PERFECT CUTS

Submitted by

TARUN

2217115

MCA

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

Under the guidance of

Mrs.Suchetha Vijay

Assistant Professor, MCA Department

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

Mrs. Shwetha

Fabric fashion

near surathkal junction

Submitted to

ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY (AIMIT)

ST ALOYSIUS COLLEGE (AUTONOMOUS)

MANGALURU, KARNATAKA

2023

ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY

ST. ALOYSIUS COLLEGE (AUTONOMOUS)

BEERI, MANGALORE – 575 022

P G DEPT OF COMPUTER APPLICATIONS & SOFTWARE TECHNOLOGY

PROFORMA OF PROJECT PROPOSAL

Note : The student has to fill this form and approval has to be sought from the Dean of the faculty before starting the BCP

[1]. Student Register Number : 2217115

[2]. Student Name : TARUN

[3]. Title of the Project PERFECT CUTS

[4]. Name & Address of the Organization : Fabric fashion near surathkal junction.

[5]. Telephone No. : 8431676913.

[6]. E-mail : shwetha48@gmail.com.

[7]. App / URL : nill.

[8]. Address of the Project Site : Fabric fashion near surthkal junction.

[9]. Telephone No. : : 8431676913..

[10]. E-mail : shwetha48@gmail.com..

[11]. Residential Address of the Student

[12]. Telephone No : 9353980418.

[13]. Mobile Number : 9353980418.

[14]. E-mail : tarun120501@gmail.com

I hereby declare that the above furnished details are true to best of my knowledge and request you to permit me to take up the Business Consultancy Project prescribed for MCA III Semester.

Dated : Signature of the Student

The Proposal for the project work has been accepted

Dean of the Faculty

CERTIFICATE OF AUTHENTICATED WORK

This is to certify that the project report entitled PERFECT CUTS

submitted to Aloysius Institute of Management and Information Technology (AIMIT), St Aloysius College, Mangalore is an original work carried out by Mr./Ms._____TARUN___Register no. _____2217115____ under my guidance. The matter embodied in this project is authentic and is genuine work done by the student and has not been submitted whether to this University, or to any other University / Institute for the fulfilment of the requirement of any course of study.

Signature of the Student:

Date:.....

Name and Address

of the student

.....

.....

.....

Register No.....

Signature of the Guide

Date: .....

Name, Designation

and Address of the Guide

.....

.....

.....

Project Proposal Report / Synopsis

for

PERFECT CUTS

TARUN

2217115

MCA

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

25-10-2023

Under the guidance of

Mrs.Suchetha Vijay

Assistant Professor, MCA Department

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

Submitted to

ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY (AIMIT)

ST ALOYSIUS COLLEGE (AUTONOMOUS)

MANGALORE, KARNATAKA

I. Title of the Project

PERFECT CUTS.

II. Statement of the Problem

In today's fast-paced world, there is a growing demand for convenient and diverse clothing options. Perfect cuts, recognizes the need to expand its services to an online platform. However, the tailoring shop faces several challenges in implementing this transition effectively. Firstly, there is a lack of a user-friendly and efficient online Ordering system, resulting in customer frustration and decreased sales. Additionally, the absence of a dedicated delivery management system hinders timely and accurate deliveries, leading to customer dissatisfaction. Furthermore, the app lacks personalization, limiting upselling opportunities and customer engagement. Moreover, competition is fierce in the online cloth delivery industry, and perfect cuts needs a strategy to stand out and attract a loyal customer base. Finally, ensuring data security and privacy is vital in today's digital landscape. In sum, the problem at hand is to develop a comprehensive online tailoring shop app for perfect cuts that addresses these issues and delivers a seamless, secure, and delightful customer experience.

III. Why this particular topic chosen?

The topic perfect cuts app was chosen due to its universal appeal and the growing trend of cloths enthusiasts seeking diverse and authentic designs. By focusing on this topic, the app aims to cater to a wide range of designs and traditional comfort clothes. It fosters a sense of culinary adventure, offering users the chance to discover new cloths and experiences from the comfort of their homes. With an emphasis on quality, variety, and convenience, perfect cuts is committed to satisfying the cravings and curiosities of fashion lovers.

IV. Objective and Scope

OBJECTIVE

The objective of the perfect cuts app is to provide a seamless and delightful culinary experience for clothes enthusiasts. Our platform aims to connect users to the shop via online app, allowing them to effortlessly Order their favorite dresses for delivery. We prioritize convenience, quality, and variety, of dresses. Our commitment is to ensure customer

satisfaction through efficient delivery services and a user-friendly interface, making it the go-to destination for culinary exploration and dress delivery convenience.

SCOPE

Perfect cuts, an online dress delivery app dedicated to the tailoring shop, holds immense potential. With today's diverse clothing culture and tech-savvy population, there's a ready market for a platform that connects dress enthusiasts with variety of collection. By curating unique experiences, promoting unique dresses, and offering seamless delivery services,perfect cuts can capture a niche market and foster loyalty among users. Additionally, the growing trend of online dress Ordering and the city's vibrant clothing culture create a favorable environment for expansion and success. With the right strategy and quality service, this venture can thrive.

V. Methodology

1. Project Overview:

The objective of this methodology is to outline the process for developing dress delivery app for perfect cuts. The app will enable users to browse the menu, place Orders, and have their favourite dresses delivered to their desired location. The methodology encompasses the key steps involved in designing, developing the app.

2. Requirement Gathering and Analysis:

- Conduct meetings and interviews with perfect cuts stakeholders to understand their goals and expectations.
- Identify the target audience, their preferences, and their expectations from the dress delivery app.
- Gather information about perfect cuts menu, pricing, and delivery process.
- Analyse competitor's apps to identify best practices and potential areas for differentiation.

3. Planning:

- Define the project scope, objectives, and deliverables.
- Create a project timeline with milestones and deadlines.
- Determine the technology stack and frameworks to be used for app development.
- Allocate resources, including design and development team members.

4. Information Architecture and Wire-framing

- Develop a sitemap to outline the app's structure and navigation flow.
- Create wireframes to visualize the layout and user interface of key app pages.

5. UI/UX Design:

- Design the app's visual identity, including logo, colour scheme, and typography, aligned with Perfect Cuts guidelines.
- Create high-fidelity mock-ups of key app pages, focusing on usability, intuitive user experience, and responsive design.
- Incorporate feedback from stakeholders and conduct usability testing to refine the design.

6. Front-end Development:

- Convert the approved design mock-ups into XML code.
- Implement responsive design techniques to ensure optimal viewing experience across devices.
- Integrate front-end frameworks and libraries as necessary.

7. Back-end Development:

- Set up a secure and scalable server infrastructure for handling app operations and data management.
- The language we use to develop the backend is JAVA.
- Develop a database schema to store menu items, user profiles, Orders and other relevant information.
- Implement user registration, login/logout, and Order processing functionalities.
- Integrate payment gateway(s) to facilitate online transactions securely.

8. Testing:

- Conducting manual testing of all app functionalities, including menu browsing, Order placement, payment processing, and user account management.
- Identify and resolve any bugs or issues discovered during testing.

9. Post-Launch Support and Maintenance:

- Monitor app performance, user feedback, and analytics data.
- Address any reported issues promptly and perform regular updates and maintenance.
- Continuously optimize the app based on user feedback and emerging trends.

VI. Process Description

Module description for Perefect Cuts

1. User module

2. Admin module

1. User modules

- Registration/login: Newcomers have the option to sign up and create an account on the app, while those who have already registered can log in.
- Home Page: Users can view the home page of the CLJ-Culinary Lover's Junction.
- About Page: This page gives information about restaurant and the people.
- Contact Us: This page contains contact information about the restaurant.
- Order: In this section, users can explore the range of clothes offerings at Perfect Cuts, where they can access detailed descriptions and pricing information for each item.
- Add to Cart: This feature allows the users to add the clothes items to the cart and customize the quantity before checkout.
- Check out: This page allows users to fill their delivery address.
- Make payment: This module allows the users to make the payment either by debit/credit cards or cash on delivery.
- Reviews and ratings: This section allows the users to give their feedback.
- My account: In this module user can view his/her profile, update his/her profile and logout from account.
- My Order: This module enables users to track the status of their clothes delivery in real time. It may include features like live tracking, estimated delivery time, and notifications to keep users updated about their Order's progress.

2. Admin modules

- Product Management: The menu management module allows restaurant admin to create and update their menus. It includes features like adding menu items, setting prices, adding descriptions, and specifying any customization options.
- Order Processing and Confirmation: Once an Order is placed, this module manages the Order processing workflow. It includes features like Order confirmation, notifying the restaurant, and updating the user about the Order status.
- Manage payment: Keep track of the amount received.
- Reviews and Ratings Management: This module allows admin to view feedback of users to make decisions of improvement to the app or his restaurant.

VII. Resources and Limitations

Resources:

- Comprehensive Database: perfect cuts boasts an extensive database filled with a wide variety of clothes, encompassing menus, reviews, ratings, and other pertinent details.
- User-Generated Content:perfect cuts's database primarily thrives on content generated by users. This means that users can actively contribute their reviews, ratings.
- Advanced Search and Filter Features: perfect cuts offers advanced search and filter capabilities, allowing users to refine their searches based on parameters like price range, and ratings, enabling them to discover the best culinary options to meet their preferences.
- Delivery and Reservation Services: perfect cuts facilitates both cloth delivery and reservations, simplifying the process for users to Order their favourite dresses.

Limitations:

- Inconsistent Reviews: The quality of user-generated reviews can fluctuate, with some reviews potentially being biased or inaccurate, which can pose challenges for users in making well-informed decisions.
- Competition: The online dress delivery market is highly competitive, with numerous established players already operating in the sector. This intensifies the challenge for a new entrant like perfect cuts to establish a significant customer base.
- Infrastructure Requirements: Operating an online clothes delivery platform demands a robust and dependable technological infrastructure, including servers, databases, and payment gateways, which can be both expensive and time-consuming to develop and maintain.
- Delivery Logistics: Managing the logistics of cloth delivery, including delivery personnel, routes, and scheduling, can be intricate and demanding. Ensuring punctual and efficient deliveries can be particularly challenging, especially during peak hours or adverse weather conditions.
- Quality Assurance: Ensuring the maintenance of dress quality and packaging integrity during delivery can be a noteworthy challenge. Guaranteeing that dress is delivered safely and with intact packaging is vital for customer satisfaction.

- Legal and Regulatory Compliance: Compliance with a range of legal and regulatory requirements, such as dress safety and hygiene standards, tax regulations, and data protection laws, is essential for online dress delivery apps like perfect cuts Ensuring adherence to these requirements can be a substantial undertaking.

