

A Project Report On
“E-commerce Fashion Website”

Submitted for partial fulfillment award of the requirements for the award of the Degree of

BACHELOR OF ENGINEERING
IN
INFORMATION TECHNOLOGY
BY

SYED IMAD ALI (160320737015)
MOHD ABDUL KHADER (160320737019)
ABDUL JABBAR (160320737022)

Under the guidance of

MS. RESHMA BEGUM

Assistant Professor

Department of IT- DCET, HYD



Department of Information Technology
Deccan College of Engineering and Technology
(Affiliated to Osmania University)
Hyderabad



DECCAN COLLEGE OF ENGINEERING AND TECHNOLOGY

(AFFILIATED TO OSMANIA UNIVERSITY)

Estd. by DAR-US-SALAM EDUCATIONAL TRUST

Dar-us-salaam, Near Nampally, Hyderabad -500 001.

CERTIFICATE

This is to certify that the project work entitled “E-commerce Fashion Website” is a Bonafide work carried out by **Syed Imad Ali (160320737015), Mohd Abdul Khader (160320737019), Abdul Jabbar (160320737022)** in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF ENGINEERING IN INFORMATION TECHNOLOGY** by the **OSMANIA UNIVERSITY**, Hyderabad, under our guidance and supervision.

The results embodied in this report have not been submitted to any other university or institute for the award of any degree or diploma.

Internal Guide

Ms. Reshma Begum

Assistant Professor
Department of IT-DCET

Co-Ordinator

Ms. Shaheen Begum

Assistant Professor
Department of IT-DCET

External Examiner

DECLARATION

This is a declaration that the work completed by us under the direction of MS. Reshma Begum, Department of Information Technology, Deccan College of Engineering and Technology, Osmania University, is documented in the current project, "E-commerce Fashion Website", which is titled. The project report work was completed by us, and no information from any other source was used to create the reports.

We officially confirm that we are the Fashion website participants. We have helped the project's development by offering concepts, recommendations, and fixes for issues that arose throughout its execution.

Except where we have properly recognized the information's source, we certify that the work submitted in this project report is our own.

We thank the other members for their contributions and affirm that we have also made a variety of contributions to the project. Any inaccuracies or omissions in this report are entirely our faults, and we stand ready to make any necessary changes.

We declare that, throughout the course of the project's development, we abided by the moral standards of honesty and academic integrity. We have adhered to all pertinent policies and rules and have not violated any in the way of academic misconduct, plagiarism, or cheating. We are conscious of the possibility of disciplinary action for any type of academic misconduct, which may include failing the unit, module, or course.

SYED IMAD ALI (160320737015)

MOHD ABDUL KHADER (160320737019)

ABDUL JABBAR (160320737022)

ACKNOWLEDGEMENTS

We would like to sincerely thank and be indebted to our project manager, Ms. Reshma Begum, for her insightful comments and keen interest in this project.

The department head's great infrastructure and welcoming environment helped us finish our job effectively, and we are grateful for that.

Finally, we'd like to use this chance to express our gratitude to our family for helping us with the task. I appreciate and thank everyone who contributed, directly or indirectly, to the success of this project.

SYED IMAD ALI (160320737015)

MOHD ABDUL KHADER (160320737019)

ABDUL JABBAR (160320737022)

LIST OF ABBREVIATIONS

Abbreviation	Full form
HTML	Hyper Text Markup Language
SQL	Structured Query Language
UI	User Interface
SSL	Secure Socket Layer
SMTP	Simple Mail Transfer Protocol
UML	Unified Modeling Language
CSS	Cascade Style sheets
API	Application Programming Interface
GUI	Graphical User Interface

LIST OF FIGURES

Fig no	Name of Figure	Page no
3.5.1.1	Software Development Life Cycle	11
4.2	Flow Chart	15
4.4.1.1	Types of UML Diagrams	18
4.4.1.2	Use case Diagram	21
4.4.1.3	Activity Diagram 1	23
4.4.1.4	Activity Diagram admin	24
4.4.1.6	Sequential Diagram	26
4.4.1.7	DFD Level 0	27
4.4.1.8	DFD Level 1	28
4.4.1.9	DFD Level 2	29
4.4.1.10	ER Diagram	30

ABSTRACT

This project aims to design and develop an E-commerce website for a clothing brand with all the necessary features which enhances customer's experience and engagement. The website will provide a user-friendly interface for customers to browse through the store, add items to the cart, search items by categories, purchase them with ease, pay through Secured integrated payment gateways and much more.

