INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT, AKURDI, PUNE

"Inventory Sell-Mart"

PG-DAC March 2024

Submitted By:

Group No: 15

Roll No.	**Name of Student**
243027	Disha Kamble
243079	Resham Pandey

Mrs. Vaishnavi Ghodke
Project Guide

Mr. Rohit Puranik
Centre Coordinator

ABSTRACT

Inventory Sell-Mart is a comprehensive platform designed to integrate inventory management with e-commerce functionalities. It allows businesses to efficiently track and manage their inventory while providing customers with a seamless online shopping experience. The platform uses technologies like React and Redux for a responsive front-end, and Spring Boot for a robust backend API. Security is enhanced through Spring Security for authentication and authorization, and payments are securely processed via Razor Pay.

Inventory Sell-Mart is a robust and scalable application designed to streamline inventory management and facilitate seamless e-commerce operations. The project aims to address the challenges faced by retailers and wholesalers in managing their product inventory and online sales through an integrated platform built with React for the frontend and Spring Boot for the backend.

ACKNOWLEDGEMENT

I take this occasion to thank God, almighty for blessing us with his grace and taking our endeavor to a successful culmination. I extend my heartfelt thanks to our esteemed guide, **Mrs. Vaishnavi Ghodke** for providing me with the right guidance and advice at the crucial juncture and showing me the right way. I sincerely thank our respected Centre Co-Ordinator, Mr. Rohit Puranik, for allowing us to use the available facilities. I would also like to thank the other faculty members at this occasion. Last but not least, I would like to thank my friends and family for the support and encouragement they have given me during our work.

Disha Kamlakar Kamble (240341220055)
Resham Pandey (2403412200156)

Table of Contents

1. <u>INTRODUCTION</u>

AtoZ: A Comprehensive Inventory Management and E-Commerce Solution. It is a comprehensive platform designed to integrate inventory management with e-commerce functionalities. It allows businesses to efficiently track and manage their inventory while providing customers with a seamless online shopping experience. The platform uses technologies like React and Redux for a responsive front-end, and Spring Boot for a robust backend API. Security is enhanced through Spring Security for authentication and authorization, and payments are securely processed via Razor-pay.

AtoZ is tailored to meet the needs of retail and wholesale businesses, ensuring real-time inventory updates, automated stock adjustments, and a user-friendly interface for customers. This integration of inventory management with e-commerce helps businesses reduce manual errors, improve operational efficiency, and provide a superior customer experience.

In the **Inventory Sell-Mart** system, both **User** and **Admin** roles play crucial roles in the management and operation of the platform. Each role comes with specific responsibilities, permissions, and access levels to ensure that the system operates smoothly and securely.

1. Admin Role

The admin role is the most powerful in the Inventory Sell-Mart system, having full control over the platform's functionalities. Admins are typically business owners, managers, or IT personnel who need comprehensive access to manage the inventory, users, and overall system settings.

❖ **Responsibilities:**

- **User Management:**
  - Create, manage, and delete user accounts.
  - Assign roles and permissions to users.
  - Monitor user activities and ensure compliance with company policies.

- **Inventory Management:**
  - Add, update, or delete product information.
  - Monitor stock levels and ensure that inventory data is up-to-date.
  - Set reorder levels for products to automate stock replenishment.

- **Order Management:**
  - View, process, and manage customer orders.
  - Oversee order fulfillment, including shipping and delivery processes.
  - Handle returns and exchanges.

- **Payment Management:**
  - Oversee and manage payments processed through Razorpay.
  - Resolve payment disputes and refund requests.

- **System Configuration:**
  - Configure system settings, including tax rates, discount rules, and shipping options.
  - Manage and update platform content, such as banners, promotions, and announcements.

- **Reporting & Analytics:**
  - ▪ Generate and analyze reports on sales, inventory levels, user activities, and financial transactions.
  - Use data to make informed business decisions.

- **Security Management:**

- Implement and enforce security policies using Spring Security.
- Monitor security logs and respond to potential security breaches.

Access Level:

- Full access to all system features and data.
- Ability to override settings and actions performed by users.
- Access to sensitive financial and personal information of users and customers.

2. User Role

Users in the Inventory Sell-Mart system typically refer to customers or employees with limited access. Their access level is more restricted compared to Admins, focusing on tasks related to inventory tracking, order placement, or basic system interactions.

Responsibilities:

- **Inventory Access (for employees):**
  - View product details, stock levels, and inventory status.
  - Update inventory data (with limited permissions).

- **Order Placement (for customers):**
  - Browse products, add items to the cart, and place orders.
  - Track order status and view order history.

- **Profile Management:**
  - Update personal information such as name, address, and contact details.
  - Manage account settings, including password changes.

- **Payments:**
  - Make payments for orders via the integrated Razorpay gateway.
  - View payment history and download receipts.

- **Support Requests:**
  - Raise support tickets for order issues, product inquiries, or account-related

problems.

- ○ Communicate with customer support for assistance.

Access Level:

- Limited access based on assigned permissions.

  Can view and interact with specific features (e.g., order placement, product browsing) but cannot alter system settings.

  No access to sensitive system configurations, financial data, or user management functionalities.

Role-Based Access Control (RBAC)

The system implements Role-Based Access Control (RBAC) using **Spring Security**. This ensures that users only have access to the functionalities they need based on their roles. For instance:

- Admins have unrestricted access to all resources and can manage both the system and its users.
- Users have limited access, often restricted to viewing and interacting with inventory items, placing orders, and managing their profiles.

Security Considerations

Both roles are secured through robust authentication mechanisms provided by Spring Security. This includes:

- **Authentication:** Verifying the identity of users before granting access to the system.
- **Authorization:** Ensuring that users can only access the resources permitted by their role.
- **Session Management:** Secure handling of user sessions to prevent unauthorized access.

AtoZ is role-based structure ensures that tasks are appropriately delegated and that the system remains secure and efficient. Admins can focus on high-level management and strategic decisions, while users interact with the system to fulfil their specific roles without compromising security.

1.1 Purpose

The purpose of the Inventory Sell-Mart system is to provide a comprehensive and efficient platform for managing and selling inventory in a retail or wholesale environment. The system is designed to streamline various business operations related to inventory management, order processing, and customer interactions, ultimately enhancing the overall productivity and profitability of the business.

1.2 Scope

The **scope of the Inventory Sell-Mart project** encompasses the design, development, deployment, and maintenance of a comprehensive inventory management and sales platform that integrates modern technologies to meet the needs of retail and wholesale businesses. This scope is defined by the functionalities that the system will provide, the technologies used, and the stakeholders involved.

1.3 Objective of Inventory Sell-Mart

The objectives of the Inventory Sell-Mart project outline the primary goals that the system aims to achieve, providing clear guidance on its design, implementation, and operational focus. These objectives ensure that the system effectively supports the needs of its users while driving business efficiency and growth.

1. Efficient Inventory Management
2. Streamlined Sales Processes
3. Enhanced User Experience
4. Robust Security and Access Control
5. Scalable and Flexible Architecture
6. Comprehensive Reporting and Analytics
7. Cost Optimization
8. Enhanced Customer Experience
9. Compliance and Audit Readiness

10. Support for Growth and Expansion

1.4 Functionalities Provided by Inventory Sell-Mart

1. User Management

- User Registration and Login:
  - Users can create accounts, log in, and manage their profiles.
  - Secure authentication and authorization using Spring Security.
- Role-Based Access Control:
  - Different user roles (e.g., Admin, Employee, Customer) with specific permissions.
  - Admins can create and manage these roles and assign them to users.
- Profile Management:
  - Users can update their personal information, such as name, email, and contact details.
  - Password management, including options for changing and resetting passwords.

2. Inventory Management

- Product Catalog Management:
  - Admins can add, update, and delete products in the inventory.
  - Manage product details such as name, description, price, SKU, category, and images.
- Stock Level Monitoring:
  - Real-time tracking of inventory levels for each product.
  - Automatic notifications for low stock levels to ensure timely reordering.
- Inventory Adjustment:
  - Manual adjustments to inventory levels to account for discrepancies, damage, or loss.
- Automated Reordering:
  - Automatic generation of purchase orders or alerts when inventory falls below a predefined threshold.

3. Sales Management

- Order Processing:
  - Customers can place orders, and the system automatically processes them, updating inventory levels accordingly.
  - Order confirmation and status updates are sent to customers.
- Payment Integration:
  - Integration with Razorpay for secure online payments, supporting multiple payment

methods (credit/debit cards, UPI, wallets).

- Order Tracking:
    - Customers can track the status of their orders, from processing to delivery.
- Invoicing and Billing:
    - Automatic generation of invoices for completed orders.
    - Customers and admins can view and download invoices.

4. Customer Management

- Customer Profiles:
    - Maintain detailed profiles for each customer, including order history, preferences, and contact information.
- Loyalty Programs and Discounts:
    - Implementation of loyalty programs or discounts to reward frequent customers and encourage repeat business.
- Customer Support:
    - Provide customer support through features like live chat, FAQs, or contact forms.
- Feedback and Reviews:
    - Customers can leave feedback and reviews on products, which can be managed by admins.

5. Reporting and Analytics

- Sales Reporting:

  Detailed reports on sales performance, including revenue, profit margins, and top-selling products.
- Inventory Reporting:

  Reports on inventory levels, turnover rates, and stock valuation.
- Customer Analytics:

  Analysis of customer behavior, purchase patterns, and demographics to inform marketing strategies.
- Financial Reporting:

  Overview of financial metrics, including sales revenue, expenses, and profitability.

6. Security and Compliance

- Data Encryption:

Encryption of sensitive data such as payment information and personal details to protect against unauthorized access.

- Audit Trails:

  Logging of significant actions within the system, such as inventory changes, order processing, and user management for accountability and compliance.

- Compliance with Legal Requirements:

  Ensuring that the system complies with relevant legal and regulatory requirements, such as data protection laws (e.g., GDPR).

7. Customer Experience Enhancement

- Search and Filtering:

  Advanced search and filtering options for customers to find products based on various criteria such as category, price, or brand.

- Wishlist and Shopping Cart:

  Features allowing customers to save products for future purchases or add them to their shopping cart.

- Responsive Design:

  A user interface that adapts to different devices, ensuring a seamless experience on desktops, tablets, and smartphones.

8. Integration and Extensibility

- API Integration:

  RESTful APIs for external systems to interact with the Inventory SellMart platform, enabling data exchange with other applications.

- Third-Party Integrations:

  Integration with third-party services such as accounting software, CRM systems, or e-commerce platforms.

2. SOFTWARE REQUIREMENT SPECIFICATION

The functional requirements for Inventory Sell-Mart outline the specific features and capabilities that the system must provide to meet the needs of its users. These requirements are essential for guiding the development process and ensuring that the final product aligns with the business objectives.

2.1 Functional Requirements for Inventory Sell-Mart

1. User Management

- **User Registration:**

  The system shall allow new users to create an account by providing personal details, such as name, email, and password.
- **User Authentication:**

  The system shall authenticate users during login using their registered email and password.
- **Role-Based Access Control:**

  The system shall support role-based access, where different users (Admin, Employee, Customer) have different permissions.
- **Profile Management:**

  Users shall be able to view and update their profiles, including personal details and passwords.

2. Inventory Management

- **Product Management:**

  The system shall allow admins to add, update, and delete products from the inventory. The system shall store product details such as name, description, SKU, price, category, and images.
- **Stock Level Monitoring:**

  The system shall monitor stock levels in real time and display current inventory levels.

- **Inventory Adjustment:**

  The system shall allow manual adjustments to inventory levels to account for discrepancies or damage.

- **Reordering:**

  The system shall automate reordering by generating purchase orders when stock levels are low.

3. Sales Management

- **Order Placement:**

  Customers shall be able to place orders for products from the inventory.

- **Order Processing:**

  The system shall process orders, updating inventory levels accordingly.

- **Payment Processing:**

  The system shall integrate with Razorpay to handle online payments securely.

  The system shall support various payment methods, including credit/debit cards, UPI, and wallets.

- **Order Tracking:**

  Customers shall be able to track the status of their orders from processing to delivery.

- **Invoicing:**

  The system shall automatically generate invoices for completed orders, available for download.

4. Customer Management

- **Customer Profiles:**

  The system shall maintain profiles for each customer, including order history and personal details.

- **Order History:**

  Customers shall be able to view their order history.

- **Loyalty Programs:**

  The system shall support the creation and management of loyalty programs for customers.

- **Feedback and Reviews:**

  Customers shall be able to leave feedback and reviews on products.

5. Reporting and Analytics

- **Sales Reports:**

  The system shall generate reports on sales performance, including revenue, profit margins, and top-selling products.

- **Inventory Reports:**

  The system shall generate reports on inventory levels, turnover rates, and stock valuation.

