T-shirt E-commerce Web App

By

Tanzeela Ghafoor **2020-GCUF-515018**

Tayyiba Sohail **2020-GCUF-515034**

Kashaf Akram **2020-GCUF-515009**

Project submitted in partial fulfillment
of the requirements for the degree of

BACHELOR OF SCIENCE

IN

COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

Government Graduate College Chowk Azam

2020-2024

In the Name of Allah, the Entirely Merciful, the Especially Merciful

Al-Fatihah [1: 1], Nobel Quran

DECLARATION

This project, a die-hard work, is productized out by Tanzeela Ghafoor, Kashaf Akram and Tayyiba Sohail under the supervision of Mr. Yasir Arfat (Lecturer, Computer Department) and Mr. Dure-Subhani (HOD, Computer Department) Community College, GC College, Chowk Azam, Pakistan. We feel please to declare that the project and contents of this project is the productive result of our hardworking, studies and research and no part of this is copied from any published source. This work has been conducted under the practical atmosphere of our studies not for the award of any other degree / diploma. The College may take action if the information provided is found guilty at any stage. Any external sources of information used in this project, including references, have been duly acknowledged through proper citations and bibliographical references. The project has not been previously submitted for any other degree or examination at any other institution. Any contributions made by others to this project, including guidance and support from faculty members, have been duly acknowledged. The software code, documentation, and any other materials presented as part of this project are the result of my own work, unless otherwise acknowledged. I take full responsibility for the authenticity and originality of the content presented in this project. I understand that any misrepresentation or falsification of information in this declaration will have serious consequences, including the possibility of disciplinary action by the institution.

Signature of Student:

Tanzeela Ghafooor _____

Registration No. 2020-GCUF-515018

Tayyiba Sohail _____

Registration No. 2020-GCUF-515034

Kashaf Akram _____

Registration No. 2020-GCUF-515209

Dedication

I dedicate this project to God Almighty, my creator, my strong pillar, my source of inspiration, wisdom, knowledge, and understanding. He has been the source of my strength throughout this program, and on His wings alone have I soared. I also dedicate this work to my friends who have encouraged me all the way and whose encouragement has ensured that I give it all it takes to finish what I have started. This project, titled T-shirt E-commerce Web App, is dedicated to the individuals who have been a source of inspiration and support throughout this journey. To my family, for their unwavering love, encouragement, and understanding. Your belief in me has been my driving force, and I am forever grateful for your constant support. To my project supervisor, [Mr. YasirArfat], for your guidance, expertise, and valuable insights. Your mentorship has been invaluable in shaping this project and in fostering my growth as a developer. To my friends and classmates, for their camaraderie, motivation, and the countless hours spent brainstorming ideas and discussing challenges. Your presence has made this project more enjoyable and fulfilling. To the staff and faculty members at [GC College], for providing a conducive learning environment and the resources that have contributed to my education and the successful completion of this project. To all the users and stakeholders who participated in the testing and evaluation of the app, providing valuable feedback and helping me improve the functionality and user experience. Lastly, I would like to express my gratitude to everyone who has played a role, big or small, in this project. Your support and belief in my abilities have been instrumental in its completion. This project is dedicated to each and every one of you. Thank you for being a part of my journey and for making it a truly enriching experience.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude and appreciation to all those who have contributed to the successful completion of my Final Year Project titled "T-shirt E-commerce Web App." This project has been an enriching and rewarding experience, and I would like to acknowledge the individuals and organizations for their support. My project supervisor, **Mr. Yasir Arfat**, for their invaluable guidance, expertise, and continuous support throughout the project. Their knowledge and insights have been instrumental in shaping the direction of this work. The faculty members of **GC College**, for providing me with a conducive learning environment and valuable resources. I am grateful for their dedication to education and their efforts in imparting knowledge and skills. My family and friends, for their unwavering support, encouragement, and understanding. Their belief in me and their motivation have been a constant source of inspiration, driving me to overcome challenges and strive for excellence. The participants who willingly volunteered their time and provided feedback during the testing and evaluation phases of the project. Their insights and suggestions have been invaluable in refining the functionality and user experience of the app. The developers and contributors of open-source libraries, frameworks, and tools that were utilized in the development of this project. Their efforts have significantly expedited the development process and enhanced the overall quality of the application. I would also like to extend my gratitude to all the train passengers, restaurant owners, and delivery personnel who may benefit from this train food ordering app. Their needs and expectations have been a driving force behind the development of this solution. Lastly, I want to express my appreciation to all the individuals, both mentioned and unnamed, who have directly or indirectly contributed to this project. Your support and assistance have played an integral role in its successful completion. In conclusion, I am truly grateful to everyone who has been a part of this project. Your contributions, encouragement, and support have been invaluable, and I am honored to have had the opportunity to work on this project with such wonderful individuals.

Tanzeela Ghafoor (Roll No. 18)

Tayyiba Sohail (Roll No. 34)

Kashaf Akram(Roll No. 09)

To

The Controller of Examinations,

Government Graduate College ,

Chowk Azam

I, the Supervisor, certifies that the contents and form of the project submitted by

Tanzeela Ghafoor (Roll no.18)

Tayyiba Sohail (Roll no.34)

Kashaf Akram(Roll no.09)

Have been found satisfactory for the award of the degree.

Internal Examiner

Name:_____

Signature:_____

Lecturer: Government Graduate College Chowk Azam Department of Computer Science

External Examiner

Name:_____

Signature:_____

Table of Contents

Table of Contents

Abstract

The popularity of online shopping is growing rapidly, and this project focuses on simplifying the process of purchasing T-shirts online. In the modern-day, almost everyone is familiar with e-commerce platforms, and the demand for convenient online shopping has increased exponentially. This project revolves around an e-commerce platform specifically designed for T-shirts, allowing customers to browse, select, and order their desired products with ease. The system provides a seamless shopping experience by offering various categories such as men, women, and kids, along with detailed product descriptions, size options, and price comparisons. This thesis aims to create a user-friendly web application that helps users effortlessly purchase T-shirts from the comfort of their homes. Customers can add products to their cart, modify selections, and securely process payments. The project consists of two major components: one for customers to browse and purchase T-shirts, and another for the admin to manage products and inventory. The application is developed using React for the frontend and Node.js, Express, and MongoDB for the backend, ensuring scalability and efficiency. It also includes a social sharing feature, allowing users to share their selected products directly to their social media accounts.

By implementing this e-commerce system, the project enhances the shopping experience for customers while also providing a reliable platform for businesses to sell their T-shirts online. The web app is designed to be responsive and accessible across different devices with various screen sizes, ensuring convenience for all users. This work utilizes modern web technologies to create a scalable and user-centric platform for online T-shirt shopping.

Keywords

T-shirt e-commerce, Online shopping, Web application, Product catalog, Shopping cart, Payment gateway, React, Node.js, MongoDB, Responsive web design

Chapter-1

Introduction to the Problem

1.1 Introduction

The main purpose of this website is to facilitate offline customers online because customers cannot spend their precious time in markets trying to find the best deal. The primary objective is to allow people to customize T-shirts. They can create their dream T-shirts that perfectly fit their style and personality. Our website offers a wide range of customization options, allowing them to design a T-shirt that is uniquely theirs. Customers can either choose their own design or customize specific designs and then provide all the details about their specifications. After that, they move on to the order page, complete their order, and proceed to the payment method. By following these steps, customers can successfully order their custom T-shirts.

1.2 Background

The traditional T-shirt retail system relies on manual processes, leading to inefficiencies and errors. To address these issues, we developed an e-commerce platform that automates browsing, selection, and purchasing. This system simplifies order management, enhances accuracy, and improves accessibility for customers, offering a seamless online shopping experience.

1.3 Purpose

The purpose of this project is to provide a platform for customers to personalize and customize their T-shirts. It allows individuals to create unique and personalized T-shirts that align with their preferences, style, and specific needs. Moreover, it helps T-shirt brands and sellers cater to a wide range of customer preferences, thereby increasing customer satisfaction and loyalty.

1.4 Scope

The scope of our project is to design a comprehensive platform that provides a safe and user-friendly environment for ordering ready-made T-shirts and customizing T-shirts. The main aim of the project is to offer an easy-to-use web portal for the services provided to customers. People often face difficulties in finding well-designed T-shirts that match their specific preferences and budget. Our web app has the following major features listed below:

- User-friendly Interface
- Providing Customization Services
- Secure from Unauthorized Access

- Allowing Returns if the Customer is Not Satisfied
- Gathering Customer Feedback throughout the Process
- Maintaining Contact with Customers Post-Delivery for Any Future Assistance or Advice about T-shirts

1.5 Goal Of Project

The goal of this project is to create a customer-centric experience that fosters loyalty, differentiation, and growth while allowing brands to gather valuable insights and adapt to evolving consumer preferences. The primary objective of this project is to provide customers with a unique and personalized shopping experience for T-shirts.

1.6 Problem Definition

The problem definition for a T-shirt e-commerce customization project typically revolves around identifying the challenges or shortcomings in the existing customer experience and addressing them through customization. These challenges may include limited design options, difficulty in finding the perfect fit, lack of personalization, and the inconvenience of shopping for unique and customized T-shirts. The project aims to solve these issues by providing a platform where customers can easily personalize their T-shirts to match their individual preferences, ensuring a better fit, style, and overall shopping experience.

Problem: Limited Personalization Options and Generic Customer Experience

1.7 Project Planning

The main purpose of this project is to facilitate customers in managing their custom T-shirt orders online, ensuring a personalized experience and providing options for retakes if customers are not satisfied. This project is supported by Supervisor prof. Yasir Arfat, who motivated and guided me in fulfilling the task efficiently. Their support helped in developing the project more effectively.

We completed this project almost within 4 months, divided into 10 main phases:

- Project Definition Selection
- Project Proposal Submission
- Project Description Submission
- Requirement Gathering
- Patent Searching
- Requirement Specification and Planning
- Designing
- Coding and Testing
- Documentation
- Implementation and Deployment

Task ID	Tasks Description	Dec	Jan	Feb	March		April		May		June
1	Project Selection										
2	Project Proposal										
3	Project Description Submission										
4	requirement Gathering										
5	Patent Seraching										
6	Requirement Specification and Planning										
7	Designing										
8	Coding And Testing										
9	Deocumentation										
10	Implementation And Deployment										

Figure 1.1 Planning

1.8 Project modules

In our T-shirt e-commerce web app, modules serve as distinct units of functionality, each comprising source files and build configurations. They enable us to compartmentalize different aspects of the project, such as user management, product catalog, customization features, order processing, payment integration, customer feedback, support systems, and administrative tools. Modules operate independently, allowing for separate development, testing, and debugging, which enhances flexibility and facilitates efficient maintenance and scalability of the application.

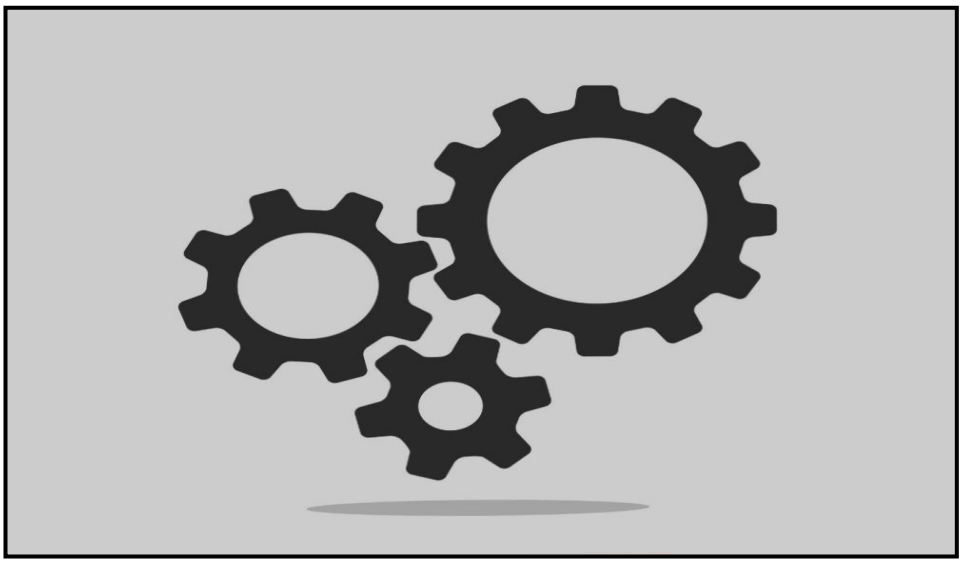

Figure 1.2 Module

1.8.1 Frontend

1. **User Interface**
    - Home page with featured products, categories, and promotions to attract users.
    - Our Website Intro Section
    - Our Collections
    - Special Dresses
2. **Product Listing Page**
    - List T-shirts with filtering options (size, color, price) for easy browsing.
3. **Cart Page**
    - Enable users to view items in their cart and proceed to checkout seamlessly.
4. **Checkout Page**
    - Facilitate the purchase process with forms for shipping information and secure payment details.
5. **User Account Page**
    - Allow users to manage their orders, addresses, and personal information securely.
    - Enable users to share purchases on social media platforms directly from the confirmation page.
    - Highlight T-shirts endorsed by influencers, enhancing user engagement and trust.

1.8.2 Backend

1. **AdminPanel**
    - Create a comprehensive admin panel to manage products, categories, orders, and user accounts.
    - Implement role-based access control to ensure secure administrative operations.
2. **Database Design**
    - Use MongoDB, Express Js, Node Js to design schemas for managing users, products, orders, reviews, categories, and influencer endorsements.
3. **API Development**
    - Create RESTful APIs to handle interactions between the front-end and back-end.
    - Implement CRUD operations for products, orders, and user accounts.
    - Develop APIs for social media sharing functionality.

4. **Payment Gateway Integration**
    - Integrate popular payment gateways (e.g., Stripe) for secure transaction handling.
5. **User Authentication**
    - Implement secure user authentication and authorization mechanisms.
    - Use JWT (JSON Web Tokens) for managing user sessions securely
6. **Payment Gateway Integration**
    - Integrate popular payment gateways (e.g., Stripe) for secure transaction handling.
7. **Order Management**
    - Develop features for managing orders, including creation, status updates, and cancellation.

Chapter-2

Background and Problem Defination

2.1 Overview

The different technologies, frameworks, and programming languages that can be used to develop this website. Explore the various types of this project that can be offered on this website. This may include clothing styles, colors, patterns, fabric choices, sizing options. Continuously gather feedback from users and make iterative improvements in this website.

2.2 Existing Technology

Not so long ago, computers revolutionized online clothing shopping. The evolution of the internet, smartphones, and other technologies now provides a significant and efficient way to purchase products and explore the latest dress designs and stitching ideas. With the convenience of online tailors, customers can find the best deals, place orders, and cancel them at any time, all while enjoying secure online transactions.

2.3 Area of study

This area focuses on understanding consumer preferences, motivations, and decision-making process in the context of this project. It involves studying market trends, consumer segmentations, branding strategies and effective marketing techniques to attract and retain customers in this project. The study area should be your area of coverage that is, your case study. Introducing information from other areas or region will have no significance on the subject matter, hence your primary focus would be on the area your research is covering.

2.3.1 Tools and Technology:

A technology tool is an electronic, digital, or physical resource that can support teachers in the delivery and testing of content. Technology tools include apps, platforms, and software, and can be used in virtual, hybrid, or traditional learning environments.

There are the following tools that use in this project:

- HTML
- CSS
- Bootstrap
- JavaScript
- React Js

- Node Js
- Express Js
- Mongo DB

2.3.2 Front-end:

Frontend development refers to the process of creating and implementing the user interface of a website or application that users interact with directly. Frontend developers typically work with HTML, CSS, Javascript and React js to design and build the visual elements and user experience of a website. They are responsible for ensuring the website or application is visually appealing, easy to navigate, and functions properly across different devices and browsers.

1. **HTML:**

HTML (Hyper Text Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).HTML markup includes special "elements" such as <head>, <title>, <body>, <header> etc.

Figure 2.1 HTML image

2. **CSS:**

CSS in Front-end development deals with developing attractive, user-friendly, simple, and engaging interfaces for any website. The user does not want to know the complexity of either the database or wants to read the code to understand what they are looking for.CSS, or Cascading Style Sheets, is used to control the look and feel of web pages. CSS allows you to add colors, fonts, and other design elements to your HTML content. With CSS, you can create visually stunning websites that are both functional and aesthetically pleasing.

