Development of a Full Stack ecommerce Web Application using Spring Boot, React, Hibernate and MySQL

GNEY PATEL
211860131

INTRODUCTION

Today, I'll be giving you a brief introduction to the development of a full-stack e-commerce web application using Spring Boot, React, Hibernate, and MySQL. In the world of online shopping, creating a seamless and user-friendly platform is crucial for success.

By leveraging these powerful technologies, we can build an efficient and scalable e-commerce application. In this presentation, we will explore the key components and their roles in the development process.

Spring Boot will serve as the backend framework, React will handle the frontend interface, Hibernate will handle the object-relational mapping, and MySQL will serve as the database management system. By combining these technologies, we can create a comprehensive solution that includes user registration, product catalog management, shopping cart functionality, order management, and much more.

So, let's dive into the details and discover how these technologies work together to build a high-performance e-commerce web application.

OVERVIEW OF TECHNOLOGY’S

1.Spring Boot: Spring Boot is a powerful Java framework that simplifies the development of web applications. It provides a comprehensive set of features, such as dependency management, auto-configuration, and a built-in web server. We will leverage the Spring Boot framework to create a scalable and modular backend for our e-commerce application.

2.React: React is a JavaScript library for building user interfaces. It enables us to create reusable UI components and efficiently update the user interface in response to data changes. With React, we can build a responsive and interactive front-end for our e-commerce application.

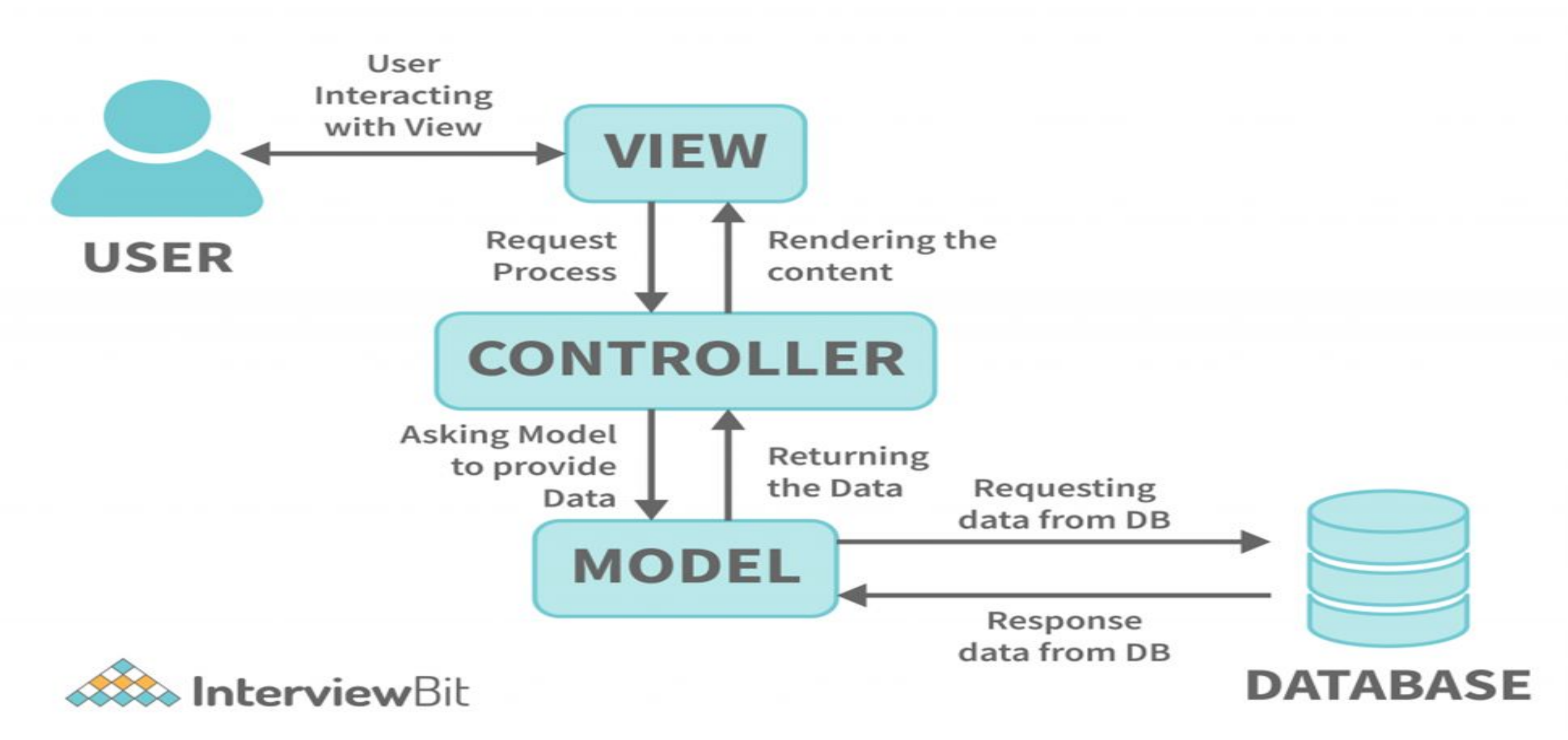
3.Hibernate: Hibernate is an object-relational mapping (ORM) framework for Java. It simplifies database operations by mapping Java objects to database tables. Hibernate handles the complexity of data persistence, allowing us to focus on the application's business logic. We will use Hibernate to integrate our application with the MySQL database.

