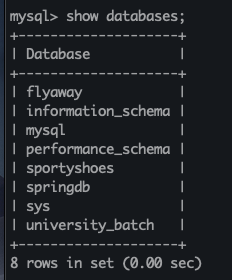
Core development concepts like exception handling, code reviews, code refactoring, versioning, and SCRUM framework are equally showcased.

**Assumptions**

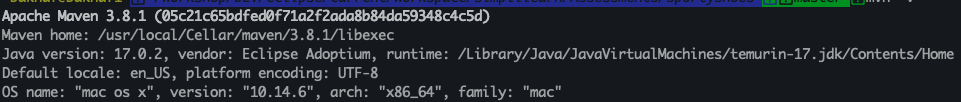
1. It is assumed that the reviewer of the source codes, the compiler, the tester and/or runner of this application is conversant with the Java Technologies and possess some level of programing background principally SpringBoots, JSPs, JSTL.
2. This application was developed in Eclipse IDE and based on JDK 17 (Temurin); it is assumed that the same environment will be available for the review and testing of this application.



1. It is assumed that the code reviewer and the tester of this application are familiar with Git and can clone the Github repository for this project.
2. The development of this application never focused on the security and boundary contexts of the underlying OS and the development database. A simple “localhost” connection to the underlying database will suffix (refer to the application.properties).
3. A Mysql RDBMS will be available in the test environment and database named: sportyshoes is created *ab initio* as shown thus,



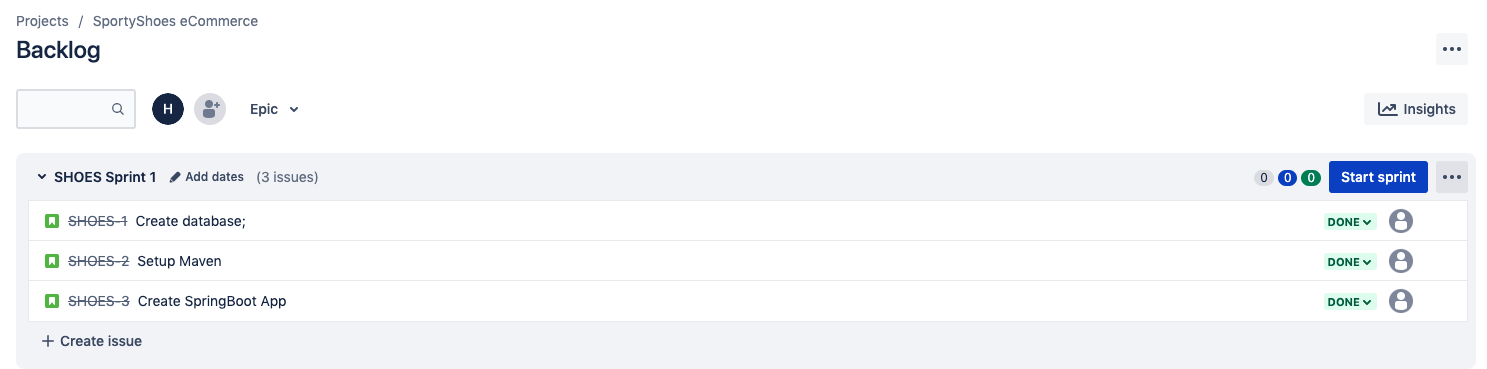
1. A maven infrastructure is required to be able to re-build / re-package the application as shown thus,