VIII. Testing Technologies used

Manual testing is an essential part of the testing process for an online cloth shopping app. Here are the steps involved in the manual testing process.

- Requirement Analysis: The initial phase is to meticulously examine the requirements and specifications specific to the perfect cuts app. Testers need a comprehensive understanding of the app's unique features and functions.
- Test Planning: Testers must formulate a test plan that delineates the scope of testing, the types of tests to be conducted, and the testing timetable, all tailored to the needs of perfect cuts.
- Test Case Development: Testers should create test cases that align with the specific requirements and characteristics of the perfect cuts app. These test cases should encompass all of the app's distinct features.
- Test Execution: Testers then proceed to execute the developed test cases, while diligently documenting any defects or issues they encounter during the testing process.
- Defect Reporting: Any defects or issues discovered during testing should be reported, utilizing a defect tracking system. Each defect should be documented in detail, with information about its severity and steps to reproduce the issue.
- Defect Resolution: The development team is responsible for addressing and rectifying the defects reported by testers. They must also verify that the fixes effectively resolve the issues.
- Retesting: Testers perform retesting to confirm that the defects identified have been effectively resolved, and that the perfect cuts app is functioning as intended.
- Regression Testing: To safeguard existing functionality, testers conduct regression testing, ensuring that the changes made during development and testing have not negatively impacted other parts of the app.

- Sign-Off: Once the testing process is complete and all defects have been addressed, testers provide their sign-off, signifying that the perfect cuts app is ready for release.

Consistent documentation and adherence to this manual testing process are crucial to ensuring that all aspects of the app are thoroughly tested, and that any defects or issues are identified and resolved promptly, thereby guaranteeing a smooth and reliable experience for culinary enthusiasts using the platform.

IX. Conclusion

In conclusion, perfect cuts is a remarkable online dress delivery app that caters to the diverse variety of culinary enthusiasts. With its wide variety of designed clothes, user-friendly interface, efficient delivery services, and commitment to quality, it thrives to become a go-to platform for dress lovers. Whether you crave exotic designs or comfort dress, perfect cuts has something to satisfy every palate. It's not just a app; it's a culinary journey that brings the different flavors to your doorstep, making dining a delightful and convenient experience for all.

Software Requirements

Specification

for

PERFECT CUTS

2217115

MCA

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

25-10-2023

Under the guidance of

Mrs. Suchetha Vijay

Assistant Professor, MCA Department

AIMIT, St Aloysius College (Autonomous)

Mangalore-575022

Submitted to

ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY (AIMIT)

ST ALOYSIUS COLLEGE (AUTONOMOUS)

MANGALORE, KARNATAKA

Table of Contents

Table of Contents

.....

Revision History

.....

1. Introduction2

1.1 Purpose

.....

1.2 Document Conventions

.....

1.3 Intended Audience and Reading Suggestions

.....

1.4 Project Scope…

.....

1.5 References

.....

2. Overall Description

2.1 Product Perspective

2.2 Product Features

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

3. System Features

3.1 System Feature 1

.....

3.2 System Feature 2 (and so on)

.....

4. External Interface Requirements

4.1 User Interfaces

.....

4.2 Hardware Interfaces

.....

4.3 Software Interfaces

.....

4.4 Communications Interfaces .....

5. Other Nonfunctional Requirements .....,

5.1 Performance Requirements .....
5.2 Safety Requirements.....
.....
5.3 Security Requirements .....
5.4 Software Quality Attributes .....

6. **Other Requirements**

Appendix A: Glossary
.....

Appendix B: Analysis Models
.....

Appendix C: Issues List
.....

Revision History

I. Introduction

1.1 Purpose

The purpose of "PERFECT CUTS" online dress delivery app is to provide a convenient platform for dress enthusiasts to explore and Order a diverse range of culinary delights.Perfect cuts is a system aimed to assist in management of tailoring activities within the industry. It will provide online services to customers such as: measurement submission to their tailors, check whether their garments are finished and also help in properkeeping of records.This will ensure availability of right information, information safety, easy storage, access and retrieval.The study aims at building a computerized tailoring management system that would be moreeffective and efficient than the existing manual system

1.2 Document Conventions

This Software Requirements Specification (SRS) document adheres to the following conventions and standards:

- Document Format: The SRS is presented in a standardized text format, using clear headings and sections for easy navigation and reference.
- Fonts: Standard fonts are used throughout the document for readability. No specific font has special significance.
- Prioritization: Each requirement statement is assigned a priority level to indicate its importance. Priorities are categorized as "High," "Medium," and "Low." Higher-level requirements are assumed to be inherited by detailed requirements, unless explicitly stated otherwise.
- Keywords: Keywords such as "must," "shall," "should," and "may" are used to indicate requirement importance and flexibility, following standard conventions in requirements engineering.
- References: Relevant external documents and standards are cited and listed in the "References" section (Section 1.5).
- Version Control: The document includes a version number and date to track revisions and updates**.**

1.3 Intended Audience and Reading Suggestions

Intended Audience:

This Software Requirements Specification (SRS) document is intended for various stakeholders involved in the development, deployment, and management of the "Krishna Traders E-Store." The primary audience includes:

1. **Developers:** Software developers and programmers responsible for building and implementing the e-commerce app.
2. **Project Managers:** Individuals overseeing project planning, scheduling, and execution.
3. **Quality Assurance/Testers:** Testing teams responsible for validating the system against specified requirements.
4. **User Interface (UI) Designers:** Design professionals responsible for creating the user interface of the app.
5. **System Architects:** Individuals involved in system design and architecture.
6. **Business Analysts:** Those responsible for understanding business needs and translating them into system requirements.

1.4 Project Scope

Perfect cuts will permit to register and deliver measurements to the tailor for the next process to follow.It also maintains clients' information and generating various reports about the tailor shop. The main users of the project are clients and system Administrator.It also enables customers to check the status of their garments i.e. if ready or not for collection.The system provides information about the cost, the fabric type the customer

want his/her dressknit from, the duration a customer wants the dress finished, the type of material to be used,quantity in terms of pairs needed and most importantly, the system computes the total cost andavails that information to the customer.However, online payment has not been achieved,but the customer is expected to pay either via mobile money transfer services like m-pesa, pesapal or cash when they come to pick their clothes.

1.5 References

C. Larman, APPLYING UML AND PATTERNS AnC. Larman, APPLYING UML AND PATTERNS An Introduction to Object-OrientedAnalysis and Design and Iterative Dev: Pearson Educatio005.

- C. Larman, APPLYING UML AND PATTERNS An Introduction to Object-OrientedAnalysis and Design and Iterative Development, 3rd ed., Massachusetts: Pearson Education, 2005.
- D. Carrington, CSSE3002 Course Notes, School of ITEE University of Queensland, 2008.
- IEEE Recommended Practice for Software Requirements Specifications, IEEE Standard 830, 1998.

2. Overall Description

2.1 Product Perspective

The Perfect cuts online dress delivery app is a web-based platform that offers a comprehensive range of features for browsing, reviewing, placing Orders, and managing the Order processing. It is accessible through internet browsers on various devices, including desktop and mobiles.

2.2 Product Features

- Allowing users to create accounts or log in using email.
- A search bar and filters to help users find dresses easily.
- Display of dresses menus with item prices, and images.
- Allowing users to add items to their cart, customize Orders, and place Orders for delivery.
- Enabling users to view, edit, and confirm their Orders before checkout through add to cart feature.

- Securely process payments online through various payment methods, such as credit/debit cards and cash on delivery.
- Allowing users to review uniquely designed dresses.
- Providing a means for users to contact customer support for assistance with Orders or issues.
- Offering a backend system for shop owners to manage their menus and Orders.
- Gathering feedback from users to improve the app's functionality and service.

2.3 User Classes and Characteristics

1. Users:

Characteristics:

- Individuals looking to Order dress online.
- They create accounts to save preferences and payment information.
- Browse menus, place Orders, and track deliveries.
- Provide reviews and ratings for dresses.

2. Administrator:

Characteristics:

- Platform operators responsible for overall management.
- Monitor and maintain the app.
- Resolve disputes and provide customer support.
- Create profiles with menus, pricing, and delivery options.
- Manage Orders, track inventory, and adjust menus.
- Access customer reviews and ratings.

2.4 Operating Environment

- **Hardware Platform:** The software is intended to run on standard web servers with hardware components that meet modern web hosting standards. This includes servers equipped with sufficient processing power, memory, and storage capacity to support the application's performance requirements.
- **Operating System:** The software is compatible with multiple operating systems, including but not limited to:
    - Microsoft Windows Server

    - Linux (e.g., CentOS, Ubuntu)
    - macOS Server
    - Web hosting environments (e.g., cPanel)
- **Web Server:** The application requires a web server to host the app. Commonly used web servers include Apache, Nginx, and Microsoft Internet Information Services (IIS).
- **Database Management System (DBMS):** The software utilizes a MongoDB database, and as such, it requires compatibility with MongoDB versions 5.0 or later. The DBMS manages data storage, retrieval, and processing for the application.
- **Web Technologies:** The application relies on web technologies such as HTML, CSS, JavaScript, and React for the user interface, as well as Node.js and Express.js for server-side operations.
- **Browser Compatibility:** End-users will access the system using web browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, Safari, and others. The software should be compatible with recent versions of these browsers to ensure a seamless user experience.
- **External Services:** The software interacts with external services, including payment gateways and shipping services. It should be capable of communicating with these services securely and efficiently.

Peaceful Coexistence:

The "PERFECT CUTS" is designed to peacefully coexist with other web applications and services running on the same server, ensuring that it does not disrupt the operation of other hosted applications.

2.5 Design and Implementation Constraints

1. **Corporate and Regulatory Policies:** Adherence to data protection, privacy, and security regulations.
2. **Hardware Limitations:** Meeting specified hardware requirements for optimal performance.
3. **Interfaces:** Integration with external applications following established protocols.
4. **Specific Technologies:** Use of MongoDB, React, and Node.js for development.
5. **Security:** Implementing industry best practices for data protection.
6. **Language:** Initial development in English; multilingual support as a future enhancement.
7. **Maintenance Responsibility:** Customer organization responsible for ongoing maintenance.

These constraints ensure compliance, performance, and security throughout the development process.

2.6 User Documentation

The "PERFECT CUTS" will be accompanied by a comprehensive set of user documentation components to assist users in effectively navigating and utilizing the platform. These components include:

1. **User Manuals:** Detailed user manuals will be provided in digital format, offering step-by-step instructions for various aspects of the application, such as registration, shopping, and Order tracking.
2. **Online Help:** The software will feature an integrated online help system, accessible directly within the application. Users can access context-sensitive help for specific tasks and features.
3. **Tutorials:** Interactive tutorials will be available to guide users through common processes, enhancing the onboarding experience for new users.
4. **Frequently Asked Questions (FAQs):** A compiled list of frequently asked questions and answers will be accessible on the platform, providing quick solutions to common user queries.