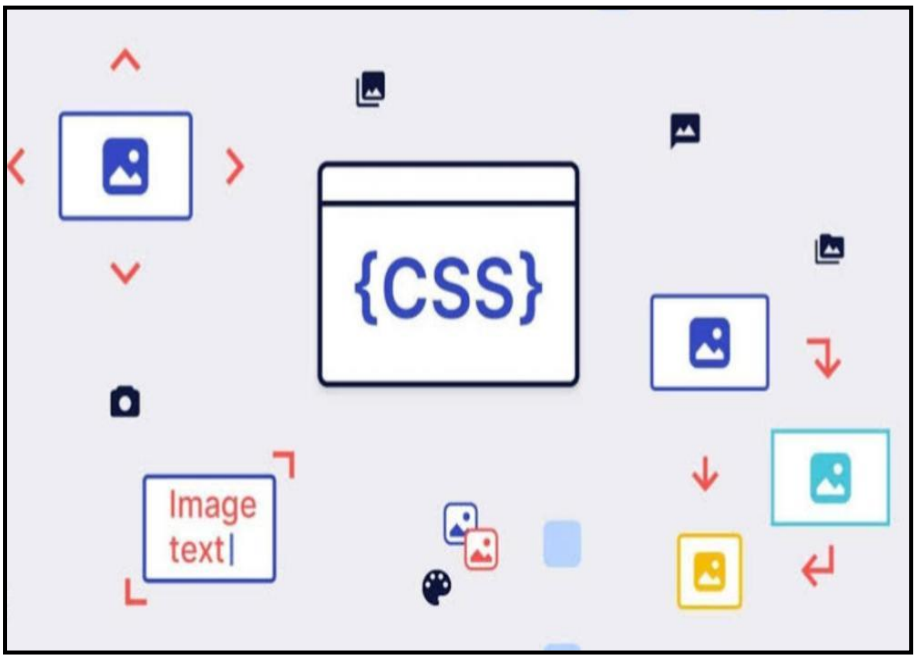

Figure 2.2 CSS image

3. Bootstrap:

Bootstrap is a free, open-source front-end development framework for the creation of websites and web apps. Designed to enable responsive development of mobile-first websites, Bootstrap provides a collection of syntax for template designs. Bootstrap is easy to learn. Due to its popularity, plenty of tutorials and online forums are available to help you get started.

Figure 2.3 Bootstrap image

4. JavaScript:

JavaScript is a scripting language used to develop web pages. JavaScript can be used extensively in this website to enhance the user experience, enable interactivity, and facilitate dynamic customization. It's important to note that JavaScript is often used in conjunction with other web technologies such as HTML and CSS to create a seamless and interactive user experience in an outfitter customization website.

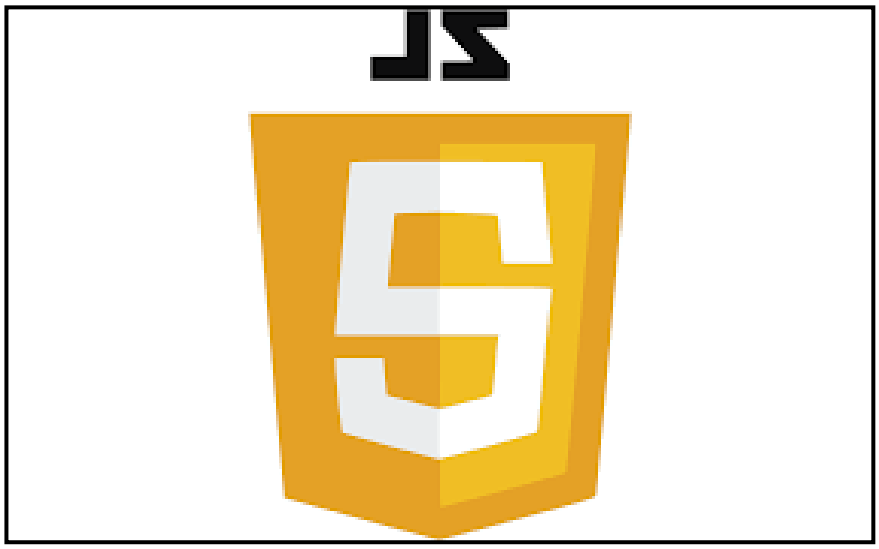

Figure 2.4 JavaScript image

5. React Js:

React.js is a powerful JavaScript library for building user interfaces, particularly single-page applications where efficient rendering and dynamic data management are crucial. It enables developers to create reusable UI components, allowing for faster development and maintenance. React.js manages the view layer of web applications, handling the complexity of state management and ensuring that the UI remains in sync with the underlying data. This makes it an ideal choice for modern web applications like e-commerce platforms, where performance and scalability are key consideration

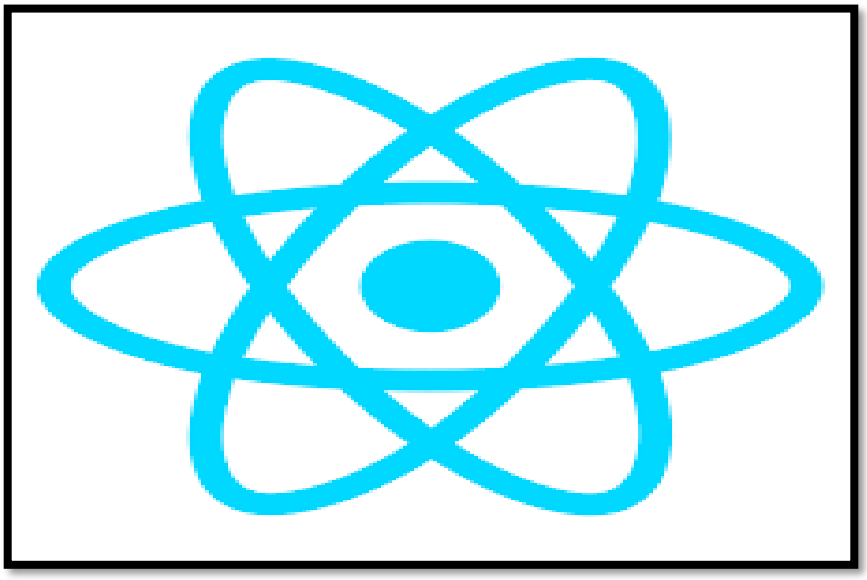

Figure 2.5 React image

2.3.3 Back-end:

The back-end refers to a part of computer application or a programs code that allow it to operate and that cannot be accessed by a user. Most data and operating system syntax are stored and accessed in back-end of computer system. Typically, the code is comprised of one or more programming languages.

- MongoDB
- Express Js
- Node Js

1. **MongoDB:**

   MongoDB is a popular NoSQL database known for its flexibility and scalability. Unlike traditional relational databases, MongoDB stores data in JSON-like documents, making it easier to model and manage unstructured data. This schema-less architecture allows for rapid development and adaptation to changing requirements. In the context of the T-shirt e-commerce web app, MongoDB is used to store and manage product information, user data, order details, and other essential data. Its ability to handle large volumes of data and perform fast queries makes it an ideal choice for the backend of an e-commerce application.

Figure 2.6 MongoDB image

2. Express.js:

Express.js is a minimal and flexible Node.js web application framework that provides a robust set of features for building web and mobile applications. It simplifies the process of handling HTTP requests, managing middleware, and routing, making it easier to build server-side applications. In the T-shirt e-commerce web app, Express.js is used to create a RESTful API that handles client requests, processes user authentication, manages shopping cart functionality, and facilitates communication between the frontend and the MongoDB database. Its simplicity and performance make Express.js a powerful tool for backend development.

3. Node.js:

Node.js is a runtime environment that allows developers to run JavaScript on the server side. It is built on the Chrome V8 JavaScript engine, enabling fast and efficient execution of code. Node.js is event-driven and non-blocking, making it suitable for building scalable and high-performance applications. In the T-shirt e-commerce web app, Node.js serves as the backbone of the backend, allowing for the seamless integration of the MongoDB database and the Express.js framework. It handles server-side logic, processes user requests, and ensures that the application can handle multiple concurrent users efficiently.

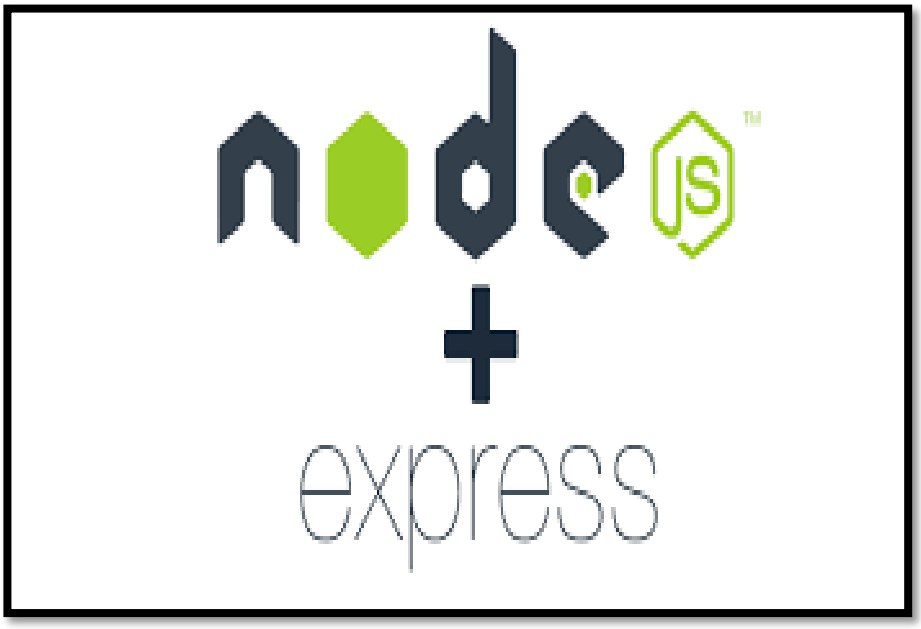

Figure 2.7 Node/Express Js image

2.4 Research Questions:

1. Why is product customization important and useful for a T-shirt e-commerce platform?
2. Can customization enhance customer involvement in the T-shirt shopping experience?
3. What factors influence consumer preferences and decision-making in T-shirt selection?
4. What are the primary motivations and drivers for consumers to engage in T-shirt customization?
5. How does the level of customization offered by the T-shirt store impact consumer satisfaction and loyalty?
6. What role does social influence play in consumers' decisions to customize their T-shirts?
7. How do consumers perceive the value of customized T-shirts compared to ready-made options?
8. How do demographic factors, such as age, gender, and socioeconomic status, influence consumers' interest in T-shirt customization?

2.5 Reasons for the Project:

The T-shirt e-commerce platform allows customers to personalize their orders, ensuring they receive products that meet their individual style and size preferences. Consumers increasingly seek personalized experiences and products that align with their unique needs.

- **Differentiation and Competitive Advantage**: Offering T-shirt customization provides a competitive edge in the e-commerce industry. By allowing users to personalize their T-shirts, the platform can differentiate itself from competitors and attract a broader customer base.

- **Enhanced Customer Engagement**: Customization offers an interactive shopping experience, where customers actively participate in designing their T-shirts, which can foster customer loyalty and increase engagement.
- **Increased Customer Satisfaction**: Allowing customers to create T-shirts that match their size, style, and color preferences leads to higher satisfaction, as they receive a product that fits their exact needs.
- **Reduced Inventory Risk**: Customization allows for on-demand production, reducing the risk of overproduction and excess inventory, as T-shirts are only produced based on specific customer orders.
- **Technological Advancements**: Leveraging tools like 3D modeling or design previews on the e-commerce site enhances the user experience, making customization more accessible and efficient. These advancements enable seamless and immersive customization experiences for customers.

Chapter-3

System Requirments Analysis

3.1 Requirement Analysis:

The project aims to develop an online platform where users can browse various categories of T-shirts (men, women, and kids), add products to their cart, proceed to checkout, and share products on social media. The requirement analysis will gather information about the specific objectives, functionalities, and constraints of this e-commerce platform. Key stakeholders include customers (users of the platform), the admin (responsible for managing product listings), and the development team. The primary users will be online shoppers who are interested in purchasing T-shirts. The user base may consist of individuals with different fashion preferences, and the platform must provide an intuitive shopping experience.

3.2 why is requirement analysis is necessary?

Requirement analysis is essential in software development as it helps identify, analyze, and document the needs and expectations of the users and stakeholders. For this T-shirt e-commerce platform, requirement analysis ensures that features such as login, product selection, cart management, and payment are aligned with user expectations. It helps in defining the project scope, avoiding costly mistakes, and ensuring quality assurance. Proper analysis ensures that the end product is user-friendly and meets the goals of the e-commerce system, including social media integration.

3.3 Steps in requirement analysis:

The requirement analysis for this e-commerce project involves gathering, documenting, and validating user needs. Here are the key steps involved:

3.3.1 Identify the purpose:

The purpose of requirement gathering for the T-shirt e-commerce platform is to ensure that the system provides users with a seamless product selection, shopping cart, and checkout experience, along with the ability to share selected items on social media platforms.

3.3.2 Identify Stakeholders:

Identify the key stakeholders involved in the project, such as users (customers shopping on the platform), the admin (responsible for managing product listings and monitoring transactions), and the development team (building and maintaining the platform).

3.3.3 Gather initial requirements:

Conduct interviews and surveys to gather high-level requirements. This includes understanding the needs for user registration, product selection, cart functionality, payment processing, order tracking, and social media integration. Stakeholder input is essential to ensure that the platform meets both customer and

admin expectations.

3.3.4 Elicit Customer Preferences:

- Capture user preferences regarding T-shirt types, sizes, colors, and designs.
- Gather functional requirements related to account creation, login, product browsing, cart management, payment methods, and order history.
- Capture the need for social media integration, allowing users to post selected T-shirts directly to their social accounts with price and website links.

3.3.5 Prioritize and Analyze Requirements:

- Review and analyze the collected requirements for consistency, completeness, and feasibility.
- Prioritize the most critical features such as user registration, cart management, secure payment, and social media sharing. Consider factors like business value, user convenience, and technical constraints.

3.3.6 Validate Requirements:

- Conduct reviews and feedback sessions with stakeholders to ensure the requirements are complete and reflect the users' needs.
- Validate that the platform's features, such as browsing, payment, and social media integration, are in line with user expectations.

3.4 System Functional Requirements:

The T-shirt e-commerce platform should support the following core functionalities:

- Users can view and filter T-shirts in three categories: men, women, and kids.
- Users can select T-shirts, add them to the cart, and proceed to checkout.
- Users can pay using methods such as Easy Paisa and cash on delivery.
- Users can track the status of their orders.
- Users can share selected T-shirts along with their price and a link to the website on social media platforms

3.4.1 User Registration and Login:

- Users should be able to create an account with a username, email, and password.

- Users should be able to securely log in with their credentials.
- User profiles should store personal information like name, email, and past orders.

3.4.2 T-shirt Selection and Customization:

- Users should be able to browse and select T-shirts based on categories (men, women, kids).
- Users should be able to select different sizes and colors for each product.
- A visual representation of the product should be available.

3.4.3 Product Management (Admin):

The admin can manage product listings through the admin panel. They can add, edit, or delete products from the catalog, ensuring that users always see updated information.

3.4.4 Add Products (Admin):

Admin can add new products by providing the following information:

- Product Title
- Category (Men, Women, Kids)
- Description
- Upload Image
- Price
- Available Quantity
- Submit button

3.4.5 Update and Delete Product Record (Admin):

Admins can update product information or remove items from the catalog by editing or deleting entries through the admin panel.

3.4.6 Order Management:

- Users should be able to view their selected T-shirts, payment information, and delivery location.
- The system should allow users to proceed to checkout and confirm their orders.

3.4.7 Payment Method

The system should support various payment methods, including:

- Script
- Easy Paisa
- Cash on delivery

3.4.8 Contact Us

Users can contact the admin using a contact form with the following fields:

- Name
- Email
- Message
- Send button

3.4.9 SRS Document (System Requirements Specification)

The SRS document defines the functional and non-functional requirements of the T-shirt e-commerce platform. It outlines the system's objectives, user needs, and technical constraints, serving as a formal agreement between the development team and stakeholders to ensure a clear understanding of the system requirements.

