MUTIKART

A PROJECT REPORT

For

Project (KCA451)

Session (2023-24)

Submitted by

MEHAK GARG

2200290140092

MUSKAN CHOUDHARY

2200290140096

Submitted in partial fulfillment of the

Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of

MS. MONIKA KANSAL

(Assistant Professor)

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS

KIET Group of Institutions, Ghaziabad

Uttar Pradesh-201206

(MAY 2024)

DECLARATION

I hereby declare that the work presented in this report entitled “Multikart", was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute. I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources. I affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name: Mehak Garg
Roll. No.: 2200290140092
Branch: MCA

Name: Muskan Choudhary
Roll. No.: 2200290140096
Branch: MCA

CERTIFICATE

Certified that **Mehak Garg 2200290140092 and Muskan Chooudhary-2200290140096** has carried out the project work having "**MULTIKART**" (**Final Project-KCA451**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision.

The project report embodies original work, and studies are carried out by the student herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date: **Mehak Garg (2200290140092)**

Muskan Choudhary (2200290140096)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Ms. Monika Kansal
Assistant Professor
Department of Computer Applications, KIET Group of Institutions, Ghaziabad

Dr. Arun Kumar Tripathi
Head
Department of Computer Applications, KIET Group of Institutions, Ghaziabad

ABSTRACT

The Multikart project aims to evaluate the performance and user experience of the Multikart e-commerce platform, which specializes in online clothing retail for men and women, featuring multiple brands. This study provides a comprehensive analysis of the platform's functionality, usability, and overall effectiveness in meeting customer needs. By leveraging various research methodologies, including user surveys, performance metrics, and comparative analysis with other e-commerce platforms, this project seeks to identify key strengths and areas for improvement in Multikart's operations. The research will assess user experience by evaluating navigation ease, search functionality, and overall user satisfaction, while also measuring performance metrics such as website loading times, transaction processing speeds, and responsiveness. Additionally, a comparative analysis with leading competitors will help identify Multikart's competitive advantages and disadvantages. Customer feedback will be gathered to gain insights into user preferences and pain points, and the diversity and availability of brands offered will be analysed. The outcomes of this project will provide valuable insights into the current state of the Multikart platform, guiding future enhancements to improve user satisfaction and competitive positioning in the e-commerce market.

ACKNOWLEDGEMENT

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Ms. Monika Kansal** for her guidance, help, and encouragement throughout my project work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi**, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help on various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Mehak Garg (2200290140092)

Muskan Choudhary (2200290140096

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Multikart is a dynamic e-commerce platform [1] designed to revolutionize online shopping. With a vast array of clothing options for both men and women from numerous brands, Multikart offers a seamless shopping experience. Its user-friendly interface, powered by Java Spring Boot for the backend, Angular for the frontend, and MongoDB [2] for the database, ensures efficient navigation and secure transactions. Multikart's commitment to quality and variety makes it a go-to destination for fashion enthusiasts. Whether you're looking for trendy outfits, accessories, or footwear, Multikart caters to diverse tastes and preferences. Its robust backend architecture ensures reliability and scalability, making it suitable for handling large volumes of customer interactions. Explore Multikart today and discover a world of fashion at your fingertips!

1.1 Project Overview

Multikart is an innovative e-commerce platform built on Java Spring Boot and MongoDB, offering a wide range of clothing options for men and women from various brands. The project aims to provide a seamless shopping experience with a user-friendly interface and secure transaction handling. Leveraging modern technologies, Multikart ensures reliability, scalability, and efficient management of customer interactions. It caters to diverse fashion preferences, making it a one-stop destination for online clothing shopping. Explore Multikart to enjoy a convenient and fulfilling shopping experience.

1.2 Background

During the pandemic Situation People found it very difficult to buy clothes for everyday use, elderly people, growing children, newborn babies, bedridden patient, etc.

There is a need for online Shopping Facility which bring customers and vendors on a common platform. These will save people from travelling to markets and establish a direct contact between clothes and accessories vendor.

We aimed to bridge the gap between sellers and buyers, facilitating easy transactions and improving user engagement. Multikart's development was driven by the need for a reliable, fast, and feature-rich e-commerce solution that could adapt to the evolving digital landscape, ultimately enhancing the overall shopping experience for users worldwide.

1.3 Key Features

Here are some key features of Multikart:

- **Wide Range of Brands**: Multikart offers clothing options from numerous brands, providing a diverse selection for shoppers.

- **User-Friendly Interface**: The platform is designed with an intuitive interface for easy navigation and a seamless shopping experience.

- **Secure Transactions**: Multikart ensures secure transactions, prioritizing the safety of customer data and payment information.

- **Robust Backend**: Built on Java Spring Boot and MongoDB, Multikart's backend architecture is robust, reliable, and scalable.

- **Customer Reviews and Ratings**: A system for customers to leave feedback and rate products, helping others make informed purchasing decisions.

- **Multi-Vendor Support:** A platform that allows multiple sellers to list their products, manage inventory, and handle orders independently, creating a diverse marketplace.

- **Responsive Customer Service**: Multikart provides responsive customer support to address queries and issues promptly, enhancing overall customer satisfaction.

- **Order Tracking and Management**: Real-time order tracking for customers and comprehensive order management features for sellers to handle shipments, returns, and exchanges.

- **Marketing and Promotion Tools**: Features for running promotional campaigns, discounts, and offers to attract and retain customers.

- **Advanced Search and Filtering**: Powerful search functionality with filters based on categories, price range, brand, and ratings to help users find products quickly and efficiently.

1.4 Objective of Project

The primary objective of Multikart is to provide a user-friendly and secure platform for online clothing shopping. It aims to offer a diverse range of clothing options

from various brands, catering to the fashion needs of both men and women. Multikart strives to ensure a seamless shopping experience through efficient navigation, secure transactions, and responsive customer service. Additionally, it aims to leverage modern technologies to enhance reliability, scalability, and overall customer satisfaction. And to Continuously innovate with new features and technologies to stay ahead in the competitive e-commerce market.

1.5 Scope

The scope of Multikart extends beyond being just an e-commerce platform; it represents a comprehensive solution for modern online shopping. With its wide range of clothing options from various brands, Multikart aims to cater to the diverse fashion preferences of customers globally. The platform's user-friendly interface and secure transaction handling make it accessible and trustworthy for users of all backgrounds. Multikart's robust backend architecture, powered by Java Spring Boot and MongoDB, ensures scalability to accommodate growing demands and maintain high performance levels.

The platform aims to support scalability, high availability, and robust security measures while ensuring a user-friendly interface and compliance with global standards and regulations. By leveraging cutting-edge technologies like AI, AR/VR, and blockchain, Multikart seeks to stay ahead of market trends and cater to a diverse, global customer base.

Additionally, Multikart's scope includes responsive customer service, reflecting its commitment to enhancing the overall shopping experience and fostering long-term customer relationships.

CHAPTER 2

FEASIBILITY STUDY

Feasibility study can help you determine whether or not you should proceed with your project. It is essential to evaluate cost and benefit of the proposed system. Various types of feasibility study are taken into consideration [3].

2.1 Market Analysis:

The e-commerce market is rapidly expanding, projected to reach $6.3 trillion globally by 2024, driven by increasing internet penetration and smartphone usage. Rising demand for personalized shopping experiences, Consumers prioritize convenience, fast delivery, and easy returns so multikart targets tech-savvy consumers aged 18-45 in urban and suburban areas. Key market trends include personalized shopping experiences, sustainability, and the rise of mobile and social commerce. While Multikart's strengths lie in its wide product range and user-friendly interface, it faces challenges in brand recognition and resource constraints.

Opportunities exist in market expansion and social media marketing, but it must navigate intense competition and cybersecurity threats. Multikart's growth strategy focuses on expanding product categories, improving customer service, and leveraging partnerships to build a loyal customer base.

2.2 Technical Feasibility:

Multikart's technical feasibility hinges on its robust infrastructure, utilizing scalable cloud solutions to handle increasing user traffic and expanding product listings efficiently. The platform's development leverages modern technologies such as microservices architecture for modularity, allowing seamless updates and maintenance. Employing secure payment gateways and adhering to stringent cybersecurity protocols ensures safe transactions. Advanced data analytics and AI are integrated for personalized recommendations, enhancing user experience. Furthermore, APIs enable smooth

integration with third-party services like logistics and payment processors. With a focus on mobile optimization, the platform ensures a seamless shopping experience across devices. While initial development requires significant investment, the long-term benefits include scalability, flexibility, and enhanced security, making the project technically viable.

2.3 Financial Feasibility:

Financial feasibility for Multikart hinges on several factors, including initial investment, revenue generation, and cost management. The startup requires substantial initial capital for website development, inventory acquisition, marketing, and logistics setup. Revenue streams will primarily come from product sales, complemented by affiliate marketing and premium membership subscriptions. Cost management is crucial, encompassing operational expenses like warehousing, shipping, employee salaries, and technology maintenance. Break-even analysis suggests that Multikart can achieve profitability within 2-3 years, provided it can scale its user base and maintain competitive pricing. Effective marketing strategies, efficient supply chain management, and leveraging economies of scale will be key to ensuring financial sustainability and growth.

2.4 Operational Feasibility:

Operational feasibility for Multikart hinges on its ability to effectively manage supply chain logistics, ensure a seamless user experience, and maintain robust IT infrastructure. Key operational aspects include establishing reliable partnerships with suppliers and logistics providers to ensure timely inventory replenishment and efficient delivery.

The platform's user-friendly interface and secure payment systems are essential for maintaining customer satisfaction and trust. Furthermore, the integration of advanced technologies, such as AI for personalized recommendations and robust cybersecurity measures, is crucial for operational success. While initial resource constraints and competition pose challenges, strategic planning, effective resource management, and continuous innovation can enhance Multikart's operational capabilities, ensuring smooth day-to-day operations and scalability for future growth.