4.MySQL: MySQL is a popular open-source relational database management system. It provides a robust and scalable platform for storing and retrieving data. We will utilize MySQL to store product information, user data, and other relevant information for our e-commerce application.

Overview Of Software

- User registration and authentication: Allow users to create accounts, log in securely, and manage their profiles.
- Product catalog: Display a wide range of products with relevant details, including images, descriptions, and pricing information.
- Shopping cart: Enable users to add products to their carts, update quantities, and proceed to checkout.
- Order management: Implement features like order history, order tracking, and payment processing.
- Admin panel: Provide an administrative interface for managing products, inventory, and user accounts.
- Search functionality: Implement a search feature that allows users to find products based on keywords or categories.
- Responsive design: Ensure the application works seamlessly across various devices, including desktops, tablets, and mobile phones.

ARCHITECTURE



WORKING

- The user open up the web site and click on register button and user will get message “successfully registered”.(as soon as user hit the register button React application direct to backend part and user controller will handle the event).
- Then the user after registering his data into database the home screen appears.
- Where user can search the product by his category or random product which he wants to purchase.
- After clicking on the product that particular product will be added into the cart item .(backend:- as soon as cart item button hit the product id will go to cart controller and that id product will fetched from product class and then it will be added to cart new id)
- After reviewing the cart item there is an option for checkout

- As soon as the user clicks on checkout the payment page will appear in this project I have implement typed payment system (dummy) but [as user provide the card details he will get the message on screen successfully placed the order and he will get random order number on screen] card details and order number will be stored in database.
- But in terms of cash checkout option then he will get successfully placed the order and he will get random order number on screen