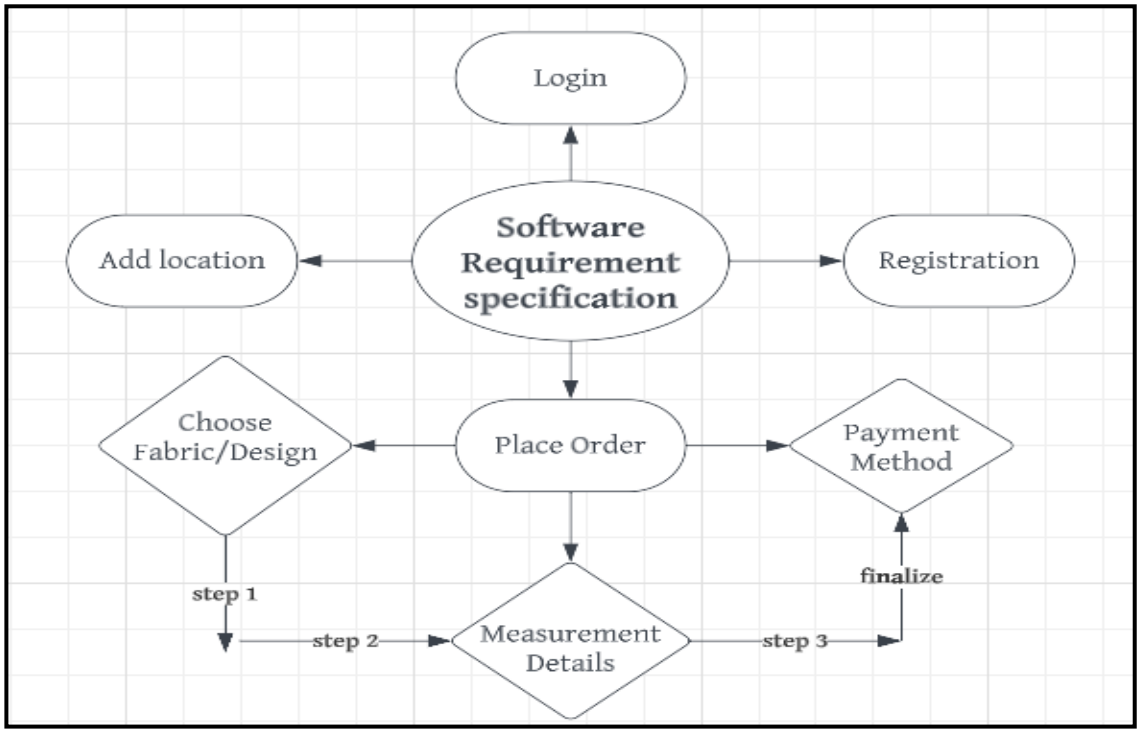

Figure 3.1 Diagram of SRS

By the names SRS report, or software document

Sr.No	Description
SRS 01	System should be able to provide login facility
SRS 02	System should be able to provide Registration facility
SRS 03	System should be able to Choose Design
SRS 04	System should be able to Choose fabric
SRS 05	System should be ab to place orders
SRS 06	System should be able to measurement details
SRS 07	System should be able to add location

Table 3.1Software Requirement Specification

3.5 Non- Functional Requirements

Non-functional requirements define the overall qualities and behavior of the system rather than specific features. For the T-shirt e-commerce platform, these requirements ensure the platform's efficiency, security, and usability.

1. **Performance:**The platform should load quickly, especially when users navigate through product categories, add items to the cart, and proceed with checkout. It should handle a high volume of concurrent users and ensure quick responses for actions like adding to the cart or sharing products on social media.

2. **Usability:**The platform should be intuitive and easy to use, with clear navigation and minimal steps to complete purchases. The process of browsing T-shirts, adding them to the cart, and checking out should be smooth, with clear instructions and user feedback.

3. **Reliability:**The platform must be reliable, ensuring that user activities like product selection, cart updates, and orders are consistently saved. Data related to products, orders, and user accounts should be accurate and up-to-date at all times.

4. **Scalability:**The platform should be scalable to accommodate a growing number of users, products, and categories without performance degradation. As the user base grows, the platform should remain responsive.

5. **Security:**The platform should ensure the protection of user data, including personal details, payment information, and order history. All sensitive information must be encrypted, and secure payment methods like Easy Paisa and Cash on Delivery should be supported.

6. **Maintainability:**The platform should be easy to maintain and update, allowing new products, categories, or features to be added with minimal downtime.

3.6 Hardware & Software Requirements

The hardware and software requirements for the T-shirt e-commerce platform will depend on the technical design and the scale of the project.

3.6.1 Software Requirements

1. **Operating System:**The platform should be compatible with popular operating systems such as Windows, macOS, or Linux for server deployment.
2. **Web Server:**A web server such as Node.js will be used to handle the server-side operations of the platform. This server will manage the back-end logic, including product management, user authentication, and payment processing.
3. **Database Management System:**MongoDB will be used as the database management system to store user data, product information, order history, and other relevant details.
4. **Programming Languages:**
    - **Frontend:**The frontend will be built using React.js, HTML, CSS, and JavaScript to ensure a dynamic and interactive user interface.
    - **Backend:**The backend will use Node.js with Express.js to handle server-side functionality, including user authentication, cart management, and social media integration.
5. **Development Tools:**Development tools like Visual Studio Code will be used for coding, along with Postman for API testing, and Git for version control and collaboration.
6. **Third-party APIs:** Integration of social media APIs to allow users to share T-shirt products, along with payment gateway APIs like Easy Paisa for secure payment handling.

3.6.2Hardware Requirements

1. **Server Infrastructure:** A cloud-based server or hosting environment (e.g., **AWS**, **Heroku**, or similar) with sufficient CPU, RAM, and storage to handle user requests and manage backend operations.
2. **Storage:** Ample storage space is necessary to store product images, user accounts, order history, and other relevant data. Cloud storage solutions can be employed for flexibility and scalability.

3. **Client Hardware:** Standard devices like laptops, desktops, tablets, or smartphones can be used by both users and admin to access the platform. The admin should also have access to a reliable internet connection to manage products, orders, and updates to the site.
4. **Input Devices:** Standard hardware such as a keyboard, mouse, and internet-enabled devices will be required for development and admin use.

Chapter-4

System Design

4.1 Usecase

A use case describes the system’s behavior from the user's standpoint, providing an essential tool for system developers to understand the requirements from the perspective of users. Use case diagrams help gather the functional requirements of a system, including internal and external influences. These requirements are primarily related to design. During system analysis, use cases are prepared, and actors are identified based on system functionality. A use case is written in simple, natural language to ensure clarity for a broad audience, including customers, users, and executives.

4.1.1 Use case diagram objects

In a use case diagram, objects or entities that interact with the system are represented as actors. These actors can include individuals, external systems, or other systems within the same environment. Use case diagrams consist of the following objects:

- Actor
- Use case
- System

1. Actor

An actor in a UML Use Case Diagram is any entity (person, organization, or external system) that performs a role within a system. In a use case diagram, actors interact with use cases. In this project, two primary actors are involved in interacting with the system.

User: Represents a customer who browses products, selects T-shirts, adds items to the cart, and proceeds to checkout. Users can also post their selected T-shirts on social media.

Figure4.1: User

Admin: Represents the admin responsible for managing the product catalog, orders, and users. Admin can add, update, or delete products and track customer orders.

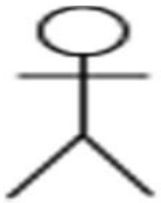

Figure4.2: Admin

2. Use case

A use case in a UML Use Case Diagram provides a visual representation of the system's business functionalities. Each use case shows how actors interact with the system. This use case demonstrates the key activities for both the customer and the admin throughout the e-commerce process, ensuring seamless navigation from browsing products to completing a purchase and sharing on social media.

Key Use Cases for User:

- **Login/Registration:**Users register or log in to their account.
- **Browse Products:** Users can navigate through the men, women, and kids categories.
- **Add to Cart:** Users select shirts and add them to their cart.
- **Checkout:** Users finalize the purchase and select a payment method.
- **Post to Social Media:** Users can share selected shirts along with price and links on social media.
- **Track Orders:** Users can view their order status.

Key Use Cases for Admin:

- **Manage Products:**Admins can add, update, or delete products from the catalog.
- **View Orders:**Admins can track and manage customer orders.
- **Manage Users:**Admins can manage user accounts and interactions.

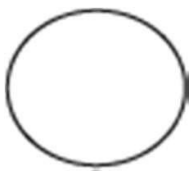

Figure4.3:Usecase

3. System

In a UML Use Case Diagram, the system is represented as a rectangle that encompasses all the use cases. This rectangle defines the system's scope, including the functionalities that the actors interact with. In this T-shirt e-commerce web app, the system interacts with both the User and Admin at various stages, allowing them to complete essential tasks like purchasing shirts, managing products, and handling orders.

Figure4.4:System

4.1.2 Use cases Model

A use case model describes the functionality of the proposed system. It represents a discrete unit of interaction between the User, Admin, and the system. This use case model outlines the steps involved in the e-commerce process for purchasing T-shirts, from the user's interaction with the system to the admin's product and order management.

List of Actors:

- **User:**This person interacts with the system to browse products, select shirts, add them to the cart, complete the purchase, and share the product on social media.
- **Admin:**This person manages the entire system, including adding products, tracking orders, and managing users.

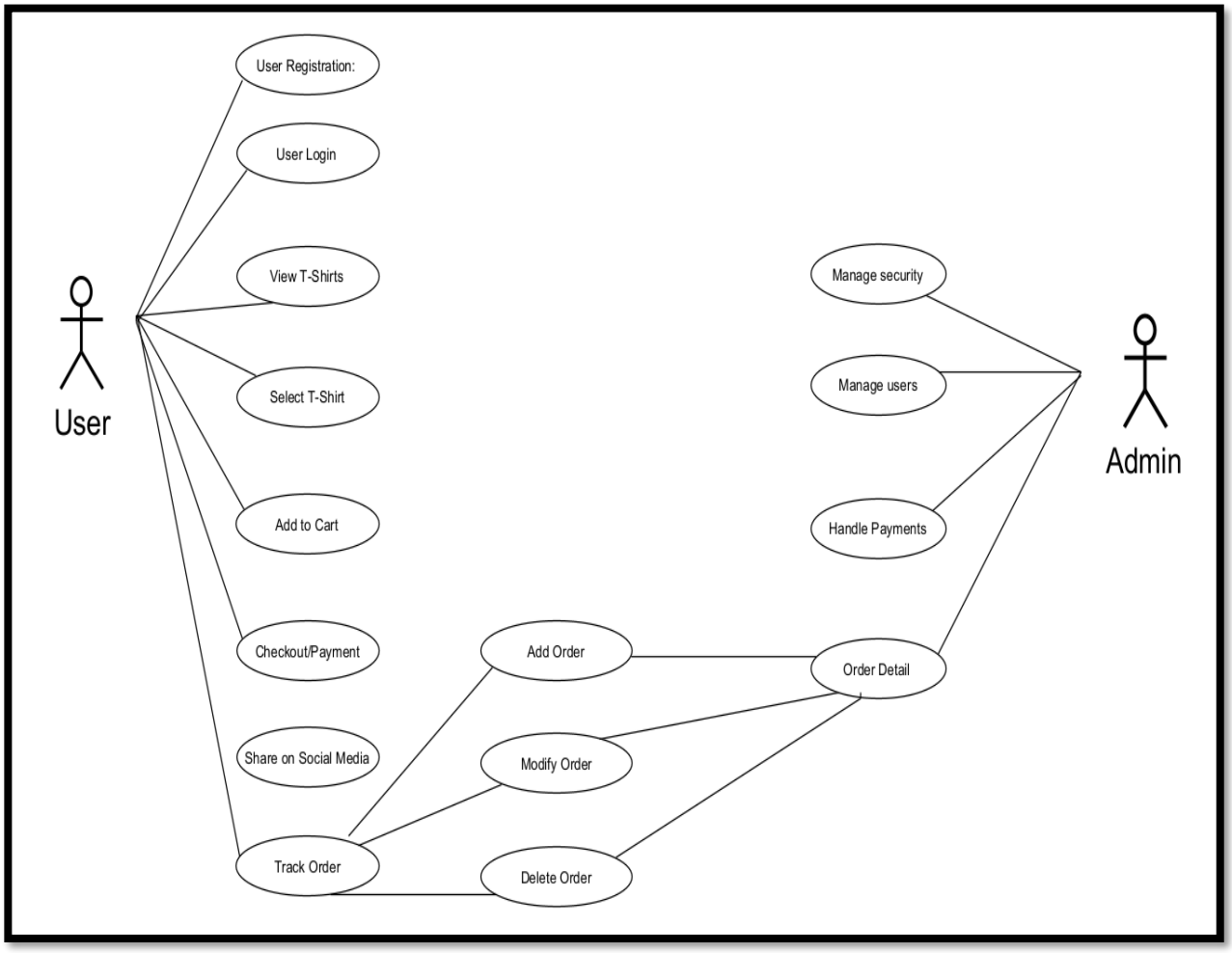

Figure 4.5: Use Case of T-shirt

Use Case Description:

1. Customer Use Cases:

- **User Registration:** Allows users to create an account on the website with their details such as username, email, and password.
- **User Login:**Allows users to log in to their account securely using their credentials.
- **View T-Shirts:**Users can view T-shirts displayed on the website's homepage, categorized into Men, Women, and Kids.
- **Select T-Shirt:**Users can choose their preferred T-shirts, including size and color options.
- **Add to Cart:**Users can add selected T-shirts to their shopping cart.
- **Checkout/Payment:** Users proceed to checkout and choose a payment method such as Easy Paisa or Cash on Delivery to complete the purchase.
- **Share on Social Media:**Users can share details of selected T-shirts, including the price and a

link to the website, on their social media accounts.

- **Track Order:**Users can track the status of their orders from placement to delivery.

2. Admin Use Cases

- **Mange Security:**Admin can add, update, or delete T-shirt listings from the catalog
- **View Orders:**Admin reviews and manages details of customer orders.
- **Manage Users:**Admin handles user account management and permissions.
- **Handle Payments**:Admin oversees payment processing and manages transaction details.
- **Track and Update Order Status:** Admin monitors and updates the status of orders to ensure timely fulfillment.