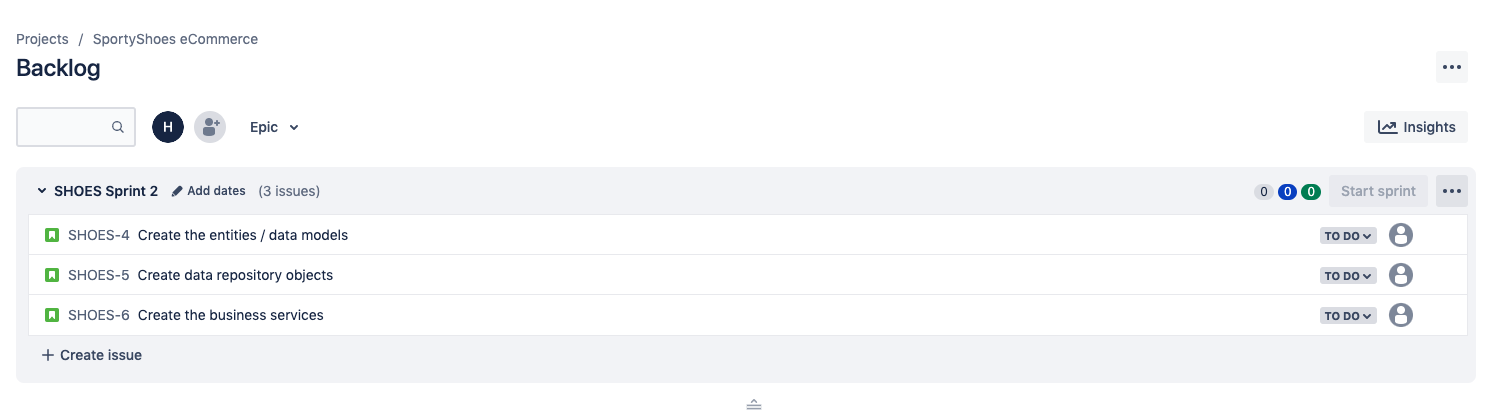
**Project Management**

This application was developed within a period of 4 weeks. I had employed the SCRUM framework for an iterative development. The user stories in the Product Backlog are high-level (without task break down for the individual story) and were implemented in four iterations as shown below.

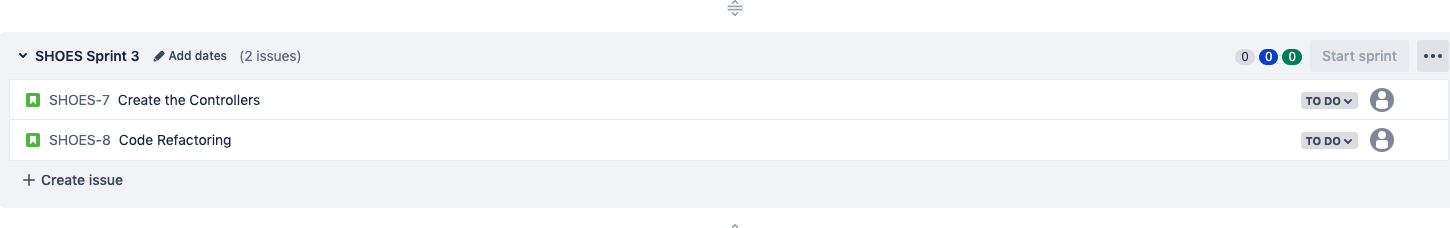
Sprint 1 Goal: Get the environment and infrastructure ready



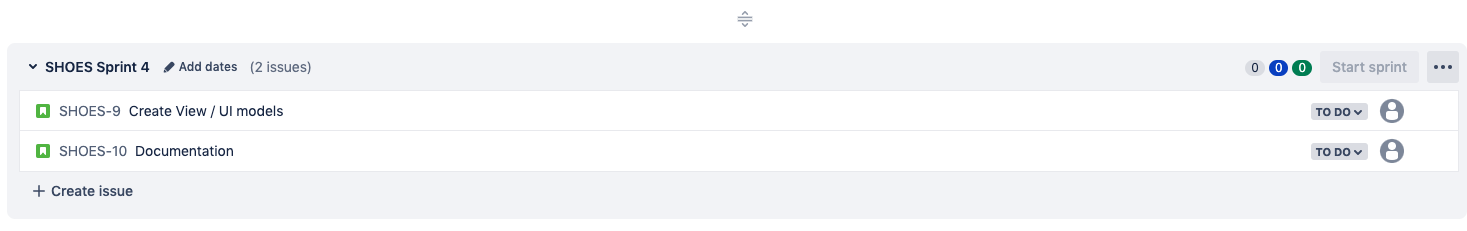
Sprint 2 Goal: Build Application Framework / Model Entities