This project also includes the admin dashboard which will help the admin to keep track of the progress of the company, manipulate the data items efficiently, and to add & remove the data items depending on the stock available, without writing any code.

We aim to serve the technology with creativity and a touch of aesthetics. The project will involve developing a database for the purchase history of the items, current delivery status etc.

This project is expected to showcase the brand on the internet to gain the customer engagement and increase its revenue. This project is used to make life easy for the customer, to buy their essentials from a large variety of area in a simplest way and to maximize the sales of the brand owner.

The project is designed with the help of Figma and developed using Custom-coding technology of HTML, CSS and JavaScript as the front-end, PHP, MYSQL as the backend and some additional tools which enhances the quality of the project at both sides are used.

The main coding language used is HTML and JavaScript.

This Website is used to browse and purchase clothing items from anywhere in the world by a single dynamic website which will help the user to know all about the brand in a single website.

TABLE OF CONTENT

CERTIFICATE	ii
DECLARATION	iii
ACKNOWLEDGEMENTS.....	iv
LIST OF ABBREVIATIONS.....	v
LIST OF FIGURES	vi
ABSTRACT	vii
CHAPTER I.....	1
INTRODUCTION	1
Introduction	1
Study of the Problem.....	2
Scope.....	3
Objective	4
Infrastructure	5
CHAPTER II	6
LITERATURE SURVEY.....	6
CHAPTER III.....	7
SYSTEM ANALYSIS	7
Objective	7
Problem Specifications.....	8
Proposed System.....	9
Applications.....	9
Modules and their Functionalities	10
Software and Hardware Requirements.....	12
CHAPTER IV	13
DESIGN.....	13
Architecture	13
Flow chart.....	15
Algorithm	16
UML Diagram	17
CHAPTER V.....	31
IMPLEMENTATION	31

Partial Code	31
Screenshots	56
Chapter VI	62
Conclusion	62
Chapter VII	63
Reference	63

CHAPTER I

INTRODUCTION

Introduction

Welcome to Quickstyle, your ultimate fashion destination! Quickstyle is a user-friendly online fashion store designed to cater to all your style needs.

With a range of functionalities, Quickstyle aims to provide you with a seamless user experience, easy shopping, and secure account verification.

Finding the perfect fashion pieces can be overwhelming, but with Quickstyle, you can leave all your fashion worries behind. Our website is built using HTML, CSS, Bootstrap, My SQLite, and JavaScript, ensuring a robust and efficient platform. Whether you're a fashion enthusiast or exploring new styles, Quickstyle is here to guide you every step of the way.

We understand the importance of a user-friendly interface when it comes to fashion discovery. That's why Quickstyle offers an intuitive and well-designed platform that makes it easy for you to explore and find various fashion options.

Whether you're searching for clothing, accessories, or footwear, Quickstyle provides a seamless shopping process that allows you to update your wardrobe with just a few clicks. Our goal is to make your fashion journey as simple and enjoyable as possible.

In today's digital world, security is paramount. That's why Quickstyle incorporates email verification to ensure the authenticity of user accounts. By implementing this feature, we provide an additional layer of protection and maintain a trustworthy environment for all our users. Your security and peace of mind are our top priorities.

Study of the Problem

In the fast-paced world of today, shoppers encounter a multitude of challenges and frustrations while navigating the realm of fashion e-commerce. The inundation of options, intricate checkout procedures, and absence of personalized guidance often contribute to bewilderment and squandered time. Conventional online shopping platforms may fall short in providing the round-the-clock assistance and user-centric experiences that modern buyers anticipate. Acknowledging these obstacles, the Quickstyle Fashion Store was established to tackle these pain points and introduce a comprehensive solution.

Choice Overload: With an abundance of fashion websites, style influencers, and social media forums, shoppers often find themselves inundated with a surplus of choices. Sifting through countless selections of clothing, accessories, footwear, and trends can prove time-consuming and perplexing.