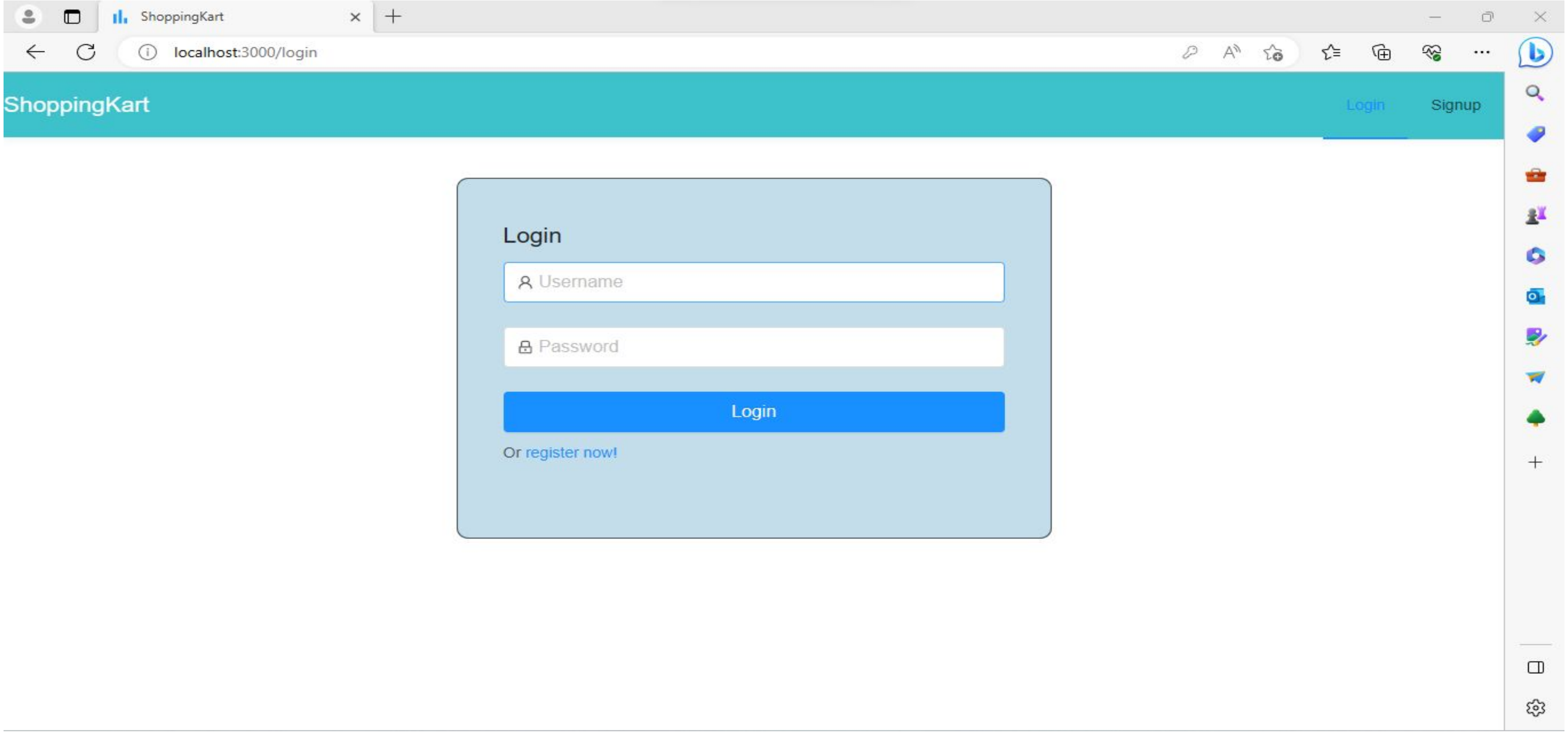
Software Requirements

- Spring STS Eclipse(backend)
- Visual Studio Code(front end)
- Npm(to run react js code)
- Postman(API testing)
- MySQL work bench or community server service

System requirements

- Intel core i3 or above processor
- Memory:- minimum 4gb ram
- Os:- any platform just required the internet
- Tools and framework used:- React,Spring ,Hibernate,API