2.5 Environmental Feasibility:

Environmental feasibility for Multikart involves evaluating the platform's impact on the environment and implementing sustainable practices. This includes assessing aspects such as packaging materials, energy consumption, waste management, and carbon emissions. Multikart can adopt environmentally friendly packaging options, such as recyclable or biodegradable materials, to reduce its carbon footprint and minimize waste generation. Implementing energy-efficient technologies and practices in data centers and warehouses can further reduce energy consumption and operational costs. Waste management strategies, such as recycling and proper disposal of electronic waste, can

also contribute to environmental sustainability. Additionally, Multikart can promote eco-friendly products and encourage suppliers to adhere to sustainable sourcing and production practices. By prioritizing environmental responsibility and implementing green initiatives, Multikart can not only reduce its environmental impact but also enhance its brand reputation and attract environmentally conscious consumers.

2.6 Cultural feasibility:

It deals with compatibility of the project with cultural environment. Cultural feasibility for Multikart involves evaluating its alignment with cultural norms, values, and preferences of its target audience. This includes considering factors such as language preferences, cultural sensitivities in marketing and product offerings, and inclusivity in catering to diverse customer demographics. Multikart must ensure that its branding, messaging, and product assortment resonate positively with different cultural backgrounds to foster trust, engagement, and loyalty among customers. Additionally, understanding local customs, holidays, and traditions can inform promotional strategies and customer engagement initiatives, enhancing cultural relevance and acceptance of Multikart in various markets. Striving for cultural sensitivity and inclusivity can contribute significantly to Multikart's success and acceptance within diverse cultural contexts.

Overall feasibility study of the project reveals that the goals of the proposed system are achievable.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Functional Requirements/Modules

Functional requirements are the backbone of any software or system, outlining the specific features and functionalities that it must possess to meet the needs of its users. These requirements define the behaviour of the system and specify what it should do, rather than how it should be implemented. Here are some common functional requirements:

3.1.1 User Registration and Login:

Users can access the registration page on the Multikart website and can easily create accounts by providing basic information such as name, email, and password. Users can also reset forgotten passwords through a secure process. The system validates the information provided, ensuring that the email address is unique and meets password complexity requirements [4].

Upon successful validation, the user's account is created, and they receive a confirmation email or message. Once registered, users can manage their profiles, view order history, and save preferences, enhancing their shopping experience on Multikart.

3.1.2 Product Catalogue:

A comprehensive catalogue of clothing items from various brands, including categories, filters, and search functionality for easy navigation are available. Each product is listed with a title, description, images, price, availability status, and specifications. Users can view multiple product images, read detailed descriptions, and check product ratings and reviews if available. Users can search for specific products using keywords or phrases in the search bar.

Multikart provides advanced filtering options, allowing users to refine search results based on price range, brand, size, colour, etc. Filters help users narrow down their choices and find products that match their preferences.

3.1.3 Wishlist:

The wishlist feature in Multikart allows users to save their favourite items for future purchase, providing a convenient way to track products they are interested in. Users can easily add or remove items from their wishlist with a simple click. From the wishlist, users can easily move items to their shopping cart for checkout when they decide to make a purchase. The system updates the cart accordingly, allowing users to proceed with the checkout process seamlessly.

Overall, the wishlist feature in Multikart enhances user experience by allowing them to curate a personalized list of desired products, receive updates, and easily transition from browsing to purchasing when ready.

3.1.4 Shopping Cart:

The shopping cart and checkout process in Multikart are designed to provide a seamless and efficient experience for users. The shopping cart allows users to add, remove, and modify quantities of items they wish to purchase, with a clear display of product details and total cost. The checkout process is streamlined, guiding users through entering shipping information, selecting payment methods, and reviewing order details. Multiple payment options, including credit/debit cards and cash on delivery, are supported to cater to diverse preferences. Multikart calculates the subtotal for all items in the cart, including taxes and shipping costs if applicable. Users can see the total cost of their order before proceeding to checkout. While viewing the cart, users can continue shopping and add more items to their cart. The cart dynamically updates to reflect any changes in quantities or new additions.

Overall, the shopping cart functionality in Multikart streamlines the buying process, allows users to review and adjust their orders before checkout, and enhances the overall shopping experience.

3.1.5 Order Management:

Order management in Multikart provides users with a comprehensive system to track and manage their purchases. Once an order is placed, users receive confirmation and can monitor the status through various stages, from processing to shipping and delivery. The platform offers detailed order history, allowing users to view past purchases and reorder items if desired. Notifications are sent for key updates, such as shipment dispatch and delivery status. This robust order management system enhances transparency and customer satisfaction by keeping users informed and in control of their orders.

3.1.6 Payment:

Multikart provides a secure and versatile payment system to facilitate smooth transactions for users. Users can choose from multiple payment options, including credit/debit cards, digital wallets, and cash on delivery, catering to diverse preferences. Additionally, users receive instant payment confirmation and can easily track payment status through their account.

3.1.7 Ratings:

Multikart's rating system allows users to provide feedback and ratings for products, enhancing transparency and helping other shoppers make informed decisions. Users can rate products based on their experiences, providing valuable insights into product quality, fit, and satisfaction. These ratings contribute to building trust and credibility within the Multikart community, empowering users to make confident purchase decisions.

3.2 Non-Functional Requirements

Non-functional requirements, also known as quality attributes or constraints, describe the characteristics of a system beyond its specific functionalities. Here are some non-functional requirements:

3.2.1 Performance:

Ensuring fast page loading times, minimal downtime, and efficient server response to handle peak traffic loads during sales events or promotions. The system should handle a large number of concurrent users without significant performance degradation. Performance metrics such as page load times, server response times, and transaction processing speeds should meet industry standards.

3.2.2 Security:

Multikart must ensure the security and confidentiality of user data, including personal information, payment details, and order history. The platform should use encryption protocols (e.g., HTTPS, SSL/TLS) to protect data in transit [5]. Implementing secure authentication mechanisms (e.g., strong passwords, two-factor authentication) is essential to prevent unauthorized access.

Regular security audits, vulnerability assessments, and penetration testing should be conducted to identify and mitigate potential security risks.

3.2.3 Usability:

Multikart should have an intuitive and user-friendly interface that is easy to navigate and understand. The platform's design should prioritize accessibility, ensuring that users with disabilities can access and use the site effectively.

Features such as search functionality, filtering options, and product categorization should be well-designed and responsive. User feedback and usability testing should be conducted regularly to identify areas for improvement and optimize the user experience.

3.2.4 Scalability:

Multikart should be scalable to accommodate growing user traffic, increasing product listings, and expanding functionalities. The system architecture should support horizontal scaling, allowing for the addition of servers or resources to handle increased demand. Load balancing techniques should be implemented to distribute traffic evenly and prevent server overload during peak periods. Scalability testing should be conducted to assess the system's ability to handle varying levels of workload and user activity.

3.2.5 Maintainability:

Multikart's codebase and infrastructure should be well-organized and documented to facilitate ongoing maintenance and updates. Version control systems (e.g., Git) should be used to manage code changes and collaboration among development teams.

Regular code reviews, code refactoring, and software updates are necessary to improve system performance, fix bugs, and address security vulnerabilities. Documentation should include system architecture diagrams, APIs, data models, and deployment procedures to aid in troubleshooting and future development efforts.

By addressing these non-functional requirements, Multikart can ensure a high-performance, secure, user-friendly, scalable, and maintainable e-commerce platform that meets the needs of its users and business goals.

CHAPTER 4

HARDWARE AND SOFTWARE SPECIFICATION

4.1 Hardware Specification

The hardware requirements for a Multikart project are relatively minimal but crucial for its functionality. Here's a breakdown of the hardware components needed:

- **Primary Servers:**
  - At least 4GB RAM (8GB recommended) for smooth performance.
  - Dual-core processors to handle various tasks efficiently.
- **Storage:**
  - Minimum 50GB SSD storage for fast read/write operations.
  - Backup servers with ample storage capacity for data redundancy.
- **Network Connectivity:**
  - Gigabit Ethernet connectivity for high-speed data transfer.
- **Load Balancer:**
  - Hardware load balancer to distribute traffic across servers evenly.
- **Operating System:**
  - Linux-based OS (e.g., Ubuntu Server, CentOS) for stability and security.4.2 Software Specification.

4.2 Software Specification

4.2.1 Frontend:

- **Operating System:**
  Any modern OS such as Windows, macOS, or Linux.
- **Web Browser:**
  Latest versions of popular browsers like Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge.

- **Angular Framework:**
  Angular CLI [6] for project scaffolding, development, and deployment and Latest stable version of Angular (e.g., Angular 12) for frontend development [7].
- **Node.js and npm:**
  Node.js runtime environment for executing JavaScript code outside a browser and npm (Node Package Manager) for managing frontend dependencies and packages.
- **TypeScript:**
  TypeScript as the primary language for writing Angular applications, providing strong typing and modern features.
- **HTML/CSS:**
  HTML5 for structuring web pages.
  CSS3 for styling and layout design.
  Bootstrap or Material Design for responsive UI components and layouts.
- **Angular Material:**
  Angular Material library for UI components, theming, and Material Design guidelines.
- **Text Editor or IDE:**
  Tools like Visual Studio Code, Sublime Text, or WebStorm for coding and development.
- **Git:**
  Version control system for managing codebase and collaboration.
- **TypeScript:**
  The primary language used in Angular development.
- **Angular Material:**
  Optional but recommended for UI components and styling.