- **Customer Analytics:**

  The system shall provide insights into customer behavior, including purchase patterns and demographics.

6. Security

- **Data Encryption:**

  The system shall encrypt sensitive data, such as payment information and personal details.

- **Authentication and Authorization:**

  The system shall enforce strong authentication mechanisms and ensure that users have access only to the functionalities allowed by their roles.

2.2 Non-Functional Requirements for Inventory SellMart

1. Performance

- Response Time:

  The system shall respond to user actions within 2 seconds under normal operating conditions.

- Scalability:

  The system shall handle an increasing number of users and transactions without performance degradation. It should support at least 10,000 concurrent users.

- Throughput:

  The system shall process at least 100 transactions per second during peak usage times.

2. Reliability

- Availability:

  The system shall have an uptime of 99.9% over a 12-month period, ensuring high availability for users.

- Fault Tolerance:

  The system shall continue to operate in the event of hardware or software failures, with minimal disruption to users.

- Error Handling:

  The system shall gracefully handle errors and provide meaningful feedback to users when issues occur.

3. Usability

- User Interface:

  The system shall have a user-friendly interface that is easy to navigate, with clear instructions and minimal learning curve.

4. Maintainability

- Modularity:

  The system shall be designed in a modular fashion, allowing for easy updates and enhancements to individual components without affecting the entire system.

- Code Quality:

  The system shall follow coding best practices, with well-documented, clean, and maintainable code.

- Testing:
  The system shall undergo thorough testing, including unit tests, integration tests, and user acceptance tests, to ensure quality and stability.

- **Other Requirements:**

Hardware and Network Interfaces:

Back-end Server Configuration:

- Intel Pentium-IV Processor
- 8 GB RAM

Front-end Client Configuration:

- AMD RYZEN 5 Processor
- 128 MB SDRAM
- 10 GB Hard Disk Drive
- 104 Keys Keyboard
- PS2 Mouse with pad

Software Interfaces:

Software configuration for back-end Services:

- Java EE
- Spring Boot, JPA, Razor Pay, Spring Authentication
- MySQL
- STS 3.9.18

Software configuration for front-end Services:

- ReactJS, Redux
- HTML, CSS, JS
- Bootstrap
- VS Code

3. DIAGRAMS

3. 1 Entity Relationship Diagram:

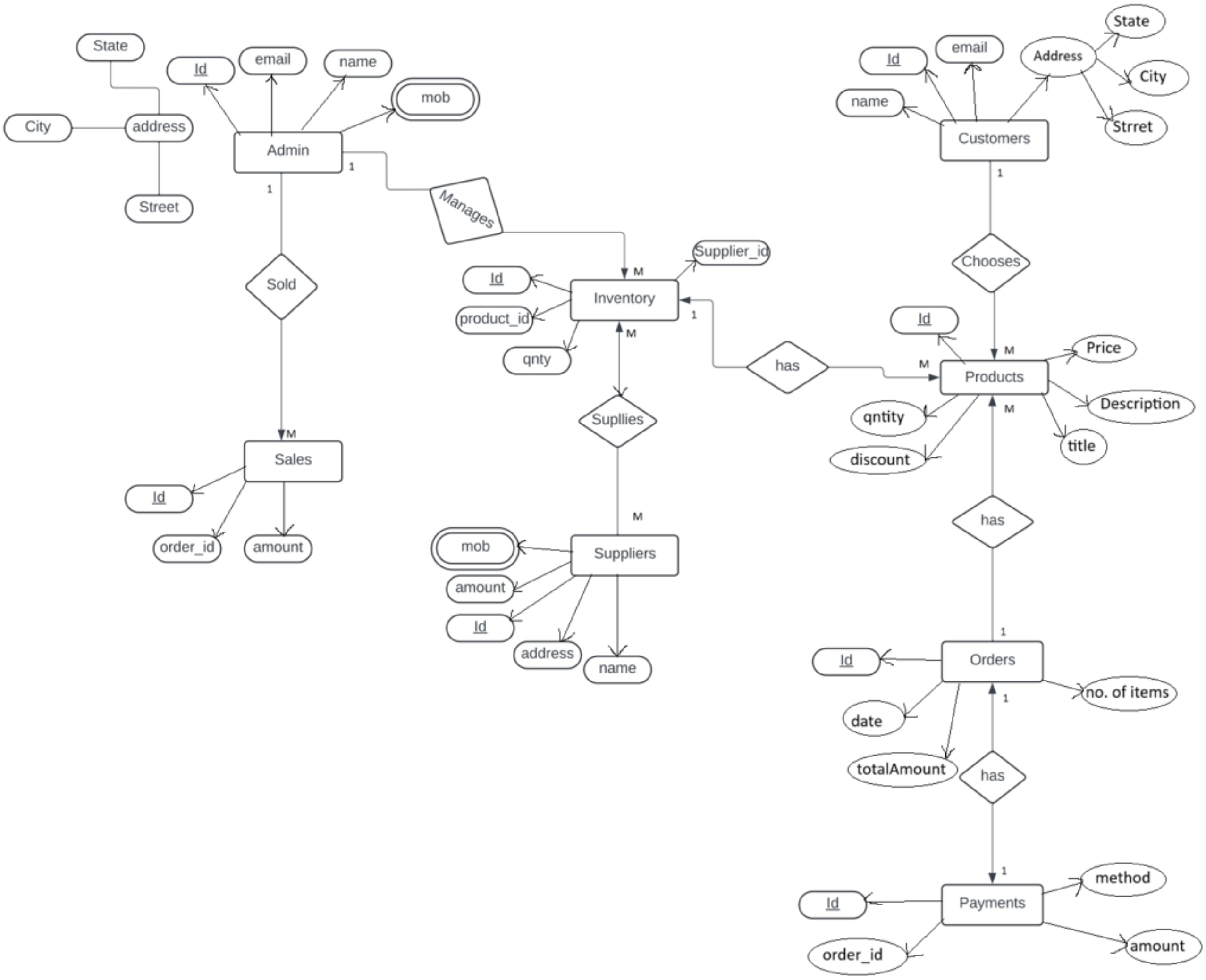


Fig. ER Diagram for Inventory Sell-Mart

3.2 Use Case Diagram:

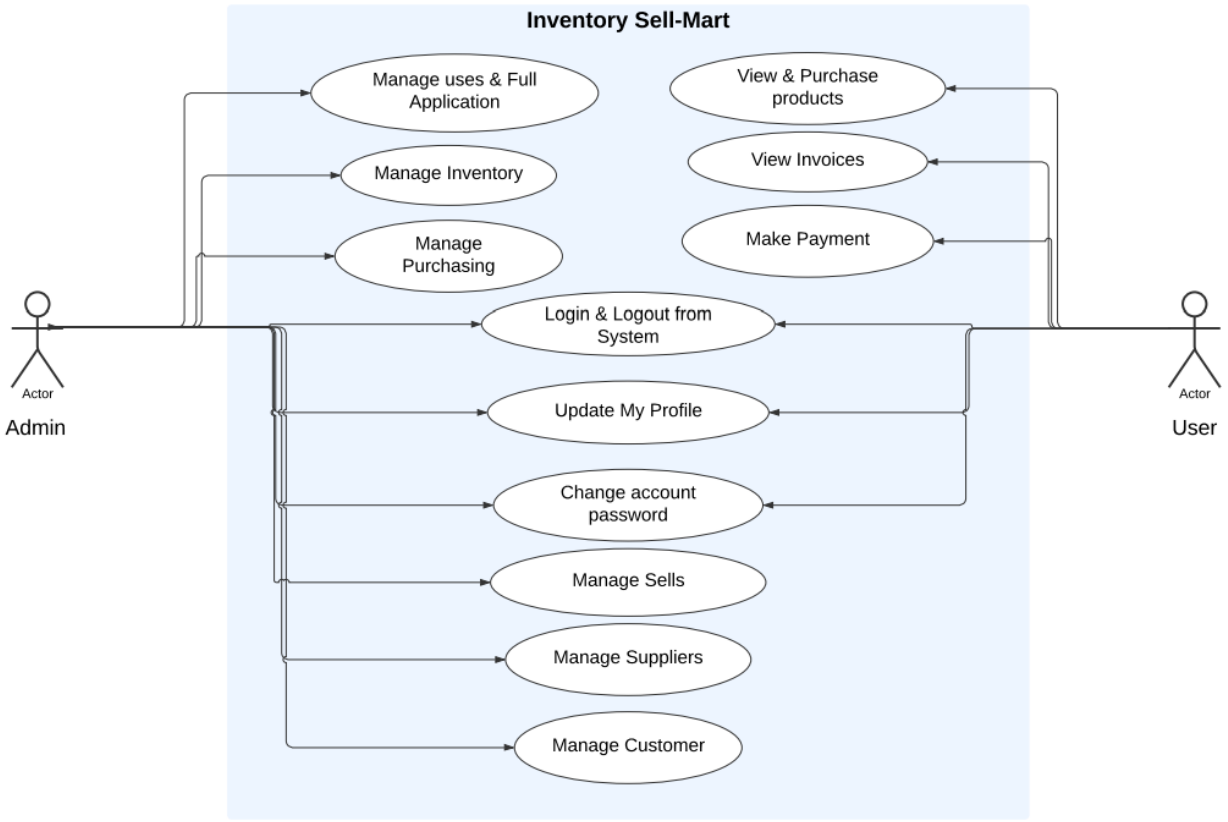
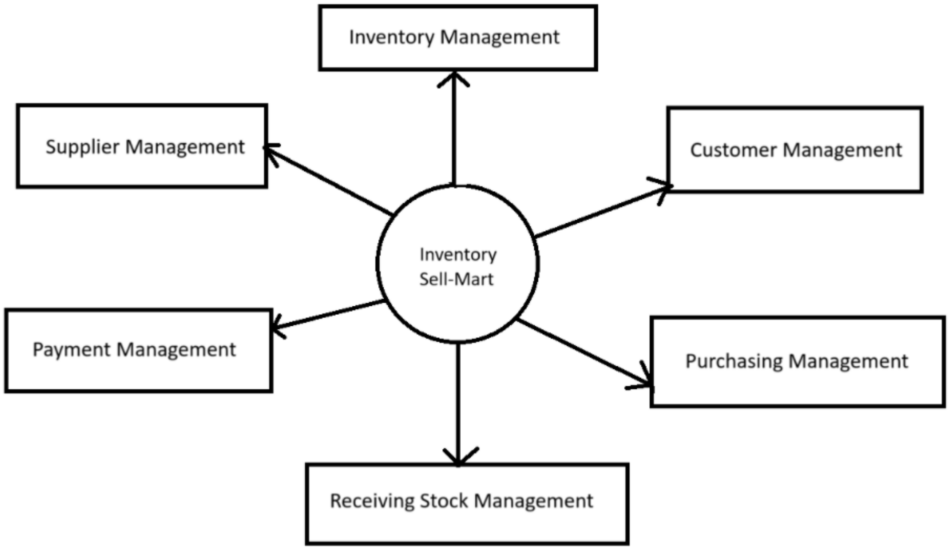


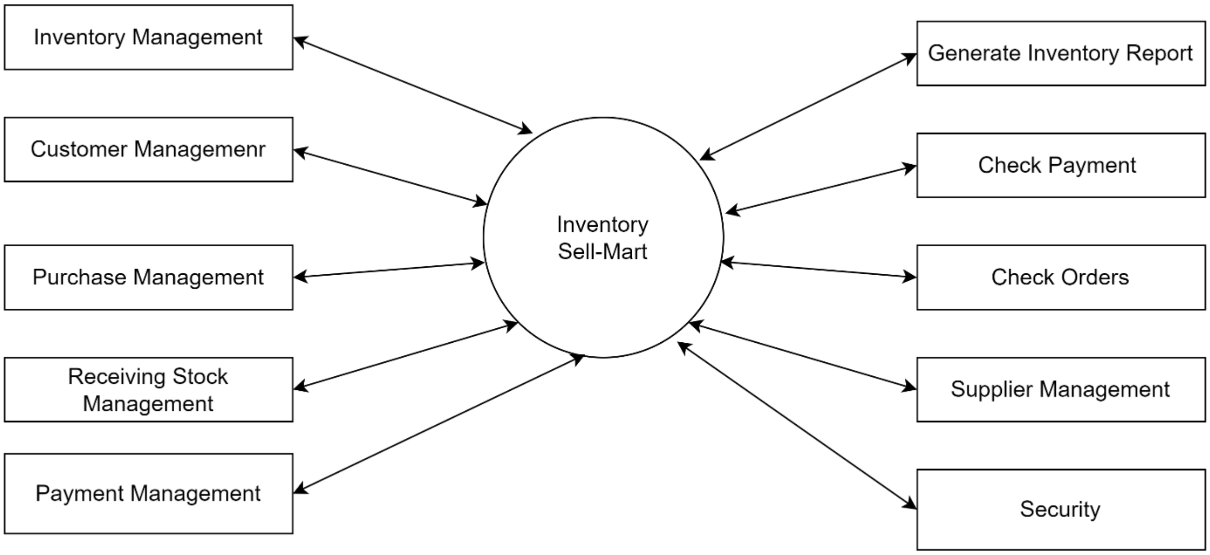
Fig. Use Case Diagram for Inventory Sell-Mart