Sprint 3 Goal: Controllers / Business Logic

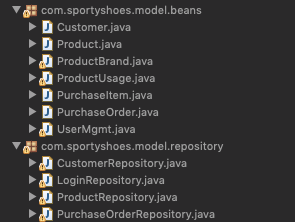


Sprint 4: Create the UIs model / Documentation

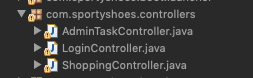


**Project Structure**

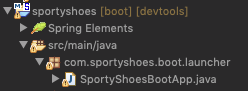
1. Backend: The backend models comprises of entities, Spring JPand MySQL database for storage.



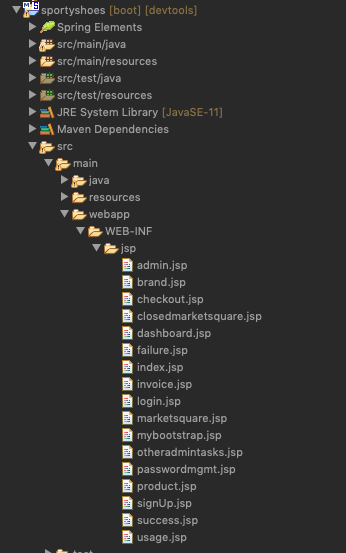
1. Business Logic: The logic is implemented using the Spring Controllers



1. Spring Boot App



1. Front end: The views are implemented with JSP with some bootstrap inclusion.



**Execution Flow**

This is a web-based application and there is some degree of free typing in some of the UI controls however, these view inputs are validated against the relationships amongst the backing entity beans.

Two types of user roles are allowed during authentication viz:

* Admin: to perform the setup and admin functionalities.
* Customer: use the application based off the setup by the admin.

To use the application, the initial default data for some of the entities are created by the Spring Boot App via the CommandLineRunner interface e.g. the UserMgmgt entity is used to setup the admin username and password as shown here thus:



Initial Admin username is: [a@a.com](mailto:a@a.com)

Initial Password: 1

Type of User: Admin

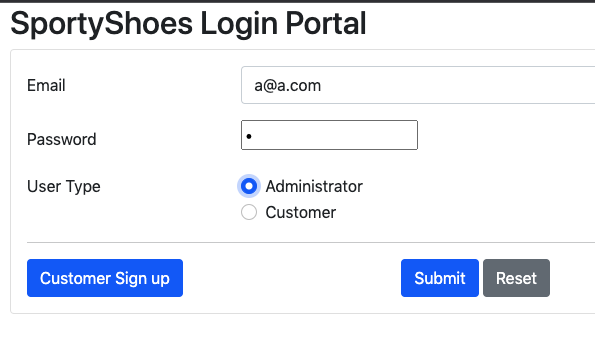
**Data Structure & Algorithms**

The business logic and the manipulations of the entities rely heavily on the collection framework and streaming.

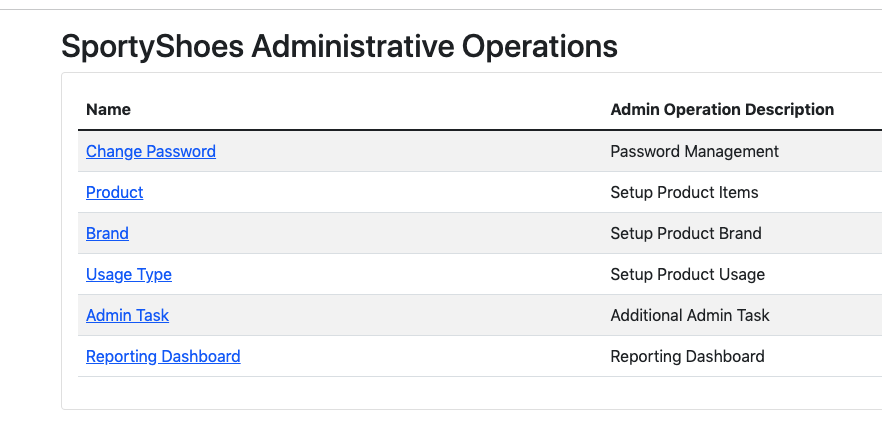
**Unit Testing**

To successfully compile and run this program, minimum of JDK 1.8 must be running in the test environment.