Complex Checkout Procedures: Completing a purchase typically involves a series of intricate steps, including product searches, price comparisons, item selections, and payment arrangements. Each stage may necessitate navigation through various interfaces, completion of lengthy forms, and engagement in payment processes. This complexity can elicit frustration and consume substantial time for shoppers.

Absence of Personalized Guidance: Many shoppers seek tailored assistance and expert advice while exploring fashion options. Conventional online stores, although offering support, often operate within specific hours, posing a challenge for shoppers to access guidance outside of regular business times. This lack of round-the-clock assistance can lead to decision-making delays and hinder the overall shopping experience.

Security Apprehensions: In the digital era, consumers are growingly concerned about the security and legitimacy of their purchases and personal information. Instances of online fraud and data breaches have underscored the necessity for secure and reliable platforms when engaging in fashion e-commerce.

Recognizing these issues, Quickstyle Fashion Store strives to offer:
an intuitive and secure shopping environment, addressing these concerns and ensuring a seamless and enjoyable fashion discovery and purchase journey

1. Comprehensive and Curated Product Information: Quickstyle Fashion Store diligently compiles relevant and current product details from various sources, presenting them in an organized and consolidated manner. Shoppers can effortlessly access comprehensive information about clothing, accessories, footwear, and trends on a single platform, streamlining the shopping process and alleviating the challenge of overwhelming options.

2. Simplified Shopping Experience: Quickstyle simplifies the shopping experience by amalgamating multiple fashion services into a unified platform. Shoppers can explore, compare, and purchase clothing, accessories, and footwear seamlessly, eliminating the need to navigate various websites. The platform also offers secure and convenient payment methods, reducing the intricacies associated with making fashion acquisitions.

3. Round-the-Clock Support via Interactive Assistance: The incorporation of Quickstyle's interactive support feature ensures that shoppers have constant access to assistance. This technology, akin to Quickstyle's fashion assistant, can comprehend and address shopper inquiries, offering real-time insights and tailored suggestions. This innovation diminishes the inconvenience of awaiting assistance and heightens the overall shopping experience.

4. Enhanced Security Measures: To assuage security apprehensions, Quickstyle implements rigorous

account verification processes. By validating shopper accounts through stringent measures, such as email authentication, the platform ensures the credibility of users and fosters a secure environment for transactions and data privacy.

Recognizing these concerns, Quickstyle Fashion Store fervently endeavors to provide an intuitive and secure shopping environment, effectively tackling these challenges and guaranteeing a seamless and gratifying fashion exploration and purchase voyage. Quickstyle aspires to empower shoppers, making their fashion journey more efficient and enjoyable.

Scope

The scope of the QuickStyle Fashion Store encompasses a diverse range of aspects related to fashion exploration, shopping, and assistance. The core elements and features within this scope include:

1.Fashion Inspiration and Information: The website offers a comprehensive array of fashion information, including trending styles, outfit ideas, fashion trends, and styling tips. Users can explore a variety of fashion inspirations and gather essential details to make well-informed style choices.

2.Personalized Shopping and Exploration: QuickStyle enables users to personalize their fashion journey by providing integrated search functionalities for clothing, accessories, and footwear. Users can search, compare options, and make purchases seamlessly within the platform. The website aims to simplify the entire shopping process, from browsing styles to completing purchases.

3.User Profile and Preferences: Users can create individual profiles on the QuickStyle website, allowing them to save their preferences, track purchases, and access tailored recommendations. User profile management includes features such as style customization, password control, and access to purchase history.

4.QuickStyle Fashion Assistant: The QuickStyle fashion assistant, leveraging advanced technology, is available around the clock to provide guidance and address user inquiries. The assistant can offer insights on styles, fashion suggestions, product details, and general fashion-related queries. It functions as a virtual style advisor, ensuring continuous support throughout the fashion discovery and shopping experience.

QuickStyle aspires to empower users in their fashion journey, making style exploration and shopping more efficient and enjoyable.

1. **Email Verification:** The website incorporates email verification as part of the account registration process. This ensures the security and authenticity of user accounts, creating a trustworthy environment for transactions and interactions.
2. **Responsive User Interface:** The QuickStyle website is designed to be user-friendly and accessible across various devices, including desktops, laptops, tablets, and mobile phones. The responsive interface ensures a seamless and consistent experience for users regardless of the device they are using.