3.3 Data Flow Diagram:

DFD Level 0:

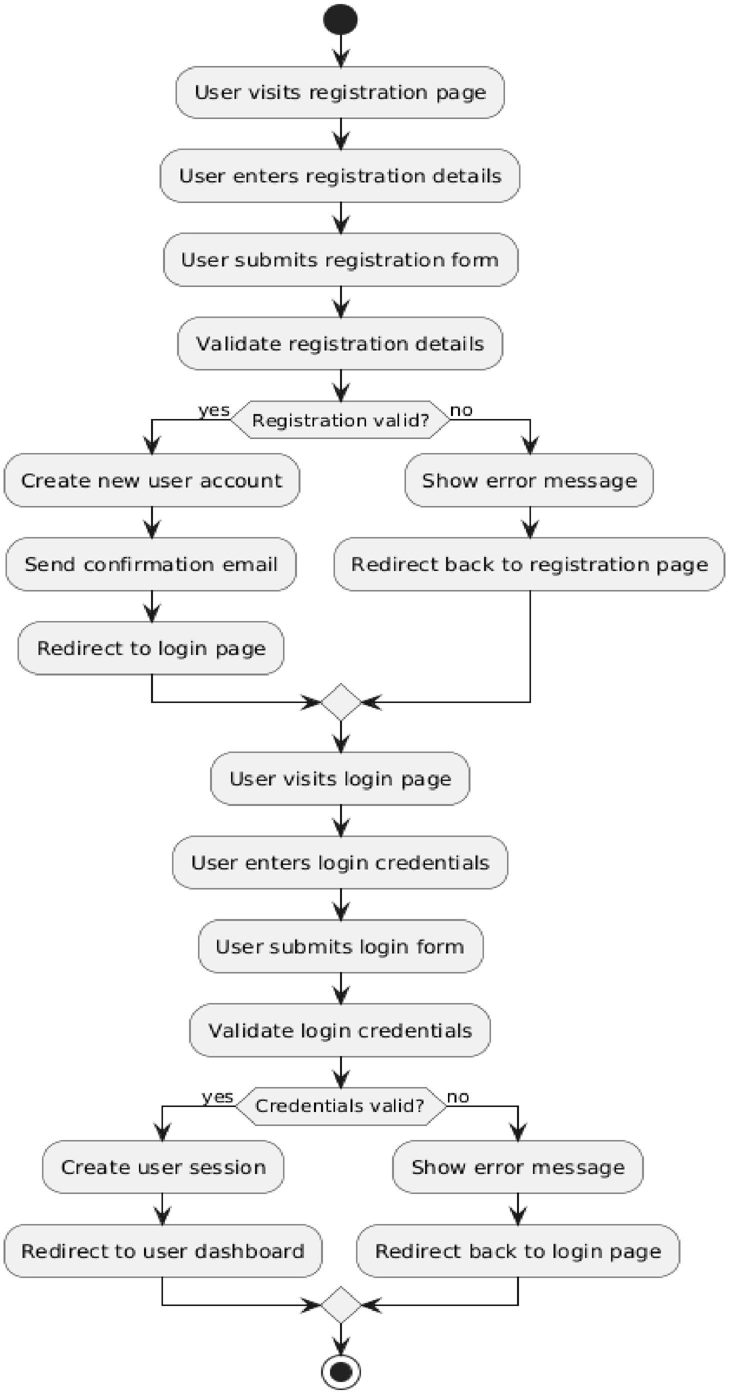


DFD level 1 :

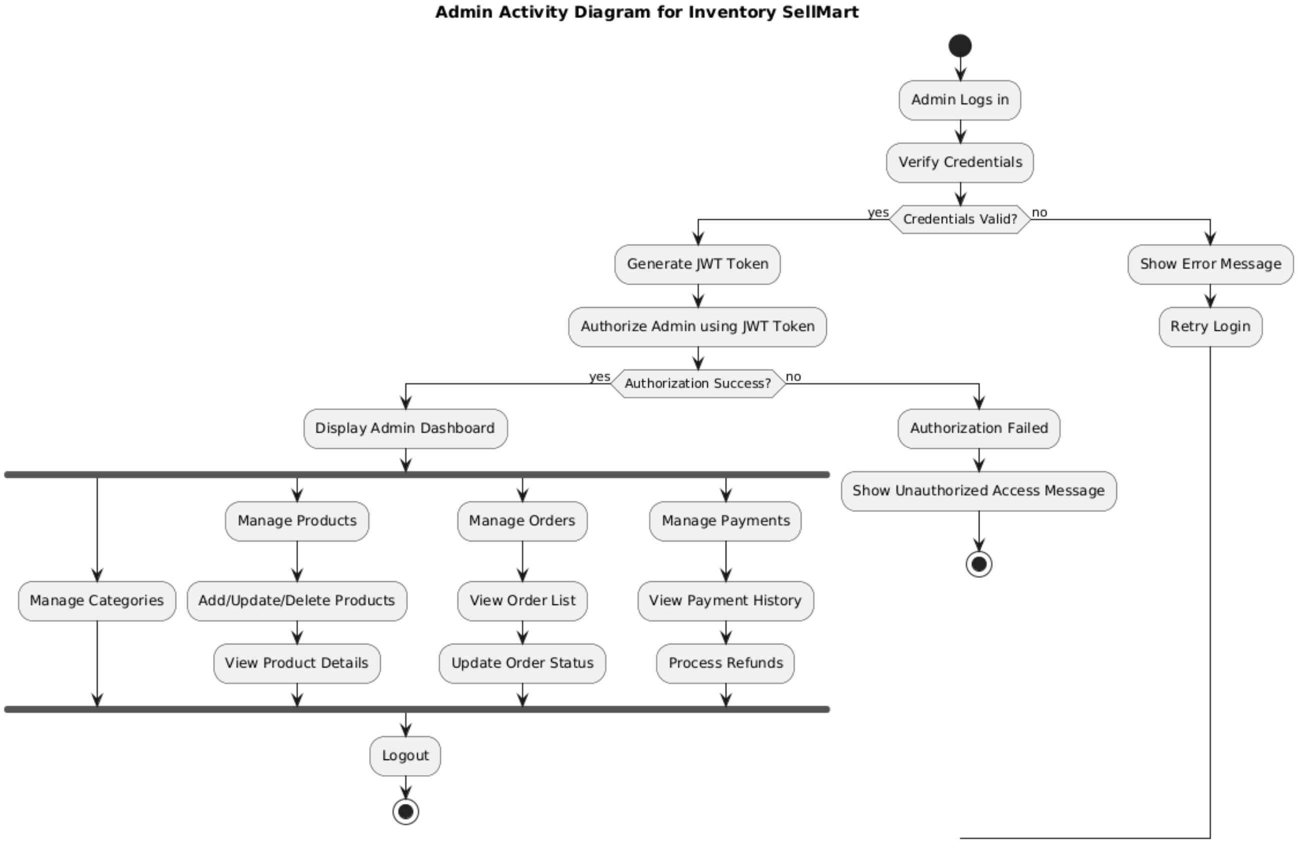


3.4 Activity Diagram :

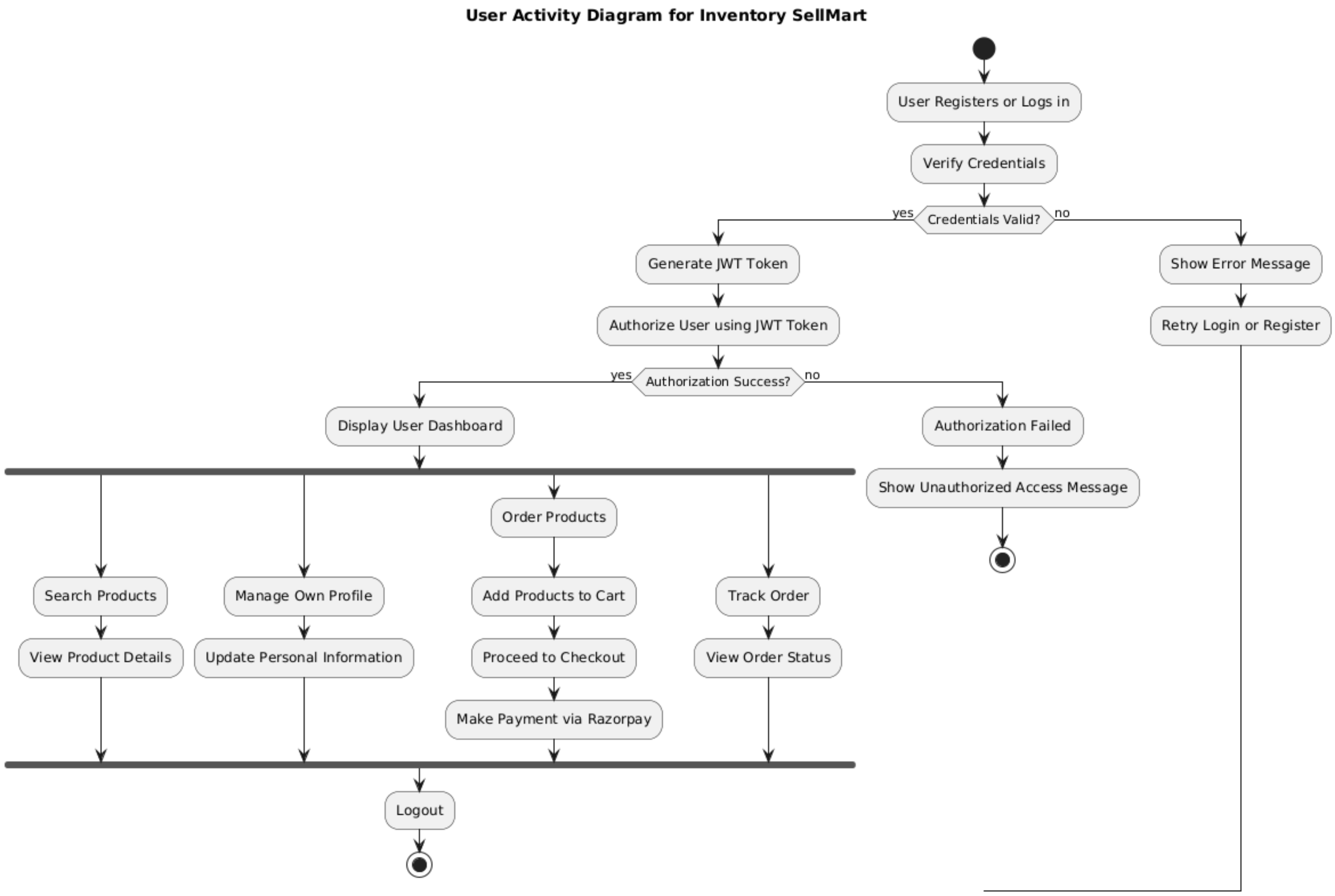
1. Login Activity Diagram


User visits registration page
User enters registration details
User submits registration form
Validate registration details
yes
Registration valid?
no
Create new user account
Show error message
Send confirmation email
Redirect back to registration page
Redirect to login page
User visits login page
User enters login credentials
User submits login form
Validate login credentials
yes
Credentials valid?
no
Create user session
Show error message
Redirect to user dashboard
Redirect back to login page