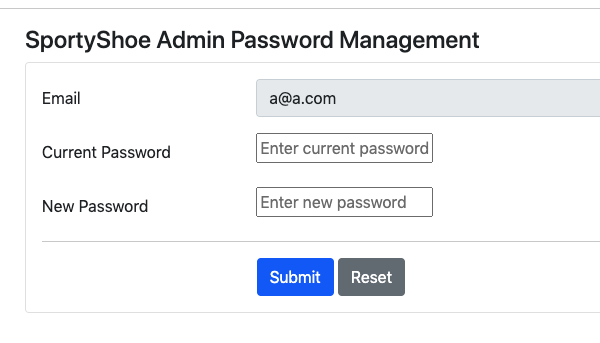
1. Login Page



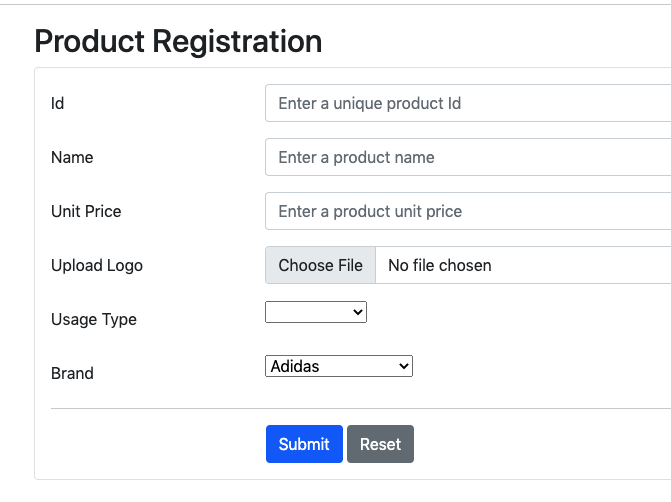
1. Admin Operation



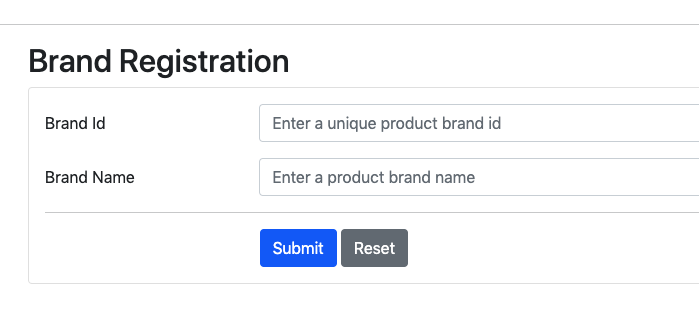