Delivery Format and Standards:

- User manuals and tutorials will be provided in PDF format for easy download and printing, ensuring accessibility for a wide range of users.
- Online help will be available in a responsive web-based format, accessible from various devices and screen sizes.
- All user documentation components will adhere to a consistent style and format to enhance usability and ensure a seamless user experience.

2.7 Assumptions and Dependencies

Assumptions:

1. **Third-Party Components:** The project assumes the reliable availability and compatibility of third-party components, including the MongoDB database and payment gateways, for seamless integration.
2. **Operating Environment:** Assumptions are made regarding the reliability and performance of the hosting environment, web server, and network infrastructure.
3. **Regulatory Compliance:** Assumptions include the understanding that the platform's compliance with data protection and privacy regulations is accurate and up to date.
4. **User Behavior:** User engagement, feedback, and interactions are assumed to align with typical e-commerce platform usage patterns.

Dependencies:

1. **External Services:** The project depends on external services, such as payment gateways and shipping providers, to facilitate secure financial transactions and Order fulfillment.
2. **Third-Party Components:** Dependencies exist on third-party components, including MongoDB and web server technologies, which are crucial for system functionality.
3. **Data Security:** The project depends on robust data security measures, as breaches could impact user trust and compliance with data protection regulations
4. **Regulatory Updates:** Dependencies on changes to data protection and privacy regulations could necessitate updates to the platform for continued compliance.

3. System Features

3.1 User Registration and Authentication

3.1.1 Description and Priority

- **Description:** This feature enables users to register for an account and log in securely.
- **Priority:** High

3.1.2 Stimulus/Response Sequences

- Users provide registration information.
- The system validates and creates user accounts.
- Users log in with their credentials.

3.1.3 Functional Requirements

- **REQ1:** User registration form with fields for username, email, password, etc.
- **REQ2:** User authentication for secure login.
- **REQ3:** Password encryption for user data security.

3.2 Product Browsing and Search

3.2.1 Description and Priority

- **Description:** This feature allows users to browse and search for Order efficiently.
- **Priority:** High

3.2.2 Stimulus/Response Sequences

- Users navigate product categories.
- Users utilize search functionality.
- The system displays product listings.

3.2.3 Functional Requirements

- **REQ4:** Product catalog with categories.
- **REQ5:** Search functionality with filters.
- **REQ6:** Product display with details and images.

3.3 Shopping Cart Management

3.3.1 Description and Priority

- **Description:** Users can add, remove, and modify items in their shopping carts.
- **Priority:** High

3.3.2 Stimulus/Response Sequences

- Users add items to the cart.
- Users adjust quantities or remove items.
- The system calculates cart totals.

3.3.3 Functional Requirements

- **REQ7:** Shopping cart management with add, remove, and update functions.
- **REQ8:** Calculation of cart totals.

3.4 Secure Payment Processing

3.4.1 Description and Priority

- **Description:** This feature enables secure online payment for purchases.
- **Priority:** High

3.4.2 Stimulus/Response Sequences

- Users select payment method.
- Users enter payment information.
- The system processes the payment securely.

3.4.3 Functional Requirements

- **REQ9:** Integration with secure payment gateways.
- **REQ10:** Encryption of payment data for security.

3.5 Order Tracking

3.5.1 Description and Priority

- **Description:** Users can track the status and location of their Orders in real-time.
- **Priority:** High

3.5.2 Stimulus/Response Sequences

- Users access Order tracking.
- The system provides Order status updates.
- Users receive Order location details.

3.5.3 Functional Requirements

- **REQ11:** Real-time Order tracking functionality.
- **REQ12:** Notification of Order status changes.

3.6 User Account Management

3.6.1 Description and Priority

- **Description:** Users can manage their accounts and profile settings.
- **Priority:** High

3.6.2 Stimulus/Response Sequences

- Users access account settings.
- Users update their profiles.
- The system saves user preferences.

3.6.3 Functional Requirements

- **REQ13:** User account settings and profile management.
- **REQ14:** Data persistence for user preferences.

3.7 Product Reviews and Ratings

3.7.1 Description and Priority

- **Description:** Users can submit reviews and ratings for Order.
- **Priority:** Medium

3.7.2 Stimulus/Response Sequences

- Users write product reviews.
- Users rate Order.
- The system displays reviews and ratings.

3.7.3 Functional Requirements

- **REQ15:** Review submission and rating functionality.
- **REQ16:** Display of reviews and ratings.

3.8 Customer Support and Feedback

3.8.1 Description and Priority

- **Description:** Users can contact customer support and provide feedback.
- **Priority:** Medium

3.8.2 Stimulus/Response Sequences

- Users access customer support.
- Users submit inquiries or feedback.
- The system handles inquiries and feedback.

3.8.3 Functional Requirements

- **REQ17:** Customer support interface.
- **REQ18:** Feedback submission and tracking.

4. External Interface Requirements

4.1 User Interfaces

The user interfaces for the "PERFECT CUTS" are designed for an intuitive and user-friendly experience, adhering to web design standards. Key features include:

- **Layout:** Clean and modern design with responsive layout.
- **Navigation:** Easy access to product categories, user accounts, and more.
- **Product Listings:** Clear presentation of product details.
- **Shopping Cart:** Efficient cart management and secure checkout.
- **User Account:** Profile management and Order tracking.

- **Search:** Quick product searches with clear results.
- **Feedback:** User-friendly customer support and feedback options.
- **Standard Elements:** Consistent buttons, error handling, and shortcuts.
- **Help:** Integrated help system for user guidance.

4.2 Hardware Interfaces

- **Device Types:** The software is designed to be accessible via standard web browsers on desktops, laptops, tablets, and mobile devices. It should be compatible with commonly used operating systems such as Windows, macOS, and various mobile platforms.
- **Data and Control Interactions:** The software interfaces with device hardware for data input, storage, and retrieval. This includes user inputs via keyboards, touchscreens, and mouse devices, as well as data storage and retrieval from hard drives and memory components.
- **Communication Protocols:** The software utilizes standard internet communication protocols, such as HTTP and HTTPS, to transmit data between the user's device and the server.

4.3 Software Interfaces

- **Operating Systems:** The software is designed to run on various operating systems, including but not limited to Windows, macOS, and Linux. It should be compatible with commonly used versions of these operating systems.
- **Web Browsers:** The E-Store is accessible through standard web browsers such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari. Compatibility is ensured for recent versions of these browsers.
- **Backend Services:** The software interfaces with server-side components, which include but are not limited to Node.js, Express.js, and MongoDB. These services handle data storage, retrieval, and server-side logic.
- **Database Management:** SQLite is the chosen database management system. The software interacts with the database to store and retrieve product information, user data, and Order details.
- **Communication Protocols:** The software utilizes standard internet communication protocols such as HTTP and HTTPS for data transmission.
- **Integrated Commercial Components:** Payment gateways and security services are integrated into the software to facilitate secure online payments and protect user data.
- **Data Sharing:** Data is shared between the frontend and backend components of the software. This includes user registration details, product listings, shopping cart contents, and Order information.
- **APIs and Libraries:** The software may utilize third-party APIs and libraries for functionalities like secure payment processing and geolocation services.

4.4 Communications Interfaces

- **Web Communication:** The primary communication channel is via web browsers using standard HTTP and HTTPS protocols. Users interact with the E-Store through web-based interfaces, making HTTP requests to access product listings, manage their shopping carts, and complete Orders.

- **E-Mail Communication:** The system may utilize e-mail for essential user communications, such as Order confirmation, Order tracking updates, and notifications. These e-mails will follow common e-mail standards and formats.
- **API Communication:** To ensure secure payment processing and other integrated services, the software communicates with third-party APIs and services using relevant protocols and message formats specified by these external services.
- **Data Transfer Rates:** Communication between the frontend and backend components occurs in real-time to provide a responsive user experience. Data transfer rates should be optimized to minimize user wait times during interactions such as product searches and cart updates.
- **Security and Encryption:** Communication, especially during payment processing and user authentication, is secured using encryption protocols (e.g., SSL/TLS). Data transmitted over the network is protected to safeguard user privacy and financial information.

Synchronization Mechanisms: Real-time synchronization between the user interface and backend services ensures that users receive timely information and updates regarding their Orders, product availability, and user account changes.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Response Time:

1. **Page Load Time:** Web pages, including product listings and checkout, should load within 3 seconds on average for standard internet connections. This ensures that users can access and interact with the platform quickly.
2. **Search Results:** Product search results should be displayed within 1 second, providing users with near-instantaneous access to product information.

Scalability:

1. **Concurrent Users:** The system should be able to handle a minimum of 100 concurrent users without a significant decrease in performance. Scalability should be ensured to accommodate increased user loads during peak periods.

Reliability:

1. **Availability:** The system should aim for 99% uptime, ensuring that users can access the E-Store without significant disruptions. Scheduled maintenance should be communicated in advance to minimize user impact.

Security:

1. **Data Protection:** Encryption protocols (e.g., SSL/TLS) must be employed to protect sensitive user data during transmission, ensuring data security.

Load Testing:

1. **Load Testing:** Regular load testing will be conducted to validate the system's performance under peak loads and to identify potential performance bottlenecks.

Caching:

1. **Caching:** Utilize caching mechanisms for frequently accessed data to reduce database queries and improve response times.

5.2 Safety Requirements

1. **Data Security:** Protect user information and payment details from unauthorized access or breaches.
2. **Transaction Security:** Ensure secure payment processing.
3. **User Privacy:** Adhere to data protection regulations.
4. **Secure Authentication:** Prevent unauthorized account access.
5. **Error Handling:** Minimize errors that may compromise data or safety.
6. **Compliance:** Meet industry-specific e-commerce and data security standards.
7. **User Guidelines:** Provide users with safety instructions and reporting procedures.
8. **Regular Audits:** Conduct periodic security assessments to identify and address vulnerabilities.