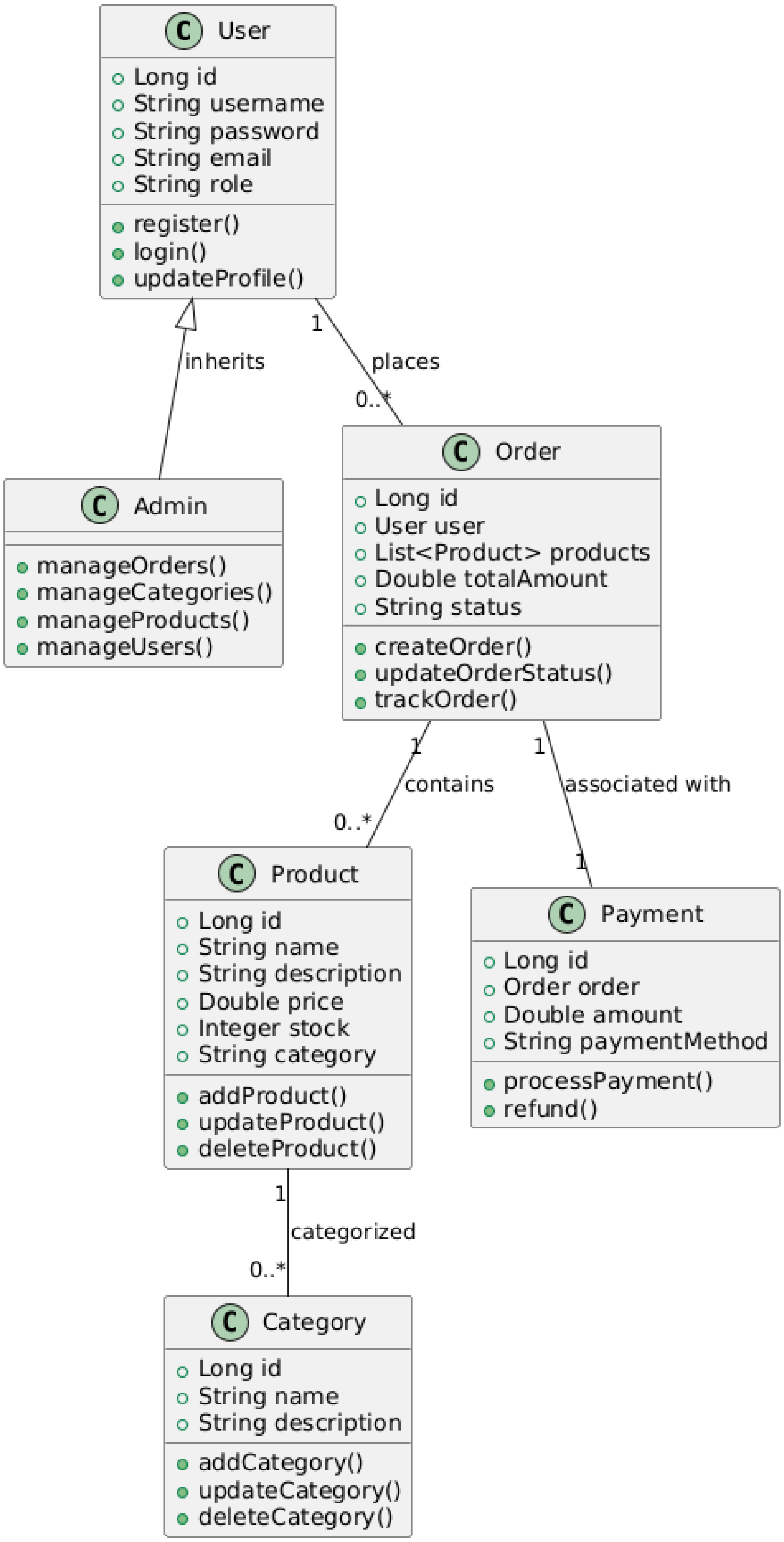
2. Admin Activity Diagram:



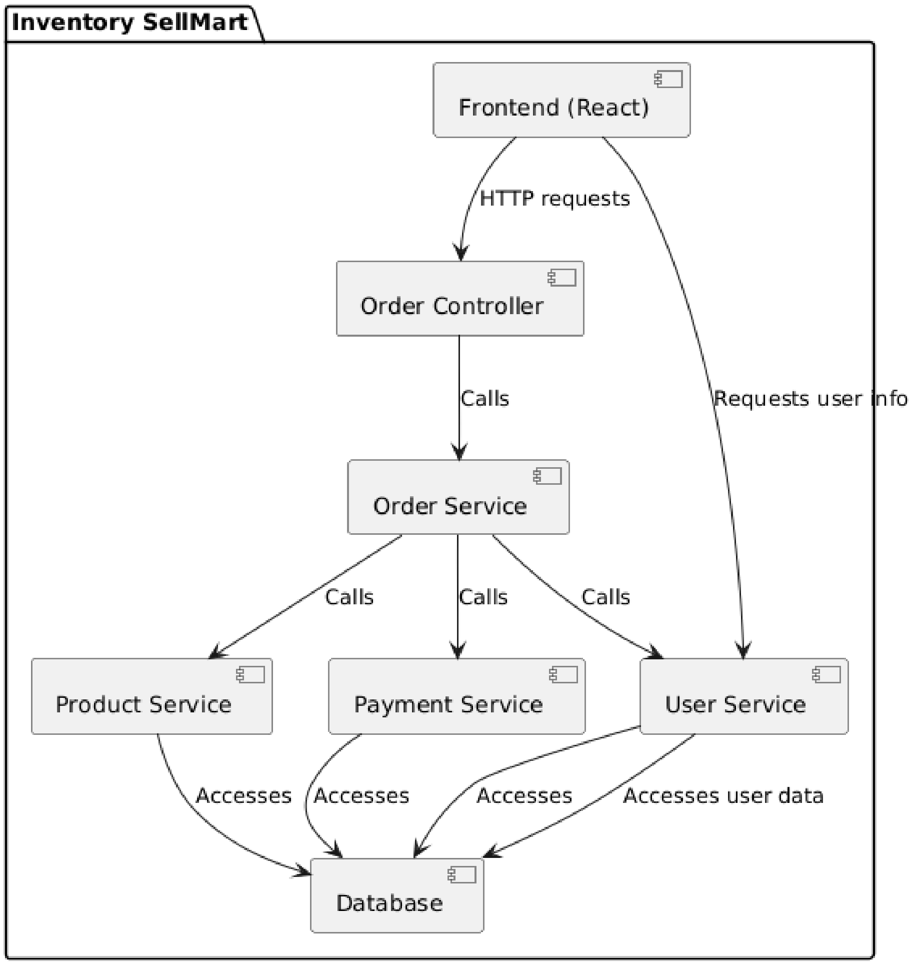
3. User Activity Diagram



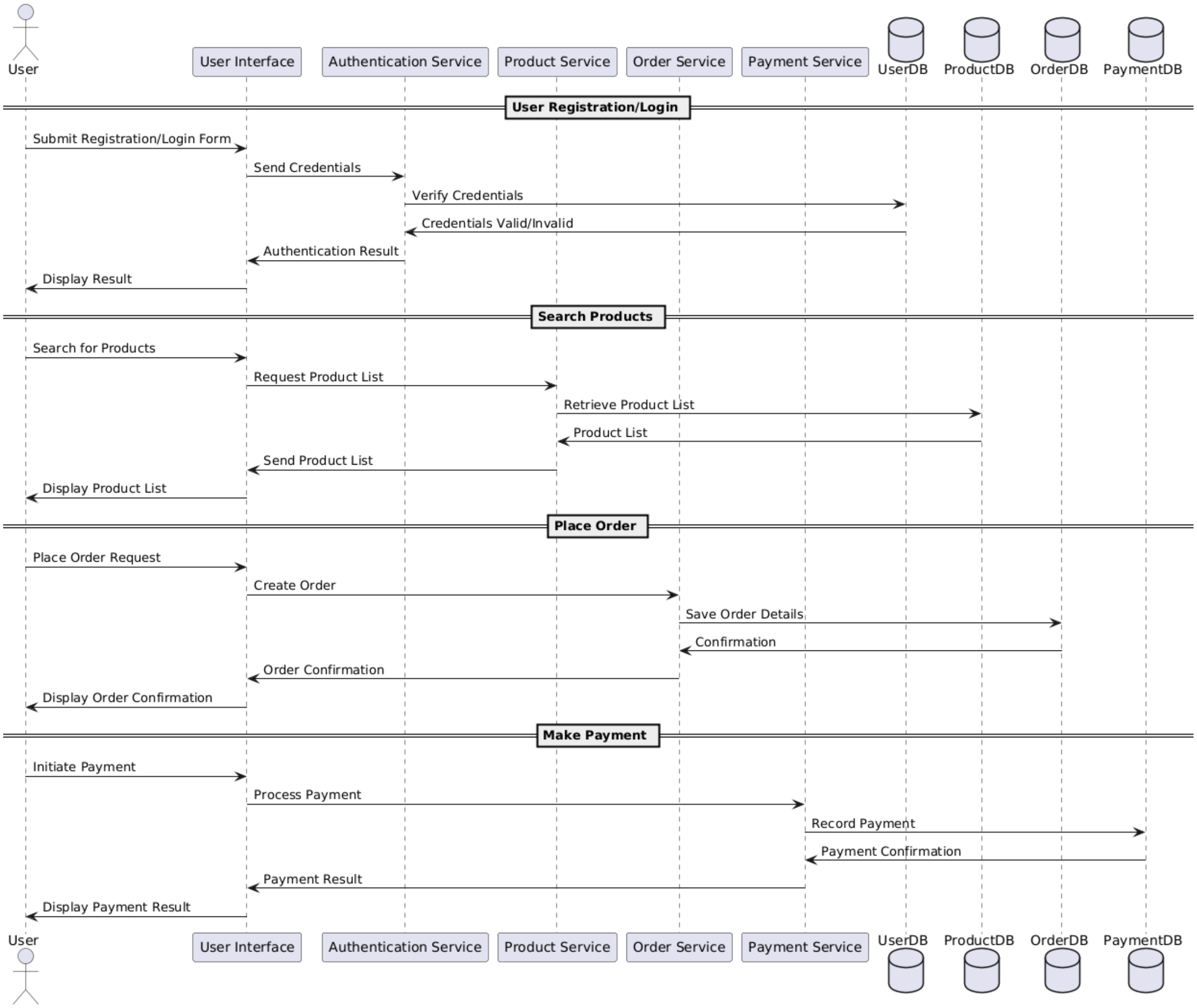
3.5 Class Diagram:



3.6 Component Diagram:

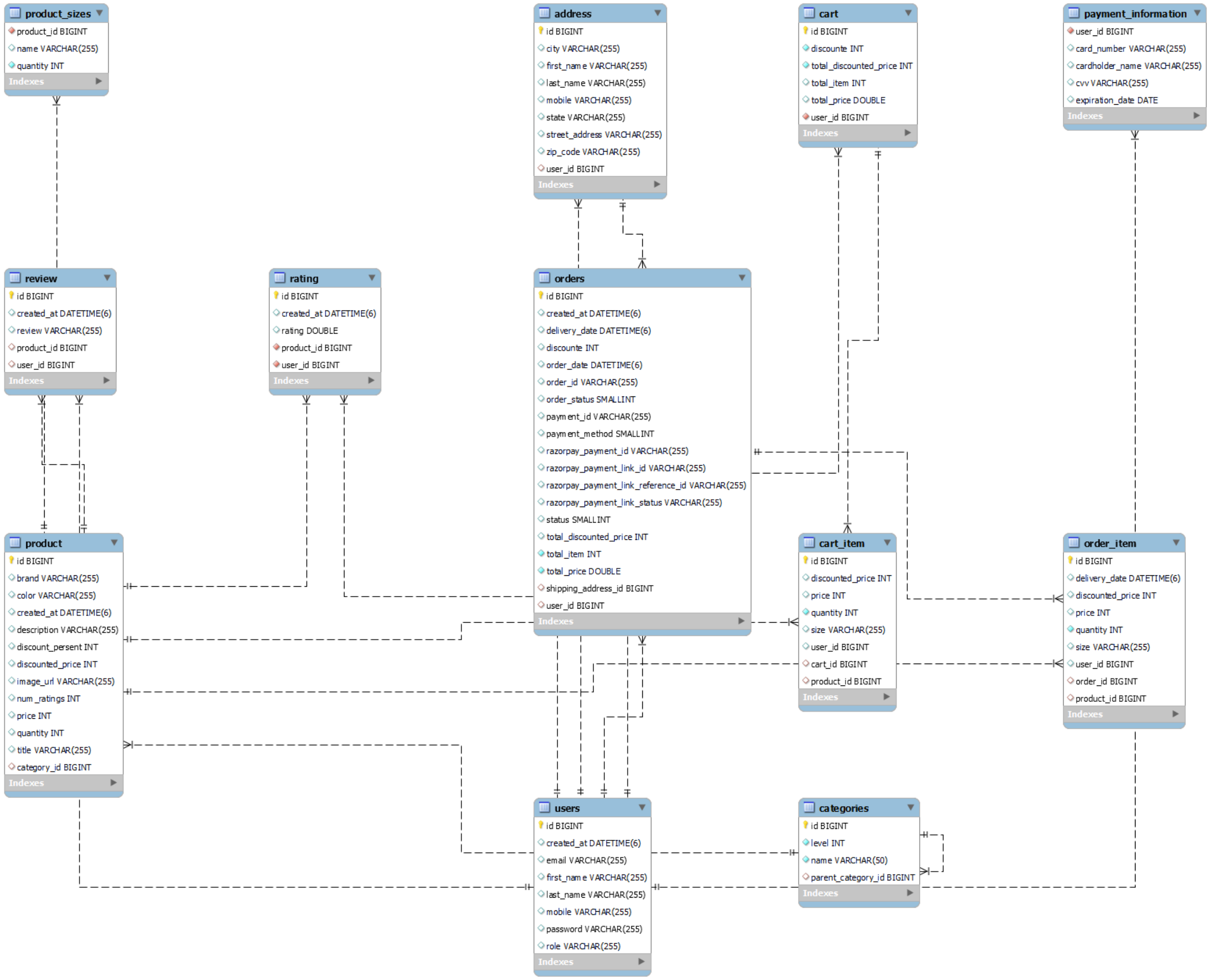


3.7 Sequence Diagram



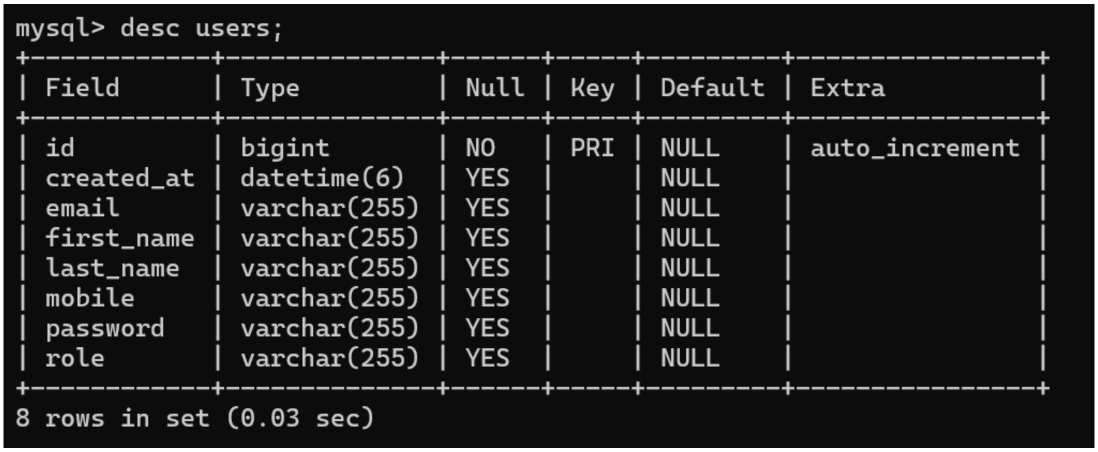
4. DATABASE DESIGN

4.1 Design:



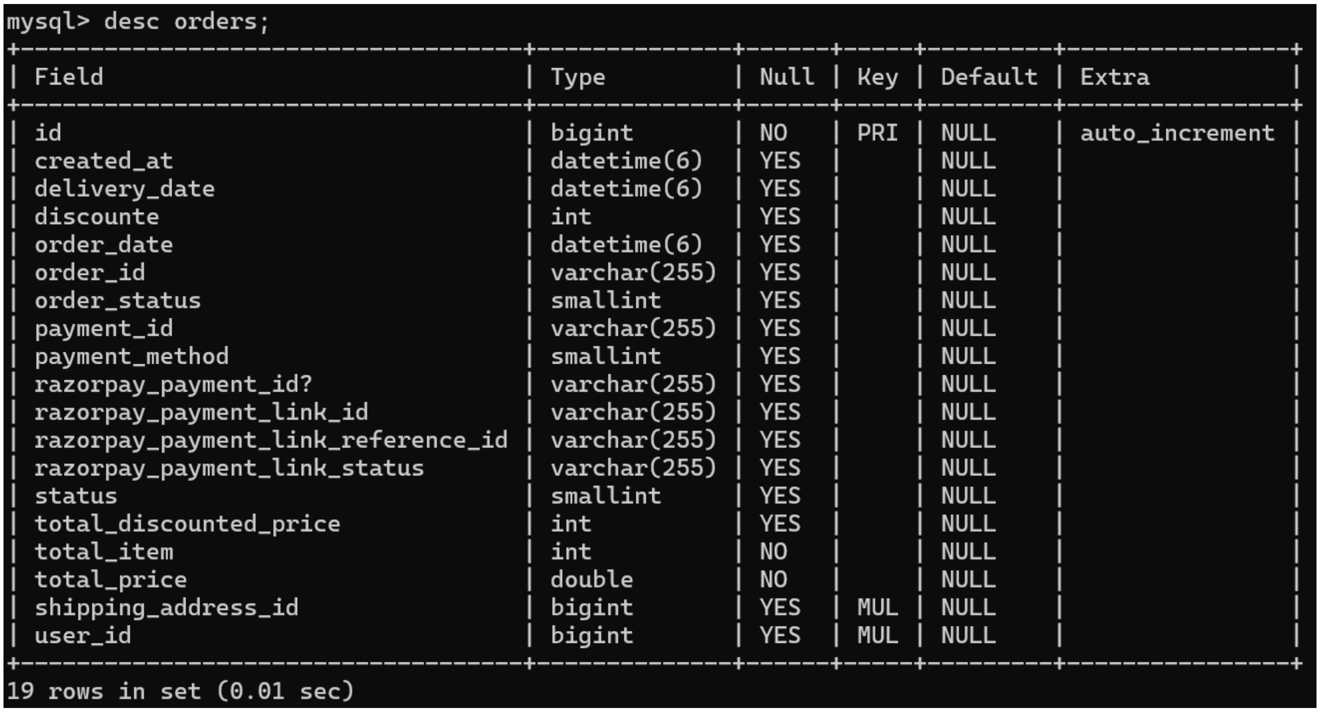
4.2 Tables:

The following table structures depict the database design.



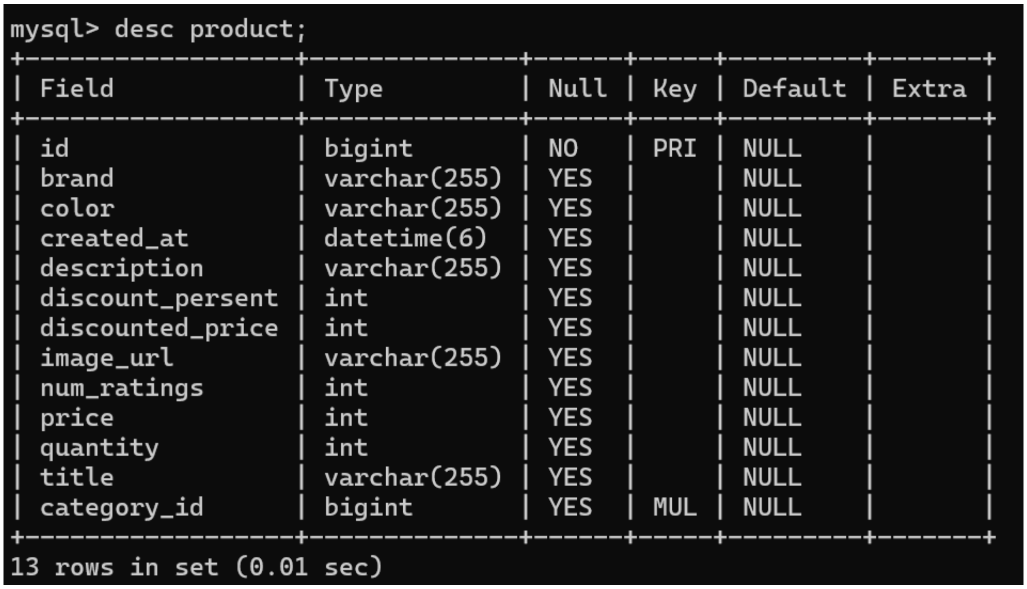
```
mysql> desc users;
+------------+--------------+------+-----+---------+----------------+
| Field      | Type         | Null | Key | Default | Extra          |
+------------+--------------+------+-----+---------+----------------+
| id         | bigint       | NO   | PRI | NULL    | auto_increment |
| created_at | datetime(6)  | YES  |     | NULL    |                |
| email      | varchar(255) | YES  |     | NULL    |                |
| first_name | varchar(255) | YES  |     | NULL    |                |
| last_name  | varchar(255) | YES  |     | NULL    |                |
| mobile     | varchar(255) | YES  |     | NULL    |                |
| password   | varchar(255) | YES  |     | NULL    |                |
| role       | varchar(255) | YES  |     | NULL    |                |
+------------+--------------+------+-----+---------+----------------+
8 rows in set (0.03 sec)
```

Table 1: user



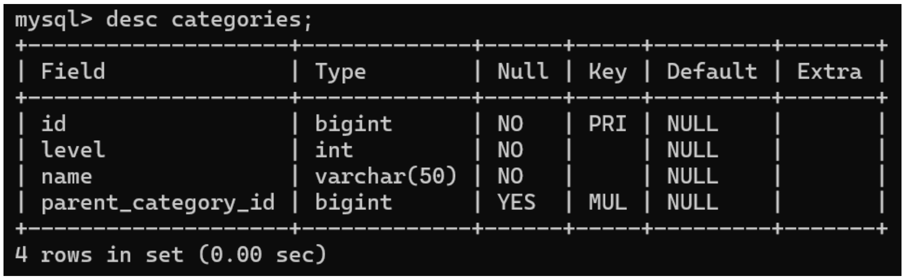
```
mysql> desc orders;
+------------------------------------+--------------+------+-----+---------+----------------+
| Field                              | Type         | Null | Key | Default | Extra          |
+------------------------------------+--------------+------+-----+---------+----------------+
| id                                 | bigint       | NO   | PRI | NULL    | auto_increment |
| created_at                         | datetime(6)  | YES  |     | NULL    |                |
| delivery_date                      | datetime(6)  | YES  |     | NULL    |                |
| discounte                          | int          | YES  |     | NULL    |                |
| order_date                         | datetime(6)  | YES  |     | NULL    |                |
| order_id                           | varchar(255) | YES  |     | NULL    |                |
| order_status                       | smallint     | YES  |     | NULL    |                |
| payment_id                         | varchar(255) | YES  |     | NULL    |                |
| payment_method                     | smallint     | YES  |     | NULL    |                |
| razorpay_payment_id?               | varchar(255) | YES  |     | NULL    |                |
| razorpay_payment_link_id           | varchar(255) | YES  |     | NULL    |                |
| razorpay_payment_link_reference_id | varchar(255) | YES  |     | NULL    |                |
| razorpay_payment_link_status       | varchar(255) | YES  |     | NULL    |                |
| status                             | smallint     | YES  |     | NULL    |                |
| total_discounted_price             | int          | YES  |     | NULL    |                |
| total_item                         | int          | NO   |     | NULL    |                |
| total_price                        | double       | NO   |     | NULL    |                |
| shipping_address_id                | bigint       | YES  | MUL | NULL    |                |
| user_id                            | bigint       | YES  | MUL | NULL    |                |
+------------------------------------+--------------+------+-----+---------+----------------+
19 rows in set (0.01 sec)
```

Table 2: orders



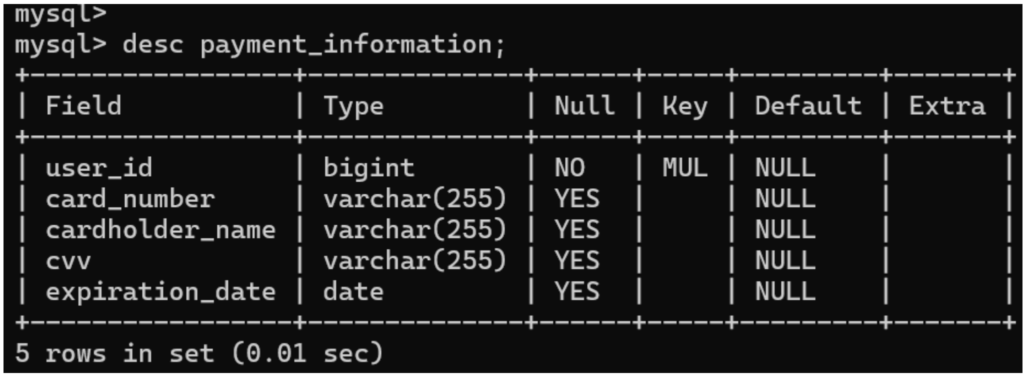
```
mysql> desc product;
+------------------+--------------+------+-----+---------+-------+
| Field            | Type         | Null | Key | Default | Extra |
+------------------+--------------+------+-----+---------+-------+
| id               | bigint       | NO   | PRI | NULL    |       |
| brand            | varchar(255) | YES  |     | NULL    |       |
| color            | varchar(255) | YES  |     | NULL    |       |
| created_at       | datetime(6)  | YES  |     | NULL    |       |
| description      | varchar(255) | YES  |     | NULL    |       |
| discount_persent | int          | YES  |     | NULL    |       |
| discounted_price | int          | YES  |     | NULL    |       |
| image_url        | varchar(255) | YES  |     | NULL    |       |
| num_ratings      | int          | YES  |     | NULL    |       |
| price            | int          | YES  |     | NULL    |       |
| quantity         | int          | YES  |     | NULL    |       |
| title            | varchar(255) | YES  |     | NULL    |       |
| category_id      | bigint       | YES  | MUL | NULL    |       |
+------------------+--------------+------+-----+---------+-------+
13 rows in set (0.01 sec)
```