1. Password Change Management



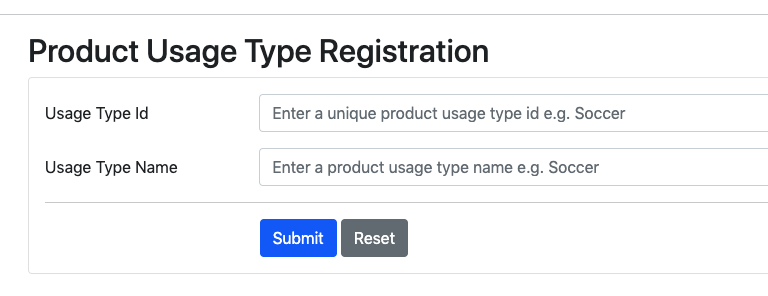
1. Product Setup



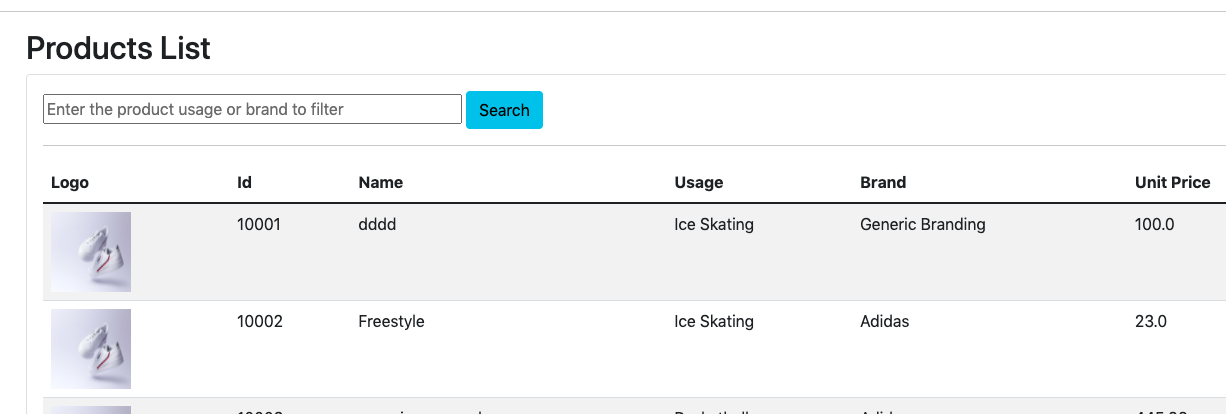