5.3 Security Requirements

1. **User Authentication:** Robust user authentication is required to ensure that only authorized individuals can access the system. Multi-factor authentication (MFA) may be implemented for sensitive transactions.
2. **Data Encryption:** All data transmitted between the user's device and the server must be encrypted using industry-standard protocols (e.g., SSL/TLS).
3. **Data Protection:** User data, including personal information and payment details, must be securely stored and protected against unauthorized access.
4. **Access Control:** Access to sensitive areas and user data should be strictly controlled and limited to authorized personnel.
5. **Security Audits:** Regular security audits and vulnerability assessments will be conducted to identify and address potential security weaknesses.
6. **Incident Response:** A robust incident response plan should be in place to address and mitigate security breaches or data leaks promptly.
7. **Compliance:** The system must adhere to industry-specific security and privacy regulations, such as GDPR, and should follow best practices.
8. **User Privacy:** Transparency about data collection, storage, and usage should be provided to users, along with clear privacy policies.

5.4 Software Quality Attributes

1. **Usability:** The system should be highly user-friendly, allowing customers to browse and make purchases with ease. A user satisfaction survey may be conducted to measure usability.
2. **Reliability:** The E-Store should be available and responsive at all times, with a target uptime of 99%. Downtime should be kept to a minimum.
3. **Maintainability:** The software must be designed for ease of maintenance and updates. Code should be well-structured, and documentation should be comprehensive.
4. **Scalability:** The system should easily scale to accommodate increased user loads, especially during peak shopping seasons.
5. **Interoperability:** The E-Store should be compatible with a wide range of web browsers and operating systems, ensuring broad accessibility.
6. **Security:** Robust security measures should protect user data and financial information. Compliance with industry-specific security standards is a priority.
7. **Performance:** The system should respond swiftly, with page load times of 3 seconds or less, and near-instant product search results.
8. **Testability:** The software should be designed for ease of testing, allowing for efficient identification and resolution of issues.

6. Other Requirements

1. **Internationalization:** The E-Store should be designed to support multiple languages and currencies to accommodate international customers.
2. **Legal Compliance:** The software must adhere to all relevant local, national, and international laws and regulations related to e-commerce, data protection, and user rights.
3. **Database Requirements:** The database should be designed for efficient data retrieval, storage, and management. It should be scalable to accommodate a growing product catalog and user base.
4. **Error Logging:** The system should log errors and exceptions for diagnostic and troubleshooting purposes, helping in the identification and resolution of issues.
5. **Documentation:** Comprehensive system documentation, including user manuals and developer guides, should be provided to assist users and developers.
6. **Backup and Recovery:** Regular data backups should be scheduled to prevent data loss. A disaster recovery plan should be in place to ensure data integrity.
7. **Reusability:** Code components should be designed with reusability in mind, allowing for the potential reuse of code in future projects.

Appendix A: Glossary

- **E-Store:** An online platform operated by Perfect cuts for the sale of dress.
- **SRS:** Software Requirements Specification - A document detailing the functional and non-functional requirements of the E-Store software.

- **GUI:** Graphical User Interface - The visual interface through which users interact with the E-Store.
- **API:** Application Programming Interface - A set of rules and protocols that allows different software applications to communicate with each other.
- **SSL/TLS:** Secure Sockets Layer/Transport Layer Security - Protocols for securing data transmission over the internet.

Appendix B: Analysis Models

1. Data Flow Diagram (DFD): The DFD illustrates the flow of data within the E-Store, including processes, data sources, data destinations, and data storage. It helps in understanding how information is processed and transferred within the system.
2. Entity-Relationship Diagram (ERD): The ERD depicts the relationships between different data entities and their attributes within the system. It provides insights into how data is organized and related.
3. Class Diagram: The class diagram defines the structure of the system in terms of classes, their attributes, and the relationships between them. It is instrumental in understanding the system's object-oriented design.

Appendix C: Issues List

1. **TBD1:** The specific encryption protocol for data security is yet to be determined.
2. **Pending Decision:** A decision is pending regarding the implementation of multi-factor authentication (MFA) methods.
3. **Information Needed:** Clarification is needed on the data retention policy for user account
4. **Conflict Awaiting Resolution:** There is a conflict in the system's error handling approach, and a resolution is required.

Software Design Document

For

PERFECT CUTS

TARUN

2217115

MCA

AIMIT

Mangalore

25/10/23

Under the guidance of

Mrs. Suchetha vijay

Professor, Department of MCA

AIMIT

Mangalore

Submitted to

ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY (AIMIT)

ST ALOYSIUS COLLEGE (AUTONOMOUS)

MANGALORE, KARNATAKA

1. Introduction

The SDD aims to provide an in-depth understanding of the system's structure, components, and functionality, ensuring that all aspects of the design are well-documented and align with the software requirements specified in the Software Requirements Specification (SRS).

Key objectives of this SDD include:

- Describing the system architecture, including its components, modules, and interactions.
- Presenting detailed design specifications for individual components.
- Explaining the system's data flow, processes, and user interfaces.
- Addressing the scalability, security, and performance considerations.
- Highlighting any design patterns, algorithms, and technologies employed in the system.

1.1 Purpose

The purpose of this Software Design Document (SDD) is to provide a detailed description of the software structure, software components, interfaces, and data essential for the implementation phase of the "Perfect Cuts". This document serves as a bridge between the Software Requirements Specification (SRS) and the actual implementation, offering a comprehensive guide to the system's design.

1.2 Scope

The Software Design Document (SDD) directly connects the Software Requirements Specification (SRS) to the development of the "Perfect Cuts" software. It establishes a clear link by mapping SRS requirements to corresponding design entities, ensuring that the design aligns with the specified needs.

The SDD covers the entire software, encompassing high-level architecture, component-level designs, user interfaces, data flow, security measures, performance considerations, and the use of design patterns and technologies. It serves as a roadmap for the development team during implementation, guaranteeing that the design meets requirements and aligns with the supermarket's objectives.

1.3 Definitions, Acronyms and Abbreviations

- SDD: Software Design Document, the document you are currently reading.
- SRS: Software Requirements Specification, the document that outlines the software requirements.
- Design Entities: The individual components, modules, interfaces, and data structures that form the software's design.
- High-level Architecture: The overall structure and organization of the software system.
- Component-level Designs: Detailed specifications for individual software components.
- User Interfaces: The graphical and interactive elements of the software that users interact with.
- Data Flow: The movement and processing of data within the software

3. Attributes of Design Entities

3.1 Identification

1. **User**: Represents the customers and administrators who interact with the e-commerce platform.
2. **Product**: Represents the items available for purchase in the online store.
3. **Shopping Cart**: Manages the items selected for purchase by the user.
4. **Order**: Represents the user's request to purchase selected items.
5. **Payment**: Handles payment processing for Orders.
6. **Inventory**: Tracks the stock levels of Order.
7. **User Profile**: Stores user-specific information, such as contact details and Order history.

3.2 Type

1. **User Interface Component**: Represents elements of the software's user interface, such as screens, forms, and navigation menus.
2. **Data Structure**: Defines the structure and format of data used by the software.
3. **Subsystem**: Refers to larger functional components or sections of the software.
4. **Service or Module**: Denotes individual software services, functions, or modules.
5. **Database Entity**: Represents tables, views, or data entities in the database schema.
6. **External Interface**: Describes interfaces with external systems, services, or APIs.
7. **Control Component**: Includes entities responsible for managing the flow and execution of processes.

3.3 Purpose

- **User Interface Component (e.g., Product Listing Page):** This entity was created to provide an interactive and user-friendly interface for customers to browse and select Order. Its primary purpose is to display product listings, facilitate search,

and enable users to view product details. It must meet performance requirements related to page load times and responsiveness as per the SRS.

- **Database Entity (e.g., Products Table):** The primary objective of this entity is to store and manage customer Orders. It is designed to meet the SRS requirements for Order data storage, retrieval, and security. It must ensure data integrity and comply with performance benchmarks for database transactions.
- **Service or Module (e.g., Payment Gateway Integration):** It serves the purpose of integrating with external payment gateways to enable secure and efficient online payments. This entity must adhere to the SRS specifications for payment processing, encryption, and transaction logging.

3.4 Function

- **User Interface Component (e.g., Product Listing Page):** The function of this entity is to transform user interactions (e.g., search queries, product selections) into visual representations of product listings and details. It enables users to explore Order and make informed choices based on their interactions.
- **Database Entity (e.g., Products Table):** This entity's function is to store and manage customer Order information, including Order details, customer data, and transaction records. It ensures data persistence and retrieval for Order processing.
- **Service or Module (e.g., Payment Gateway Integration):** The function of this entity is to transform payment requests from customers into secure, processed payment transactions. It encrypts payment data, communicates with external payment gateways, and logs transaction information.

3.5 Subordinates

The "Subordinates" attribute identifies the entities that make up the design entity. This information is crucial for tracing requirements to design entities and understanding the parent-child structural relationships through software system decomposition.

- **User Interface Component (e.g., Product Listing Page):** This entity may have subordinates such as search filters, product detail views, and customer reviews. These subordinates collectively contribute to the functionality of the product listing page.
- **Database Entity (e.g., Products Table):** Subordinate entities may include Order items, customer profiles, and transaction logs, which collectively constitute the Products Table's data structure.
- **Service or Module (e.g., Payment Gateway Integration):** Subordinate components may encompass encryption services, payment validation, and transaction logging, which are essential for the payment gateway integration module.

3.6 Dependencies

The "Dependencies" attribute defines the relationships between the design entity and other entities. It specifies the nature of each interaction, including initiation, Order of execution, data sharing, creation, duplication, usage, storage, or destruction of other entities.

- **User Interface Component (e.g., Product Listing Page):** This entity has dependencies on the product database entity for retrieving product information. It initiates data retrieval during user interactions and shares product data for display. It also depends on user input for search and selection.
- **Database Entity (e.g., Products Table):** Dependencies include data dependencies on user input and service modules, such as the payment gateway integration. It stores Order-related information, sharing data with other entities during Order processing.
- **Service or Module (e.g., Payment Gateway Integration):** This entity depends on user interactions for payment initiation and on external payment gateways for processing. It also stores transaction data in the Order database.

3.7 Interface

The "Interface" attribute explains how other entities interact with this design entity. It provides a description of the methods of interaction, including rules governing those interactions. This section outlines input ranges, the meaning of inputs and outputs, the type and format of each input or output, and any relevant output error codes.

- **User Interface Component (e.g., Product Listing Page):** This entity interfaces with users through web-based graphical user interfaces (GUIs). It accepts user input for search queries, product selection, and navigation. It returns visual representations of product listings and details as output. Input data may include search keywords, user preferences, and interaction events. Output includes rendered product images, descriptions, and user interface feedback.
- **Database Entity (e.g., Products Table):** The database entity interfaces with service modules and user interface components. It accepts and stores data related to customer Orders, including Order details, customer information, and transaction records. Interfaces with service modules include data retrieval and storage operations. Input data may include Order details, customer information, and transaction data. Output involves stored data retrieval.
- **Service or Module (e.g., Payment Gateway Integration):** This entity interfaces with user interface components and external payment gateways. It accepts payment initiation requests and returns transaction status. Input data includes payment details, user information, and transaction information. Output consists of payment processing results and transaction confirmation.