Table 3: Products



```
mysql> desc categories;
+--------------------+-------------+------+-----+---------+-------+
| Field              | Type        | Null | Key | Default | Extra |
+--------------------+-------------+------+-----+---------+-------+
| id                 | bigint      | NO   | PRI | NULL    |       |
| level              | int         | NO   |     | NULL    |       |
| name               | varchar(50) | NO   |     | NULL    |       |
| parent_category_id | bigint      | YES  | MUL | NULL    |       |
+--------------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

Table 4: Categories

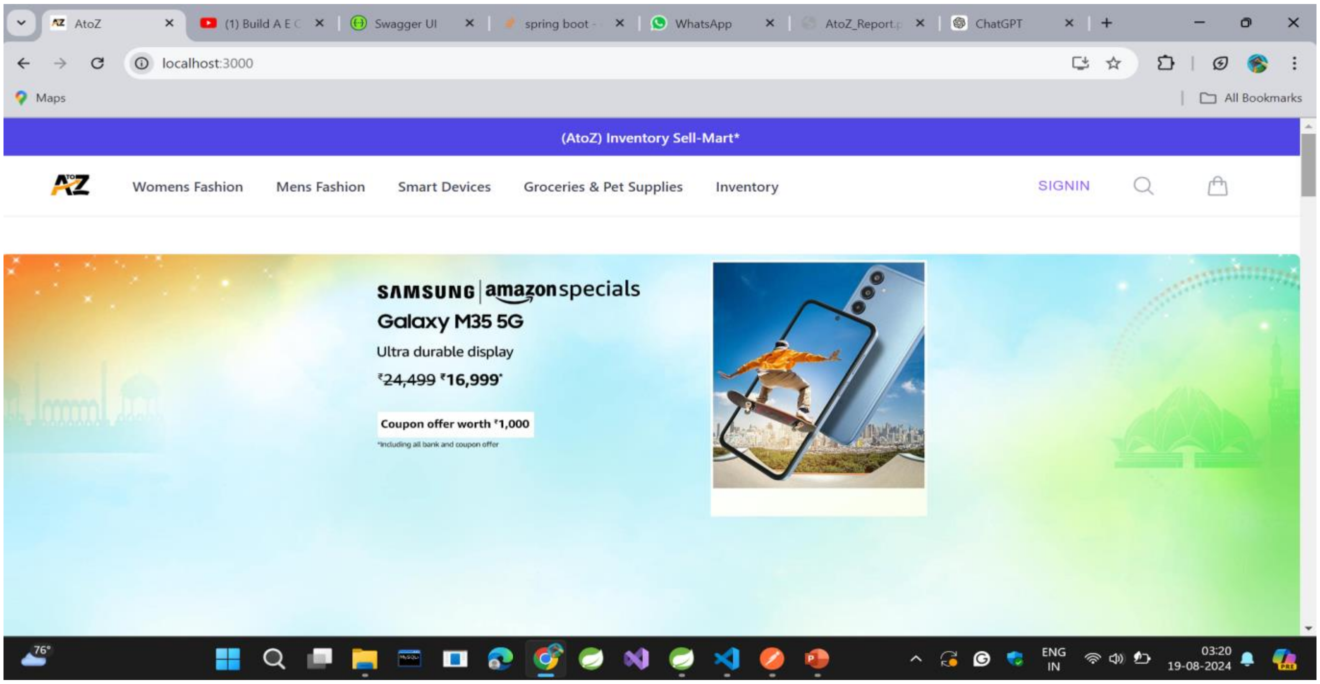


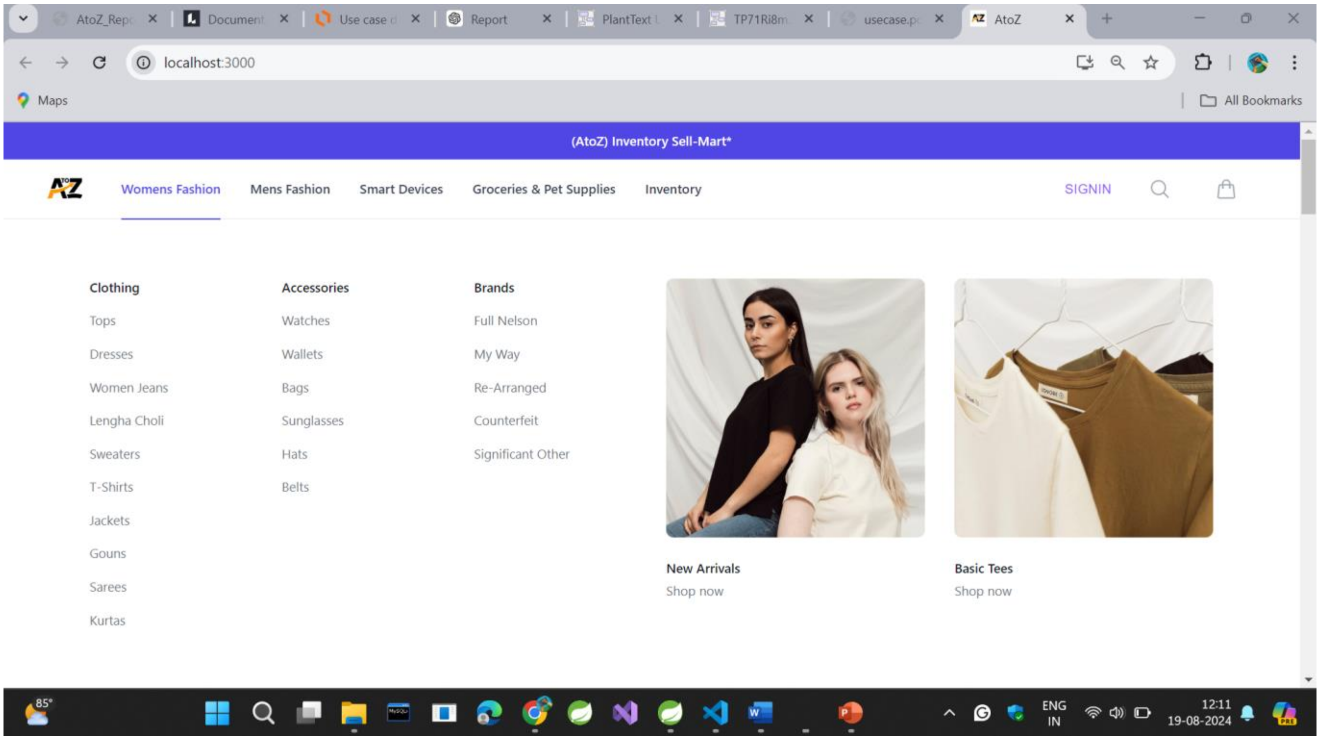
```
mysql>
mysql> desc payment_information;
+-----------------+--------------+------+-----+---------+-------+
| Field           | Type         | Null | Key | Default | Extra |
+-----------------+--------------+------+-----+---------+-------+
| user_id         | bigint       | NO   | MUL | NULL    |       |
| card_number     | varchar(255) | YES  |     | NULL    |       |
| cardholder_name | varchar(255) | YES  |     | NULL    |       |
| cvv             | varchar(255) | YES  |     | NULL    |       |
| expiration_date | date         | YES  |     | NULL    |       |
+-----------------+--------------+------+-----+---------+-------+
5 rows in set (0.01 sec)
```

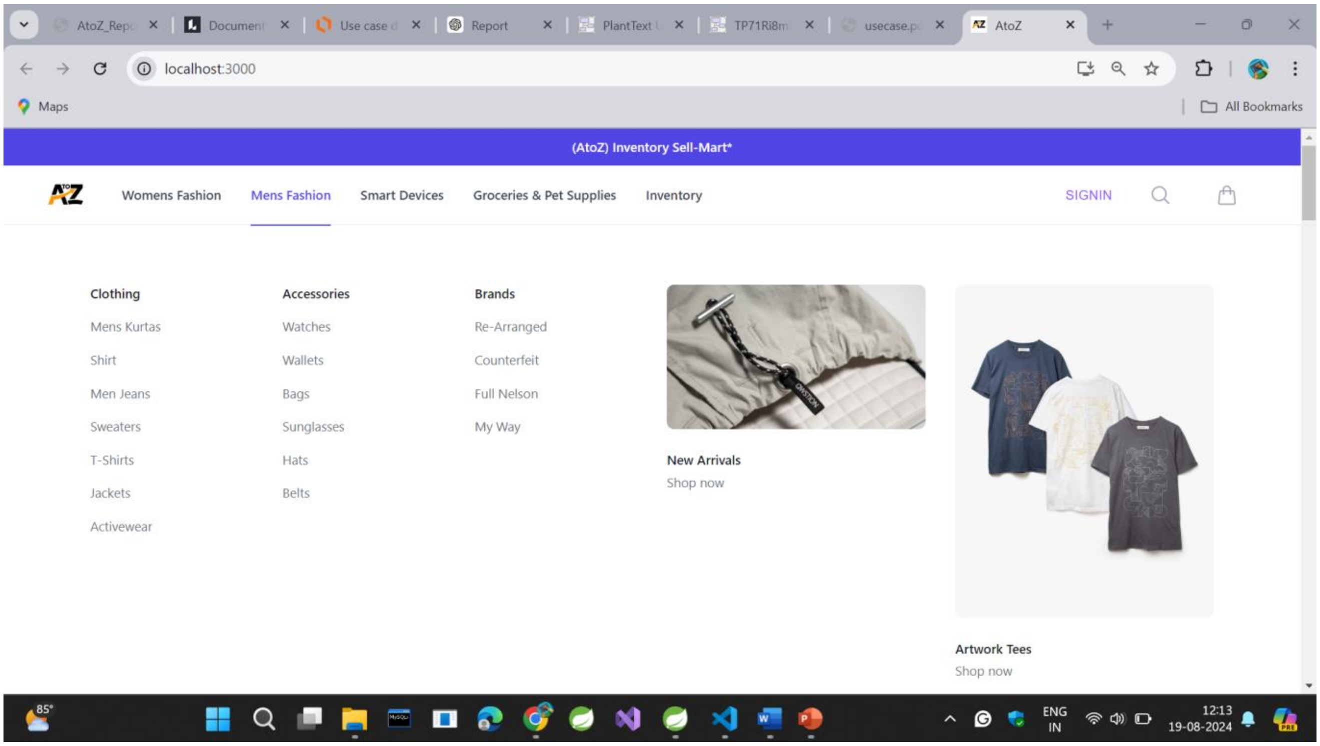
Table 5 : payment_information

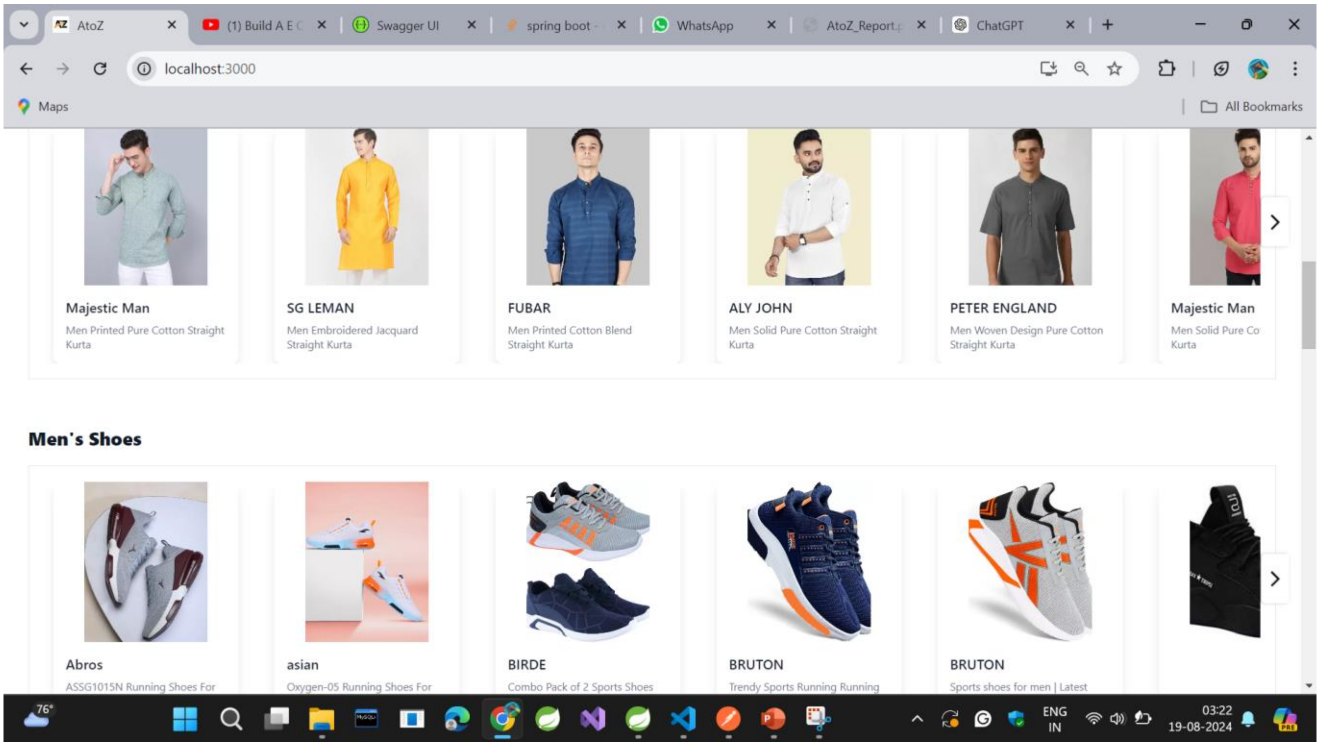
5. SNAPSHOTS

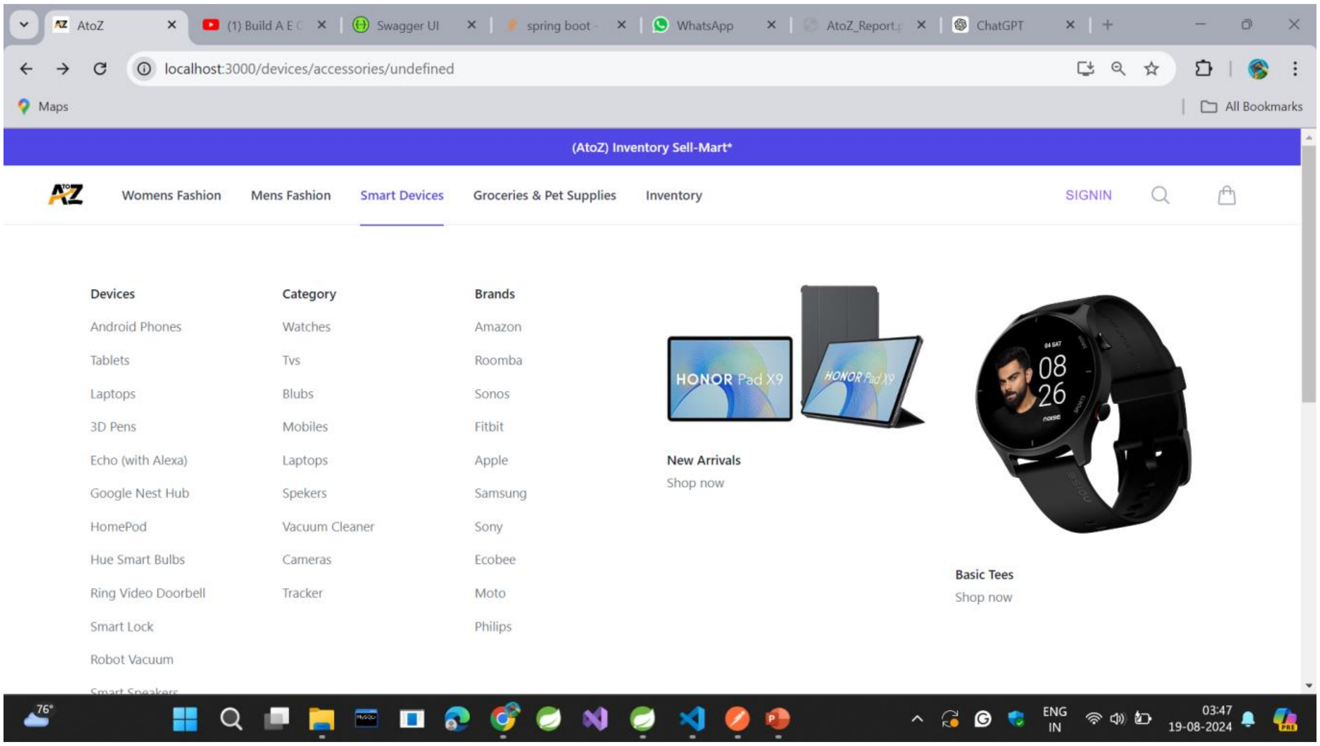
Home Page:

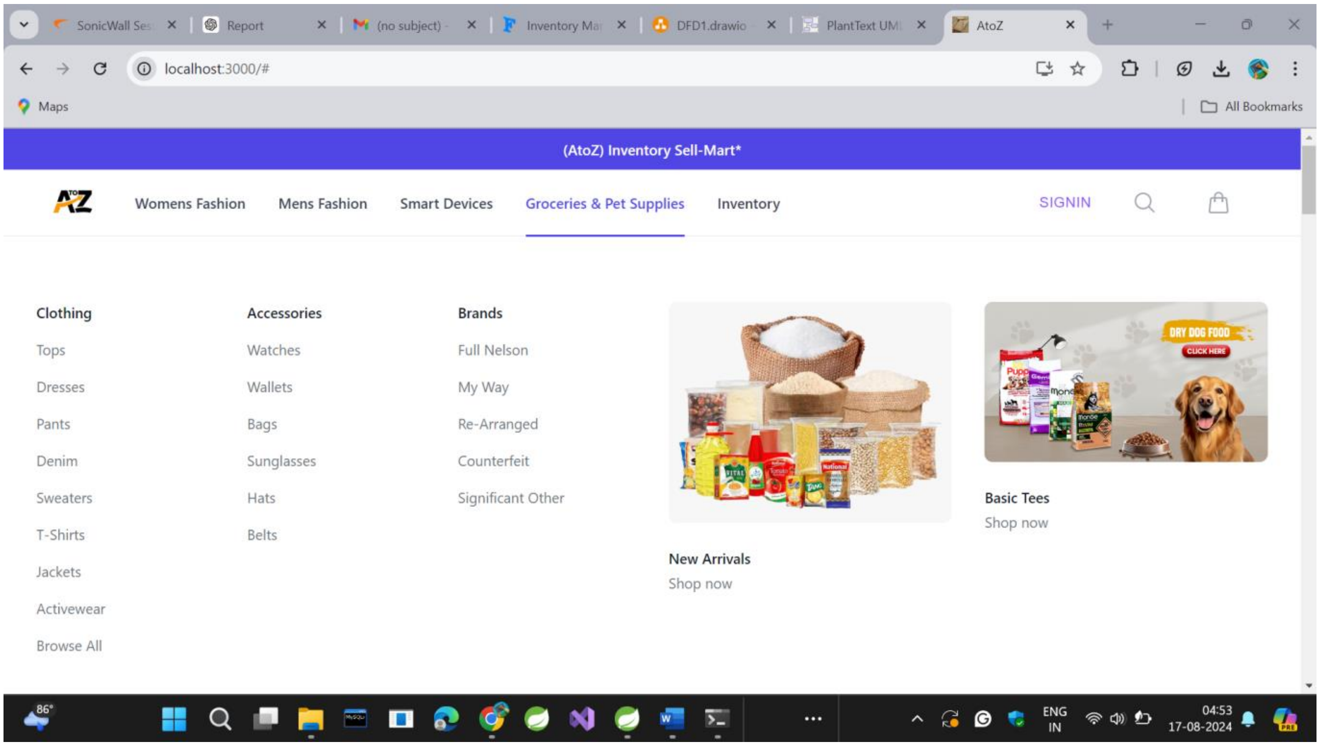




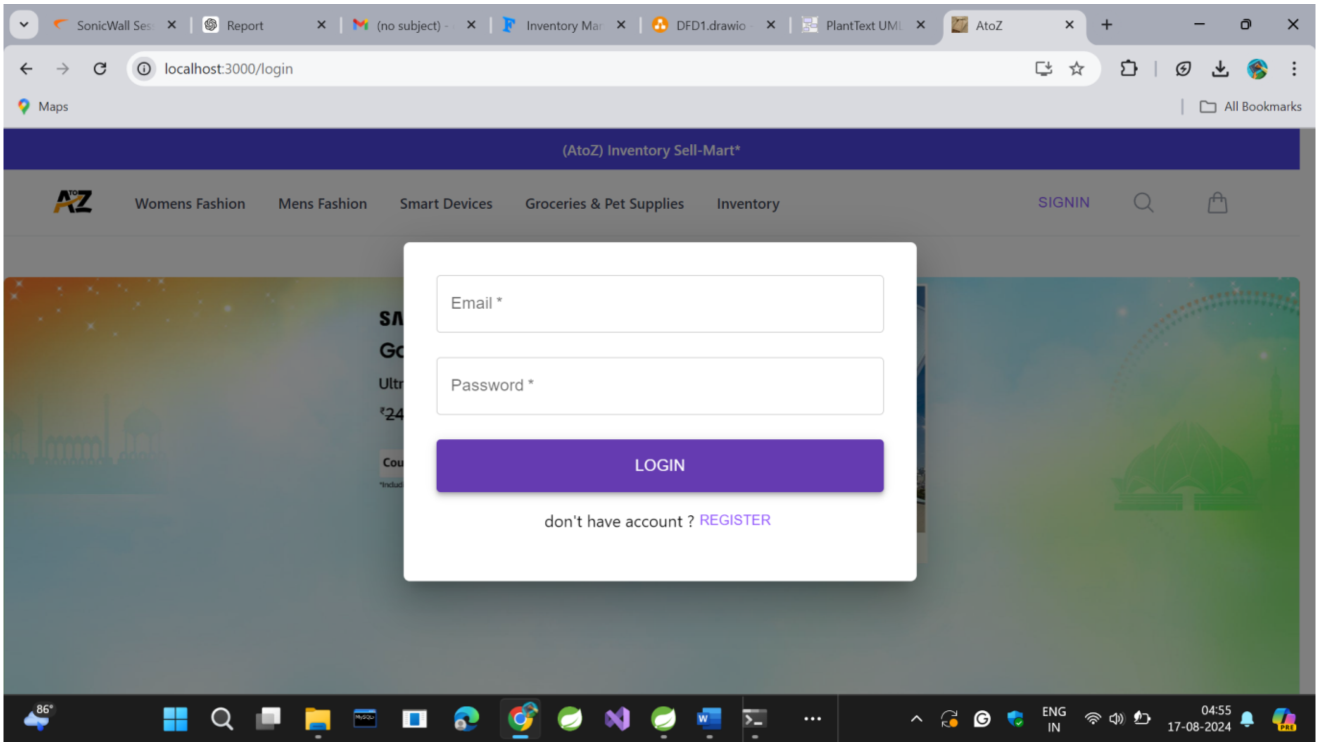

localhost:3000
(AtoZ) Inventory Sell-Mart*
Womens Fashion
Mens Fashion
Smart Devices
Groceries & Pet Supplies
Inventory
SIGNIN
Clothing
Mens Kurtas
Shirt
Men Jeans
Sweaters
T-Shirts
Jackets
Activewear
Accessories
Watches
Wallets
Bags
Sunglasses
Hats
Belts
Brands
Re-Arranged
Counterfeit