UI



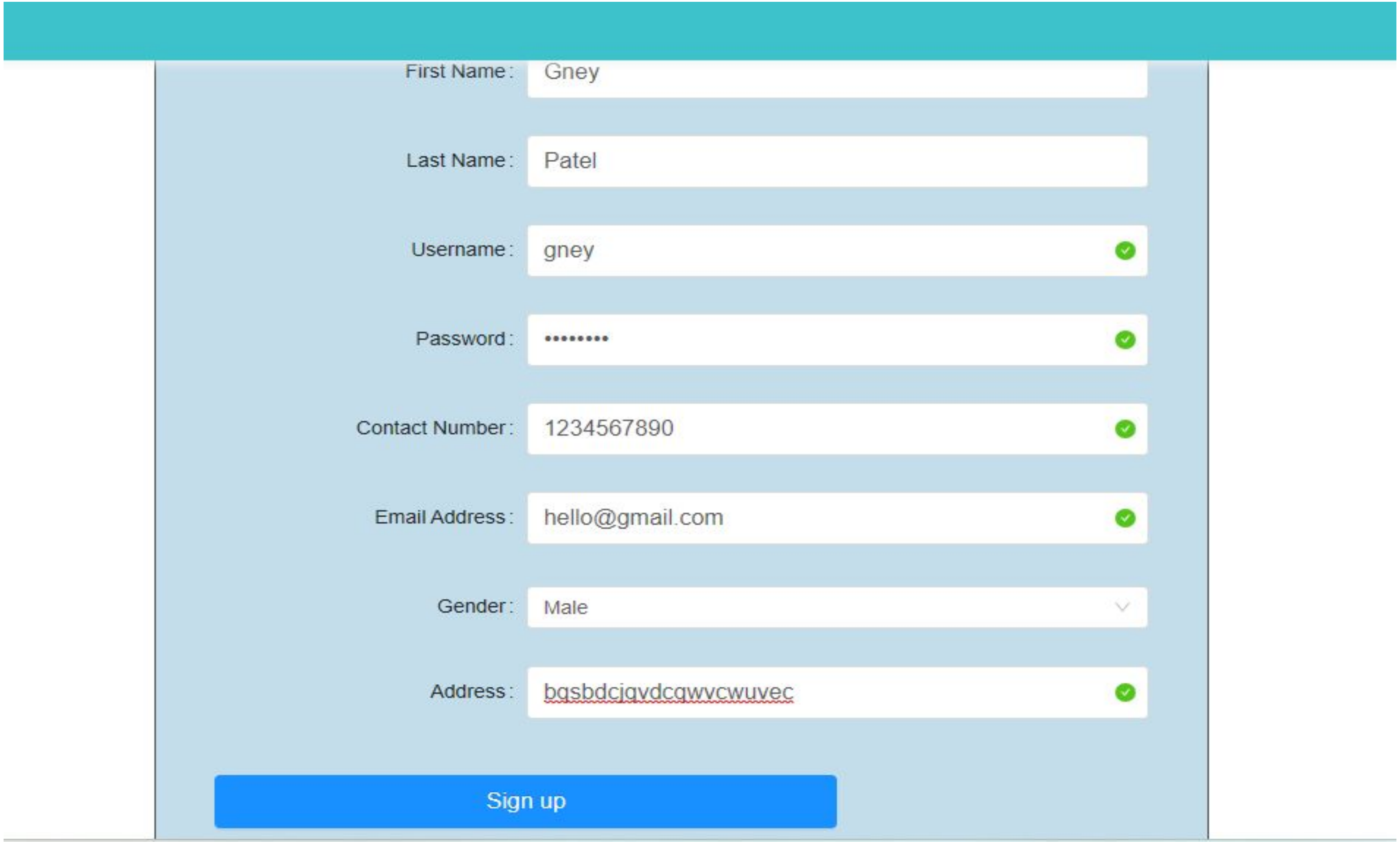
Login

Username

Password

Login

Or register now!