4.2.2 Backend:

- **Operating System:**
  Linux-based OS (e.g., Ubuntu Server, CentOS) is commonly used for hosting Java applications, but Windows or macOS can also work.
- **Java Development Kit (JDK):**
  JDK 11 or higher is recommended for Java development with Spring Boot.
- **Spring Boot Framework:**
  Spring Boot [8] for creating RESTful APIs, handling business logic, and integrating with the frontend and Spring Data MongoDB for data access and interaction with MongoDB.
- **Maven or Gradle:**
  Maven or Gradle as build automation tools for managing dependencies and project build processes.
- **Integrated Development Environment (IDE):**

IntelliJ IDEA, Eclipse, or Visual Studio Code for coding, debugging, and testing Java applications or Spring Tool Suite (STS) as IDEs for Java development with Spring Boot .

- **Database:**
  MongoDB as the NoSQL database for storing data in a scalable and flexible manner.
- **Version Control:**
  Git for version control and collaboration among development teams.
- **Deployment Tools:**
  Docker for containerization, Kubernetes for orchestration, and Jenkins for CI/CD pipelines.

4.2.3 Database:

- **MongoDB:**
  MongoDB as the NoSQL database for storing product data, user information, orders, and other e-commerce-related data.

  MongoDB Compass or Robo 3T for database administration and management.

- **MongoDB Driver:**
  MongoDB Java driver for connecting Spring Boot backend with MongoDB database.

CHAPTER 5

SYSTEM DESIGN

5.1 Flowchart

A flowchart for a Multikart is designed to illustrate the various stages of interaction between the user and the Website [9], as well as the processes involved in fulfilling user requests. Here's a description of the flowchart:

- **User Interaction:**
  - User visits the Multikart platform and Choses to browse products.

- **Product Exploration**:
  - User navigates through product categories.
  - Selects a specific product for detailed view.

- **Viewing Product Details:**
  - User reviews product specifications, images, and pricing.
  - Considers adding the product to the shopping cart.

- **Adding to Shopping Cart:**
  - User clicks on "Add to Cart" for selected products.
  - Products are added to the shopping cart.

- **Shopping Cart Management:**
  - User reviews the items in the shopping cart.
  - Makes changes to quantities or removes items if necessary.

❖ **Proceeding to Checkout:**
- User decides to proceed to checkout.
- Clicks on the checkout button.

❖ **Checkout Process:**
- User is prompted to log in or create an account if not already logged in.
- Enters shipping information, billing details, and selects a payment method.

❖ **Payment Processing:**
- User confirms the order and proceeds to payment.
- Enters payment details (credit/debit card, digital wallet, etc.).

❖ **Order Confirmation:**
- System processes the payment and confirms the order.
- User receives an order confirmation email or notification.

❖ **Order Fulfilment:**
- Backend systems initiate order processing.
- Products are prepared for shipment.

❖ **Shipment and Delivery:**
- Order is shipped to the provided shipping address.
- User may track the order status through the platform.

❖ **Order Receipt and Feedback:**
- User receives the product.
- Optionally, leaves a review or rating for the purchased product.

❖ **Order Completion:**
- Order is marked as completed in the system.
- User may choose to continue shopping or log out.

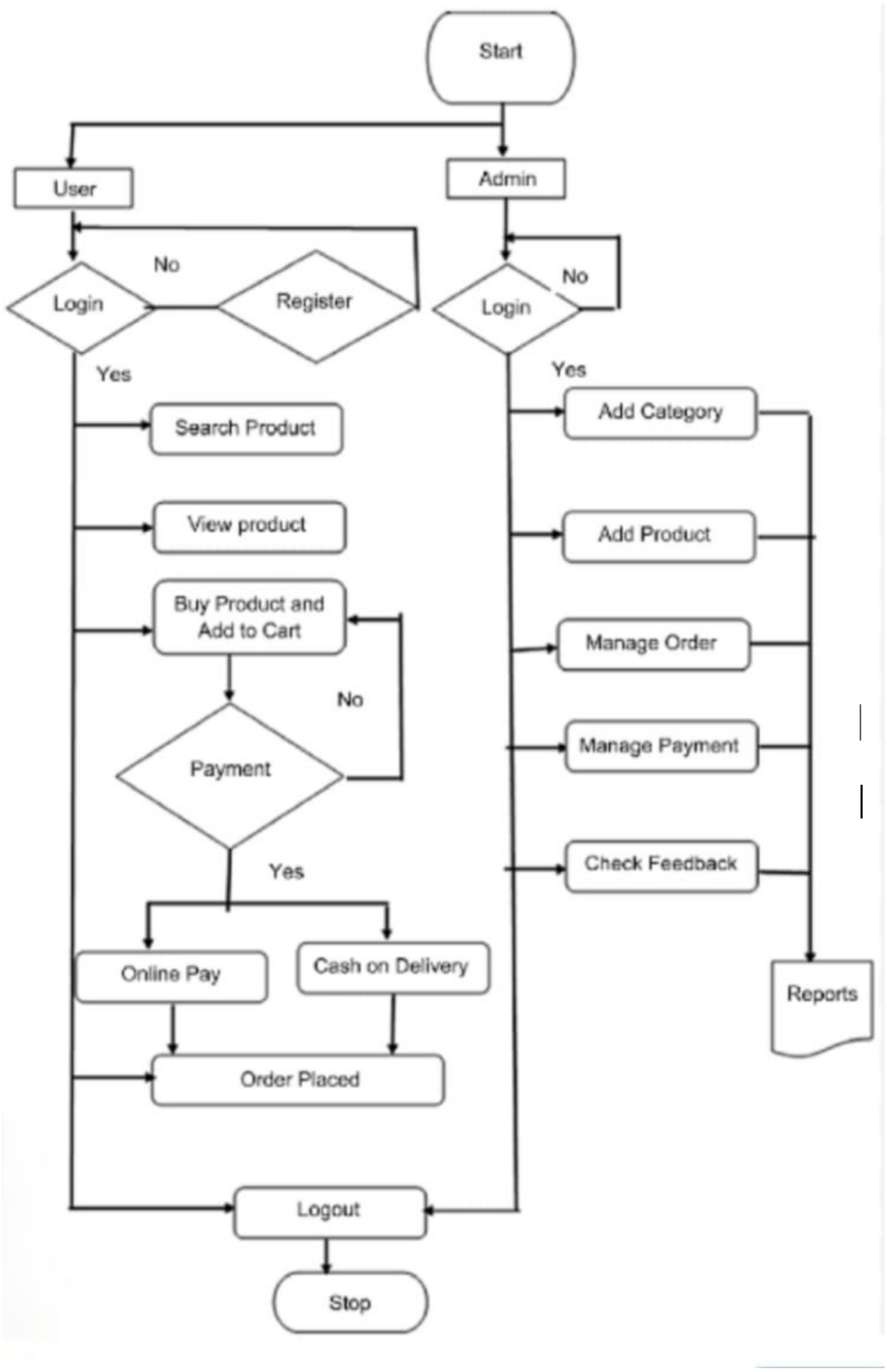

Fig 5.1: Flowchart

5.2 ER-Diagram

In this ER diagram:

- **Identify Entities:**

  Identify the main entities in Multikart, such as

  - User
  - Product
  - Order
  - Category
  - and Payment

- **2. Define Attributes:**

  For each entity, define attributes that describe them. For example:

  - User: UserID, FirstName, LastName, Email, Password, Address, Phone.
  - Product: ProductID, Name, Description, Price, StockQuantity, CategoryID.
  - Order: OrderID, UserID, OrderDate, TotalAmount, Status.
  - Category: CategoryID, Name, Description.
  - Payment: PaymentID, OrderID, PaymentDate, Amount, PaymentMethod.

- **3. Identify Relationships:**

  Determine the relationships between entities. For example:

  - User can place many Orders (One-to-Many).
  - Product belongs to one Category, but a Category can have many Products (Many-to-One/One-to-Many).
  - Order can have many Products (OrderDetails) and one Payment (One-to-Many).
  - Payment is associated with one Order (One-to-One).