The scope may expand over time as the website evolves and incorporates additional features and services to enhance the user experience. Continuous updates and improvements will be made to ensure the QuickStyle website remains a reliable and innovative platform for fashion heads worldwide.

Objective

This project aims to design and develop an E-commerce website for a clothing brand with all the necessary features which enhances customer's experience and engagement. The website will provide a user-friendly interface for customers to browse through the store, add items to the cart, search items by categories, purchase them with ease, pay through Secured integrated payment gateways and much more.

This project also includes the admin dashboard which will help the admin to keep track of the progress of the company, manipulate the data items efficiently, and to add & remove the data items depending on the stock available, without writing any code.

We aim to serve the technology with creativity and a touch of aesthetics. The project will involve developing a database for the purchase history of the items, current delivery status etc.

This project is expected to showcase the brand on the internet to gain the customer engagement and increase its revenue. This project is used to make life easy for the customer, to buy their essentials from a large variety of area in a simplest way and to maximize the sales of the brand owner.

The project is designed with the help of Figma and developed using Custom-coding technology of HTML, CSS and JavaScript as the front-end, PHP, MYSQL as the backend and some additional tools which enhances the quality of the project at both sides are used.

The main coding language used is HTML and JavaScript.

This Website is used to browse and purchase clothing items from anywhere in the world by a single dynamic website which will help the user to know all about the brand in a single website

Infrastructure

The infrastructure required to support a Fashion E-Commerce Website project encompasses the hardware, software, and resources essential for its functionality. This includes hardware components like computers, servers, and storage units that contribute to the system's operation.

The foundation of the website is built upon a robust infrastructure, incorporating technologies such as HTML, CSS, Bootstrap, and My SQLite. This forms the core of the development process, ensuring the creation of a secure and scalable online platform. PHP serves as the primary backend driving the backend functionalities that power the website's operations.

The user interface is thoughtfully designed using HTML, CSS, and Bootstrap, resulting in an engaging and responsive layout that seamlessly adapts to different devices. This strategic approach aims to enhance the overall user experience and facilitate easy navigation and interaction.

The integration of My SQLite as the database management system allows for efficient storage and retrieval of user-related data, product details, and transaction information. JavaScript is utilized to introduce interactivity and dynamic features, enhancing the website's overall functionality and user engagement.

In summary, the QuickStyle Fashion E-Commerce Website leverages a solid infrastructure and advanced technologies to provide users with a user-friendly interface, dynamic features, and seamless shopping capabilities. By delivering a smooth and enjoyable shopping experience, QuickStyle aims to redefine the way users explore and engage with fashion in the digital realm.

CHAPTER II

LITERATURE SURVEY

User-Centric Interface Design: Extensive research underscores the significance of user-centric interface design in fashion websites. Scholarly work emphasizes that intuitive and aesthetically pleasing interfaces positively influence user engagement and satisfaction. Elements such as seamless navigation, visually appealing imagery, and efficient information arrangement have been identified as pivotal components in the success of fashion website interfaces.

Personalization and Style Recommendations: Personalization and style recommendation systems are pivotal in elevating user experiences within fashion websites. By harnessing user data, preferences, and browsing history, these systems can propose tailored fashion choices, trends, and outfit inspirations. Studies underscore the potential of machine learning algorithms and collaborative filtering techniques in enhancing recommendations and delivering individualized style journeys.

User-Generated Reviews and Style Ratings: User-generated reviews and ratings have become indispensable factors in fashion decision-making. Scholarly insights indicate that the integration of user reviews and ratings into fashion websites cultivates user trust and confidence. Furthermore, sentiment analysis methods can be harnessed to extract valuable insights from user feedback, facilitating informed fashion selections.

Mobile Responsiveness and Cross-Device Accessibility: Given the ubiquity of mobile devices, ensuring mobile responsiveness and cross-device accessibility is paramount. Research affirms that responsive design principles, mobile-friendly interfaces, and optimized performance significantly contribute to user contentment and interaction. The adoption of mobile applications and responsive web design practices is recommended to ensure seamless experiences across diverse devices.