3.8 Resources

The "Resources" attribute identifies and describes all the external resources required by this design entity to perform its function. It includes information about physical devices, software services, and processing resources necessary for the entity's operation.

- **User Interface Component (e.g., Product Listing Page):** This entity requires resources such as web servers, internet connectivity, and graphical rendering libraries. It relies on the availability of web browsers on user devices. Processing resources are essential for rendering product images and user interface interactions.
- **Database Entity (e.g., Products Table):** Database management systems, storage devices, and data backup resources are required for this entity. The software services include database engines and query processing capabilities. Sufficient memory allocation and processing power are necessary for efficient data storage and retrieval.
- **Service or Module (e.g., Payment Gateway Integration):** External resources include secure communication channels with external payment gateways. This entity relies on payment gateway services for payment processing. It may also require encryption services for secure data transmission

3.9 Processing

The "Processing" attribute describes the rules and algorithms used by the design entity to achieve its intended function. It includes details about the sequencing of events, process steps, conditions, termination criteria, and any specific algorithms applied to perform a particular task.

- **User Interface Component (e.g., Product Listing Page):** This entity processes user input by parsing search queries, applying search algorithms, and retrieving relevant product data from the database. It handles user interactions, navigation, and the rendering of product listings. Processing also includes event handling for user actions, such as product selection and Order placement.
- **Database Entity (e.g., Products Table):** The database entity processes data storage and retrieval operations. It follows database management algorithms to store customer Order details, validate data integrity, and retrieve Order information upon request. Data indexing and search algorithms may be applied for efficient data access.
- **Service or Module (e.g., Payment Gateway Integration):** This entity processes payment transactions by communicating with external payment gateways. It applies encryption algorithms for secure data transmission and employs payment processing logic to handle payment requests, authorization, and transaction confirmations.

3.10 Data

The "Data" attribute describes how internal data is represented, its initial values, utilization, format, and acceptable values within each design entity.

- **User Interface Component (e.g., Product Listing Page):** This entity represents data related to product listings, user interactions, and user preferences. Initial values include empty search queries or default settings. Data is presented in a visual format, including product images, descriptions, and pricing. Acceptable values for user interactions include valid search queries and user selections.
- **Database Entity (e.g., Products Table):** The database entity stores data related to customer Orders, including Order details, customer information, and transaction records. Initial values may be null or empty. Data is stored in structured tables with predefined formats. Acceptable values are based on data integrity constraints, such as valid Order information and unique identifiers.
- **Service or Module (e.g., Payment Gateway Integration):** This entity deals with transaction data, including payment details, user information, and transaction information. Initial values may include empty or default payment information. Data formats conform to payment gateway requirements. Acceptable values are defined by payment processing standards, ensuring valid and secure transactions.

4. Decomposition Description

4.1 General Structure

1. **User Registration and Login Feature**
    - **User Registration Component**
        - **Identification:** REG-01
        - **Type:** User Interface Component
        - **Purpose:** To facilitate user registration and capture user information.
        - **Function:** Manages user registration forms, user data validation, and communication with the database.
    - **User Login Component**
        - **Identification:** LOG-01
        - **Type:** User Interface Component
        - **Purpose:** To authenticate users and grant access to their accounts.
        - **Function:** Manages user login forms, authentication processes, and session management.
2. **Product Catalog Feature**
    - **Product Catalog Component**
        - **Identification:** CAT-01
        - **Type:** User Interface Component
        - **Purpose:** To display and organize Order for user browsing.
        - **Function:** Presents product listings, filters, and navigational options.
3. **Shopping Cart Feature**
    - **Shopping Cart Component**
        - **Identification:** CART-01

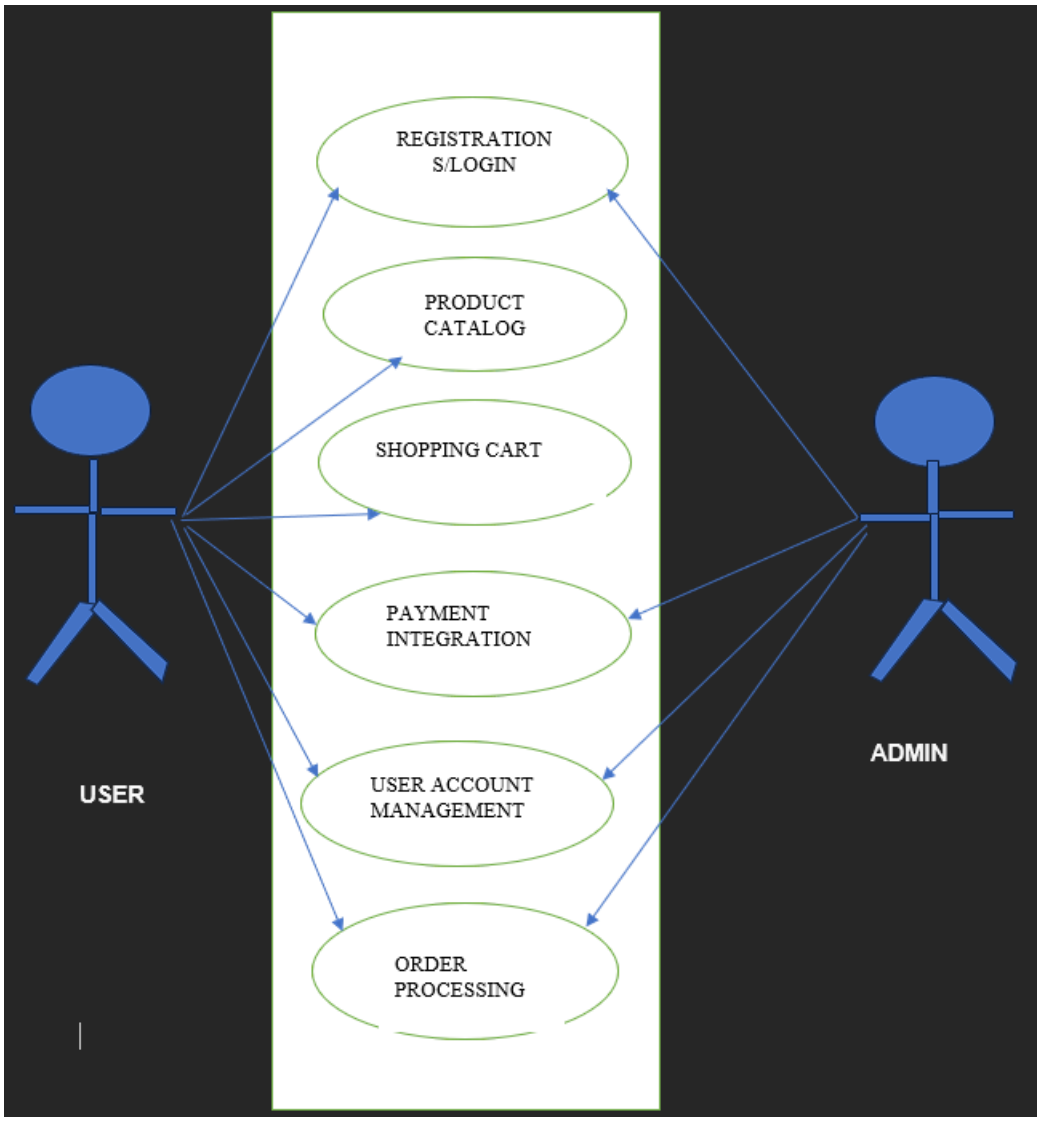
        - **Type:** User Interface Component
        - **Purpose:** To enable users to add and manage selected Order for purchase.
        - **Function:** Manages the shopping cart view, item addition/removal, and Order calculation.
4. **Order Processing Feature**
    - **Order Processing Component**
        - **Identification:** ORD-01
        - **Type:** Service or Module
        - **Purpose:** To process customer Orders, validate Order data, and prepare Orders for payment.
        - **Function:** Validates Order details, calculates totals, and prepares Order data for payment processing.
5. **Payment Integration Feature**
    - **Payment Gateway Integration Component**
        - **Identification:** PAY-01
        - **Type:** Service or Module
        - **Purpose:** To integrate with external payment gateways and securely process payment transactions.
        - **Function:** Communicates with payment gateways, ensures secure transaction processing, and handles payment data.
6. **User Account Management Feature**
    - **User Account Management Component**
        - **Identification:** ACCT-01
        - **Type:** User Interface Component
        - **Purpose:** To allow users to manage their account settings, addresses, and preferences.
        - **Function:** Handles user profile updates, address management, and preference settings.
7. **Order Tracking and Confirmation Feature**
    - **Order Tracking Component**
        - **Identification:** TRACK-01
        - **Type:** User Interface Component
        - **Purpose:** To provide Order tracking information to users.
        - **Function:** Displays Order status, tracking details, and Order confirmation.