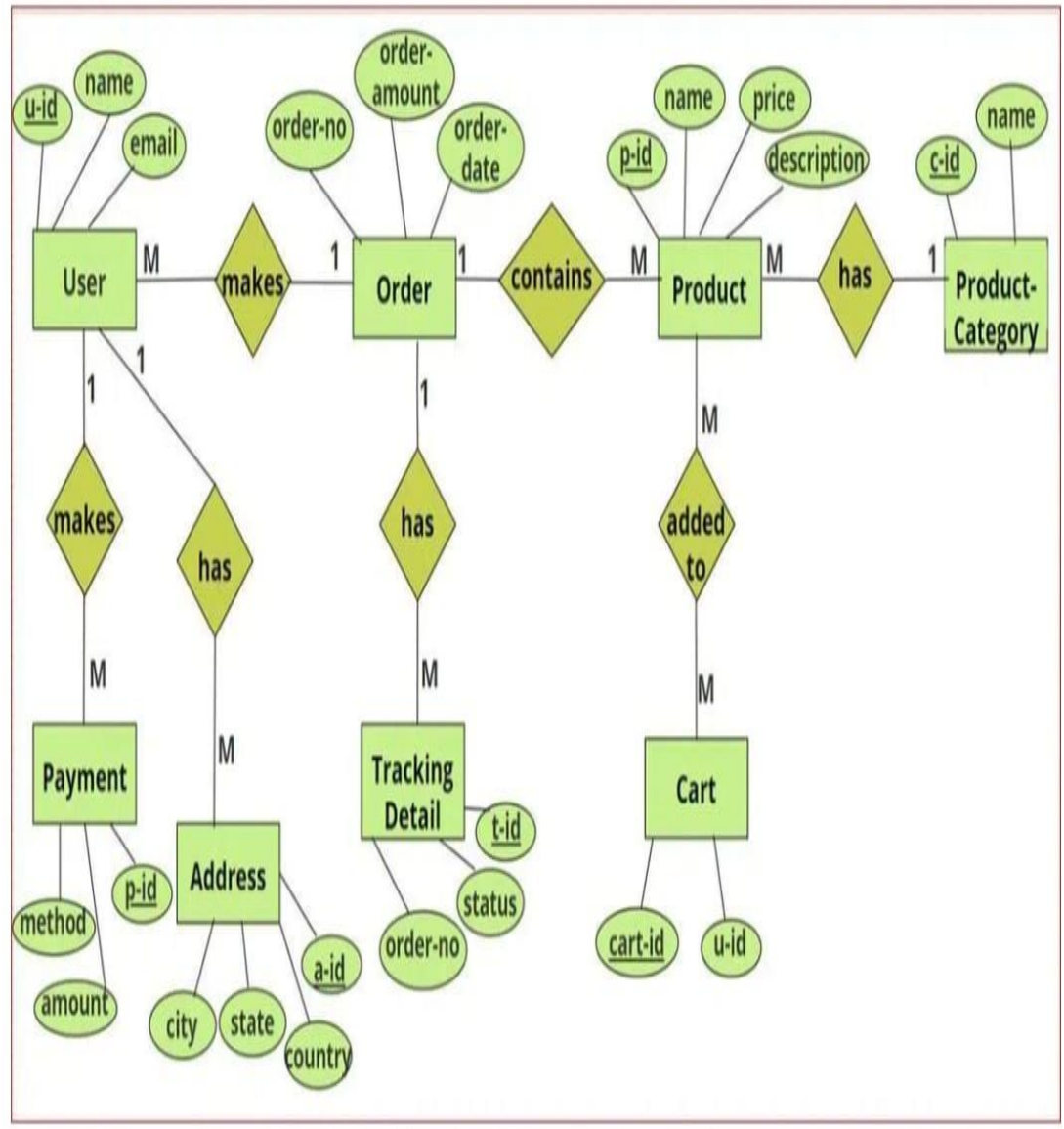

Fig 5.2: ER Diagram

5.3 Use Case Diagram

- **Identify Actors:**

Identify the actors interacting with the system. In Multikart, actors could include:

- ❖ User (Buyer)
- ❖ Admin (Administrator)
- ❖ Guest (Non-registered user)

- **Identify Use Cases:**

Determine the actions or tasks that each actor can perform within the system [10]. For example:

- ❖ User:
  - ✓ Search for Products
  - ✓ View Product Details
  - ✓ Add Product to Cart
  - ✓ Manage Cart
  - ✓ Checkout
  - ✓ Track Order
  - ✓ Leave Review/Rating

- ❖ Admin:
  - ✓ Add/Edit/Delete Product
  - ✓ Manage Orders
  - ✓ Manage Users
  - ✓ Generate Reports

- ❖ Guest:
  - ✓ Browse Products
  - ✓ Register/Login

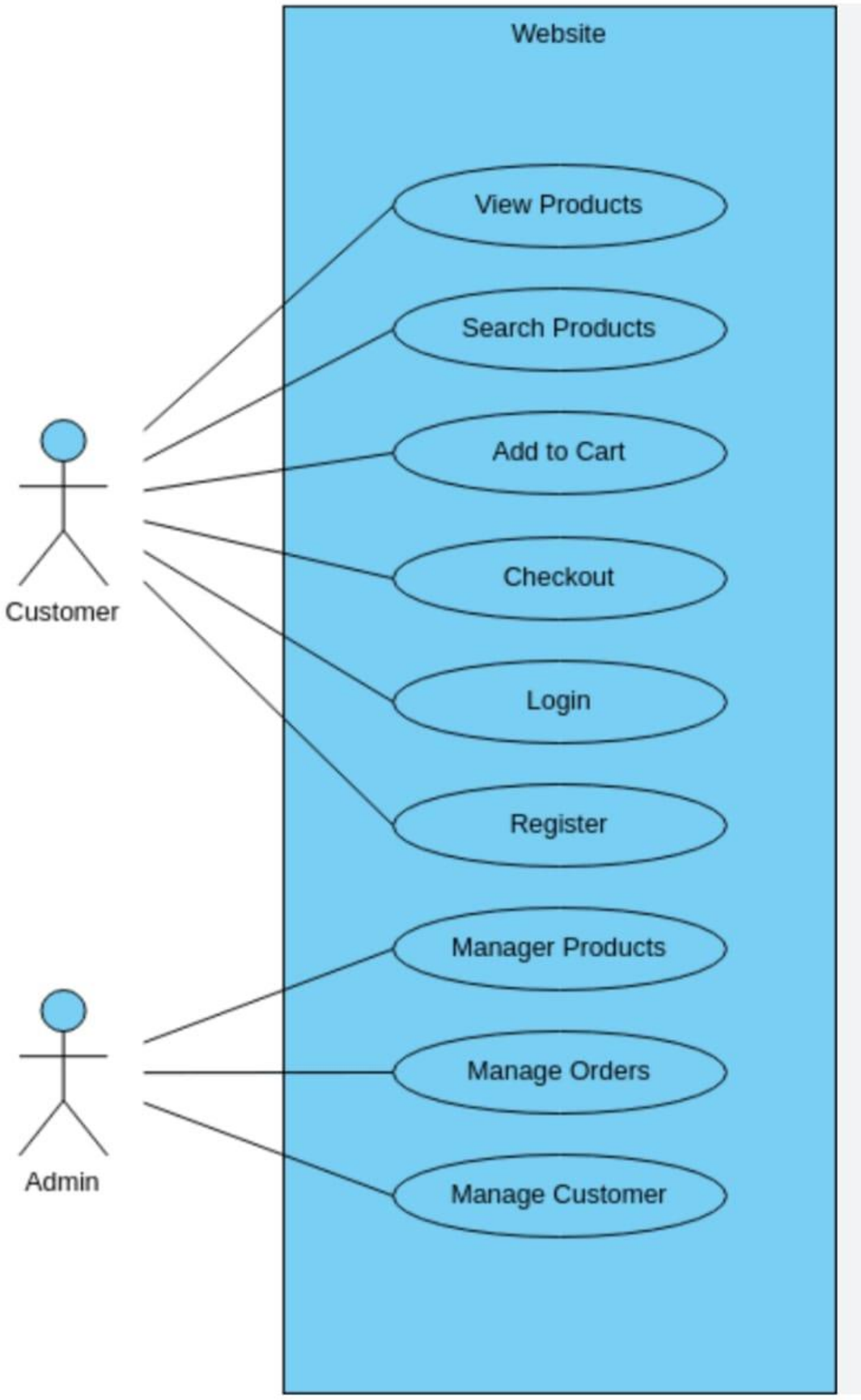

Fig 5.3: Use Case Diagram

CHAPTER 6

TESTING

Testing is a crucial aspect of developing a Project to ensure its reliability, accuracy, and performance. Here's an overview of testing methods that has employed in this Multikart project:

6.1 Unit Testing:

- Test individual components and functions of the codebase to ensure they work as expected.
- Use frameworks like JUnit for Java backend testing and Jasmine for Angular frontend testing [11].
- Unit tests can validate the functionality of user registration, login, and access control features.
- Test cases would cover scenarios such as valid credentials, invalid credentials, password resets, and role-based permissions.

6.2 Integration Testing:

- Test the integration of different modules and services to ensure they interact correctly. The focus is on verifying interactions and interfaces between different components to detect integration issues, data flow problems, and system-level defects.
- Verify data flow, API integrations, and communication between frontend and backend.

6.3 System Testing:

- It focuses on testing the system's behaviour, performance, reliability, security, and usability in a simulated production environment.
- Test the entire system to validate its functionality, including browsing products, adding to cart, checkout, and order management.
- Perform end-to-end testing of user workflows and scenarios.

6.4 Performance Testing:

- Performance testing is a type of software testing that evaluates how a system performs under different workloads and scenarios.
- Conduct load testing, stress testing, and performance profiling to identify and address performance bottlenecks.

6.5 Security Testing:

- Security testing for Multikart is crucial to ensure the protection of user data, transactions, and system integrity.
- The testing process includes assessing vulnerabilities, threats, and potential risks to the platform's security posture. This involves conducting various types of security tests, such as penetration testing, vulnerability scanning, authentication testing, authorization testing, encryption testing, and security configuration testing.
- The goal is to identify and mitigate security weaknesses, prevent unauthorized access, data breaches, and cyberattacks. Security testing also involves compliance checks with industry standards, regulations (such as GDPR, PCI DSS), and best practices for data protection and privacy. Multikart's security testing aims to build trust with users, enhance the platform's reputation, and ensure compliance with legal and regulatory requirements.
- Identify and mitigate security vulnerabilities such as SQL injection, XSS attacks, and data breaches.
- Conduct penetration testing, vulnerability assessments, and security audits to ensure data protection and system security.

6.6 Usability Testing:

- Evaluate the user interface (UI) and user experience (UX) to ensure intuitive navigation and user satisfaction.
- Define realistic user scenarios, such as searching for products, adding items to the cart, and completing checkout.
- Gather feedback from real users, conduct usability tests, and address usability issues to improve user experience.
- Measure the ease of completing common tasks, such as finding products, applying filters, and accessing account settings.
- Assess the clarity and efficiency of navigation menus, breadcrumbs, and links for seamless browsing.
- Test the responsiveness and usability of Multikart on different devices and screen sizes to ensure a consistent experience.

6.7 Regression Testing:

- Test the system after updates, changes, or bug fixes to ensure that new changes do not introduce new issues or regressions.
- Automate regression tests and create test suites to streamline testing processes.

- Regression test suites include automated tests using tools like Selenium or JUnit, as well as manual tests to validate user interfaces and complex workflows.
- Test data for regression testing includes typical user inputs, boundary values, invalid inputs, and scenarios that previously caused issues or defects.