Full Nelson
My Way
New Arrivals
Shop now
Artwork Tees
Shop now


localhost:3000
Majestic Man
Men Printed Pure Cotton Straight Kurta
SG LEMAN
Men Embroidered Jacquard Straight Kurta
FUBAR
Men Printed Cotton Blend Straight Kurta
ALY JOHN
Men Solid Pure Cotton Straight Kurta
PETER ENGLAND
Men Woven Design Pure Cotton Straight Kurta
Majestic Man
Men Solid Pure Co Kurta
Men's Shoes
Abros
ASSG1015N Running Shoes For
asian
Oxygen-05 Running Shoes For
BIRDE
Combo Pack of 2 Sports Shoes
BRUTON
Trendy Sports Running Running
BRUTON
Sports shoes for men | Latest

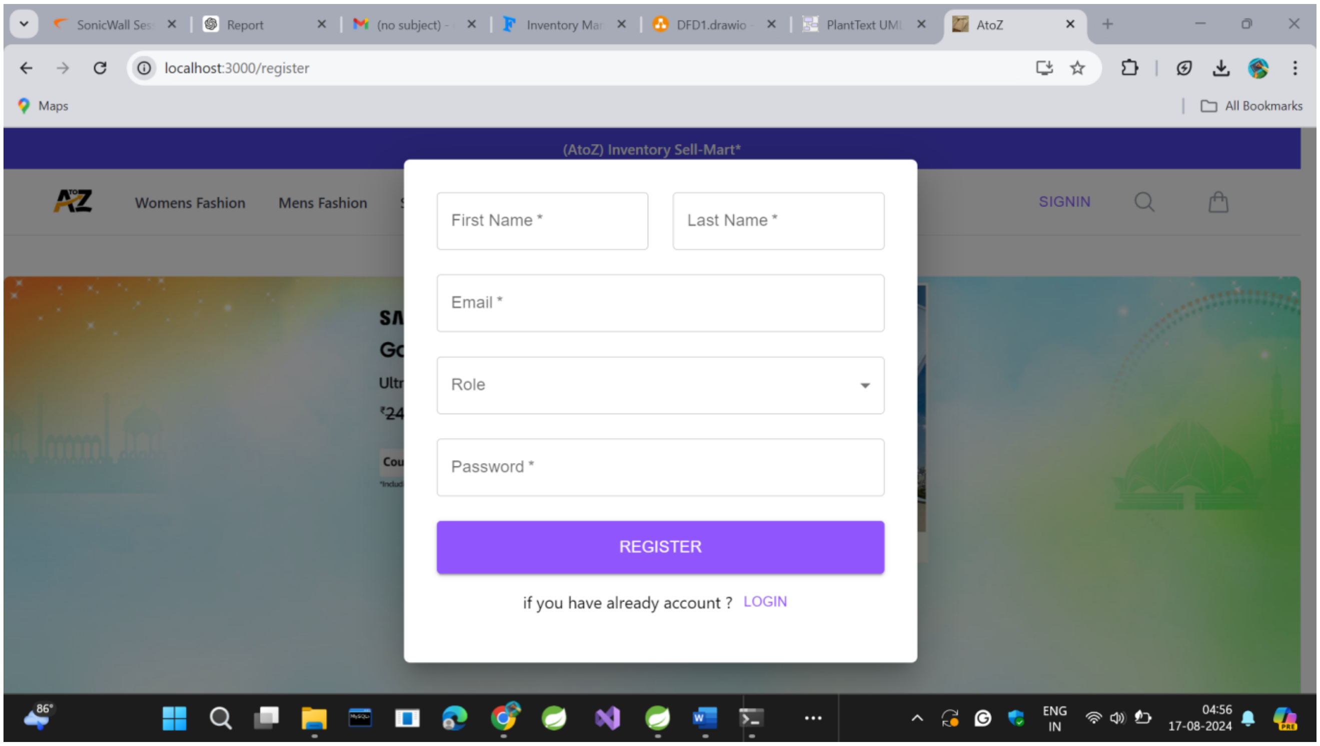

AtoZ
(1) Build A E C
Swagger UI
spring boot -
WhatsApp
AtoZ_Report.
ChatGPT
localhost:3000/devices/accessories/undefined
Maps
All Bookmarks
(AtoZ) Inventory Sell-Mart*
Womens Fashion
Mens Fashion
Smart Devices
Groceries & Pet Supplies
Inventory
SIGNIN
Devices
Android Phones
Tablets
Laptops
3D Pens
Echo (with Alexa)
Google Nest Hub
HomePod
Hue Smart Bulbs
Ring Video Doorbell
Smart Lock
Robot Vacuum
Smart Speakers
Category
Watches
Tvs
Blubs
Mobiles
Laptops
Spekers
Vacuum Cleaner
Cameras
Tracker
Brands
Amazon
Roomba
Sonos
Fitbit
Apple
Samsung
Sony
Ecobee
Moto
Philips
HONOR Pad X9
New Arrivals
Shop now
08
26
Basic Tees
Shop now
76°
ENG
IN
03:47
19-08-2024


SonicWall Se
Report
(no subject) -
Inventory Ma
DFD1.drawio
PlantText UM
AtoZ
localhost:3000/#
Maps
All Bookmarks
(AtoZ) Inventory Sell-Mart*
Womens Fashion
Mens Fashion
Smart Devices
Groceries & Pet Supplies
Inventory
SIGNIN
Clothing
Tops
Dresses
Pants
Denim
Sweaters
T-Shirts
Jackets
Activewear
Browse All
Accessories
Watches
Wallets
Bags
Sunglasses
Hats
Belts
Brands
Full Nelson
My Way
Re-Arranged
Counterfeit
Significant Other
New Arrivals
Shop now
DRY DOG FOOD
CLICK HERE
Basic Tees
Shop now
86°
ENG
IN
04:53
17-08-2024

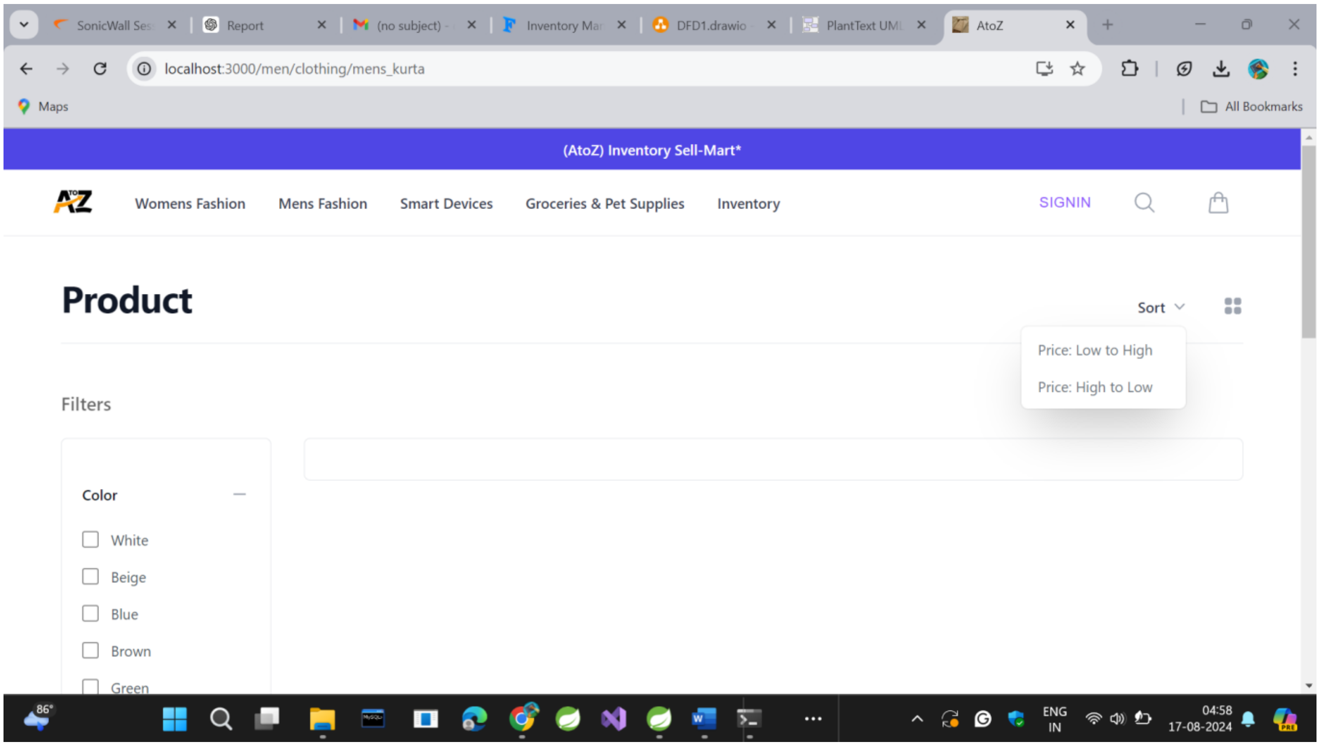
Login:



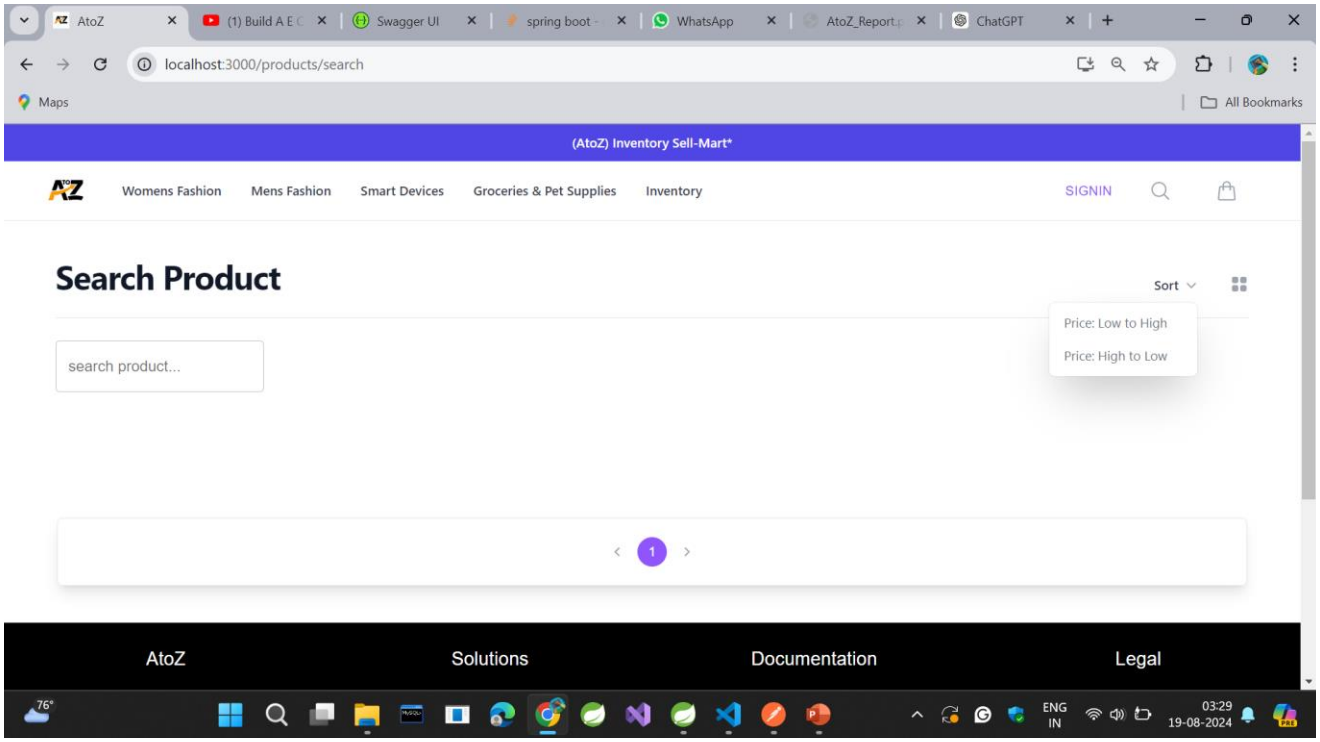
Registration:



Filtered Products:



Search Products:



6. CONCLUSION

In conclusion, Inventory Sell-Mart addresses key requirements for modern inventory and order management systems, providing a solid foundation for further development and improvement. Its modular design and adherence to security best practices position it well for continued success and adaptation in the evolving e-commerce landscape.

The **Inventory Sell-Mart** project exemplifies a sophisticated approach to inventory and order management by leveraging cutting-edge technologies and methodologies. This application is designed to meet the needs of both users and administrators, ensuring a seamless experience for managing products, orders, and payments while adhering to stringent security standards.

Its blend of modern technologies, robust security, and user-centric design positions it as a valuable tool for businesses seeking to enhance their e-commerce capabilities. The project's architecture and features provide a strong foundation for ongoing development, making it adaptable to future technological advancements and market trends.

7. REFERENCES

1. https://docs.spring.io/spring-security/reference/

2. https://docs.spring.io/spring-boot/index.html/

3. https://www.amazon.com/Agile-Software-Development-Principles-Patterns/dp/0135974445

4. Garcia-Molina, H., Ullman, J. D., & Widom, J. (2008). *Database Systems: The Complete Book*. Prentice Hall.

5. Codd, E. F. (1970). *A Relational Model of Data for Large Shared Data Banks*. Communications of the ACM, 13(6), 377-387.

6. Fowler, M. (2003). *Patterns of Enterprise Application Architecture*. Addison-Wesley.

7. Evans, E. (2003). *Domain-Driven Design: Tackling Complexity in the Heart of Software*. Addison-Wesley.

8. Soni, D., Nord, R. L., & Hofmeister, C. (1995). *Software Architecture in Industrial Applications*. Proceedings of the 17th International Conference on Software Engineering, 196-207.