1. Product Brand Setup

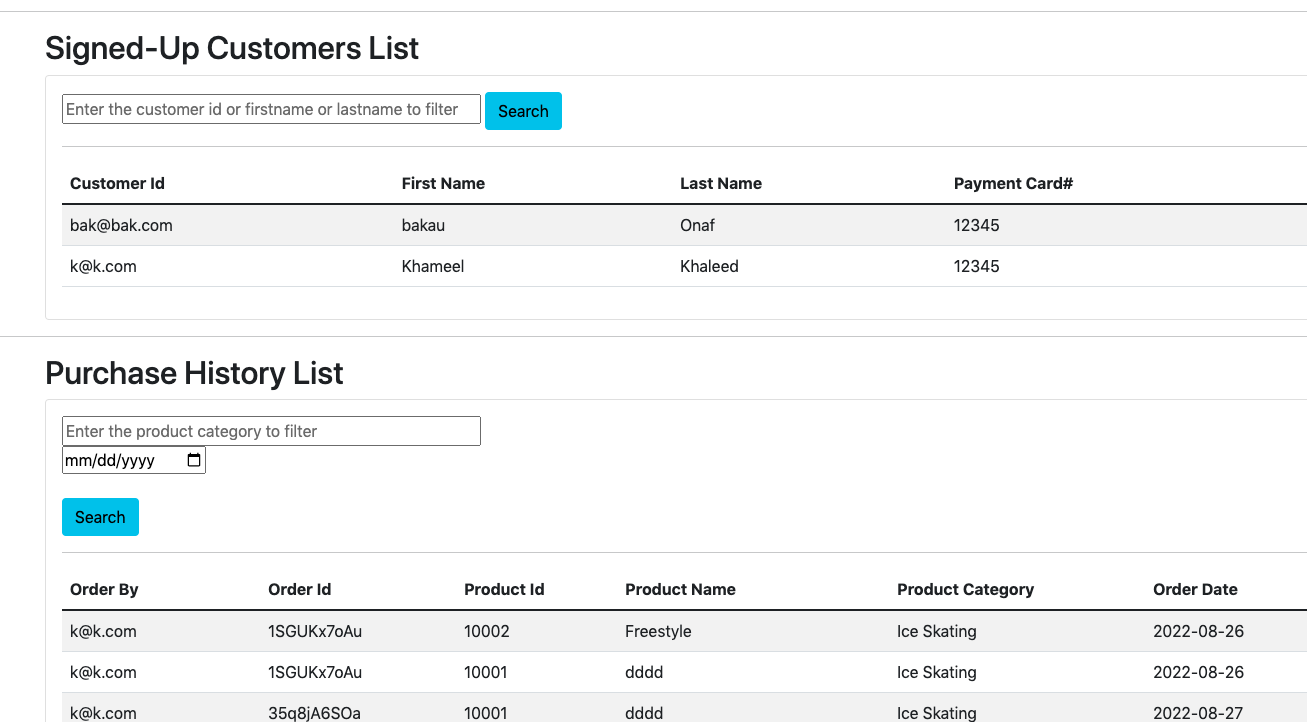


1. Product Usage Type

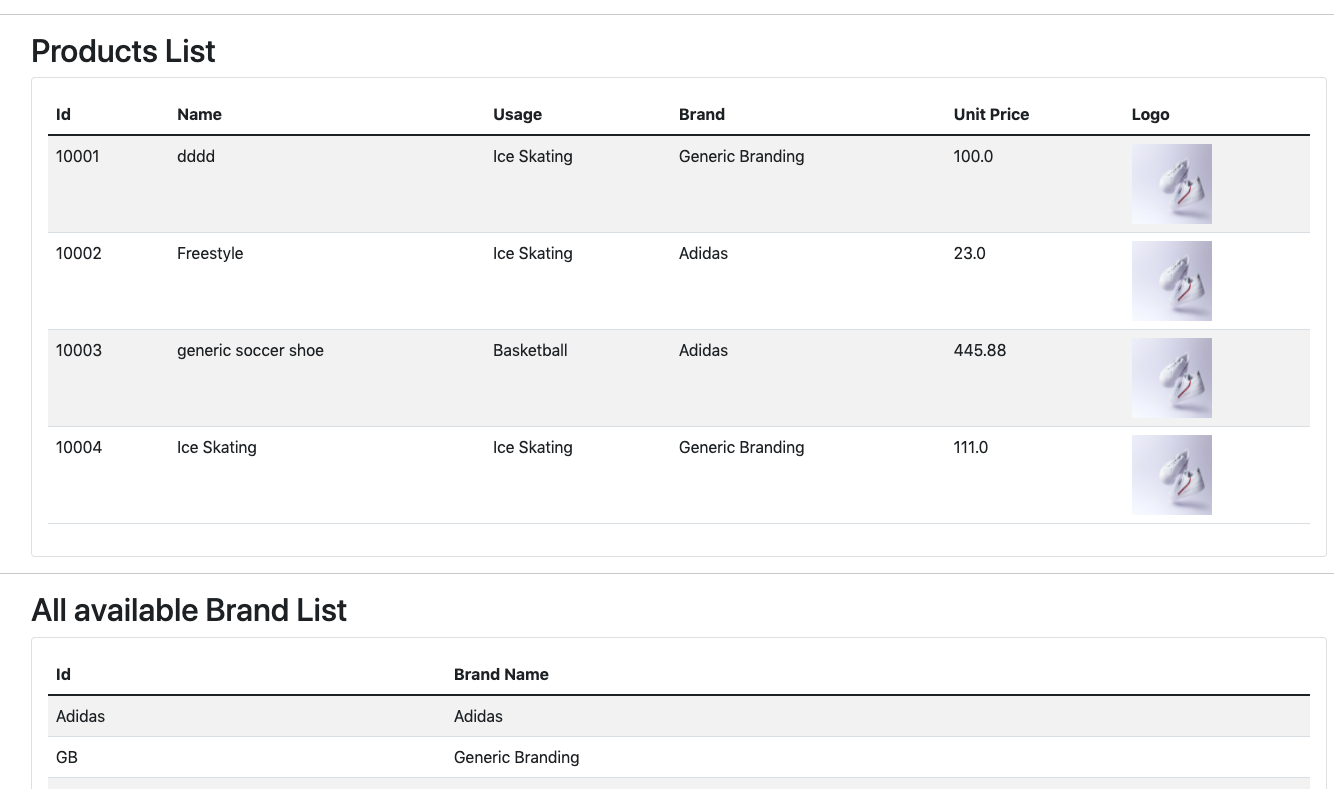


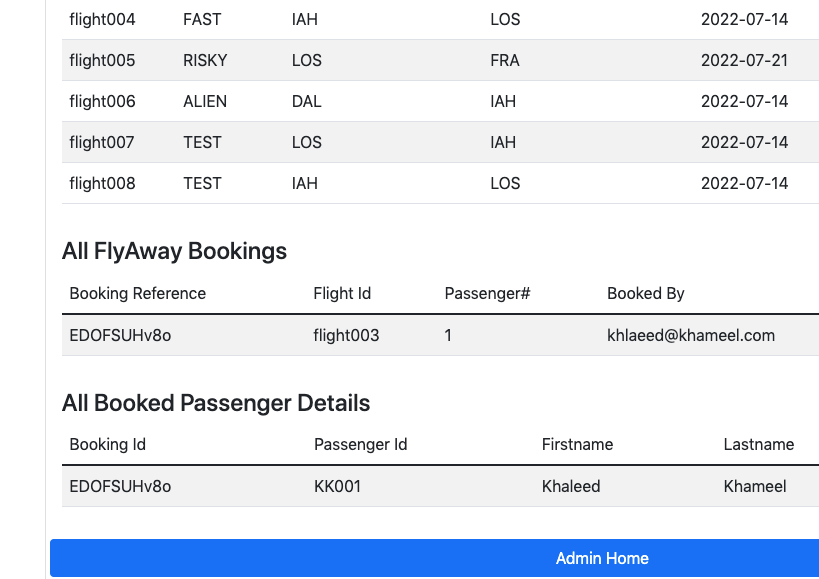