4.2 Procedural Approach

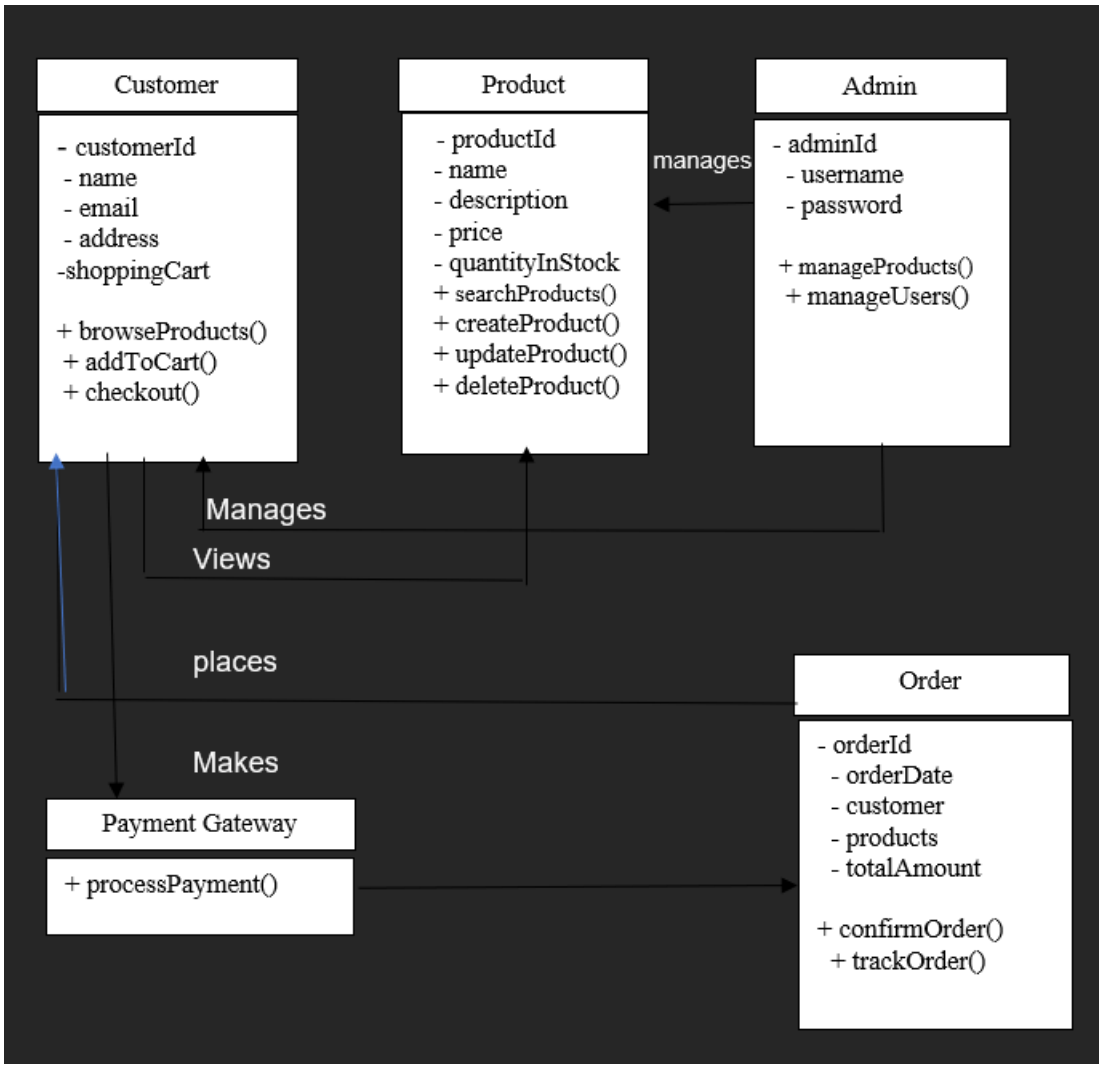
Not Applicable

4.3 Object-Oriented Approach

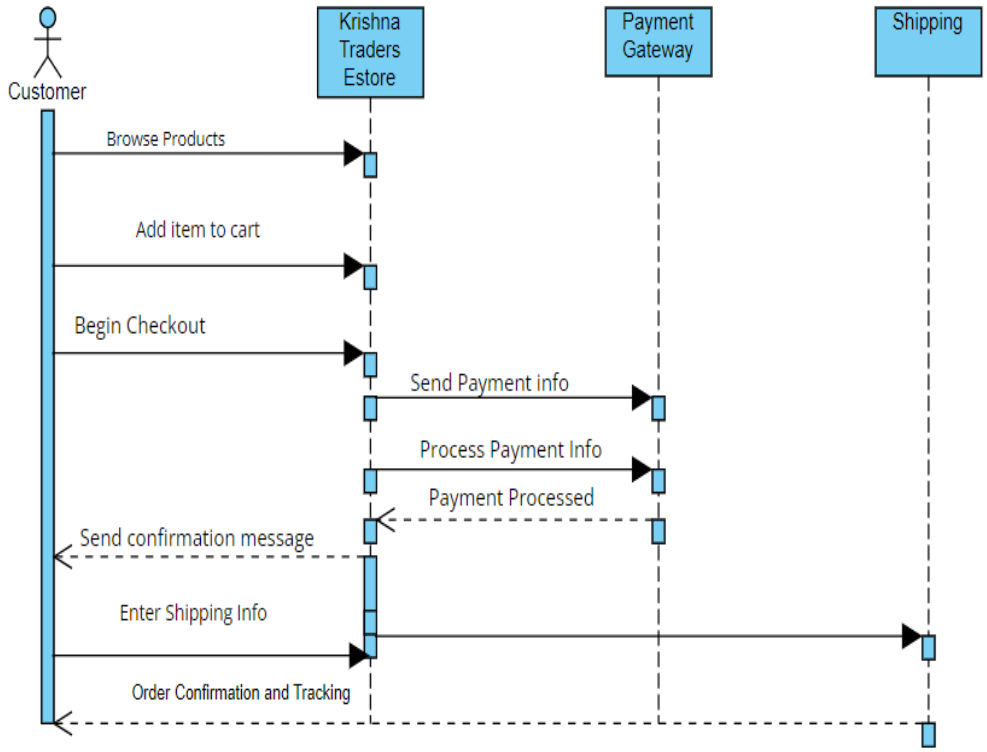
4.3.1 Use Case Diagrams



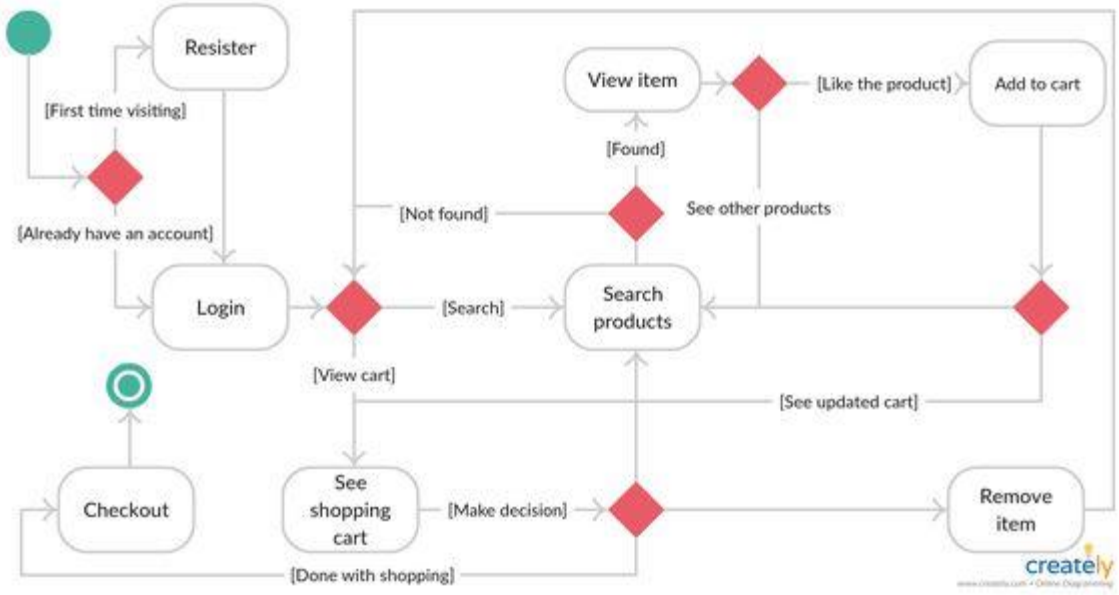
4.3.2 Class Diagrams



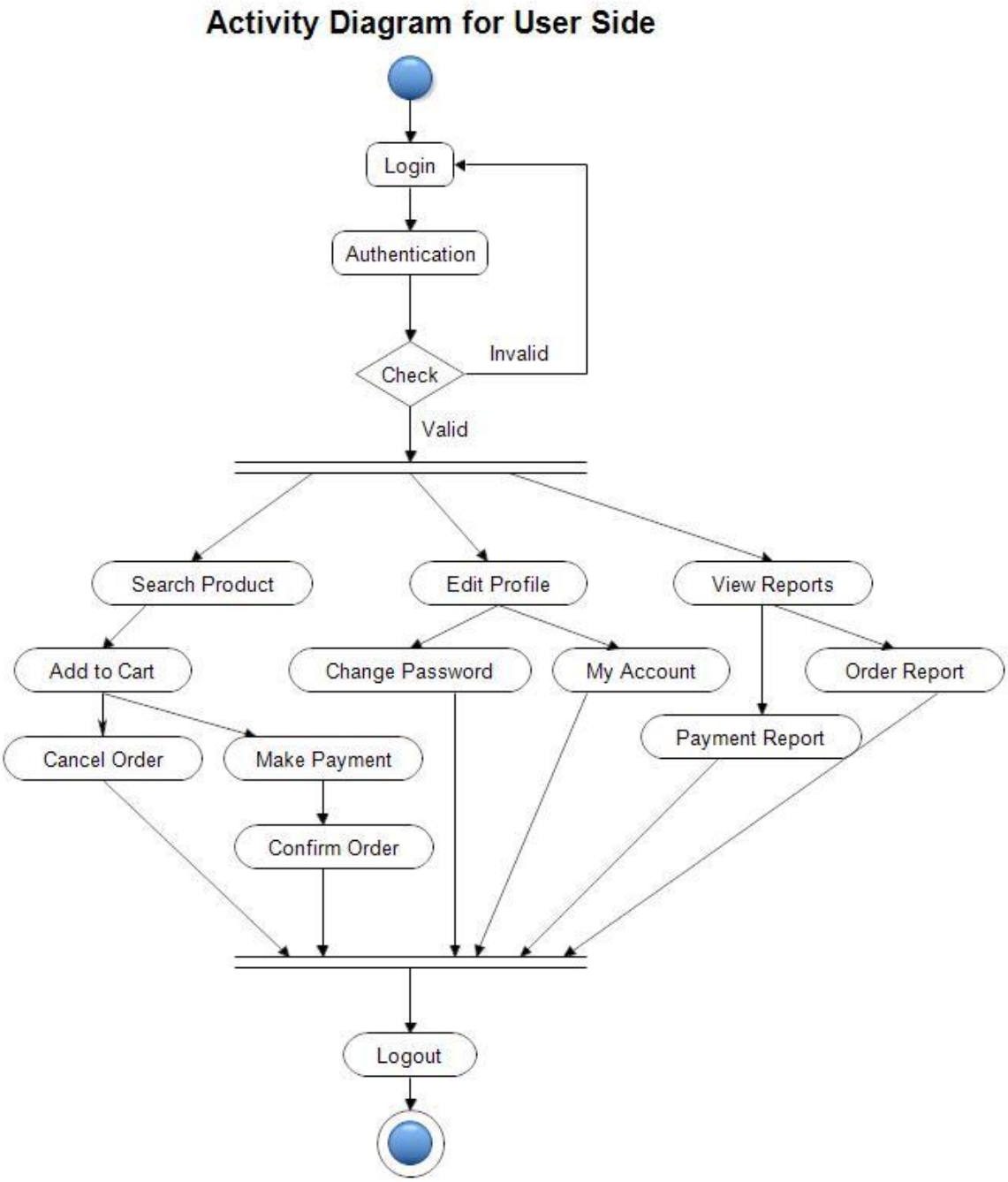
4.3.3 Sequence Diagrams

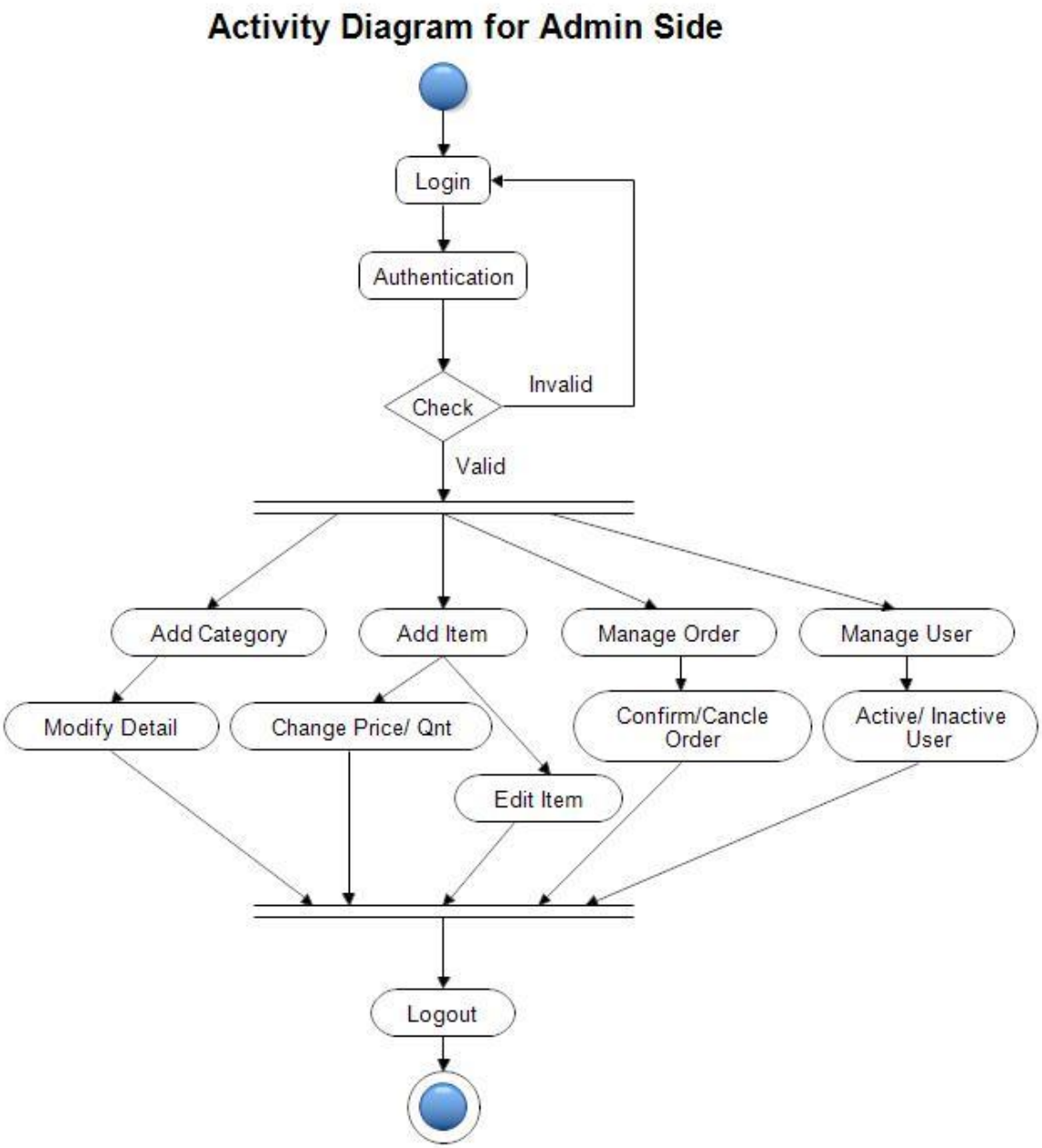


4.3.4 State chart Diagrams



4.3.5 Activity Diagrams





5. Dependency Description

Entity: User Authentication

- **Type: Module**
- **Purpose: Handles user registration and login.**
- **Dependencies:**

- **Depends on the database for storing user information.**
- **Used by other modules to verify user identity.**
- **Resources:**
- **Requires access to the user database.**
- **Processing:**
- **Provides user registration and login functionality.**
- **Data:**
- **Stores user information such as username and password.**

Entity: Product Catalog

- **Type: Module**
- **Purpose: Manages the list of available Order.**
- **Dependencies:**
- **Retrieves product information from the database.**
- **Used by the shopping cart for product selection.**
- **Resources:**
- **Requires access to the product database.**
- **Processing:**
- **Retrieves and displays product information.**
- **Data:**
- **Stores product details like name, price, and availability.**

Entity: Shopping Cart

- **Type: Module**
- **Purpose: Stores selected Order for checkout.**
- **Dependencies:**

- **Depends on the product catalog for product information.**
- **Resources:**
- **Utilizes the product catalog.**
- **Processing:**
- **Adds, removes, or updates selected Order.**
- **Data:**
- **Maintains a list of selected Order and quantities.**

Entity: Checkout

- **Type: Module**
- **Purpose: Manages the Order and payment process.**
- **Dependencies:**
- **Depends on the shopping cart for selected items.**
- **Resources:**
- **Requires access to payment gateways.**
- **Processing:**
- **Initiates the payment process.**
- **Verifies and confirms the Order.**
- **Data:**
- **Stores Order details and payment information.**

Entity: Order Tracking

- **Type: Module**
- **Purpose: Allows users to track their Orders.**
- **Dependencies:**
- **Relies on the Order and payment processing.**

- **Resources:**
- **Accesses Order and payment information.**
- **Processing:**
- **Provides Order status updates to users.**
- **Data:**
- **Contains Order tracking details.**

Entity: Customer Support

- **Type: Module**
- **Purpose: Provides customer assistance.**
- **Dependencies:**
- **May require access to Order and user information.**
- **Resources:**
- **Requires access to customer data.**
- **Processing:**
- **Supports users with inquiries and issues.**
- **Data:**
- **Stores user and Order information for support purposes.**

Entity: Product Reviews

- **Type: Module**
- **Purpose: Allows users to review and rate Order.**
- **Dependencies:**
- **May depend on the product catalog for product details.**
- **Resources:**
- **Accesses user reviews and ratings.**

- **Processing:**
- **Collects and displays product reviews.**
- **Data:**
- **Contains user-generated product reviews and ratings.**

Entity: Search Functionality

- **Type: Module**
- **Purpose: Provides a search feature for product discovery.**
- **Dependencies:**
- **Utilizes the product catalog for searching.**