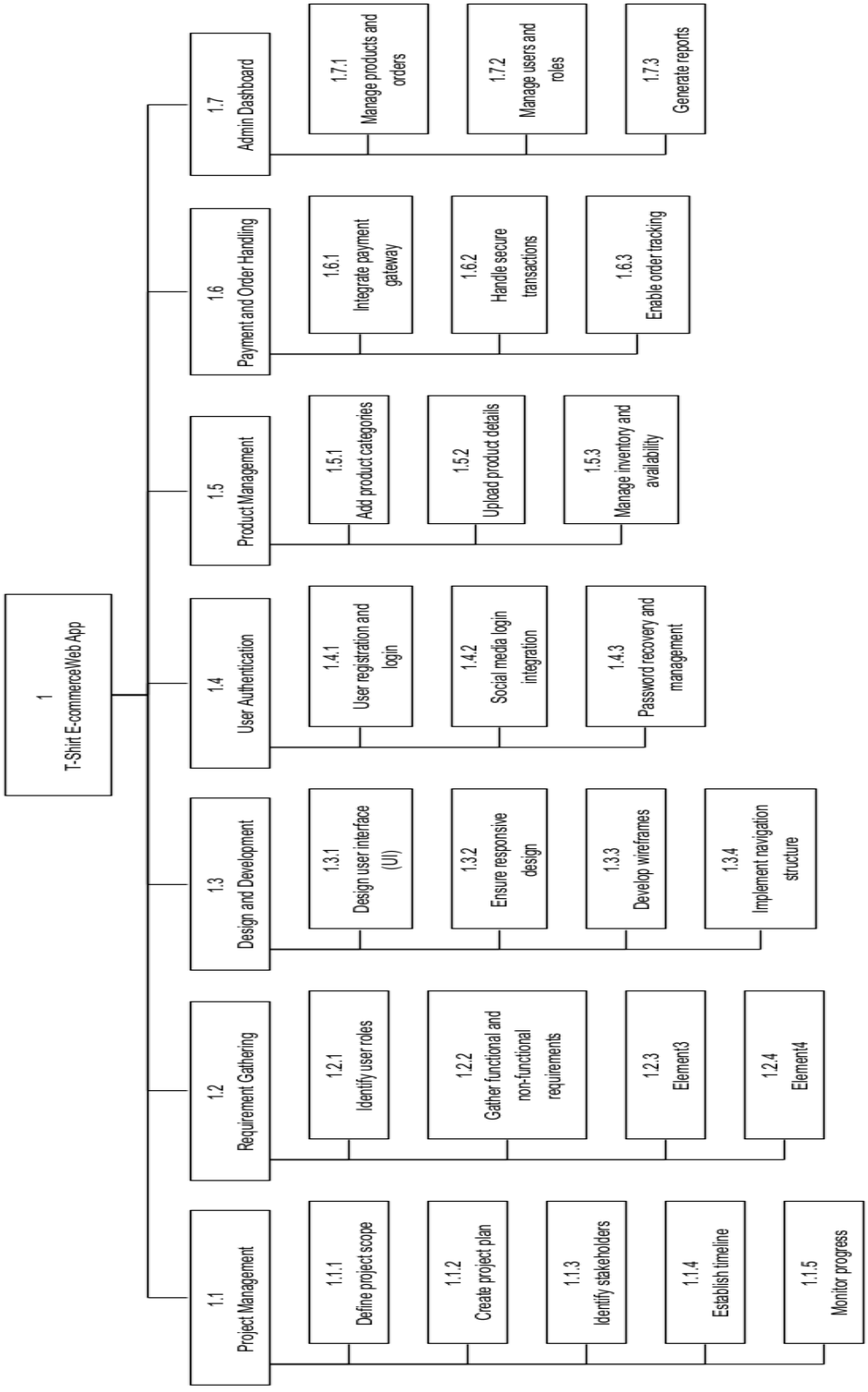

Figure 4.6:WBS T-shirt

1. Project Management			
WBS #:	1.1.1	**Task:**	Project scope and objectives
Est. Level of Effort:	1 week	**Owner:**	Project manager
Resources Needed:	Project team and stakeholders	**Work Products:**	Project scope statement and project objectives
Description of Task:	Collaborate with stakeholders to define the scope of the e-commerce project, including the platform features (product display, cart, payment, social media integration), and establish clear objectives aligned with the project goals.		
Input:	Stakeholders' requirements, project charter, e-commerce market analysis		
Dependencies:	None		
Risk:	Misalignment between stakeholder expectations and project scope		
WBS #:	1.1.2	**Work Item:**	Create project plan
Est. Level of Effort:	2 weeks	**Owner:**	Project manager
Resources Needed:	Project team, project management tools	**Work Products:**	Project plan, work breakdown structure, project schedule
Description of Task:	Develop a comprehensive project plan that outlines the tasks, timelines, resource allocations, and deliverables required for the development of the T-shirt e-commerce web app.		
Input:	Project scope and objectives, stakeholder requirements, resources availability		
Dependencies:	Completion of project scope and objectives task		
Risk:	Inaccurate estimation of effort, resources, or timelines		
WBS #:	1.1.3	**Task:**	Identify project stakeholders
Est. Level of Effort:	1 week	**Owner:**	Project manager
Resources Needed:	Project team, project management tools	**Work Products:**	
Description of Task:	Identify all individuals, groups, or organizations involved or affected by the project (e.g., developers, testers, end-users, admin, marketing).		
Input:	Project charter, organizational structure, stakeholder analysis templates		
Dependencies:	Completion of project scope and objectives task		
Risk:	Inadequate stakeholder engagement or communication		
WBS #:	1.1.4	**Work Item:**	Establish project timeline
Est. Level of Effort:	1 week	**Owner:**	Project manager
Resources Needed:	Project team, project scheduling tools	**Work Products:**	Project timeline, Gantt chart, milestone schedule
Description of Task:	Develop a detailed timeline for key development activities (e.g., frontend development, backend integration, testing, deployment, and marketing).		
Input:	Project plan, task dependencies, resources availability		
Dependencies:	Completion of project plan task		
Risk:	Inaccurate estimation of task duration or dependencies		

WBS #:	1.1.5	**Work Item:**	Monitor and control project scope
Est. Level of Effort:	Ongoing	**Owner:**	Project manager
Resources Needed:	Project team, project management tools	**Work Products:**	Project status reports, issue log, change requests
Description of Task:	Regularly track the project's progress, resolve issues, and manage changes to ensure the T-shirt e-commerce app development stays on track to achieve its objectives.		
Input:	Project plan, project performance metrices, stakeholder feedback		
Dependencies:	Completion of project plan task		
Risk:	Lack of timely project progress tracking or ineffective issue resolution		

2. Requirements Gathering

WBS #:	1.2.1	**Task:**	Identify key users' role
Est. Level of Effort:	1 week	**Owner:**	Business analyst
Resources Needed:	Business analyst, project team	**Work Products:**	User role documentation
Description of Task:	Identify the different types of users involved in the system, such as customers, administrators, and potentially guest users. Understand their roles and responsibilities in the system, specifically how they interact with the e-commerce functionalities (e.g., browsing products, adding to the cart, managing orders).		
Input:	Stakeholder Analysis, Project Charter		
Dependencies:	Project Initiation		
Risk:	Missing key user roles or incomplete understanding of user roles		
WBS #:	1.2.2	**Task:**	Gather Functional and Non-functional Requirements
Est. Level of Effort:	2 weeks	**Owner:**	Business analyst
Resources Needed:	Business Analyst, Subject Matter Experts	**Work Products:**	Requirements Documentation
Description of Task:	Collect and document both functional and non-functional requirements. Functional requirements include product search, shopping cart, checkout process, and order management. Non-functional requirements cover system performance, scalability, security, and usability.		
Input:	User Role Documentation, Stakeholder Feedback		
Dependencies:	Completion of User Role Identification		
Risk:	Ambiguous or incomplete requirements; overlooked non-functional aspects (e.g., security or performance)		
WBS #:	1.2.3	**Task:**	Analyze and Prioritize Requirements
Est. Level of Effort:	1 weeks	**Owner:**	Business analyst
Resources Needed:	Business Analyst, Project Team	**Work Products:**	Prioritized Requirements List

Description of Task:	Review and analyze the gathered requirements to ensure completeness. Prioritize these requirements based on their impact on business value and project feasibility, focusing on critical aspects like user authentication, payment integration, and scalability.		
Input:	Requirements Documentation, Stakeholder Feedback		
Dependencies:	Completion of Requirements Gathering		
Risk:	Misaligned priorities or incomplete analysis		
WBS #:	1.2.4	**Task:**	Document Requirements
Est. Level of Effort:	1 week	**Owner:**	Business Analyst, Technical Writers
Resources Needed:	Business Analyst, Technical Writers	**Work Products:**	Requirements Specification Document
Description of Task:	Formalize the functional and non-functional requirements into a comprehensive requirements specification document. This will serve as a reference for the development team and project stakeholders throughout the development lifecycle.		
Input:	Prioritized Requirements List, Stakeholder Feedback		
Dependencies:	Completion of Requirements Analysis and Prioritization		
Risk:	Incomplete or unclear documentation, leading to potential confusion in later stages of the project		

3. Design and Development

WBS #:	1.3.1	**Task:**	Design User Interface (UI)
Est. Level of Effort:	2 weeks	**Owner:**	UI/UX Designer
Resources Needed:	UI/UX Designer, Graphic Designer, Project Team	**Work Products:**	UI Design Mockups
Description of Task:	Create a visually appealing and intuitive user interface for the e-commerce platform, ensuring that the design is user-friendly and aligns with the overall brand identity. Focus on key areas such as the homepage, product pages, cart, and checkout process.		
Input:	Requirements Specification Document, Stakeholder Feedback		
Dependencies:	Completion of Requirements Documentation		
Risk:	Poor UI design leading to user dissatisfaction or confusion, misalignment with brand identity		
WBS #:	1.3.2	**Task:**	Ensure Responsive Design
Est. Level of Effort:	1 weeks	**Owner:**	UI/UX Designer, Frontend Developer
Resources Needed:	UI/UX Designer, Frontend Developer, Testing Tools	**Work Products:**	Responsive Design Mockups
Description of Task:	Ensure the platform is responsive and optimized for different screen sizes, including mobile phones, tablets, and desktops. Design for accessibility and usability across various devices to provide a seamless shopping experience.		
Input:	UI Design Mockups, Functional Requirements		
Dependencies:	Completion of UI Design Task		

Risk:	Inconsistent user experience across devices, poor accessibility		
WBS #:	1.3.3	**Task:**	Develop Wireframes
Est. Level of Effort:	1 week	**Owner:**	UI/UX Designer
Resources Needed:	Wire framing Tools (e.g., Figma, Adobe XD), UI/UX Designer, Project Team	**Work Products:**	Low-fidelity Wireframes
Description of Task:	Develop wireframes for key screens, including the homepage, product pages, and user account sections. These wireframes will serve as a visual blueprint for the development team, outlining the layout and basic functionality without detailed styling. **Input:** UI Design, Functional Requirements		
Input:	UI Design, Functional Requirements		
Dependencies:	UI Design Completion		
Risk:	Incomplete or unclear wireframes, leading to delays in development		
WBS #:	1.3.4	**Task:**	Implement Navigation Structure
Est. Level of Effort:	3 weeks	**Owner:**	Frontend Developer
Resources Needed:	Frontend Developer, UI/UX Designer	**Work Products:**	Navigation Menus, Sitemap
Description of Task:	Implement the navigation structure based on the wireframes and design. Ensure easy access to different sections, including product categories (men, women, kids), cart, user account, and checkout. Test for user-friendliness and logical flow. **Input:** Wireframes, UI Design		
Input:	Wireframes, UI Design		
Dependencies:	Completion of Wireframes		
Risk:	Confusing or complex navigation structure, leading to poor user experience		
4. User Authentication			
WBS #:	1.4.1	**Task:**	User Registration and Login
Est. Level of Effort:	1 week	**Owner:**	Backend Developer
Resources Needed:	Backend Developer, Database, Authentication APIs	**Work Products:**	User Registration and Login System
Description of Task:	Implement a user registration and login system allowing customers to create accounts and securely log in to the e-commerce platform. Ensure validation and security measures. **Input:** Functional Requirements, Database Schema		
Input:	Functional Requirements, Database Schema		
Dependencies:	Requirements Documentation, Database Setup		
Risk:	Poor security implementation or data breaches		
WBS #:	1.4.2	**Work Item:**	Social Media Login Integration
Est. Level of Effort:	2 weeks	**Owner:**	Backend Developer

Resources Needed:	Social Media APIs (e.g., Facebook, Google), Backend Developer	**Work Products:**	Social Media Login Functionality
Description of Task:	Integrate social media login options (e.g., Google, Facebook) into the user registration and login process, allowing users to log in through their social accounts.		
Input:	Social Media API Documentation, Authentication APIs		
Dependencies:	Completion of User Registration and Login System		
Risk:	API integration failures or security risks		
WBS #:	1.4.3	**Task:**	Password Recovery and Management
Est. Level of Effort:	3 weeks	**Owner:**	Backend Developer
Resources Needed:	Backend Developer, Email Service Provider	**Work Products:**	Password Recovery and Reset System
Description of Task:	Implement password recovery functionality, allowing users to reset their passwords securely via email.		
Input:	Email Integration, User Data		
Dependencies:	Completion of User Registration and Login System		
Risk:	Delays in email delivery, security breaches		

5. Product Management

WBS #:	1.5.1	**Task:**	Add Product
Est. Level of Effort:	1 weeks	**Owner:**	Backend Developer, Admin
Resources Needed:	Database, Product Category Definitions	**Work Products:**	Product Categories in Database
Description of Task:	Create and add product categories for men, women, and kids to the database, enabling classification of items.		
Input:	Functional Requirements, Admin Input		
Dependencies:	Database Setup		
Risk:	Incorrect product categorization		
WBS #:	1.5.2	**Task:**	Upload Product Details
Est. Level of Effort:	2 week	**Owner:**	Admin, Backend Developer
Resources Needed:	Product Data (images, descriptions, prices), Admin Interface	**Work Products:**	Product Detail Pages
Description of Task:	Upload product details (e.g., images, descriptions, prices) for all available products via the admin dashboard.		
Input:	Product Data, Admin Interface		
Dependencies:	Add Product Categories		
Risk:	Incorrect or missing product details		
WBS #:	1.5.3	**Task:**	Manage Inventory and Availability
Est. Level of Effort:	2 weeks	**Owner:**	Admin, Backend Developer

Resources Needed:	Inventory Management Tools, Database	**Work Products:**	Inventory Management System
Description of Task:	Implement an inventory management system to track stock levels and availability for each product, ensuring accurate product listings.		
Input:	Inventory Data, Product Database		
Dependencies:	Upload Product Details		
Risk:	Incorrect inventory tracking, out-of-stock items		

6. Payment and Order Handling

WBS #:	1.6.1	**Task:**	Integrate Payment Gateway
Est. Level of Effort:	2 weeks	**Owner:**	Backend Developer
Resources Needed:	Payment Gateway API (e.g., Stripe, PayPal), Backend Developer	**Work Products:**	Payment Gateway Integration
Description of Task:	Integrate a payment gateway to securely handle payments for purchases made on the platform.		
Input:	Payment Gateway API Documentation		
Dependencies:	User Registration and Login System		
Risk:	Payment processing failures or security issues		
WBS #:	1.6.2	**Task:**	Handle Secure Transactions
Est. Level of Effort:	1 week	**Owner:**	Backend Developer
Resources Needed:	Secure APIs, Payment Gateway	**Work Products:**	Secure Transaction System
Description of Task:	Implement secure transaction handling, ensuring data encryption and fraud prevention during payments.		
Input:	Payment Gateway API, Security Guidelines		
Dependencies:	Integrate Payment Gateway		
Risk:	Security vulnerabilities, fraudulent transactions		
WBS #:	1.6.3	**Task:**	Enable Order Tracking
Est. Level of Effort:	1 weeks	**Owner:**	Backend Developer, Frontend Developer
Resources Needed:	Order Tracking System, Admin Interface	**Work Products:**	Order Tracking Feature
Description of Task:	Allow users to track their order status from purchase to delivery, providing updates on shipping and delivery status.		
Input:	Order Data, Shipment Integration		
Dependencies:	Secure Transactions		
Risk:	Incorrect order tracking information		

7. Admin Dashboard

WBS #:	1.7.1	**Task:**	Manage Products and Orders
Est. Level of Effort:	2 weeks	**Owner:**	Admin, Backend Developer

Resources Needed:	Admin Dashboard, Backend System	**Work Products:**	Admin Dashboard for Products and Orders
Description of Task:	Allow the admin to manage products, add new products, update existing product information, and manage orders (add, modify, delete orders).		
Input:	Product and Order Data		
Dependencies:	Product Management System, Order System		
Risk:	Admin errors, product mismanagement		
WBS #:	1.7.2	**Task:**	Manage Users and Roles
Est. Level of Effort:	2 week	**Owner:**	Admin, Backend Developer
Resources Needed:	Admin Dashboard, User Data	**Work Products:**	User Management System
Description of Task:	Implement user and role management functionality within the admin dashboard, allowing the admin to manage user accounts, assign roles, and handle permissions.		
Input:	User Data, Role Definitions		
Dependencies:	User Authentication		
Risk:	Incorrect role assignments, security issues		
WBS #:	1.7.3	**Task:**	Generate Reports
Est. Level of Effort:	1 weeks	**Owner:**	Admin, Backend Developer
Resources Needed:	Reporting Tools, Data Analytics	**Work Products:**	Sales, Inventory, and User Reports
Description of Task:	Implement functionality that allows the admin to generate reports on sales, product inventory, and user activity for informed decision-making.		
Input:	Transaction Data, Inventory Data		
Dependencies:	Product and Order Management, Payment System		
Risk:	Inaccurate reports, missing data		

4.3 System sequence diagram:

A sequence diagram is a Unified Modeling Language (UML) diagram that illustrates the interactions between the customer and the e-commerce platform during the T-shirt ordering process. This sequence diagram focuses on viewing T-shirts, selecting a T-shirt, and confirming the order in the context of your project.