1. Admin Report with Filtering



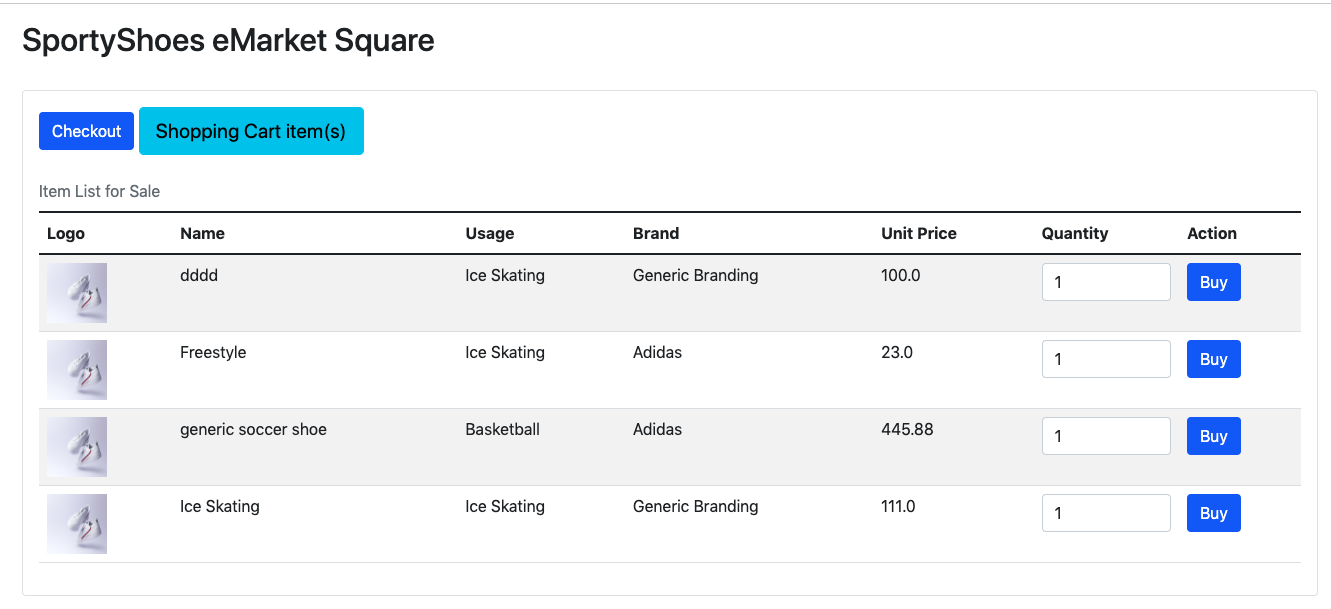


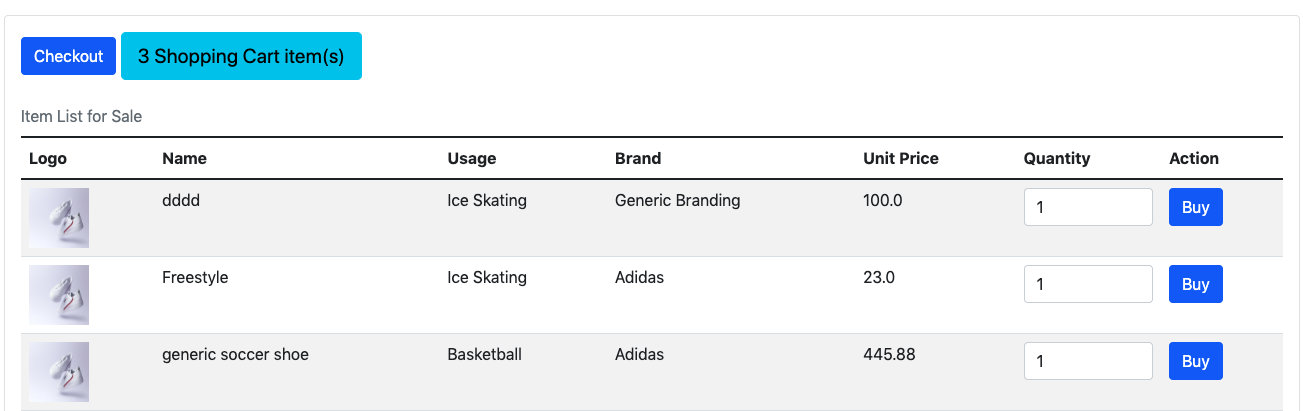