6.8 Compatibility Testing:

- Compatibility testing for Multikart involves evaluating its functionality and performance across different devices, browsers, and operating systems to ensure a consistent and seamless user experience to ensure consistent performance and user experience.
- Verify responsiveness, layout, and functionality across various platforms and configurations.
- This testing verifies that Multikart is compatible with popular web browsers like Chrome, Firefox, Safari, and Edge, as well as mobile browsers on iOS and Android devices. It also includes testing on various screen sizes, resolutions, and orientations to ensure responsive design and layout adaptability.

6.9 Acceptance Testing:

- Acceptance testing for Multikart involves evaluating the e-commerce platform's functionality, usability, and compliance with business requirements to ensure it meets stakeholders' expectations [12].
- Conduct user acceptance testing (UAT) with real users or stakeholders to validate that the system meets business requirements and user expectations.
- Address feedback and make necessary adjustments before final deployment.

By performing these testing types thoroughly, Multikart can ensure a reliable, secure, and user-friendly e-commerce platform that meets quality standards and delivers a seamless shopping experience to its users.

CHAPTER 7

SCREENSHOTS

❖ **Home page (Bright Mode)**

It is the page where user land when first open the website, it contains

- ✓ The header features a clean, white background includes the Multikart logo, navigation menu, search bar, and account/login options for registered users.
- ✓ A search bar is centrally positioned, allowing users to quickly find products.
- ✓ On the left, navigation links to key sections such as "Categories," and on right, "My Account," "Wishlist," and "Cart" are easily accessible.
- ✓ A visually appealing banner section showcasing high-quality images of featured products, promotions, or brand messages displays featured products with large, clear images, product names, prices, and quick add-to-cart buttons.
- ✓ Newsletter subscription option for users to stay updated with the latest news, offers, and promotions [13].
- ✓ The white background ensures the product images and details pop, enhancing visual appeal and focus.
- ✓ The footer section provides links to important information such as "Customer Service," "About Us," "Privacy Policy," and "Contact Us."

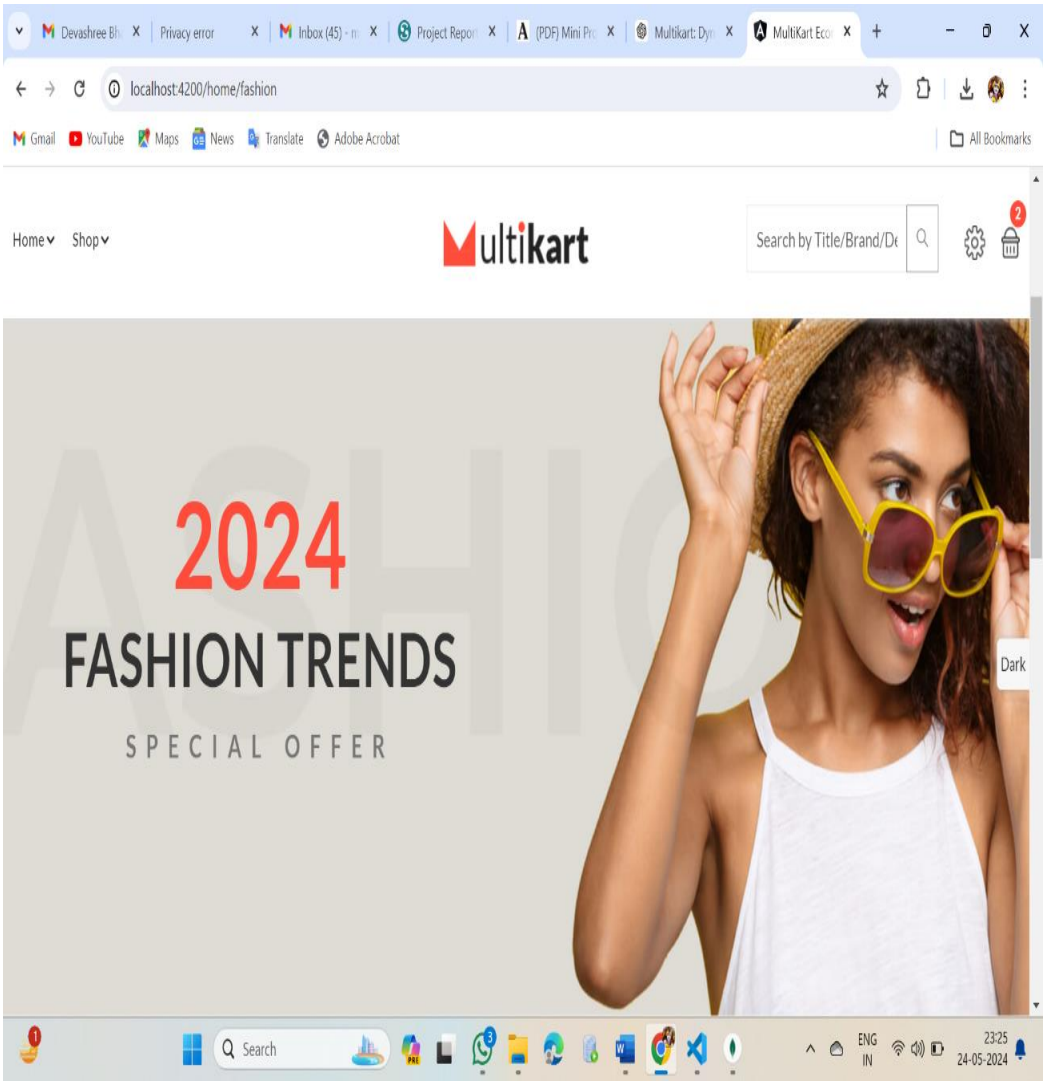

Fig 7.1: Home Page (Bright Mode)

❖ **Home Page (Dark Mode)**

- ✓ This is the same landing homepage just available in dark mode for eye relaxing or customizing your shopping experience.
- ✓ Here, with a combination of black themed homepage and white colour menu it is even more appealing but off course you can choose whichever mode you want to apply.

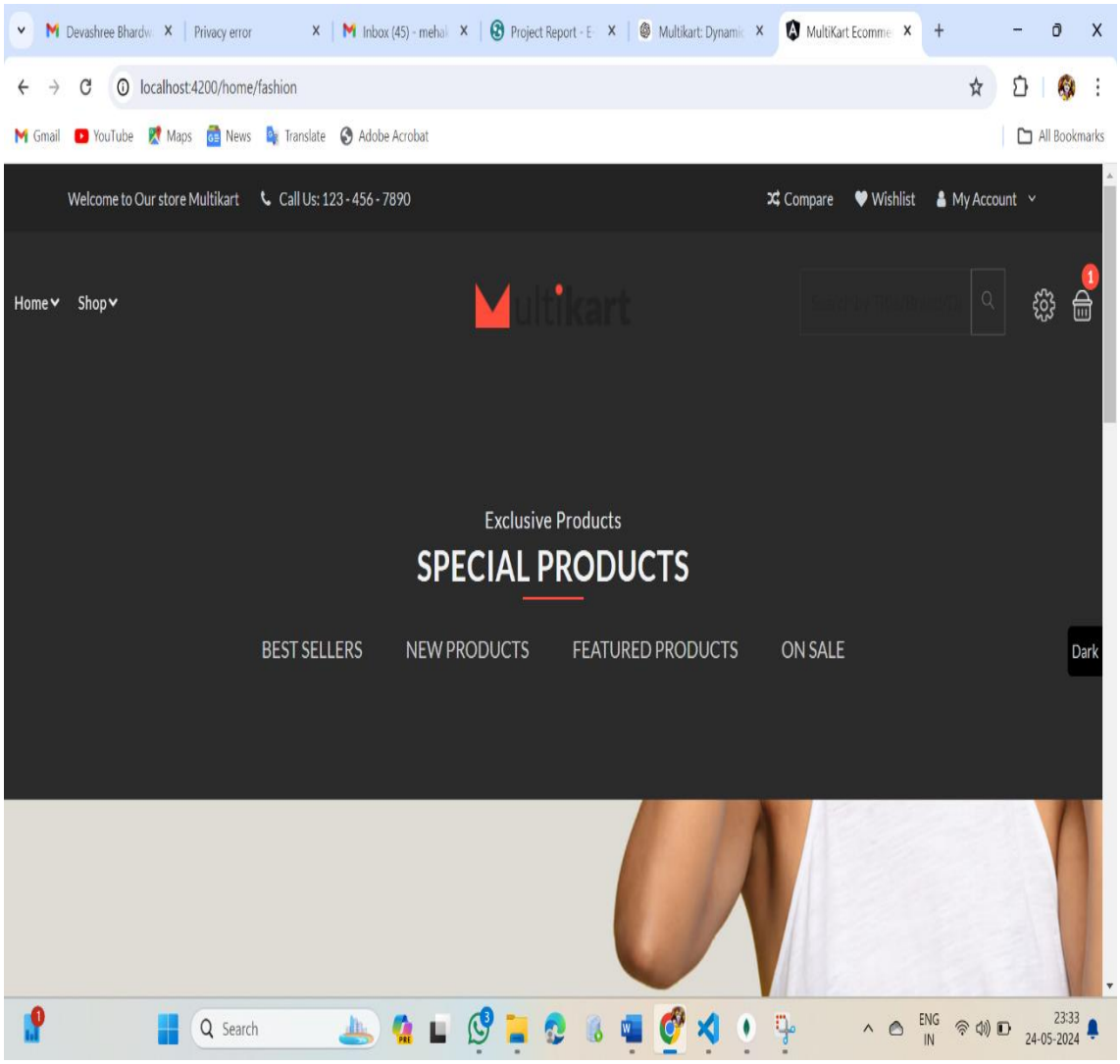

Fig 7.2: Home Page (Dark Mode)

❖ **Products (Women)**

- ✓ These are the Products available when you choose women from home page menu, all the available women clothing will show up.
- ✓ You can type in the search bar if you are finding something specific and then that particular product will show up.
- ✓ In the left side you can filter option there you can choose the brand, colour, size of product you want and then click on apply filter it will show only that particular product.

That is another way for searching specific product.