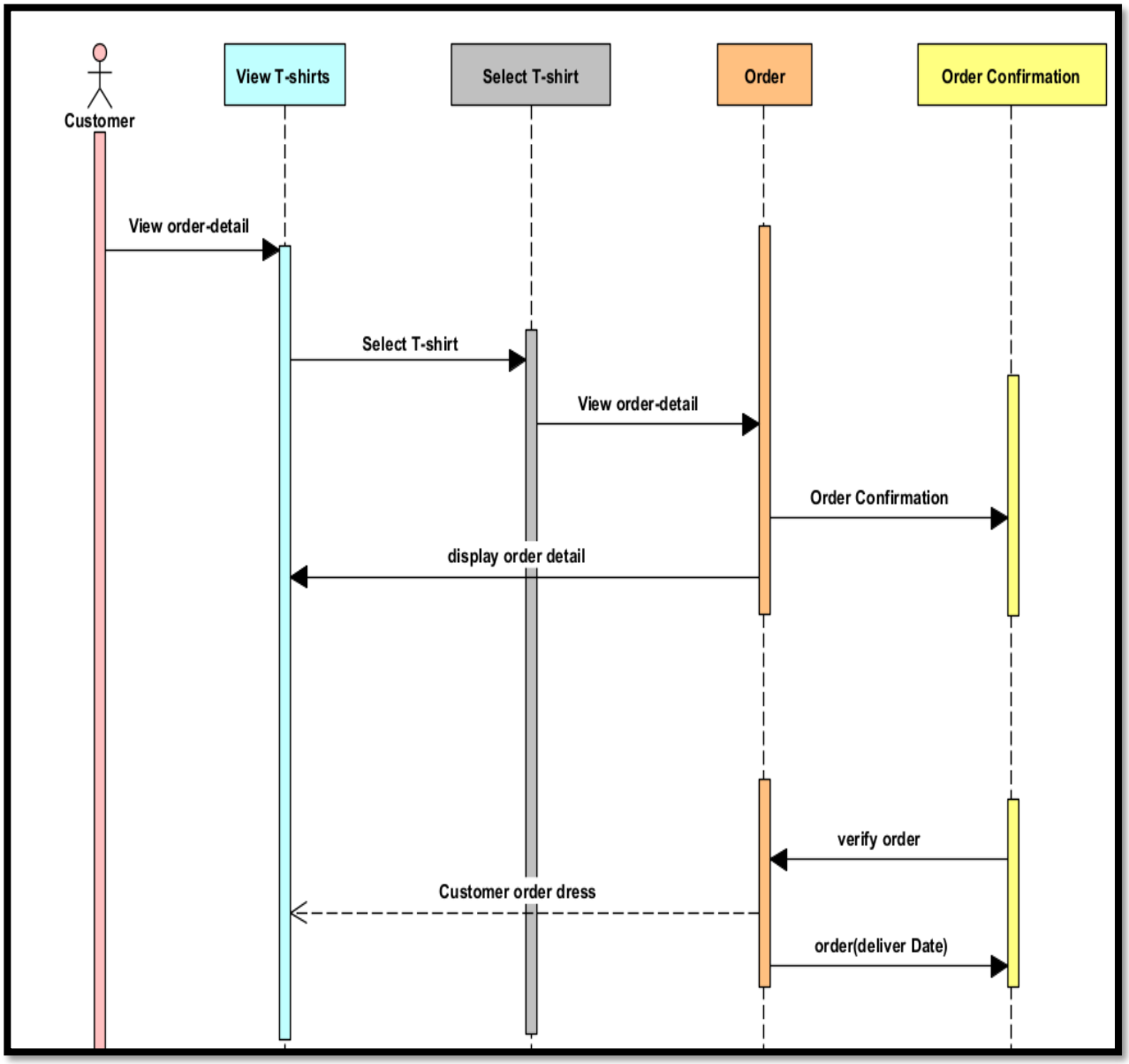

Figure 4.7: Sequence Diagram of T-shirt

Description of sequence diagram:

1. View T-shirts:

The customer interacts with the web platform to search for T-shirts. The system retrieves a list of available T-shirts from the database and displays them to the customer for browsing. The customer can explore the website's collection by selecting specific categories like gender (men, women, kids), size, color, or fabric type. The system ensuresthat the available stock is correctly shown according to the chosen filters.

2. Select T-shirt:

After viewing the available options, the customer selects a specific T-shirt. The selection is transmitted to the platform's backend, which retrieves and displays detailed product information such as price, available sizes, colors, and a high-resolution image of the T-shirt. This step allows the customer to review the product before adding it to their cart.

3. **Order:**

Once the customer selects the T-shirt and adds it to their cart, they proceed to the order confirmation phase. The system checks the availability, calculates the total cost (including taxes and shipping fees), and presents an order summary. The customer then provides their shipping address, selects a payment method, and completes the checkout process.

4. **Order confirmation:**

After the order is placed, the system generates an order confirmation, which includes details such as the order number, estimated delivery date, and a confirmation email. The system also verifies the payment and sends a tracking number to the customer for real-time updates on the order's delivery status.

4.4 Class Diagram:

This class diagram provides a simplified representation of the classes involved in this project. It depicts the high-level structure and relationships between the customer, outfitter, and database components.

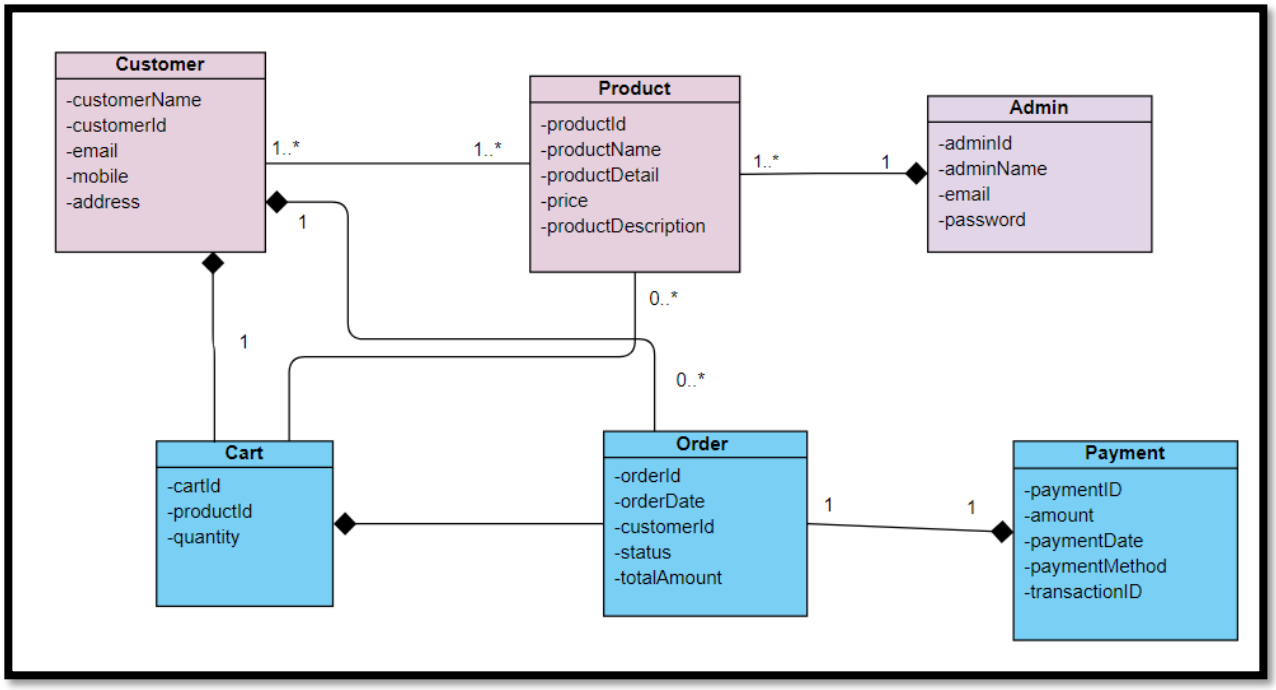

Figure: 4.8: Class diagram of T-shirt

Description:

- **Customer:**Represents a user of the app. It contains attributes such as the customer's name, email, phone number, and address. The customer can register, log in, and update their profile on the platform.

- **Admin:**The Admin class represents the system administrator responsible for managing the platform's operations. This includes overseeing the product catalog, which involves adding new products, updating existing ones, and removing outdated items. Admins also handle order processing

and customer management, ensuring that the system runs smoothly and efficiently.

- **Product:**The Product class represents the items available for purchase on the platform. It includes attributes such as product name, description, price, and the quantity in stock. Products are displayed to customers who can view detailed information and add them to their shopping cart.

- **Orde:**The Order class captures the details of a purchase made by a customer. It includes attributes like the order ID, date of purchase, total amount, status (e.g., pending, shipped, delivered), and payment method. Orders are generated when customers complete their purchases and are tracked through various stages of fulfillment.

- **Cart:**The Cart class holds information about the items a customer intends to buy. It includes attributes such as the cart ID, the list of items added, and the total price of the cart's contents. Customers can view, update, and manage their cart before proceeding to checkout.

- **Payment:**The Payment class represents the transaction details for an order. It includes attributes such as payment ID, amount, payment date, payment method, and transaction ID. Payment information is used to process transactions and verify that payments are completed successfully.

4.5 ER model:

The ER(Entity-Relationship) model for T-shirt represents the structure and relationships between entities in the system.

Description:

The final output of the ER diagram shows the relationships between the key entities in the T-shirt e-commerce system. First, a customer provides their personal information to register or log in. Once logged in, they can browse products, add items to their cart, and place an order. The customer's order includes selecting a payment method orders, and processing payments. Additionally, the admin ensures that all orders are tracked, managed, and fulfilled properly and completing the checkout process. The admin manages various tasks, including overseeing customer registration and login, updating the product catalog, handling

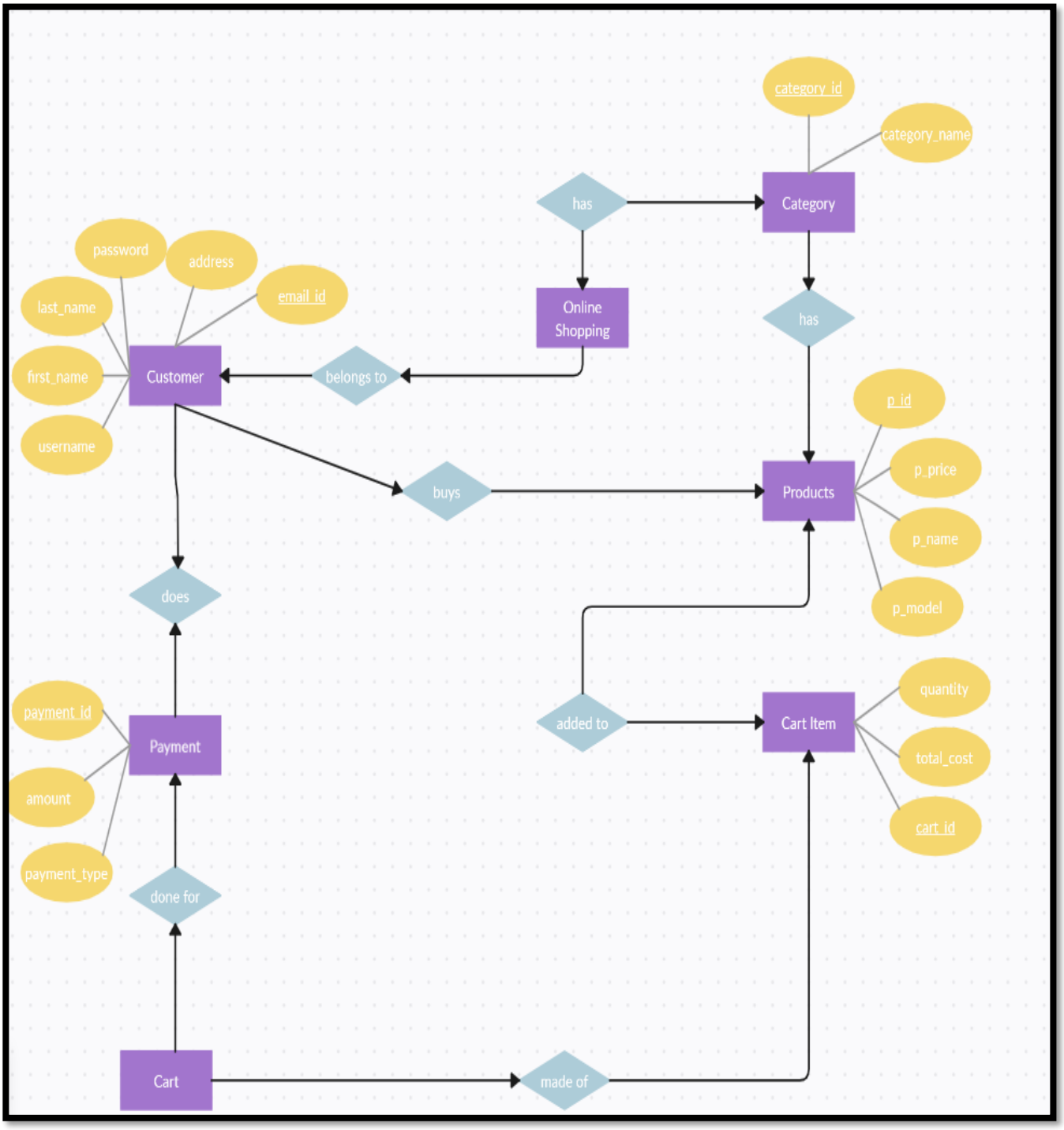

Figure 4.9: ERD of T-shirt

4.6 Data flow diagram:

The final output of the ER diagram shows the relationships between the key entities in the T-shirt e-commerce system. First, a customer provides their personal information to register or log in. Once logged in, they can browse products, add items to their cart, and place an order. The customer's order includes selecting a payment method and completing the checkout process. The admin manages various tasks, including overseeing customer registration and login, updating the product catalog, handling orders, and processing payments. Additionally, the admin ensures that all orders are tracked, managed, and fulfilled properly.There are three level of data flow diagram level 0, 1, 2.

4.6.1 Context or zero level DFD:

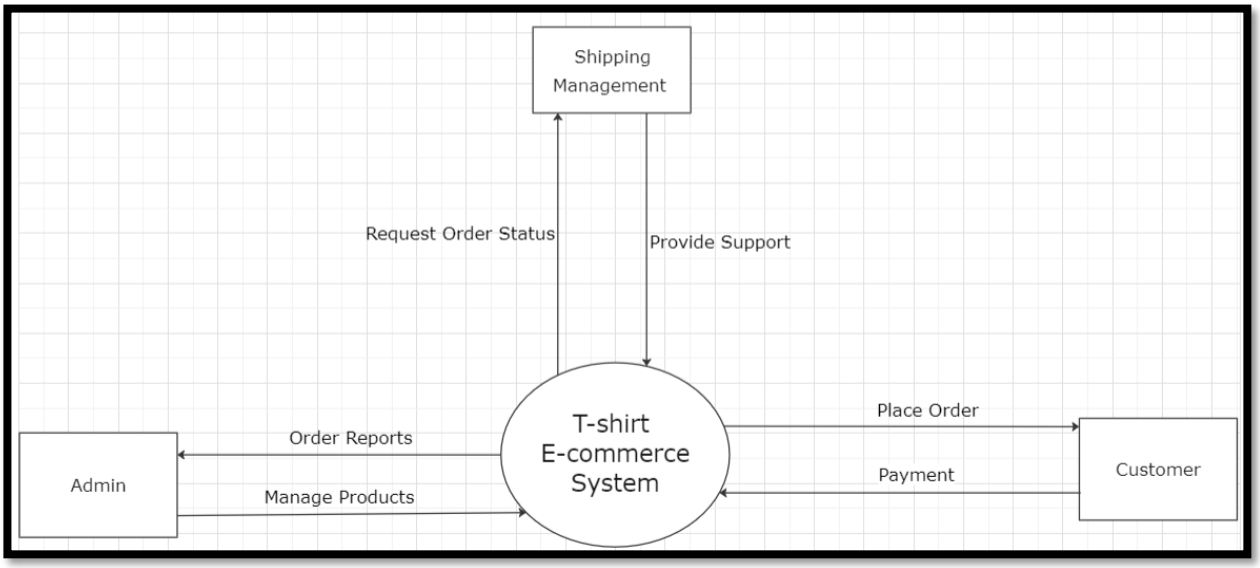

Figure 4.10: Context level DFD of T-shirt E-commerce

Description of context level DFD:

This context-level DFD provides an overview of the major components and their interactions, without delving into the internal processes of each entity. The customer browses the product catalog and places an order request through the T-shirt e-commerce system. The system processes the order and sends the necessary information to the admin. The admin manages the products, updates product information, and processes the orders in the system. The customer can view available products, track their order status, and proceed to payment. The system also handles the payment method to confirm the order.