CHAPTER III

SYSTEM ANALYSIS

Objective

In the context of the QuickStyle Fashion E-Commerce Website, the goal of system analysis is to comprehensively comprehend and assess the existing processes, recognize strengths and weaknesses, and outline the requisites and aspirations for the new system.

1. **Understanding the Current System:** The primary objective is to gain a thorough understanding of the current fashion e-commerce system. This involves scrutinizing the workflows, data flows, and interactions between various stakeholders such as shoppers, fashion experts, and administrators. By dissecting the current system, areas that demand enhancement or streamlining can be pinpointed.
2. **Identifying Shopper Requirements:** System analysis seeks to identify and document the distinct requirements and anticipations of shoppers and stakeholders. This entails conducting interviews, surveys, and obtaining user feedback to gather insights into their needs, style preferences, and pain points. The aim is to comprehensively understand what functionalities and features the new system should incorporate to align with users' expectations.
3. **Defining System Goals and Aims:** System analysis aids in defining the goals and objectives that the new fashion e-commerce system should attain. This encompasses establishing desired outcomes such as an enriched shopping experience, efficient browsing and selection, secure transactions, and personalized style recommendations. Clear-cut goals and objectives provide a framework for the formulation and execution of the new system.
4. **Analyzing System Constraints and Boundaries:** It is imperative to identify any constraints or boundaries that could impact the design and rollout of the new system. These constraints may encompass budgetary restrictions, time limitations, technical thresholds, or legal and regulatory prerequisites. Analyzing these factors facilitates the establishment of pragmatic expectations and informed decision-making during the system design phase.
5. **Developing System Requirements:** The analysis phase strives to compile and document system requirements derived from the examination of the present system, shopper requisites, and defined goals and objectives. These requirements outline the new system's objectives, encompassing functional requirements (like features, interfaces, and workflows) and non-functional requirements (such as performance, security, and scalability).

In conclusion, the objective of system analysis within the context of the QuickStyle Fashion E-Commerce Website is to establish a robust basis for the creation and implementation of the fashion platform, catering to the unique needs and expectations of modern shoppers.

Problem Specifications

In the realm of fashion e-commerce, problem specifications within system analysis entail identifying and documenting distinct challenges, limitations, and issues inherent to the current system, necessitating resolution in the new system. These problem specifications furnish a lucid comprehension of the obstacles to be overcome and serve as the foundation for delineating the requisites of the novel system. Some prevalent problem specifications in the context of the QuickStyle Fashion E-Commerce Website could comprise:

1. **Complex and Tedious Shopping Experience:** Recognize intricate and time-consuming aspects of the existing shopping process. This may encompass convoluted navigation, redundant actions, and cumbersome steps. The problem specification could underscore the demand for a streamlined and user-centric shopping journey.
2. **Absence of Real-time Assistance:** Identify the inadequacies of the prevailing customer support framework, such as restricted service hours or delayed responsiveness. Specify the necessity for a real-time assistance feature, ensuring immediate user support and insights through intelligent chatbot integration.
3. **Overwhelming Choices and Decision-Making Challenges:** Document user struggles in sifting through an extensive array of fashion options. Specify the requirement for enhanced information presentation, personalized curation, and refined recommendations to facilitate informed and confident decision-making.
4. **Security and Privacy Apprehensions:** Emphasize potential security and privacy vulnerabilities inherent to the current system, including insecure transaction processes or inadequate data safeguarding. Specify the need for fortified security measures, secure payment gateways, and robust data protection mechanisms.
5. **Inconsistent User Experience:** Identify any incongruities or usability drawbacks within the present system leading to suboptimal user experiences. Specify the necessity for a coherent and user-centric interface design, adaptable layouts for various devices, and streamlined user interactions.
6. **Limited Personalization and Customization:** Recognize existing restrictions in user personalization and customization options. Specify the requirement for features enabling users to personalize preferences, track selections, and obtain tailored recommendations based on their style inclinations and shopping history.

In summation, within the QuickStyle Fashion E-Commerce Website, problem specifications serve as a roadmap for overcoming challenges and shaping a refined and enhanced fashion exploration and shopping experience for users.