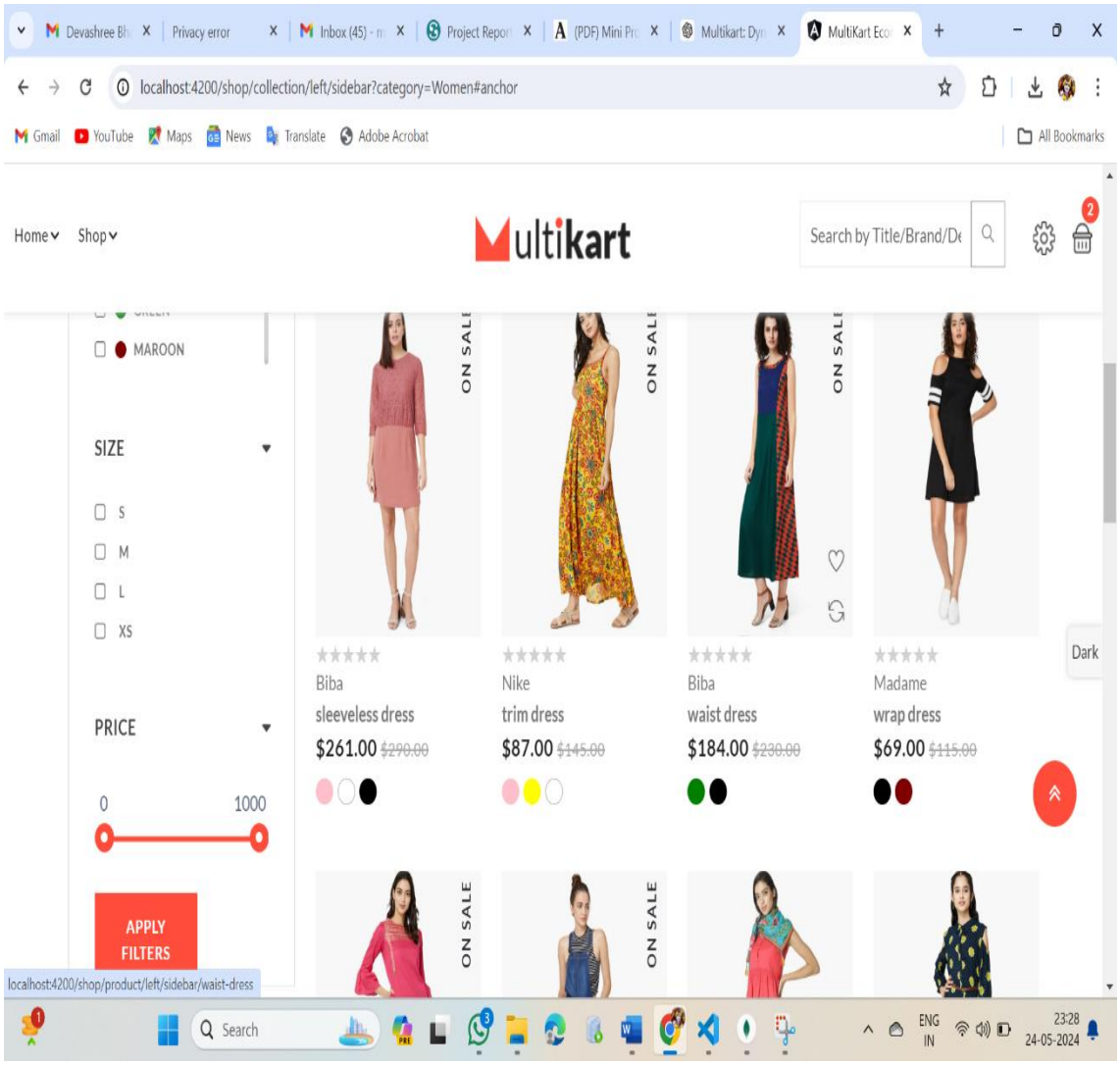

Fig 7.3: Products (Women)

❖ **Products (Men)**

✓ These are the Products available when you choose men from home page menu, all the available men clothing will show up.

Rest of the things are same as mentioned in women products.

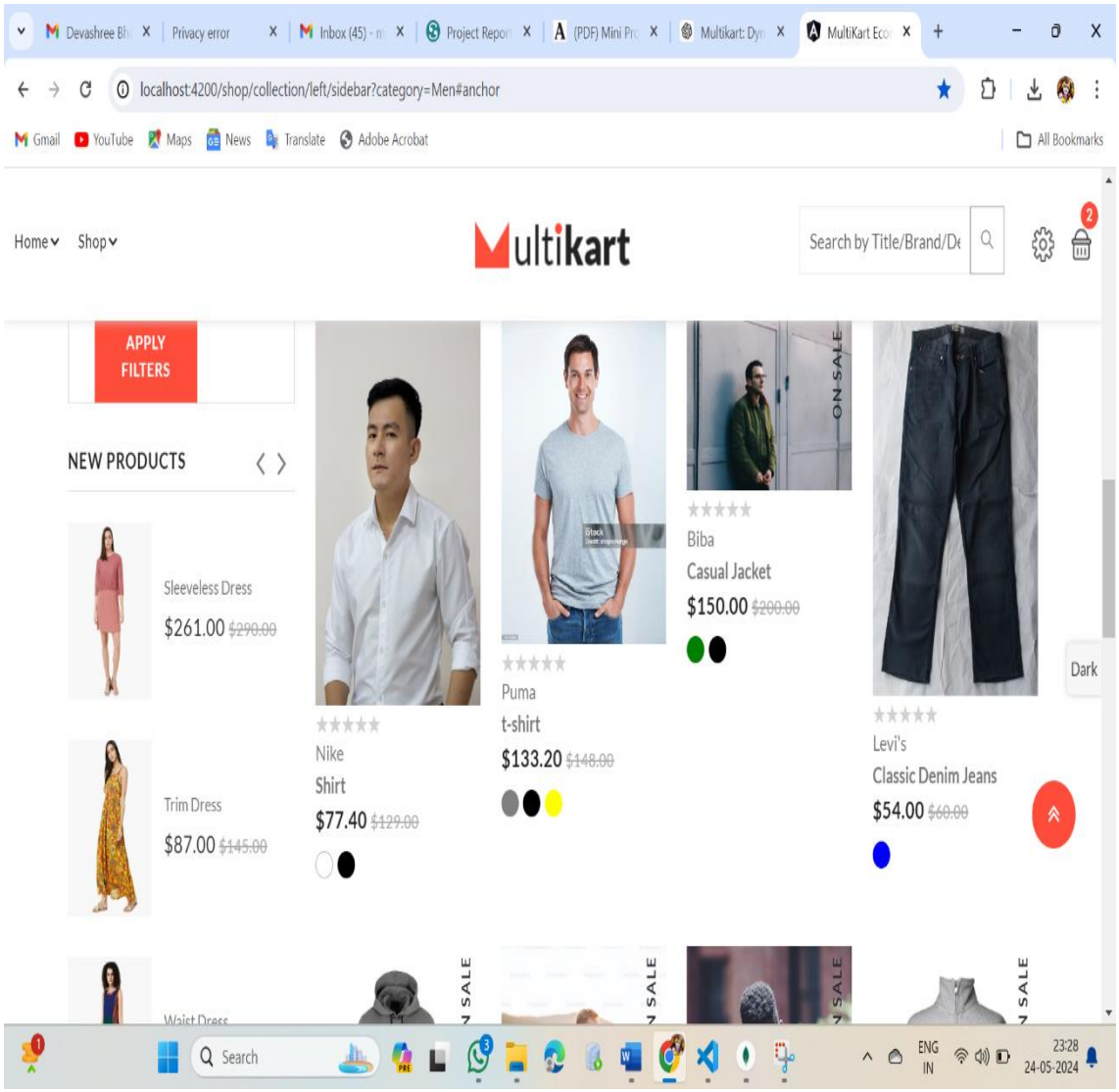

Fig 7.4: Products (Men)

❖ **Wishlist**

✓ With the wishlist feature, users can easily save products they're interested in with just a click.

✓ User need to click on wishlist option available at the right front corner of homepage.

✓ It allows users to plan their purchases and prioritize items they want to buy later. It's perfect for creating wishlists for special occasions or keeping track of items for future needs.

✓ Users can opt to receive notifications when items in their wishlist go on sale, come back in stock, or have any updates. This keeps them informed about changes and helps them make informed buying decisions.

✓ The wishlist is fully customizable. Users can add, edit quantities, or remove items as their preferences change and can checkout the items directly from there. It's a simple yet powerful tool for staying organized.