4.6.2 Level 1 DFD:

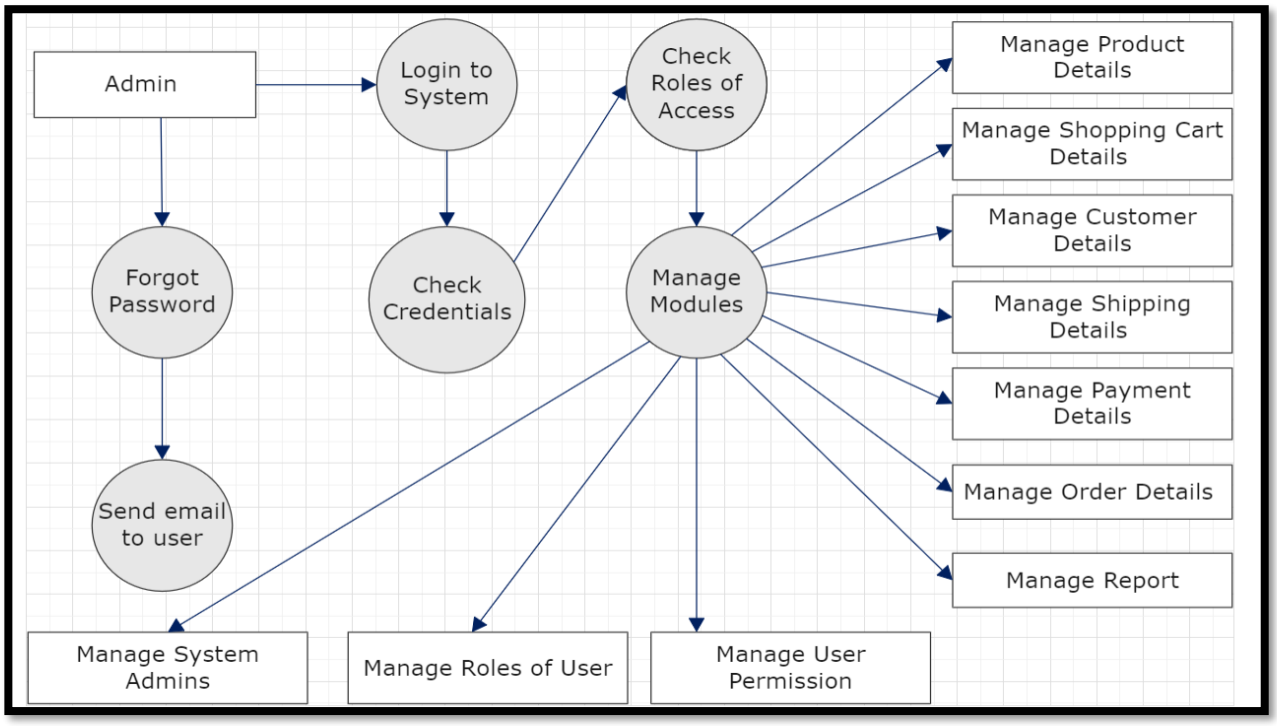

Figure 4.10.1: level 1 DFD of outfitter customization

Description of level 1 DFD:

This Level 1 DFD provides a detailed view of the main processes handled by the admin in the T-shirt e-commerce system. The admin plays a critical role in ensuring that the system functions smoothly. The following processes are managed:

The following processes are managed:

1. **Login to System:**The admin logs in to the system using their credentials, and the system checks the validity of their login information.
2. **Check Roles of Access:**After logging in, the system checks the admin's access level to determine the roles they can perform within the system.
3. **Manage Modules**: The admin has access to various modules to manage critical aspects of the e-commerce system:
    - **Manage Product Details:** This allows the admin to add, update, or remove product information (like t-shirt details such as size, color, and price).
    - **Manage Shopping Cart Details:** Admins can monitor and resolve any issues related to customers' shopping carts.
    - **Manage Customer Details:** Admins have access to customers' personal information, such as names, emails, and addresses.
    - **Manage Shipping Details:** Admins oversee shipping addresses and shipping status updates for customer orders.
    - **Manage Payment Details:** The admin manages customer payment records and ensures payments are correctly processed for each order.
    - **Manage Order Details:** Admins can view all customer orders and track their status (e.g., pending, completed).
    - **Manage Reports:** Admins can generate reports related to sales, customer activity, and system performance.
4. **Forgot Password:**If an admin forgets their password, the system allows them to reset it by sending an email to recover their credentials.
5. **Check Credentials:**The system verifies the admin's credentials when they log in or request a password reset.

6. **Manage Roles of User:**The admin manages user roles, assigning different levels of access and permissions to other admins or users.
7. **Manage User Permissions:**Admins can assign and manage specific permissions for other users, controlling what they can access or modify within the system.
8. **Manage System Admins:**This process allows the admin to add or remove other system administrators, giving them access to manage the platform.

Chapter-5

Implementation and Testing

Implementation and Testing

System testing methods involve various strategies used to ensure that the T-shirt e-commerce platform works as expected, covering both the front-end and back-end. These methods include unit testing, integration testing, and system testing. During implementation, developers use programming languages, frameworks, and tools to build the software based on the design and requirements.

5.1 Overview

Implementation refers to translating design specifications into a working software system. This process involves coding the core components such as product management, shopping cart functionality, user authentication, and secure payment integration. Testing ensures the system meets the desired requirements and performs reliably. The phases of implementation and testing are closely interconnected, ensuring smooth platform functionality and user satisfaction.

5.2 Testing method

Testing in the T-shirt e-commerce system follows an iterative process to ensure functionality, usability, and reliability. The following steps outline the methods used:

- **Define System Components:**Identify the core features of the platform, including product listings, shopping cart, checkout process, payment gateway, and user authentication.
- **Develop a Testing Plan:**This plan outlines the key areas to evaluate during testing, covering both functional aspects (adding products to the cart, completing purchases) and non-functional aspects (system performance and load handling).
- **Create a Prototype:**Develop a working prototype of the e-commerce platform to simulate the final product. This allows for quick feedback and iteration before full deployment.
- **Usability Testing:**Have users interact with the platform prototype, focusing on how easily they can browse products, add items to the cart, and complete the purchase process. Gather feedback to improve the overall user experience.
- **Collect Data:**Use surveys or analytics tools to gather data on user satisfaction, system performance, and any issues encountered during testing.
- **Iterate and Refine:**Based on feedback, make improvements and retest the system until it meets all requirements.

- **Final Evaluation:**Once all major issues are resolved, conduct a final round of testing with a larger user base to validate the system and ensure it's ready for deployment.

5.3 Functional and Non-functional testing:

These requirements ensure that the platform performs well, is user-friendly, secure, compatible with different devices and browsers, reliable, scalable to handle increasing users and products, and accessible to a diverse range of customers.

5.3.1 Functional Testing

1. Unit Testing

Unit tests verify the functionality and behavior of individual units of code, such as components, functions, or modules, in isolation. In the context of the T-shirt e-commerce platform, unit tests ensure that key features like product display, shopping cart, user authentication, and payment processing work correctly without causing unexpected issues. Each unit is tested independently to confirm it performs as intended before integrating with the rest of the system.

2. Integration Testing

Verifying the integration and interaction between different components, modules, and systems within the T-shirt e-commerce platform. This includes testing the data flow, communication, and compatibility between various subsystems, such as product management, shopping cart, order processing, and payment gateway. Integration testing ensures that when a user adds a product to the cart, it correctly updates during checkout, and that all modules work seamlessly together.

3. System Testing

System testing of the T-shirt e-commerce platform involves testing the entire system to ensure it functions as intended and meets the specified requirements. This testing focuses on verifying the integration, functionality, and performance of key features such as product display, shopping cart, user login, and secure payment processing. It ensures that the platform delivers a seamless shopping experience from product selection to checkout.

4. Acceptance Testing

Acceptance testing of the T-shirt e-commerce platform involves validating that the system meets the requirements and expectations of the end-users (customers and admin) and stakeholders. It ensures that the platform functions as intended, providing a smooth user experience from product browsing to checkout. This is the final phase of testing before the platform is deployed live.

5.3.2 Non- functional Testing

Non-functional testing for the T-shirt e-commerce platform focuses on evaluating system aspects that are not directly related to its functional requirements. This includes assessing the platform's performance, security, reliability, scalability, and user experience to ensure it operates efficiently

under different conditions and provides a seamless and secure shopping experience for customers.

1. **Performance Testing:**
   - **Load Testing:**Evaluate the app's performance with multiple users accessing the homepage, browsing categories, and completing purchases.
   - **Stress Testing:**Test the app under extreme conditions, such as high user volume, to ensure stability.
   - **Response Time Testing:**Measure the speed of user interactions, including page loads, category navigation, and checkout processes.

2. **Security Testing:**
   - **Authentication and Authorization Testing**:Ensure proper user authentication and permissions for accessing accounts and performing actions.
   - **Data Protection:**Verify that customer data is securely stored and transmitted, with encryption and best practices in place.

3. **Usability Testing:**
   - **User Interface Testing:**Check the ease of use and consistency across the homepage, category navigation, and checkout process.
   - **Accessibility Testing:**Confirm that the app is accessible to users with disabilities, meeting relevant accessibility standards.

4. **Scalability and Capacity Testing:**
   - **Scalability Testing:**Assess the app’s ability to handle increased user activity by scaling resources.
   - **Capacity Planning:**Identify the maximum load capacity and any potential performance bottlenecks.

5.4 Test Cases:

To ensure the T-shirt e-commerce web app operates correctly and meets user expectations, a variety of test cases must be executed. Start by verifying that users can successfully log in with valid credentials and are directed to the homepage. Test the navigation through the "men," "women," and "kids" categories to ensure that users can view the appropriate T-shirt listings. Check that users can select a T-shirt, view its details, and add it to the cart without issues. Assess the cart functionality by ensuring users can view, modify, and remove items as needed. Confirm that the checkout process functions smoothly, allowing users to proceed,

select payment methods, and complete their purchases. Additionally, verify that users can successfully post their selected T-shirt, including its price and a link to the website, to their social media accounts. Finally, ensure that the system handles errors properly by displaying appropriate messages for invalid login attempts, out-of-stock items, or failed payments.

5.4.1 Data and database:

For the T-shirt e-commerce web app, the data and database requirements are crucial for handling various functionalities and ensuring smooth operations. The data model and database solution must be designed to meet the needs of the system efficiently. The key data elements include:

Data/Database	Description
User Data	Stores user data and authentications credentials.
Product Catalog	Contains details about available T-shirts, including images, descriptions, sizes, colors, pricing, and measurements.
Cart Data	Records the items added to the cart by users.
Order Details	Stores information about placed orders, including user details, selected T-shirts, billing information, and delivery status.
Payment Information	Securely holds payment details for transactions.
Relational Database	Utilizes a relational database management system (DBMS) for structured data storage and retrieval.
Database Schema	Defines the logical structure of the database, including tables, fields, and relationships between entities.
Data Privacy and Security	Ensures compliance with data protection regulations, implements encryption, and performs regular backups.

5.4.2 Functional testing:

Functional testing for the T-shirt e-commerce web app involves verifying that all features work correctly and meet the required functionality. The following test cases should be customized and expanded based on specific requirements:

Test case Description	Test steps	Expected Result
Product Selection	• Browse available T-shirts. • Select a T-shirt to add to the cart.	The selected T-shirt should be added to the cart and available for review.
Cart Functionality	• View the cart. • Modify the quantity of items. • Remove an item from the cart	The cart should update correctly to reflect the changes in item quantities and removal of items.

Cost Calculation	• Add T-shirts to the cart. • Proceed to checkout. • Verify the total cost displayed.	The calculated total cost should match the sum of the item prices and any applicable taxes or discounts.
Order Placement	• Add T-shirts to the cart. • Provide necessary customer information (e.g., shipping address, payment details). • Complete the purchase..	An order confirmation should be displayed with the order details, including a summary of the purchased items and estimated delivery date
Invalid Input Handling	• Add items to the cart. • Attempt to proceed with incomplete or invalid payment information..	An error message should be displayed indicating that the payment information is incomplete or invalid.

5.4.3 user interface Testing

User interface (UI) testing for the T-shirt e-commerce web app focuses on evaluating the visual aspects, usability, and interaction of the website:

Test case description	Test steps	Expected Result
Layout and Design	Verify the overall layout, design, and visual presentation of the homepage, product pages, cart, and checkout interface.	The UI should have an appealing and user-friendly design, with consistent styling and clear navigation
Responsive Design	Test the website on different screen sizes and devices (e.g., desktop, mobile, tablet).	The UI should adapt and display properly across various devices, maintaining usability and visual integrity.
Navigation and User Flow	Navigate through the website, including browsing categories, selecting products, managing the cart, and completing the checkout process.	Navigation should be intuitive and guide the user smoothly through each step of the process, from product selection to order confirmation.
Responsiveness of Real-Time Updates	• Add items to the cart and proceed through the checkout process. • Observe real-time updates in the cart and order details.	Updates should be immediate and accurately reflect changes in the cart and order details.
Usability and Ease of Interaction	Interact with UI elements such as buttons, dropdowns, sliders, and other interactive components.	UI elements should be user-friendly, responsive, and function as intended, providing a smooth user experience.

5.4.4 performance testing of login page

To ensure the login page of the T-shirt e-commerce web app performs well under various conditions, the following test cases should be considered:

Test case description	Test steps	Expected result
Valid Login Credentials	• Enter a valid email address and password. • Click the login button.	The user should be successfully logged in and redirected to the homepage.
Empty Email or Password	• Leave the email field empty and enter a valid password. • Click the login button.	An error message should be displayed indicating that the email field is required.
Forgotten Password Functionality	Click on the "Forgot password?" link.	The user should be redirected to the password recovery page or presented with password reset options.
Account Logout Handling	• Enter invalid login credentials multiple times, exceeding the maximum allowed attempts. • Click the login button.	The user should be prevented from logging in and receive a message indicating that the account has been locked due to multiple failed attempts.

Chapter-6

User Manual

6.1 Login Page

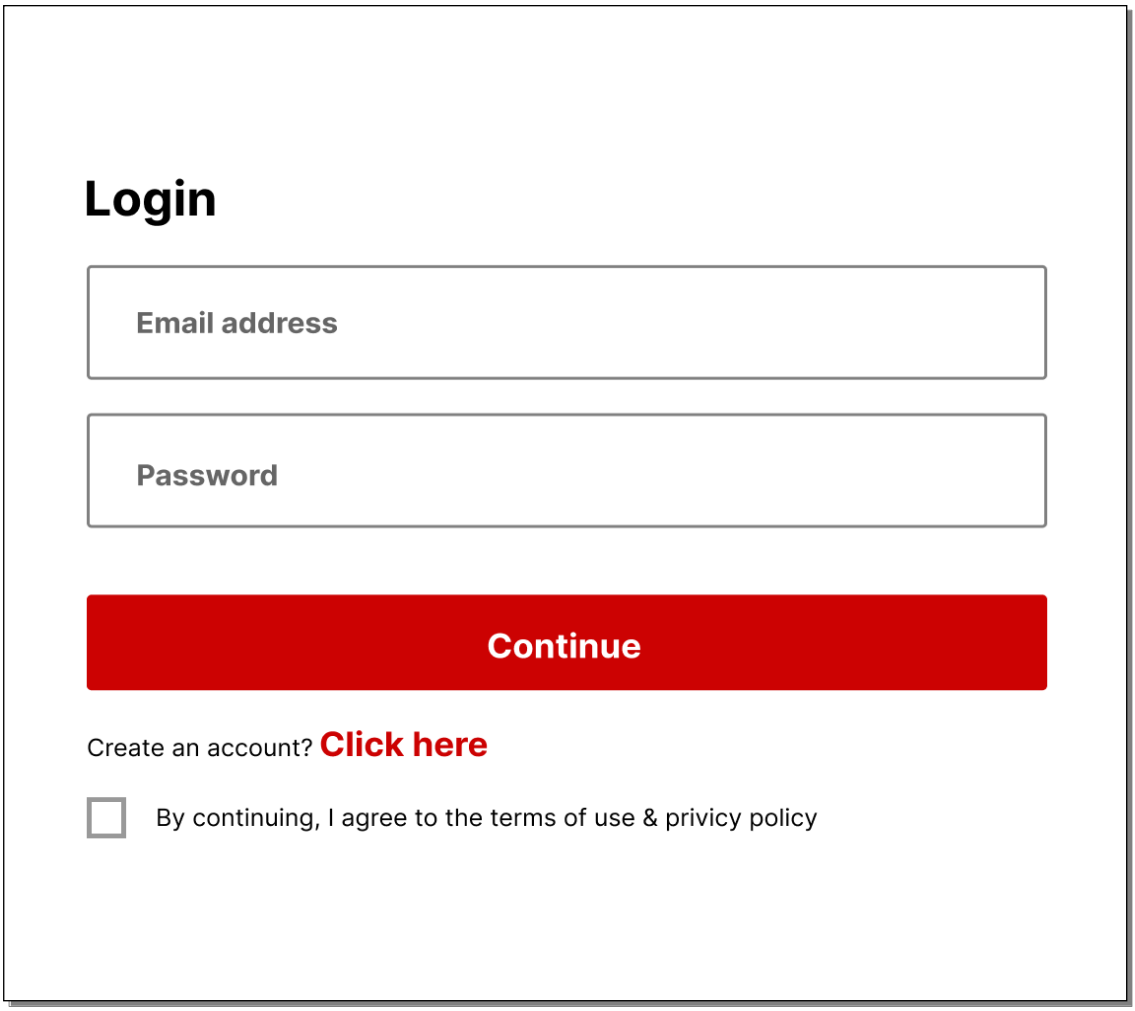

Figure 6.1: Login Page of outfitter customization

Description:

- When a user visits the homepage of the T-shirt e-commerce web app, the **login page** is the first screen that appears. Every user must log in to proceed with browsing the shirts and purchasing.
- On the **login page**, the user enters their email address and password. The system checks these credentials against the registered user data in the database. If the credentials are correct, the user is successfully authenticated and granted access to the app.
- Upon entering the email and password, an OTP (One-Time Password) is sent to the user's registered email for an additional layer of authentication. The user must input the correct OTP to continue.
- If the user is inactive after logging in, their session will expire after a certain time, defaulting to **2 hours**. This ensures security, and the user will be automatically logged out. After this period, they will need to log in again to continue using the app.