- **Resources:**
- **Requires access to the product catalog.**
- **Processing:**
- **Enables users to search for Order.**
- **Data:**
- **Manages search queries and results.**

These descriptions outline the key dependencies between the entities in our system design, considering the eight features and their interactions. The entities collaborate to create a seamless e-commerce experience for users.

6. Interface Description

Entity: User Authentication

- **Function: Manages user registration and login.**
- **Internal Interface:**
- **Functions for user registration and login.**
- **Validates user credentials and interacts with the user database.**

Entity: Product Catalog

- **Function: Manages the list of available Order.**
- **Internal Interface:**
- **Functions for retrieving and displaying product information.**
- **Connects to the product database to fetch data.**

Entity: Shopping Cart

- **Function: Stores selected Order for checkout.**
- **Internal Interface:**
- **Functions for adding, removing, and updating selected Order.**
- **Interacts with the product catalog for product details.**

Entity: Checkout

- **Function: Manages the Order and payment process.**
- **Internal Interface:**
- **Functions for initiating payment and Order confirmation.**
- **Interfaces with payment gateways for payment processing.**

Entity: Order Tracking

- **Function: Allows users to track their Orders.**
- **Internal Interface:**
- **Functions for providing Order status updates to users.**
- **Retrieves Order and payment information for tracking.**

Entity: Customer Support

- **Function: Provides customer assistance.**
- **Internal Interface:**
- **Functions for addressing user inquiries and issues.**

- **May access user and Order information for support purposes.**

Entity: Product Reviews

- **Function: Allows users to review and rate Order.**
- **Internal Interface:**
- **Functions for collecting and displaying product reviews.**
- **Manages user-generated product reviews and ratings.**

8. Detailed Design

1. USER AUTHENTICATION

1.1 ALGORITHM

STEP 1: START

STEP 2: USER OPENS THE APPLICATION.

STEP 3: APPLICATION DISPLAYS THE LOGIN AND REGISTRATION OPTIONS.

STEP 4: USER SELECTS THE "REGISTER" OPTION TO CREATE A NEW ACCOUNT OR "LOGIN" TO ACCESS AN EXISTING ACCOUNT.

STEP 5: IF THE USER SELECTS "REGISTER":

STEP 5.1: USER PROVIDES REGISTRATION DETAILS (E.G., NAME, EMAIL, PASSWORD).

STEP 5.2: APPLICATION VALIDATES THE REGISTRATION DATA (E.G., EMAIL FORMAT, PASSWORD STRENGTH).

STEP 5.3: IF THE DATA IS VALID:

STEP 5.3.1: APPLICATION CREATES A NEW USER ACCOUNT.

STEP 5.3.2: USER IS REDIRECTED TO THE LOGIN PAGE (GO TO STEP 6).

STEP 5.4: IF THE DATA IS INVALID:

STEP 5.4.1: AN ERROR MESSAGE IS DISPLAYED.

STEP 6: IF THE USER SELECTS "LOGIN":

STEP 6.1: USER PROVIDES LOGIN CREDENTIALS (EMAIL AND PASSWORD).

STEP 6.2: APPLICATION VALIDATES THE LOGIN DATA.

STEP 6.3: IF THE CREDENTIALS ARE CORRECT:

STEP 6.3.1: USER IS GRANTED ACCESS TO THEIR ACCOUNT.

STEP 6.3.2: USER IS REDIRECTED TO THE HOME PAGE (GO TO STEP7).

STEP 6.4: IF THE CREDENTIALS ARE INCORRECT:

STEP 6.4.1: AN ERROR MESSAGE IS DISPLAYED, AND THE USER CAN RETRY.

STEP 7: USER CAN NOW ACCESS THEIR ACCOUNT AND USE THE APPLICATION FEATURES.

STEP 8: END

2. PRODUCT CATALOG

2.1 ALGORITHM

STEP 1: START

STEP 2: USER OPENS THE APPLICATION.