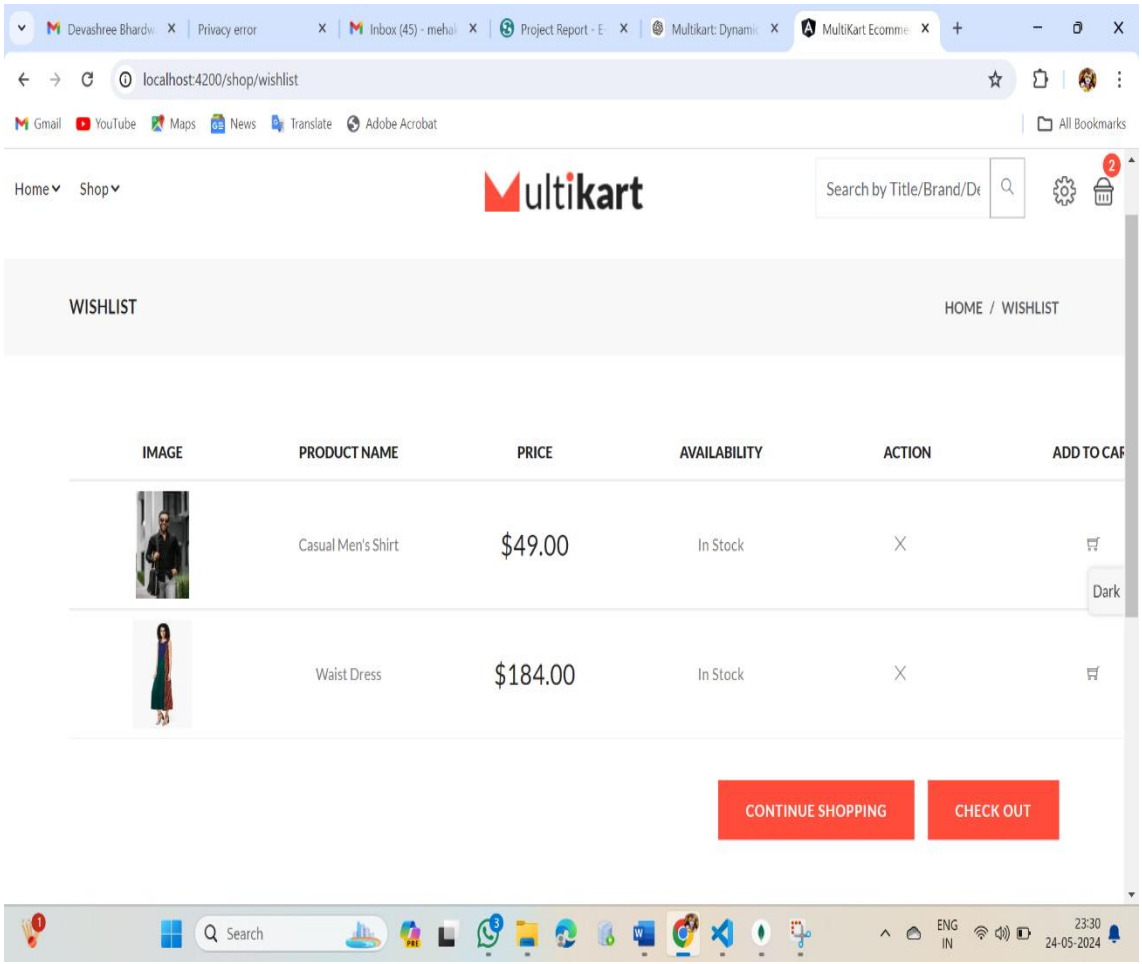

Fig 7.5: Wishlist

❖ **Cart**

✓ Users can add products to their cart with a single click while browsing the catalogue. Each product listing includes an "Add to Cart" button or icon.

✓ Users can easily view the contents of their cart, including product details, quantities, prices, and total cost. They can also edit quantities, remove items, or apply coupons/promotional codes.

✓ The shopping cart dynamically calculates the total cost based on the items added, including taxes, shipping fees, and discounts. This gives users a clear view of their expected expenses.

✓ Users can move items from their cart to a "Save for Later" section if they're not ready to purchase immediately but want to keep them for future consideration.

✓ When users are ready to checkout , they can proceed to the checkout page from the cart. Here, they enter shipping details, select payment methods, and review their order before completing the purchase.

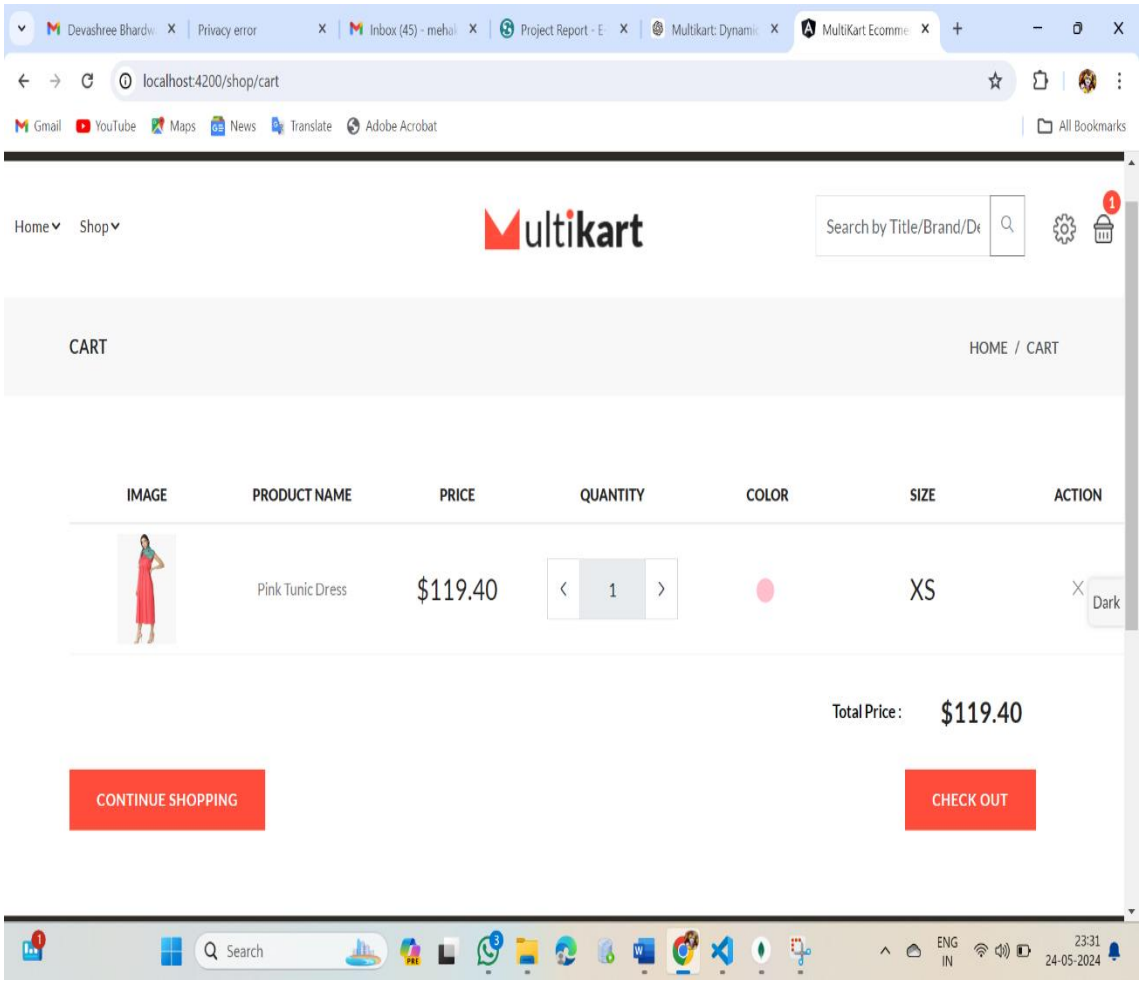

Fig 7.6: Cart

❖ **Checkout**

✓ Before proceeding to checkout, users can review the contents of their shopping cart, including product details, quantities, prices, and total cost. They can edit quantities, remove items, or apply any applicable discounts or promotional codes.

✓ Then add the details like Name, Phone, email, Country, Location, Pincode, State, so basically address details to which the product need to be delivered.

✓ Before finalizing the purchase, users have the opportunity to review their order summary, including the items in their cart, shipping details, selected shipping method, and total cost (including taxes and shipping fees).

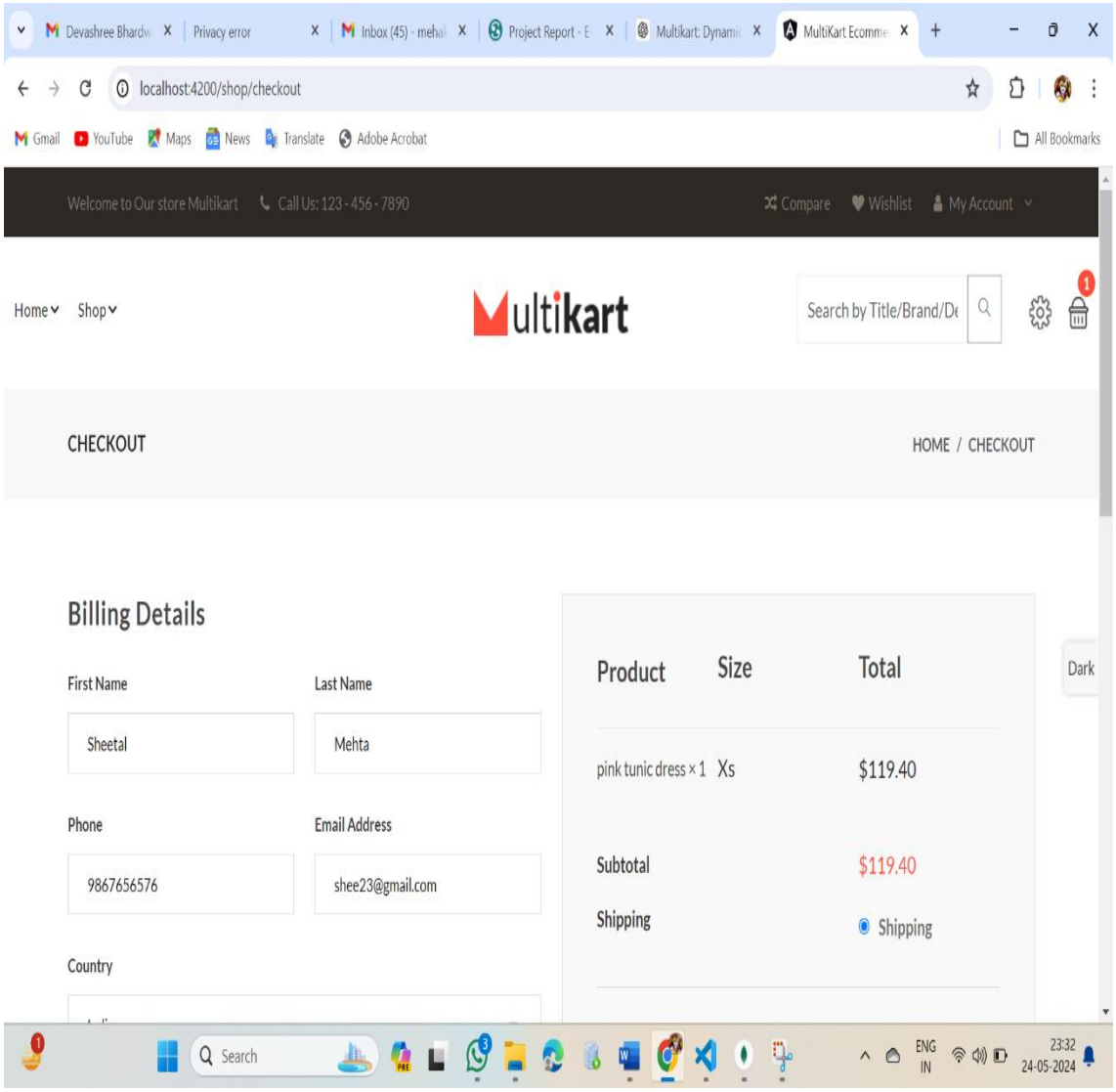

Fig 7.7: Checkout

❖ Place Order

After confirming all details and ensuring accuracy, users proceed to place their order. A confirmation screen is displayed, acknowledging the successful order placement and providing an order number for reference.