- If a user tries to navigate directly to any URL without logging in, they will be redirected back to the **login page**. This ensures that all actions within the web app are performed by authenticated users only.

- If a logged-in user attempts to access the **login page** by typing the URL, they will be redirected to the **homepage**, as they are already authenticated.

6.2 Register Page:

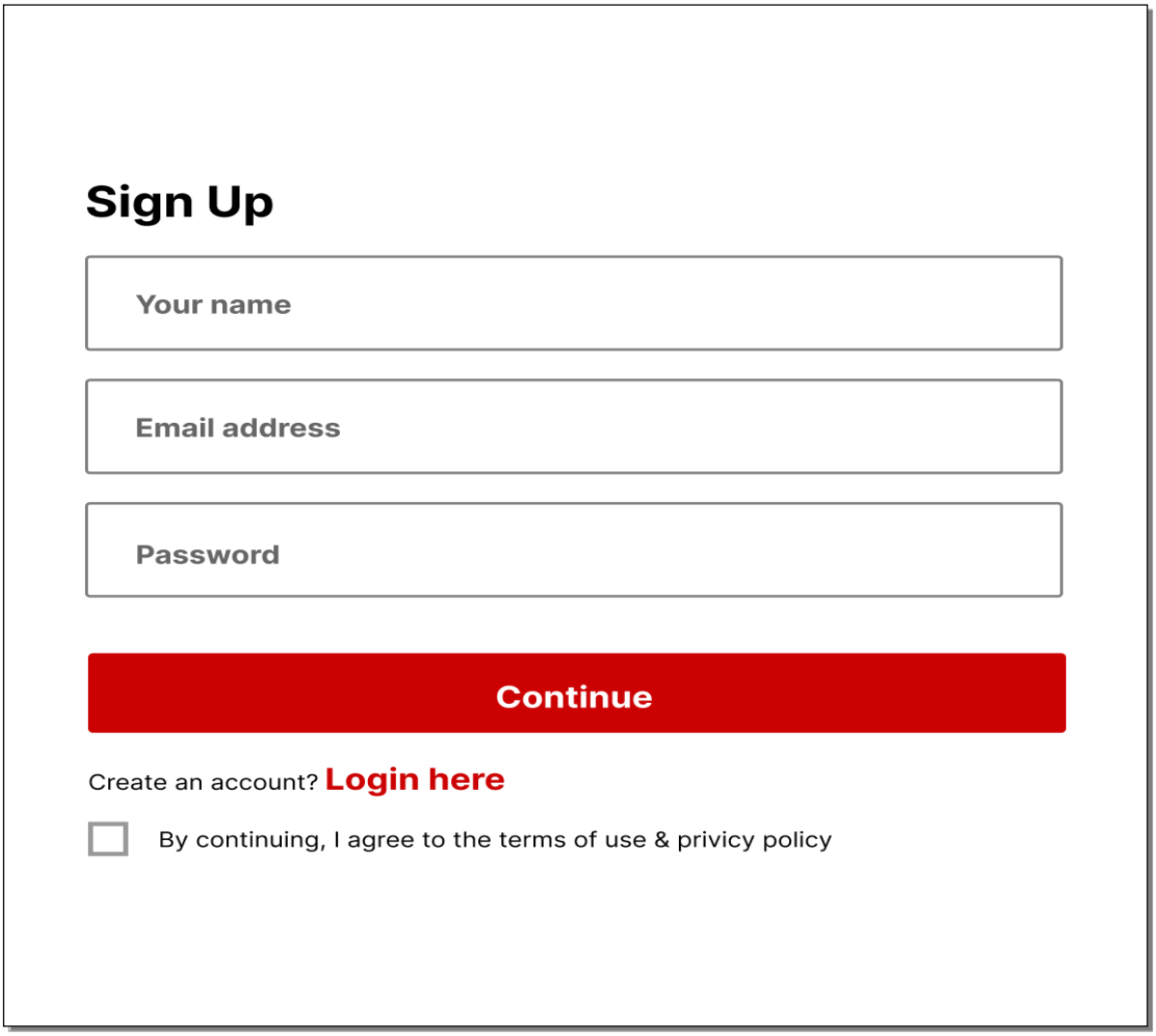

Figure 6.2: Register Page of outfitter customization

Description:

- A register page consists of a form with fields for users to enter their information, such as their full name, email address, and password.
- The design of a register page focuses on simplicity and usability, aiming to guide users through the registration process smoothly. It often includes instructions or tooltips to help users understand the required information and any specific formatting guidelines for the fields.

6.3: User classes and characteristics:

6.3.1 Home Page:

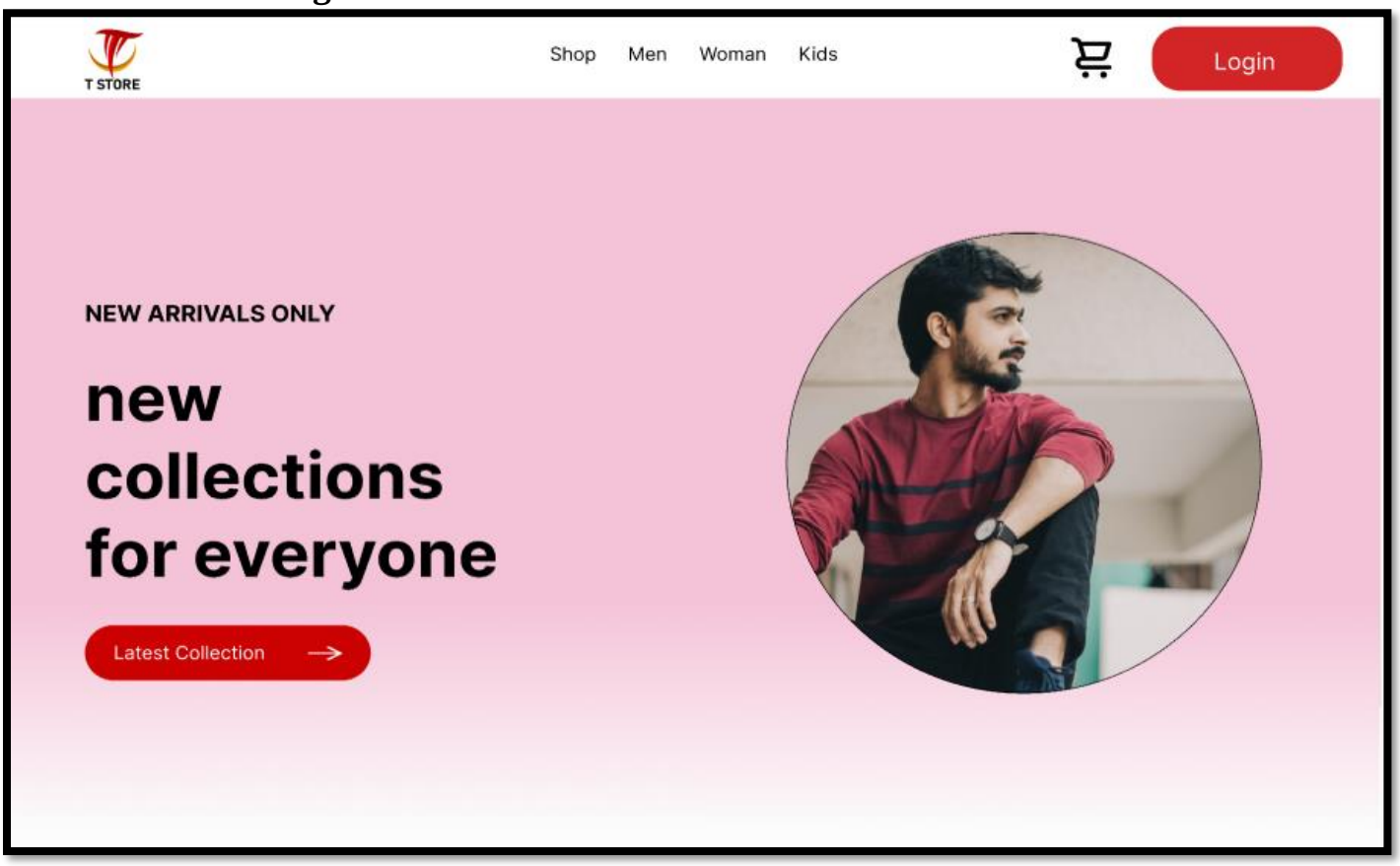

Figure: 6.3 Home page of T-shirt E-commerce

- Home page of a website serves as the main entry point and typically sets the tone for the entire site. It is designed to provide an overview of the website's content, features, and navigation options, while also engaging and directing users to other important areas of the site.
- On Home page users see many features and having multiple pages linked on this page like login and register page.
- When user click on specific page then directly go to that page.
- The top section of the home page usually contains the website's logo, and prominent navigation menu. In menu having some other pages which is for category Men, Women and kids.
- Some important content of our website mention in home page on hero section having one main button for new Collection.
- The specific layout, design, and content of the home page may vary depending on the website's purpose and target audience.

6.3.1.1 Hero Section:

The hero section is a visually appealing and attention-grabbing area located prominently on the home page. We have added some designs of clothing on our website to make it easier for users to understand our services and place orders conveniently.

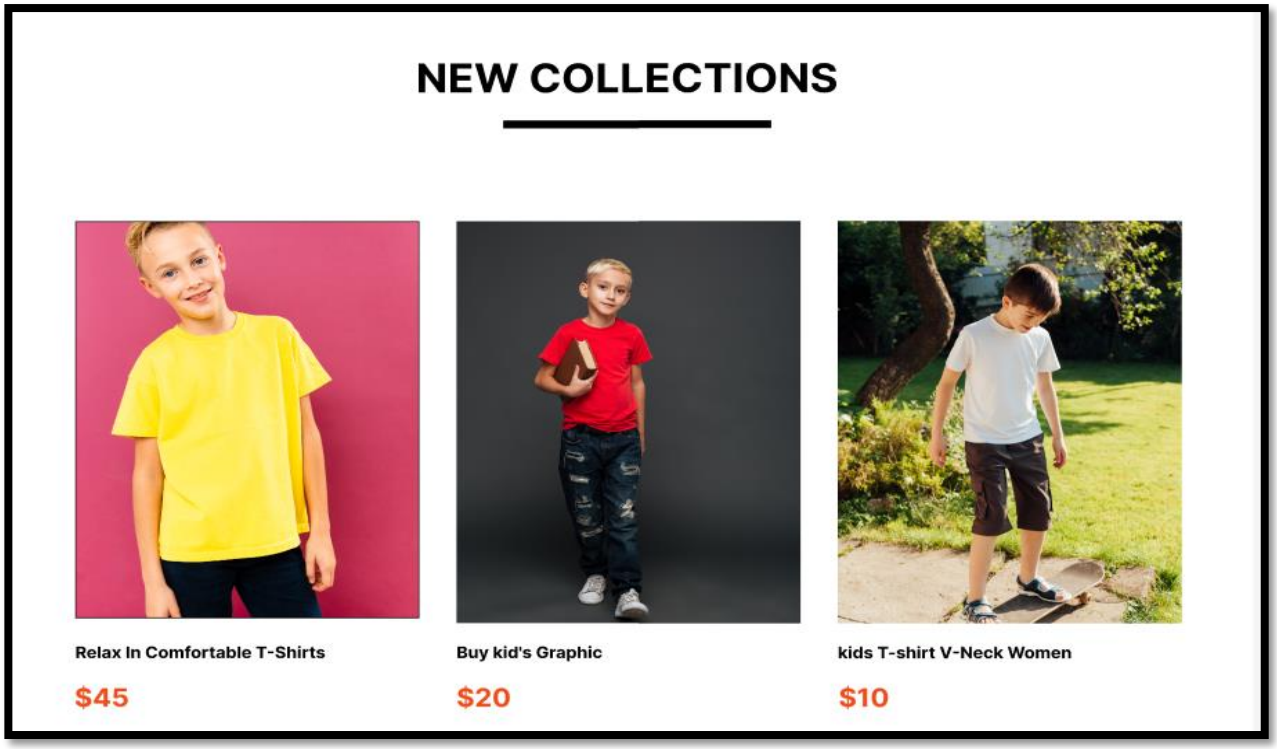

Figure 6.4: Hero section of T-shirt

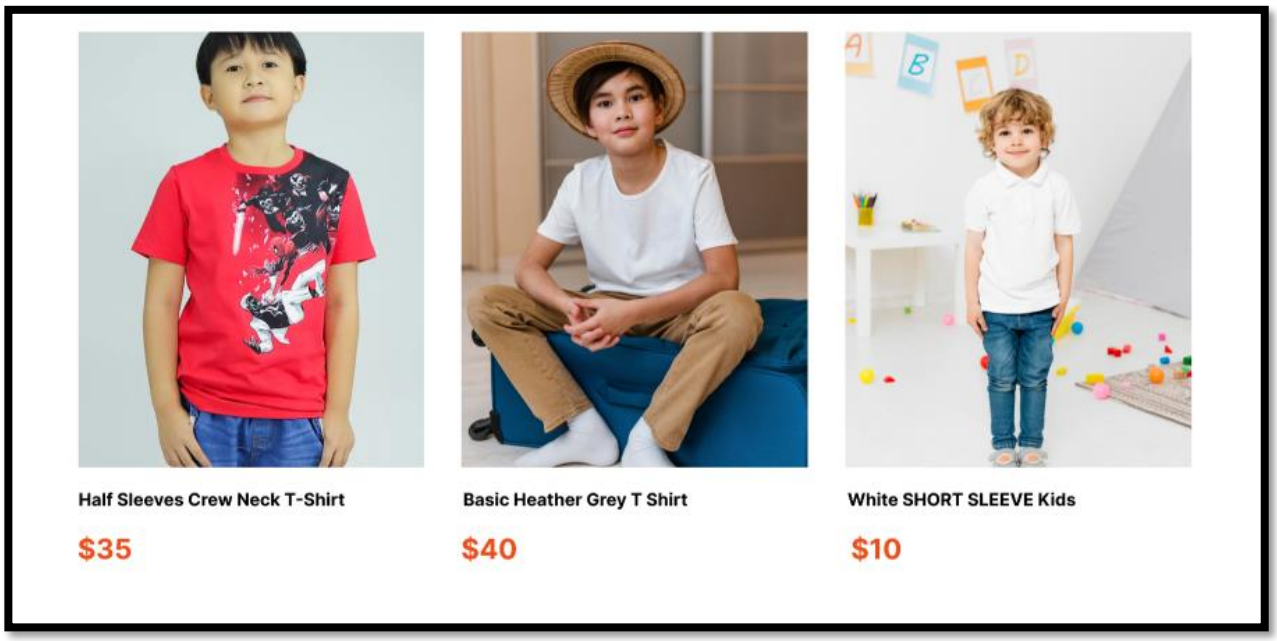

Figure 6.5 Hero section of T-shirt

Figure 6.6 Hero section of outfitter customization

6.3.1.2 Footer:

The bottom section of the home page typically contains the website's footer, which includes additional navigation links, contact information, terms of services, and copyright information. It provides essential details and ensures easy access to social media platform as well as newsletter having email functionality where user directly approach to us.