1. Admin Report Dashboard Readonly



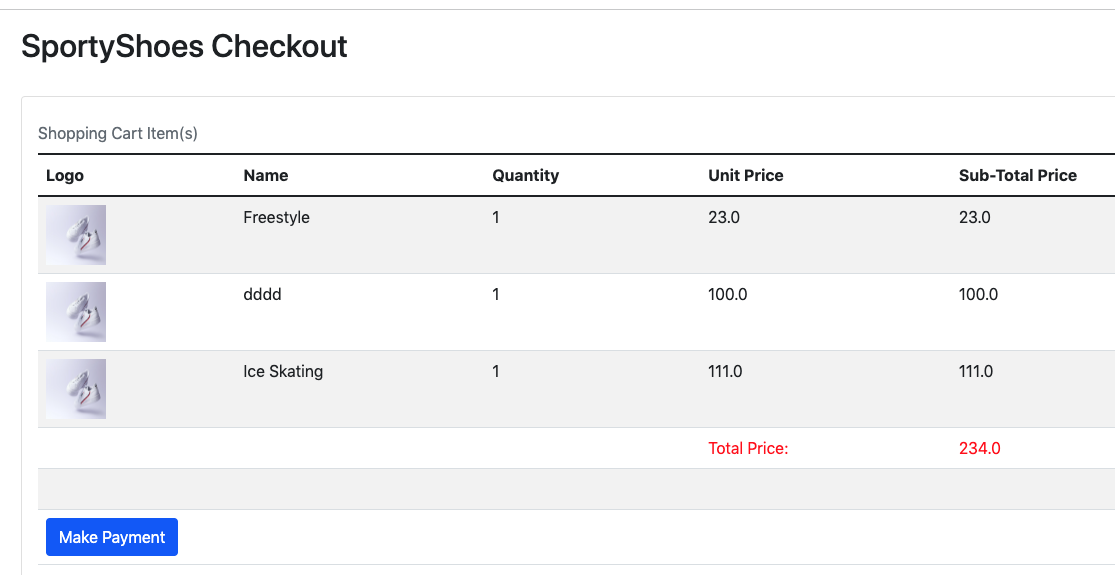


1. Customer Shopping





1. Customer Checkout / Payment



1. Customer Invoice