Users receive an order confirmation email or notification, confirming that their order has been successfully placed. The email includes order details, shipping information, estimated delivery date, and a link to track the order status.

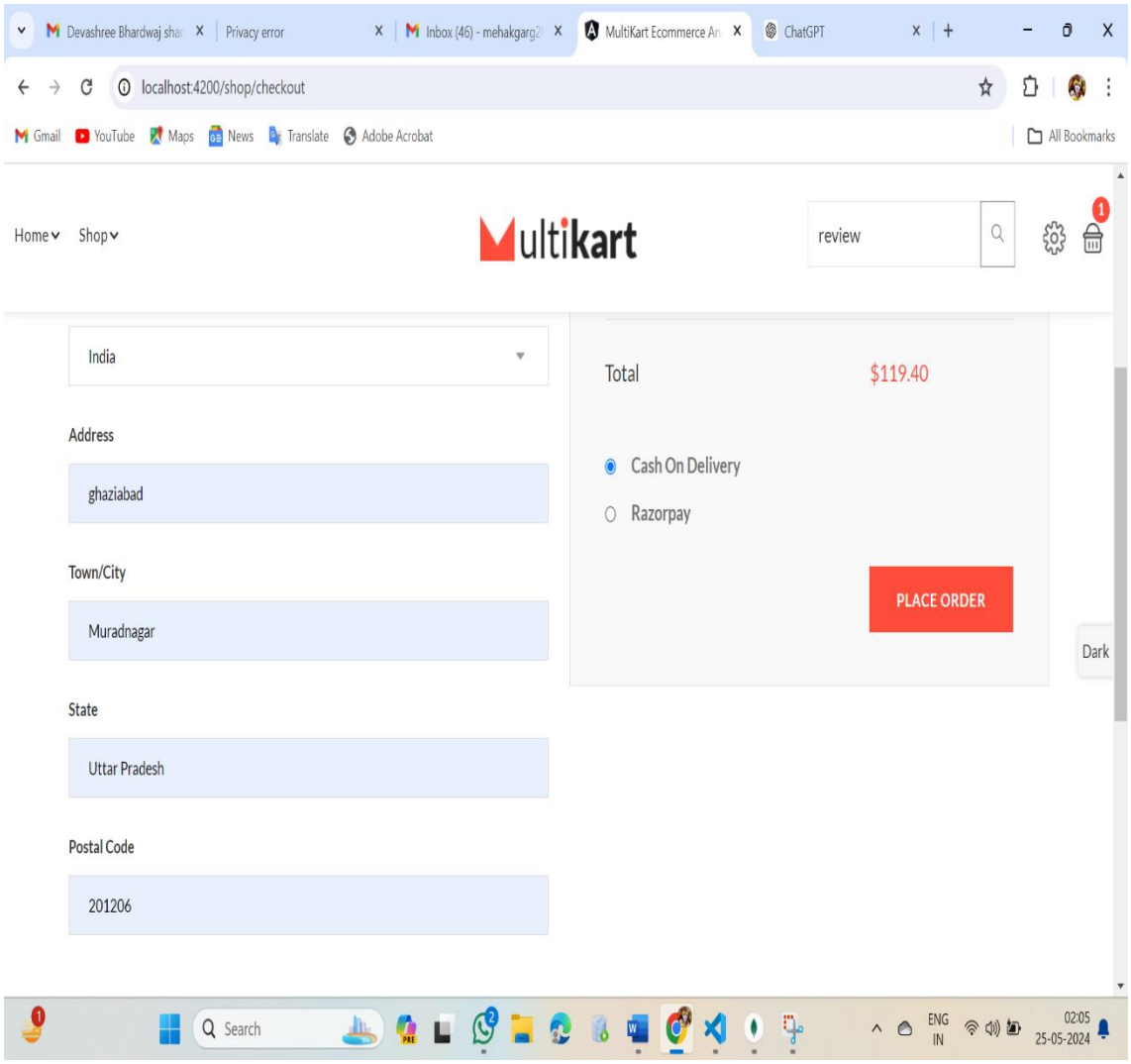

Fig 7.8: Place Order

CHAPTER 8

CONCLUSION & FUTURE SCOPE

6.1 Conclusion

In conclusion, Multikart represents a robust and user-friendly e-commerce platform designed to provide a seamless shopping experience for users. Through a combination of advanced technologies, including Angular for the frontend, Java Spring Boot for the backend with microservices architecture, and MongoDB for the database, Multikart offers a scalable and efficient solution for online retail.

Key features such as user registration and login, product catalogue management, shopping cart functionality, checkout process, order management, and user reviews and ratings contribute to the platform's overall functionality and user engagement. The use of microservices allows for better scalability, flexibility, and maintainability of the system, ensuring smooth operations even during peak loads.

Multikart also prioritizes security, performance, and usability through comprehensive testing methodologies, including unit testing, integration testing, system testing, performance testing, security testing, and usability testing. This ensures that the platform is secure, responsive, and user-friendly across different devices and platforms.

By leveraging modern technologies, adhering to best practices in software development and testing, and continuously improving user experience, Multikart stands as a competitive and reliable choice in the e-commerce landscape, catering to the needs of both customers and businesses alike.

6.2 Future Scope

The future scope for Multikart is promising, with opportunities [14] to enhance and expand the platform in several key areas :

- **Artificial Intelligence (AI) Integration:**
  Integrating AI technologies such as machine learning algorithms for personalized product recommendations, predictive analytics for inventory management, and chatbots for customer support can enhance user experience and operational efficiency.
- **Augmented Reality (AR) and Virtual Reality (VR):**
  Implementing AR and VR features for virtual product trials, interactive product showcases, and immersive shopping experiences can differentiate Multikart and attract tech-savvy consumers.
- **Blockchain Integration:**
  Exploring blockchain technology for secure transactions, transparent supply chain management, and counterfeit prevention can enhance trust, transparency, and security within the platform.
- **Enhanced Mobile Experience:**
  Focusing on optimizing the mobile experience with responsive design, progressive web apps (PWAs), and mobile-first strategies to cater to the growing trend of mobile shopping.
- **Subscription Services and Loyalty Programs:**
  Introducing subscription-based services, membership tiers, and loyalty programs to incentivize repeat purchases, enhance customer retention, and foster brand loyalty.
- **Sustainability Initiatives:**
  Incorporating eco-friendly practices, green packaging options, and sustainability-focused product lines to appeal to environmentally conscious consumers and align with corporate social responsibility (CSR) goals.
- **Voice Commerce and IoT Integration:**
  Exploring voice commerce capabilities with voice assistants like Amazon Alexa or Google Assistant, and integrating IoT devices for smart home integration, automated reordering, and personalized shopping experiences.
- **Global Market Expansion**
  International Shipping: Expand shipping capabilities to cater to a global market.
  Localized Shopping Experiences: Create region-specific versions of the platform to cater to local tastes and preferences.

By embracing these future-focused initiatives and staying abreast of emerging technologies and market trends, Multikart can continue to innovate, grow its user base, and maintain its position as a leading e-commerce platform [15] in the competitive digital landscape.

REFERENCES

The following links were searched and exploited extensively for the project development and implementation

[1]. https://www.coursera.org/in/articles/ecommerce-platforms

[2]. https://en.wikipedia.org/wiki/MongoDB

[3]. https://www.yo-kart.com/blog/10-parameters-to-evaluate-feasibility-of-your-ecommerce-business-idea/

[4]. https://www.geeksforgeeks.org/spring-security-registration-and-login-process/

[5]. https://scaledagileframework.com/nonfunctional-requirements/

[6]. https://github.com/nielsutrecht/jwt-angular-spring

[7]. https://github.com/sadanandpai/frontend-learning-kit

[8]. https://stackoverflow.com/questions/21083170/how-to-configure-port-for-a-spring-boot-application

[9]. https://www.figma.com/community/file/1289180191491829885/e-commerce-flow-chart

[10]. https://online.visual-paradigm.com/diagrams/templates/use-case-diagram/e-commerce-website-/;VPSESSIONID=A76F51CE43B98253A80C55D8975BA2B0

[11]. https://spring.io/guides/tutorials/rest

[12]. https://www.testingxperts.com/blog/ecommerce-user-acceptance-testing

[13]. https://www.optimonk.com/best-ecommerce-homepage-design/

[14]. https://www.optimonk.com/ecommerce-case-studies/

[15]. https://valasys.com/future-scope-of-ecommerce/