STEP 3: APPLICATION DISPLAYS A LIST OF PRODUCT CATEGORIES AND SEARCH OPTIONS.

STEP 4: USER CAN BROWSE ORDER IN THE FOLLOWING WAYS:

STEP 4.1: SELECT A PRODUCT CATEGORY FROM THE LIST.

STEP 4.2: USE THE SEARCH BAR TO ENTER KEYWORDS FOR PRODUCT SEARCH.

STEP 5: IF THE USER SELECTS A PRODUCT CATEGORY (STEP 4.1):

STEP 5.1: APPLICATION DISPLAYS A LIST OF ORDER WITHIN THE CHOSEN CATEGORY.

STEP 5.2: USER CAN CLICK ON A PRODUCT TO VIEW DETAILED INFORMATION (GO TO FEATURE 3).

STEP 5.3: USER CAN ADD ORDER TO THEIR SHOPPING CART (GO TO FEATURE

3. SHOPPING CART

3.1 ALGORITHM

STEP 1: START

STEP 2: USER ADDS ITEMS TO THE SHOPPING CART.

IF USER WANTS TO ADD AN ITEM:

STEP 2.1: USER SELECTS AN ITEM.

STEP 2.2: USER INDICATES QUANTITY OF THE SELECTED ITEM.

STEP 2.3: APPLICATION ADDS THE ITEM TO THE CART.

STEP 3: USER VIEWS THE CONTENTS OF THE SHOPPING CART.

IF USER WANTS TO VIEW THE CART:

STEP 3.1: APPLICATION DISPLAYS THE ITEMS IN THE CART.

STEP 3.2: USER CAN SEE ITEM DETAILS, QUANTITY, AND TOTAL COST FOR EACH ITEM.

STEP 4: USER CAN MODIFY CART CONTENT.

IF USER WANTS TO MODIFY THE CART:

STEP 4.1: USER CAN CHANGE ITEM QUANTITIES OR REMOVE ITEMS.

STEP 4.2: APPLICATION UPDATES THE CART ACCORDING TO USER'S ACTIONS.

STEP 5: USER PROCEEDS TO CHECKOUT.

IF USER WANTS TO CHECK OUT:

STEP 5.1: USER CONFIRMS CART CONTENTS.

STEP 5.2: APPLICATION CALCULATES THE TOTAL COST INCLUDING TAX AND SHIPPING FEES.

STEP 5.3: USER PROVIDES SHIPPING INFORMATION AND PAYMENT DETAILS.

STEP 5.4: USER CONFIRMS THE ORDER.

STEP 6: ORDER IS SUBMITTED.

IF USER CONFIRMS THE ORDER:

STEP 6.1: APPLICATION CREATES AN ORDER WITH THE SELECTED ITEMS.

STEP 6.2: APPLICATION PROCESSES PAYMENT TRANSACTION.

STEP 6.3: APPLICATION GENERATES AN ORDER CONFIRMATION.

STEP 6.4: USER RECEIVES AN ORDER CONFIRMATION.

STEP 7: ORDER COMPLETION.

IF PAYMENT IS SUCCESSFUL:

STEP 7.1: APPLICATION UPDATES INVENTORY.

STEP 7.2: APPLICATION NOTIFIES USER ABOUT SHIPPING DETAILS.

STEP 8: USER CAN CONTINUE SHOPPING OR LOG OUT.

STEP 9: END

4. CHECKOUT

4.1 ALGORITHM

STEP 1: START

STEP 2: USER SELECTS TO PROCEED TO CHECKOUT.

IF USER WANTS TO CHECK OUT:

STEP 2.1: APPLICATION DISPLAYS THE SHOPPING CART CONTENTS.

STEP 2.2: APPLICATION CALCULATES THE TOTAL COST INCLUDING TAX AND SHIPPING FEES.

STEP 2.3: USER PROVIDES SHIPPING INFORMATION AND PAYMENT DETAILS.

STEP 3: USER CONFIRMS THE ORDER.

IF USER CONFIRMS THE ORDER:

STEP 3.1: APPLICATION VALIDATES THE PAYMENT INFORMATION.

STEP 3.2: IF PAYMENT INFORMATION IS VALID:

STEP 3.2.1: APPLICATION CREATES AN ORDER.

STEP 3.2.2: APPLICATION PROCESSES PAYMENT TRANSACTION.

STEP 3.2.3: APPLICATION GENERATES AN ORDER CONFIRMATION.

STEP 3.2.4: USER RECEIVES AN ORDER CONFIRMATION.

IF PAYMENT INFORMATION IS INVALID:

STEP 3.2.5: USER RECEIVES A PAYMENT ERROR MESSAGE.

STEP 4: ORDER COMPLETION.

IF PAYMENT IS SUCCESSFUL:

STEP 4.1: APPLICATION UPDATES INVENTORY.

STEP 4.2: APPLICATION NOTIFIES USER ABOUT SHIPPING DETAILS.

STEP 5: USER CAN CONTINUE SHOPPING OR LOG OUT.

STEP 6: END

5. ORDER TRACKING

5.1 ALGORITHM

STEP 1: START

STEP 2: USER INITIATES ORDER TRACKING.

IF USER WANTS TO TRACK AN ORDER:

STEP 2.1: USER ENTERS THE ORDER NUMBER.

STEP 3: APPLICATION SEARCHES FOR THE ORDER.

IF ORDER NUMBER IS VALID:

STEP 3.1: APPLICATION RETRIEVES ORDER DETAILS.

STEP 3.2: APPLICATION DISPLAYS ORDER STATUS AND SHIPPING INFORMATION.

IF ORDER NUMBER IS INVALID:

STEP 3.3: APPLICATION DISPLAYS AN ERROR MESSAGE.

STEP 4: USER CAN CHOOSE TO TRACK ANOTHER ORDER OR EXIT.

STEP 5: END

6. CUSTOMER SUPPORT

6.1 ALGORITHM

STEP 1: START

STEP 2: USER INITIATES CUSTOMER SUPPORT REQUEST.

IF USER WANTS TO CONTACT CUSTOMER SUPPORT:

STEP 2.1: USER SELECTS THE "CONTACT SUPPORT" OPTION.

STEP 3: APPLICATION PROVIDES CONTACT OPTIONS.

STEP 3.1: APPLICATION DISPLAYS CONTACT METHODS (e.g., phone, email, chat).

STEP 3.2: USER SELECTS A PREFERRED CONTACT METHOD.

STEP 4: USER SUBMITS SUPPORT REQUEST.

IF USER SELECTS "PHONE":

STEP 4.1: APPLICATION PROVIDES A CUSTOMER SUPPORT PHONE NUMBER.

STEP 4.2: USER CALLS THE PROVIDED PHONE NUMBER FOR SUPPORT.

IF USER SELECTS "EMAIL":

STEP 4.3: APPLICATION PROVIDES A SUPPORT EMAIL ADDRESS.

STEP 4.4: USER COMPOSES AN EMAIL REQUEST FOR SUPPORT AND SENDS IT.

IF USER SELECTS "CHAT":

STEP 4.5: APPLICATION INITIATES A LIVE CHAT SESSION WITH A SUPPORT AGENT.

STEP 5: CUSTOMER SUPPORT AGENT RESPONDS.

IF PHONE:

STEP 5.1: AGENT ANSWERS THE CALL AND ASSISTS THE USER.

IF EMAIL:

STEP 5.2: SUPPORT AGENT RECEIVES THE EMAIL REQUEST AND RESPONDS ACCORDINGLY.

IF CHAT:

STEP 5.3: SUPPORT AGENT ENGAGES IN A LIVE CHAT CONVERSATION TO ASSIST THE USER.

STEP 6: SUPPORT INTERACTION CONTINUES UNTIL THE USER'S ISSUE IS RESOLVED.

STEP 7: USER CONFIRMS ISSUE RESOLUTION.

IF USER'S ISSUE IS RESOLVED:

STEP 7.1: USER ACKNOWLEDGES THE RESOLUTION.

IF USER'S ISSUE IS NOT RESOLVED:

STEP 7.2: USER AND SUPPORT AGENT CONTINUE TO WORK ON THE ISSUE UNTIL RESOLVED.

STEP 8: USER ENDS THE SUPPORT INTERACTION.

IF USER IS SATISFIED:

STEP 8.1: USER THANKS THE SUPPORT AGENT AND ENDS THE INTERACTION.

IF USER IS NOT SATISFIED:

STEP 8.2: USER MAY PROVIDE FEEDBACK OR REQUEST FURTHER ASSISTANCE.

STEP 9: END

7. PRODUCT REVIEWS

7.1 ALGORITHM

STEP 1: START

STEP 2: USER INITIATES PRODUCT REVIEW.

IF USER WANTS TO WRITE A PRODUCT REVIEW:

STEP 2.1: USER NAVIGATES TO THE PRODUCT REVIEW SECTION.

STEP 3: USER SELECTS A PRODUCT FOR REVIEW.

IF MULTIPLE ORDER AVAILABLE:

STEP 3.1: APPLICATION DISPLAYS A LIST OF ORDER.

STEP 3.2: USER SELECTS THE PRODUCT TO REVIEW.

STEP 4: USER COMPOSES THE PRODUCT REVIEW.

STEP 4.1: APPLICATION PROVIDES A REVIEW FORM.

STEP 4.2: USER ENTERS REVIEW DETAILS, INCLUDING RATING, TITLE, AND DESCRIPTION.

STEP 4.3: USER MAY UPLOAD IMAGES OR VIDEOS RELATED TO THE PRODUCT.

STEP 5: USER SUBMITS THE REVIEW.

STEP 5.1: USER CONFIRMS REVIEW DETAILS.

STEP 5.2: USER SUBMITS THE REVIEW TO THE APPLICATION.

STEP 6: APPLICATION RECEIVES AND PUBLISHES THE REVIEW.

STEP 6.1: APPLICATION VALIDATES THE REVIEW CONTENT.

STEP 6.2: IF VALID, APPLICATION DISPLAYS THE REVIEW ON THE PRODUCT PAGE.

STEP 7: OTHER USERS VIEW AND INTERACT WITH THE REVIEW.

STEP 7.1: USERS CAN READ AND RATE THE REVIEW.

STEP 7.2: USERS MAY PROVIDE COMMENTS OR QUESTIONS ON THE REVIEW.

STEP 8: AUTHOR OF THE REVIEW RESPONDS (OPTIONAL).

IF REVIEW AUTHOR WANTS TO RESPOND TO COMMENTS:

STEP 8.1: AUTHOR ENTERS RESPONSES TO USER COMMENTS.

STEP 9: REVIEW INTERACTION CONTINUES AS NEEDED.

STEP 10: END