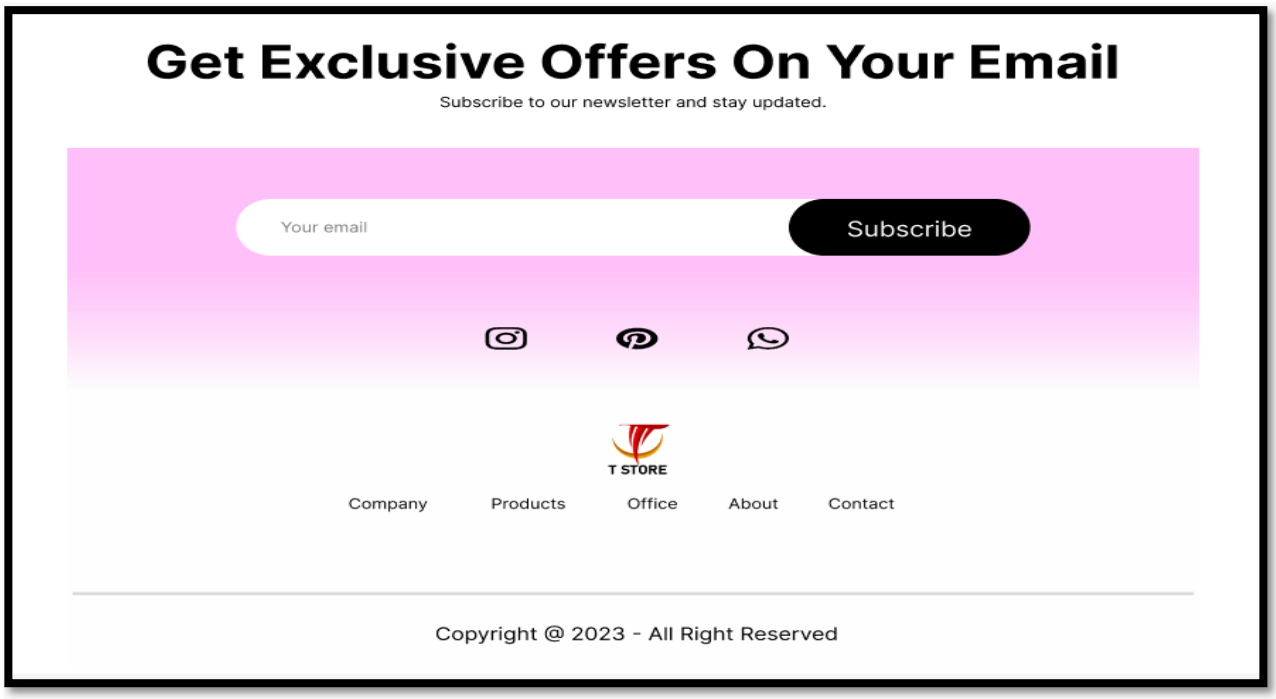

Figure 6.8-Footer of outfitter customization

6.3.2 Services:

Figure 6.9 Services of T-shirt

Description:

- In service page having slider at the top of this section to define customized services where customer directly move to new Brand page.
- Additionally, there are check now buttons on this page. Clicking on these buttons allows to go to the specific page where they can place an order otheir selections.

6.3.3 Shirt details page:

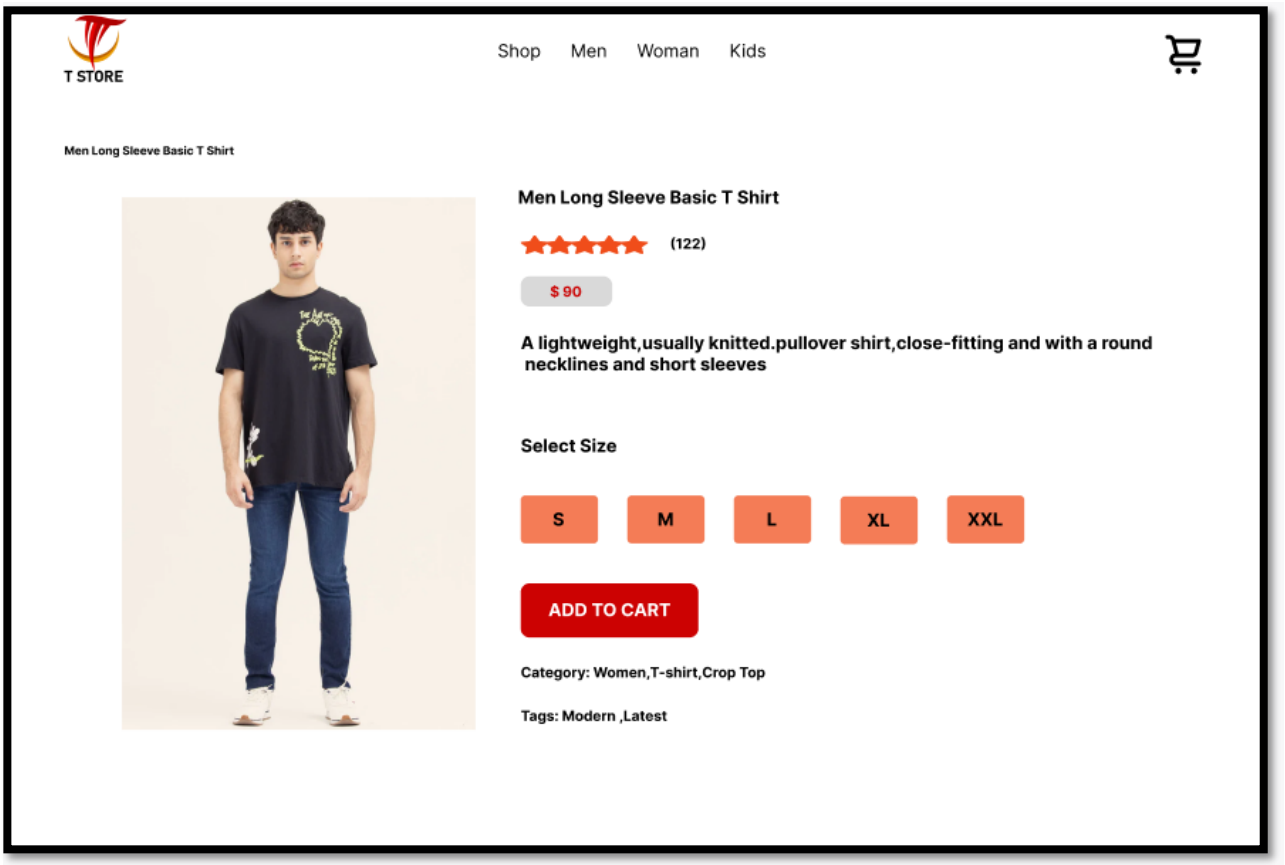

Figure 6.9 Shirt detail page of T-shirt

Description:

- Displaying the T Store logo on the top left.
- Navigation bar includes links to Shop, Men, Women, and Kids categories.
- Cart icon on the top right with a notification showing the number of items in the cart.
- The user browses through the available shirts on any Category Page
- Upon selecting a shirt, the user clicks on it to view more details.
- The user is redirected to a detailed page for the selected shirt.
- The page displays the shirt's size options, price, and description.
- The user can view the available sizes and choose a different size if needed.
- Once the size is selected, the user clicks on the "Add to Cart" button.

6.3.6 Order page

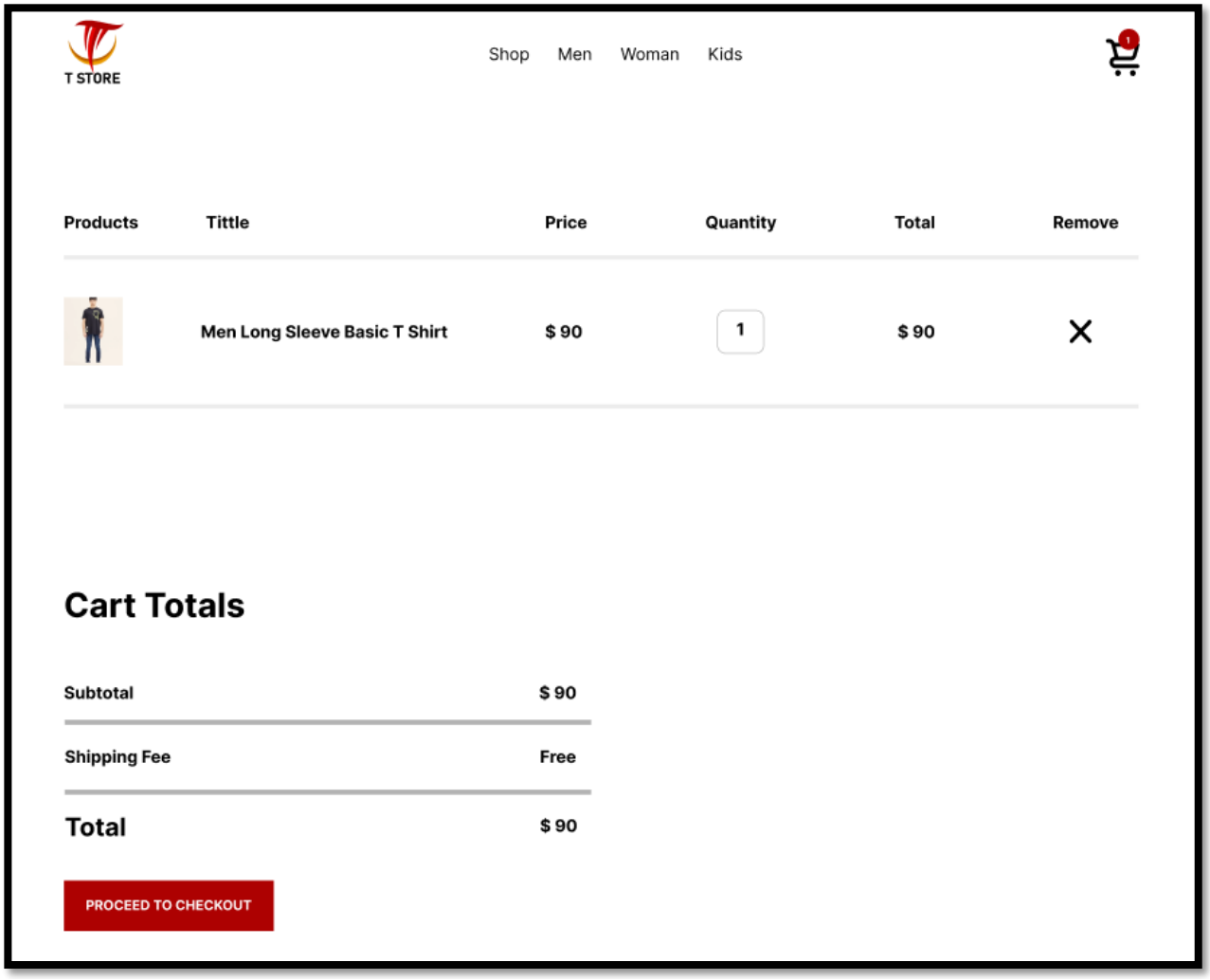

Figure 6.18 Order page of T-shirt

Descrition:

- After clicking the "Add to Cart" button, the user is directed to the Cart Page.
- The page displays the T-shirt details (product image, title, price) along with a quantity selector to adjust the number of items.
- The total price updates dynamically based on the selected quantity.
- The user can remove items from the cart by clicking the "X" button next to the product.
- The Cart Totals section shows the subtotal, shipping fee (free), and the final total amount.
- Once satisfied, the user clicks "Proceed to Checkout" to complete the purchase.

Chapter-7

Tools and Technologies

7.1 Programming Languages

The programming languages used in this application are:

- **JavaScript(React.js):**

  React.js is a JavaScript library used to build the user interface (UI) for web applications. It is particularly suited for single-page applications where dynamic data needs to be reflected on the frontend without reloading the page. React.js offers a component-based structure, enabling reusability and efficient UI rendering.

- **Node.js(Express.js):**

  Node.js is a runtime environment for executing JavaScript on the server side. Express.js is a minimal and flexible Node.js web application framework that provides a robust set of features for building web applications, including routing, middleware, and HTTP utilities, making it ideal for backend development.

- **MongoDB:**

  MongoDB is a NoSQL database that stores data in flexible, JSON-like documents. It is highly scalable and allows for rapid development, making it perfect for e-commerce web applications where the product catalog and user data can change dynamically.

7.2 Operating Environment

The operating environment for our T-shirt e-commerce web application is:

- **WebBrowsers**:

  The application runs in modern web browsers (such as Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari), ensuring cross-platform compatibility. As a web app, users can access it from any desktop or mobile device with an internet connection and a browser.

- **Node.jsServer:**

  The backend runs on a Node.js environment, which can be deployed on cloud platforms like AWS, Heroku, or any server capable of running Node.js applications.

- **MongoDBDatabase**:

  MongoDB is used to store user data, product information, orders, and transactions. It operates on cloud-based environments such as MongoDB Atlas or local MongoDB servers.

APPENDIX A:

USER DOCUMENTATION

User Documentation for Store Owner

Introduction

- Welcome to the T-shirt e-commerce web app user documentation for store owners.
- This guide will help you manage your online store, track orders, and manage products efficiently.

Getting Started

- Open the T-shirt e-commerce web app and sign up for a new account or log in if you already have one.
- Provide your store details, including store name, contact information, and address.
- Set up your product categories (Men, Women, Kids) and upload T-shirt details like images, sizes, descriptions, and prices.

Product Management

- Add, edit, or remove T-shirt products from your store.
- Include detailed descriptions, sizes, colors, and prices for each product.
- Manage stock levels and product availability based on demand.

Order Management

- Receive and process incoming orders from customers in real-time.
- View order details, including product, quantity, and customer information.
- Update the order status (e.g., processing, shipped, delivered) and handle order returns or cancellations if necessary.

Managing Offers and Discounts

- Create special offers or discounts on selected products to boost sales.
- Specify terms and conditions for promotional offers and track their performance.

User Documentation for Customers

Introduction

- Welcome to the T-shirt e-commerce web app user documentation for customers.
- This guide will help you understand how to browse, select, and purchase T-shirts through the app.

Getting Started

- Open the T-shirt e-commerce web app and sign up for a new account or log in if you already have one.
- Enter your personal details, including your name, shipping address, and contact number.

Browsing Products and Categories

- Explore different product categories (Men, Women, Kids) on the app.
- Browse through the available T-shirts and select the desired items by adding them to your cart.
- View detailed product information, including sizes, colors, and prices.

Managing the Shopping Cart

- After adding items to the cart, review the product details and adjust the quantity if needed.
- Remove items from the cart if you change your mind.
- Once satisfied with your selection, proceed to checkout to complete the purchase.

Order Tracking and Updates

- After placing your order, you can track its status in real-time through the app.
- Receive notifications on order processing, shipping, and delivery.

Sharing on Social Media

- Easily share your favorite T-shirts, along with product details and prices, to your social media accounts directly from the app.
- Your followers can click on the shared link to visit the website and explore more products.

User Documentation for Delivery Personnel

Introduction

- Welcome to the T-shirt e-commerce web app user documentation for delivery personnel.
- This guide will help you understand how to use the app to efficiently deliver customer orders.

Getting Started

- Open the T-shirt e-commerce web app and sign up for a delivery personnel account or log in if you already have one.
- Provide your personal details, including name, contact information, and identification documents.

Accepting Orders

- Receive notifications for new delivery orders in real-time.
- Review order details, including the pickup location and customer delivery address.

Pickup and Delivery Process

- Arrive at the designated store or warehouse to pick up the ordered items.
- Verify the order details and ensure the correct T-shirts are packed.
- Use the app's map integration to navigate to the customer’s delivery address.

Delivery Confirmation

- Upon reaching the customer’s address, confirm your arrival through the app.
- Hand over the order to the customer, ensuring it matches the details mentioned in the app.
- Collect any payment due, if applicable, based on the app’s instructions.

Order Tracking and Updates

- Keep the app updated with your progress during the delivery process.
- Update the order status as "picked up," "en route," and "delivered" in real-time.

APPENDIX B:

References

1. Shopify is a widely used e-commerce platform that allows businesses to set up online stores to sell products. It supports product management, inventory tracking, and various payment methods, and is available on both Android and iOS platforms.
2. WooCommerce is an open-source e-commerce plugin for WordPress. It provides complete e-commerce functionalities such as product listings, payment gateways, and cart management, making it a popular choice for small to medium-sized online businesses.
3. BigCommerce is a scalable e-commerce solution that enables businesses to build online stores and manage product sales, payments, and customer orders. The platform supports integration with various payment gateways and shipping services.
4. Stripe is a widely used online payment gateway that supports secure payment processing for e-commerce businesses. It provides APIs to integrate payment functionalities into web applications, including handling credit card transactions.
5. **P**ayPal is one of the most popular payment gateways worldwide. It allows users to make online payments, manage transactions, and track orders. It is integrated with many e-commerce platforms to provide seamless checkout experiences.
6. Magento is an open-source e-commerce platform that allows businesses to create customizable online stores. It includes features for product management, payment integration, and order tracking, supporting both small and large businesses.
7. React is a JavaScript library for building user interfaces, widely used in modern web development, including e-commerce platforms. React allows for fast rendering and dynamic content updates, improving user experience.
8. The MERN stack (MongoDB, Express, React, Node.js) is a popular full-stack development approach used to build dynamic, high-performance web applications, including e-commerce websites. Each part of the stack serves a specific role in the development of modern applications.