First Name : Gney

Last Name : Patel

Username : gney ✓

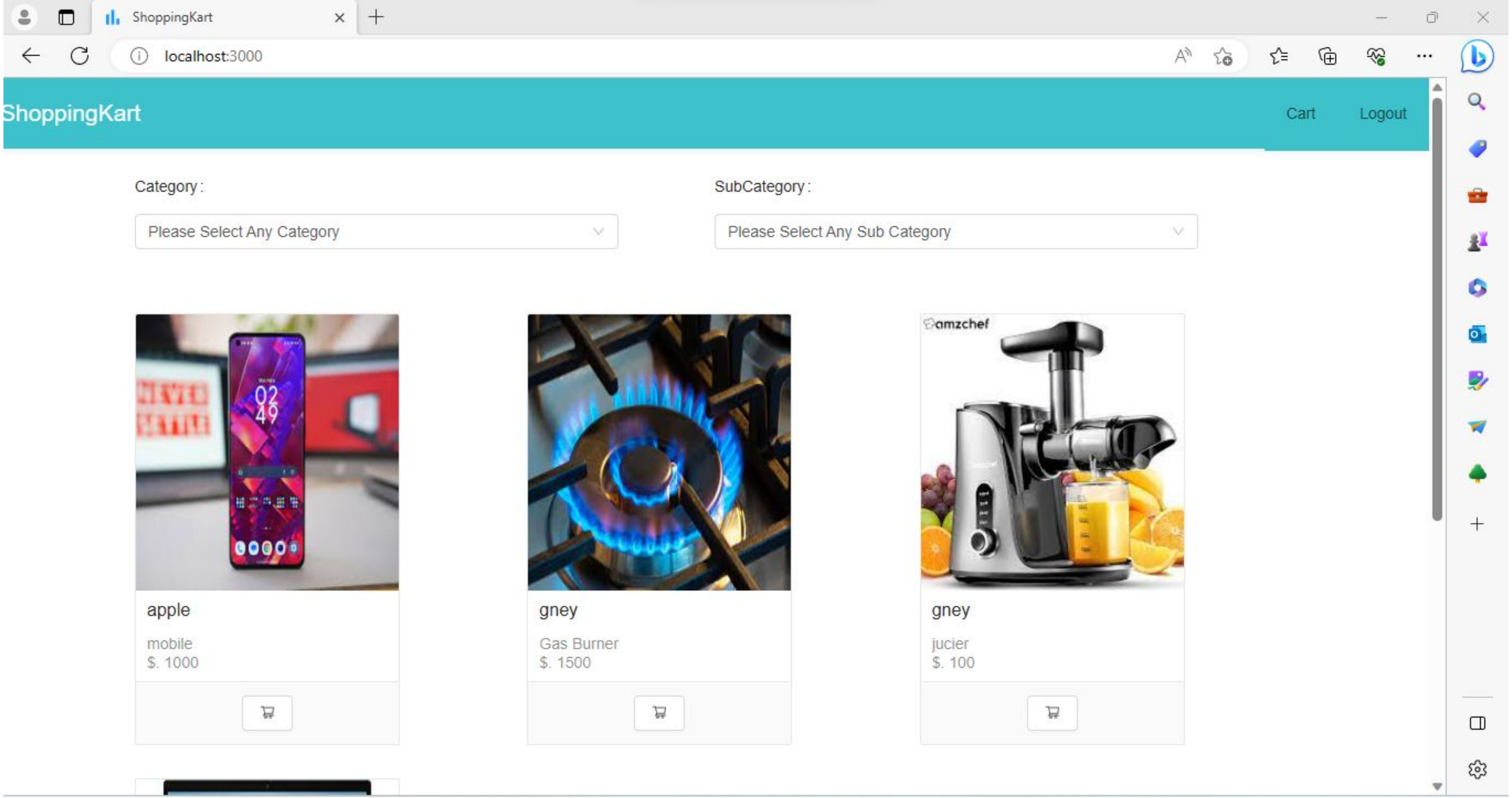
Password : ••••••••• ✓

Contact Number : 1234567890 ✓

Email Address : hello@gmail.com ✓

Gender : Male

Address : bqsbdcjqvdcqwvcwuvec ✓



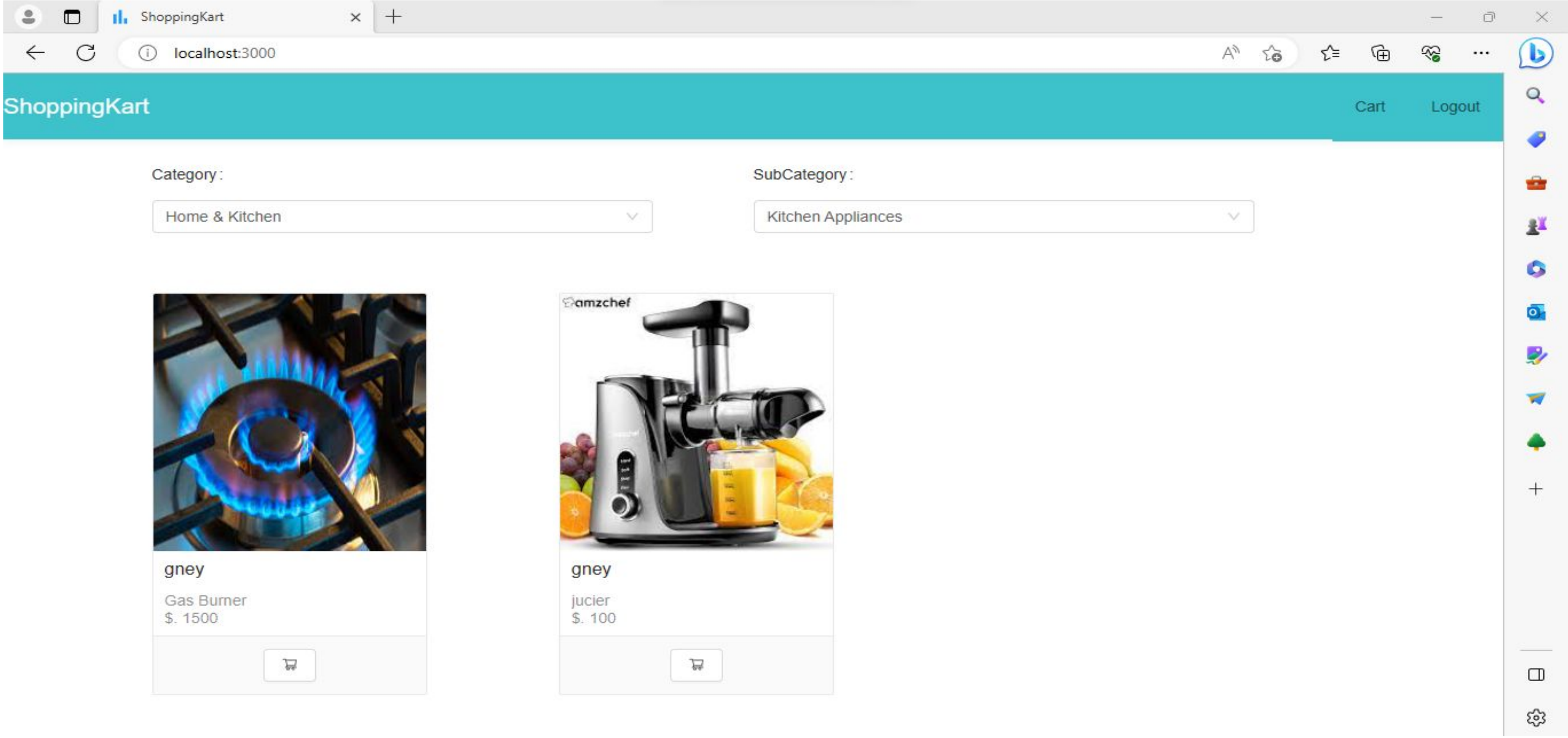
Category : Please Select Any Category

SubCategory : Please Select Any Sub Category

apple
mobile
$. 1000

gney
Gas Burner
$. 1500

gney
jucier
$. 100



Category : Home & Kitchen

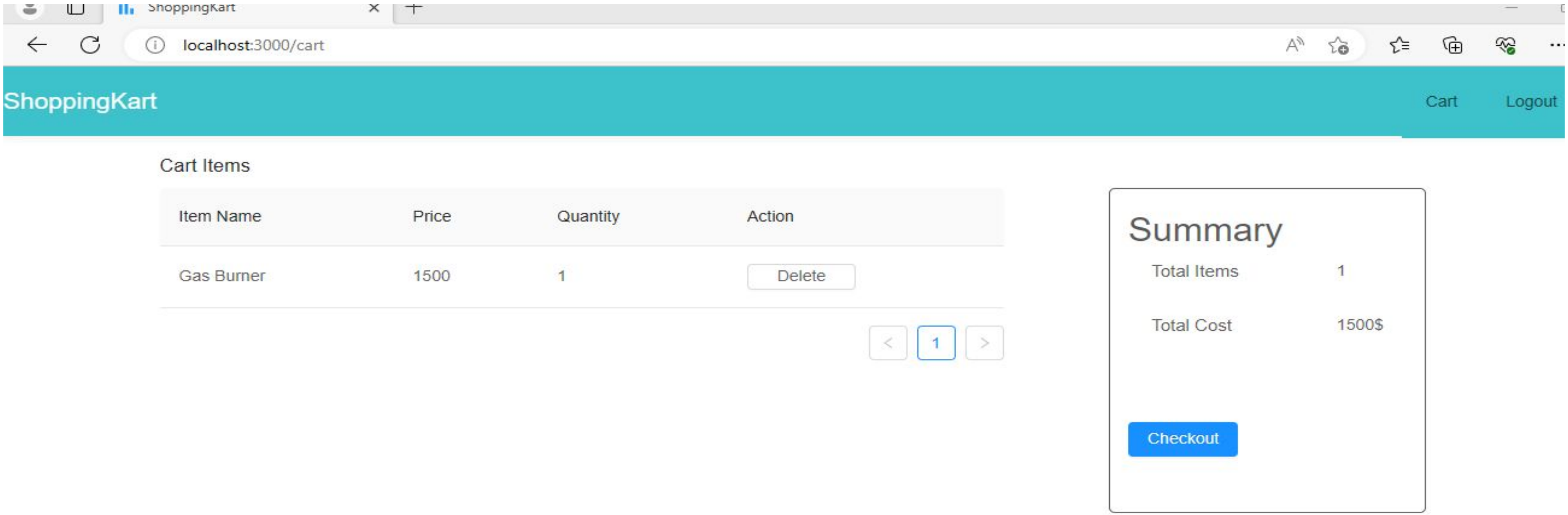
SubCategory : Kitchen Appliances

gney

Gas Burner
$. 1500

gney

jucier
$. 100

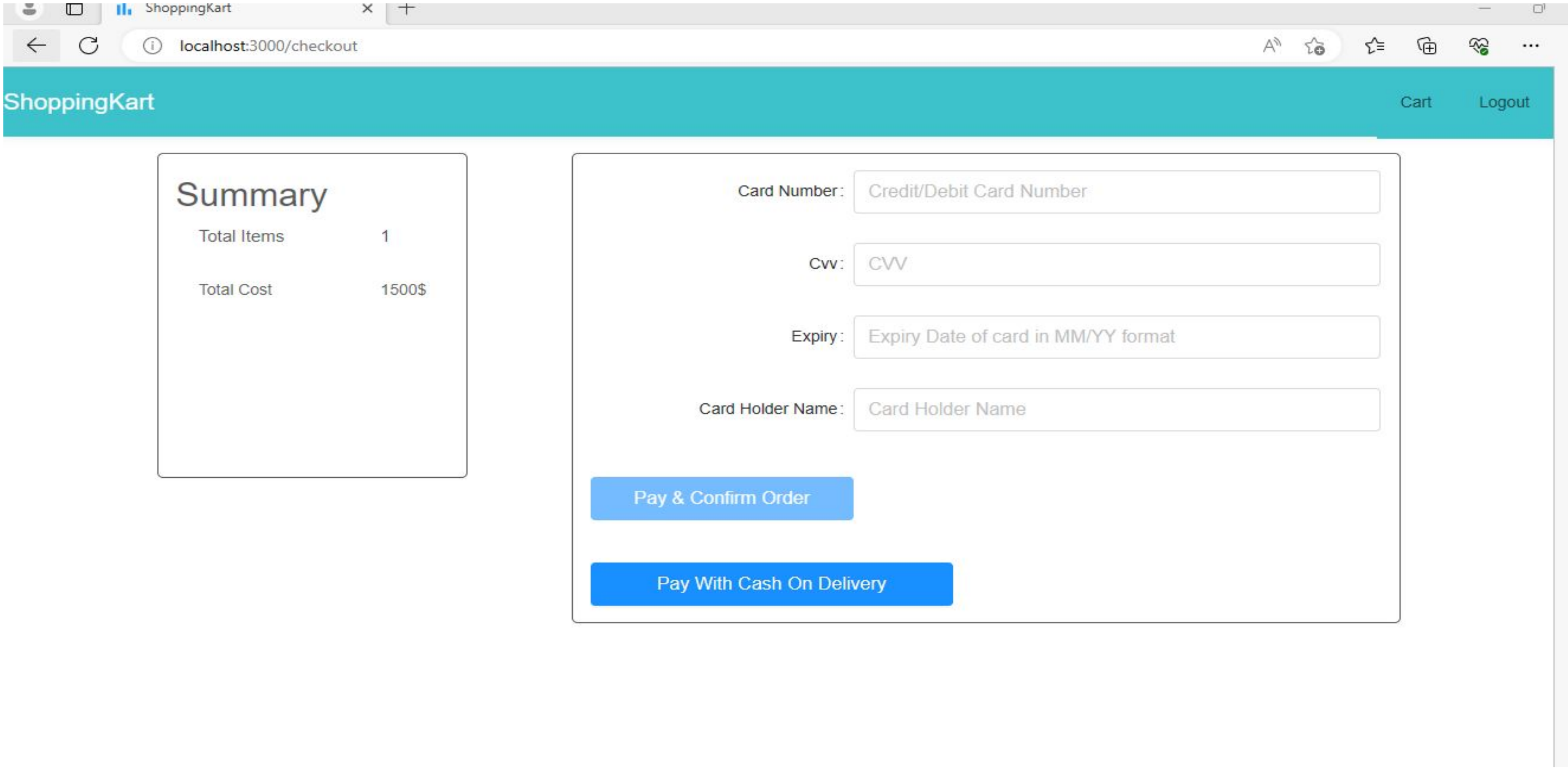


Cart Items

Item Name	Price	Quantity	Action
Gas Burner	1500	1	Delete

Summary

Total Items	1
Total Cost	1500$

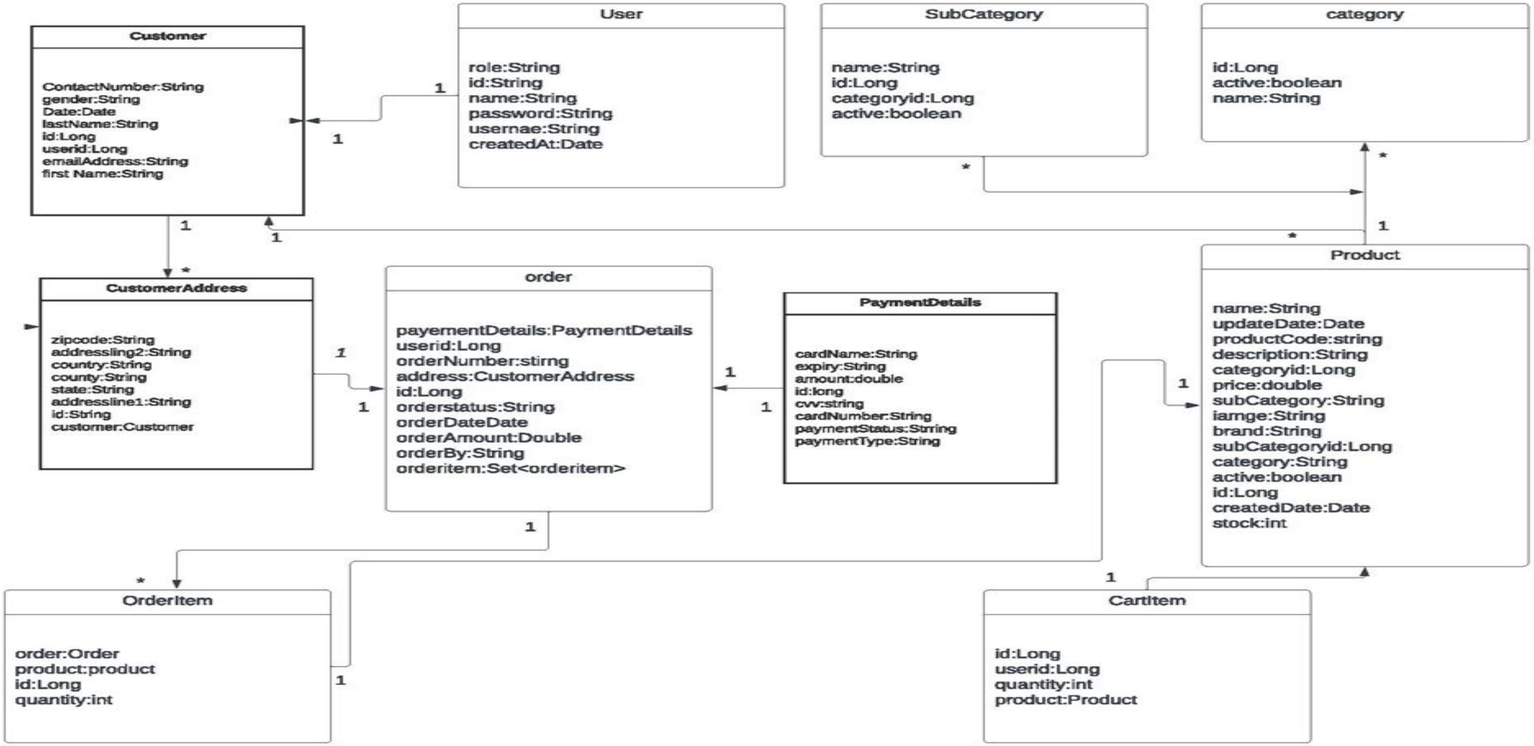


Summary

Total Items	1
Total Cost	1500$

Card Number :	Credit/Debit Card Number
Cvv :	CVV
Expiry :	Expiry Date of card in MM/YY format
Card Holder Name :	Card Holder Name

CLASS DIAGRAM



Customer

ContactNumber:String
gender:String
Date:Date
lastName:String
id:Long
userid:Long
emailAddress:String
first Name:String

User

role:String
id:String
name:String
password:String
usernae:String
createdAt:Date

SubCategory

name:String
id:Long
categoryid:Long
active:boolean

category

id:Long
active:boolean
name:String

CustomerAddress

zipcode:String
addressling2:String
country:String
county:String
state:String
addressline1:String
id:String
customer:Customer

order

payementDetails:PaymentDetails
userid:Long
orderNumber:stirng
address:CustomerAddress
id:Long
orderstatus:String
orderDateDate
orderAmount:Double
orderBy:String
orderitem:Set<orderitem>

PaymentDetails

cardName:String
expiry:String
amount:double
id:long
cvv:string
cardNumber:String
paymentStatus:Strring
paymentType:String

Product

name:String
updateDate:Date
productCode:string
description:String
categoryid:Long
price:double
subCategory:String
iamge:String
brand:String
subCategoryid:Long
category:String
active:boolean
id:Long
createdDate:Date
stock:int

OrderItem

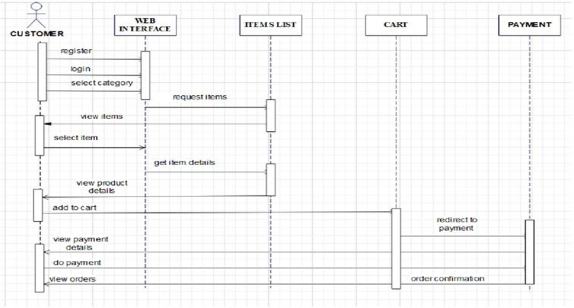
order:Order
product:product
id:Long
quantity:int

CartItem

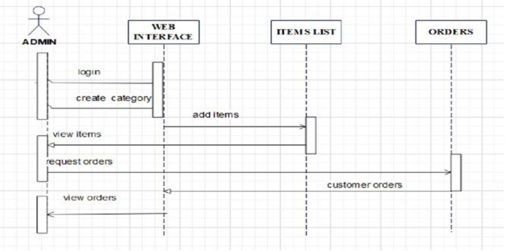
id:Long
userid:Long
quantity:int
product:Product

SEQUENCE DIAGRAM

Customer:



Admin:



Future Scope

- Progressive Web Applications (PWAs): Implementing PWA features such as offline functionality and push notifications to enhance the user experience across different devices and platforms.
- Microservices Architecture: Adopting a microservices architecture to enable scalability and independent development, deployment, and management of individual services.
- Integration with AI and Machine Learning: Leveraging AI and ML algorithms for personalized recommendations, fraud detection, inventory management, and sales forecasting.
- Payment Gateway Integration: Integrating popular and secure payment gateways, including support for digital wallets, cryptocurrencies, and alternative payment methods.
- Improved Analytics and Reporting: Incorporating advanced analytics and reporting capabilities to gain insights into customer behavior, sales performance, inventory management, and marketing campaigns.
- Enhanced Security and Compliance: Strengthening security measures to protect sensitive customer information and complying with industry regulations.

Conclusion

In conclusion, building a full-stack e-commerce web application using Spring Boot, React, Hibernate, and MySQL offers a powerful and efficient solution for creating a robust online shopping platform. By leveraging the strengths of each technology, we can achieve a scalable, responsive, and feature-rich application that meets the demands of modern e-commerce.

Thank you for your attention, and I'm excited to address any questions you may have about this development approach or any specific aspects of the application.

THANK YOU