Proposed System

To debug the current system, remove processes that duplicate data, and correct the navigational order. To build strong password mechanisms, information about audits at various levels is provided, along with a reflection of the current work status based on the organization, auditor, or date.

a) Goals:

- To streamline the blood donation and reception process.
- To enhance the current system.
- To create a scalable system.
- To be very accessible.

b) Scope:

Make sure the program is straightforward to use and that it covers all the functions of a Fashion Website.

Applications

The QuickStyle Fashion E-Commerce Website boasts a range of key applications, enhancing the fashion exploration and shopping experience:

1.Intuitive Style Discovery and Selection: The central application of the QuickStyle Fashion E-Commerce Website lies in aiding users to effortlessly discover and select their preferred fashion items. Users can explore a vast array of clothing, accessories, and footwear, compare options, and directly make purchases through the platform. The streamlined checkout process ensures a seamless journey, and the website's integration with responsive design guarantees consistent usability across multiple devices.

2.Enhanced User Experience: QuickStyle prioritizes delivering an enriched user experience by offering intuitive navigation, coherent workflows, and an appealing interface. Users can readily browse through diverse fashion choices, access comprehensive product details, and proceed with purchases with utmost ease. The website's responsiveness ensures a uniform and engaging encounter across various devices, spanning desktops, laptops, tablets, and smartphones.

3.24/7 Fashion Guidance: The QuickStyle Fashion E-Commerce Website introduces a constant fashion companion through an integrated fashion assistant. Users can access continuous assistance facilitated by this interactive feature, employing advanced technology to understand user inquiries and furnish rapid, precise responses. This real-time support empowers users to seek style insights, make informed decisions, and elevate their overall fashion satisfaction.

4.Secure User Authentication: QuickStyle implements robust user authentication practices, heightening security and trust. This application ensures the legitimacy of user accounts, mitigating potential risks of unauthorized access and unauthorized activities. Secure user authentication fosters a dependable platform, cultivating user confidence and sustaining a protected environment.

5.Fashion Insights and Trends: The website presents a wealth of fashion insights, including emerging trends, outfit inspirations, and styling advice. Users can access these valuable resources to remain well-informed about prevailing fashion trends, make astute style choices, and curate personalized looks that resonate with their individual preferences.

In summary, the QuickStyle Fashion E-Commerce Website presents an array of applications that converge to amplify the fashion discovery and shopping encounter, aligning with modern users' style aspirations and requirements.

Modules and their Functionalities

Project Phase and Modules

Software Development Life Cycle

SDLC stands for Software Development Life Cycle. It is a structured and systematic approach to software development that outlines the processes and phases involved in building high-quality software systems. SDLC provides a framework for software development teams to plan, design, develop, test, deploy, and maintain software applications. The primary goal of SDLC is to deliver software that meets the needs and requirements of the stakeholders while ensuring quality, reliability, and efficiency throughout the development process.

The typical phases of the SDLC include:

1. **Requirements Gathering and Analysis:** In this phase, project stakeholders, including clients, users, and development teams, identify and document the functional and non-functional requirements of the software. This involves understanding the problem domain, gathering user needs, and defining system specifications.

2. **System Design:** In this phase, the software architecture and system design are developed based on the requirements. The design may include high-level architectural decisions, data models, user interface design, and software component specifications.
3. **Implementation (Coding):** In this phase, the software is developed based on the design specifications. Coding involves writing the source code using programming languages, implementing algorithms, and integrating various software components.
4. **Testing:** The software is thoroughly tested to identify and fix defects, ensure functionality, and validate the software against the defined requirements. Testing includes unit testing, integration testing, system testing, and acceptance testing.
5. **Deployment:** In this phase, the software is deployed to the target environment, which may include production servers, client machines, or cloud platforms. The deployment process involves installation, configuration, and ensuring the software is ready for end-users.
6. **Maintenance:** After deployment, the software enters the maintenance phase, where updates, bug fixes, and enhancements are made based on user feedback and changing requirements. Maintenance aims to keep the software reliable, secure, and up-to-date.

Throughout the SDLC, various methodologies and approaches can be used, such as Waterfall, Agile, or DevOps, depending on the project requirements and team preferences. These methodologies provide different strategies for managing the development process, collaboration, and iteration.

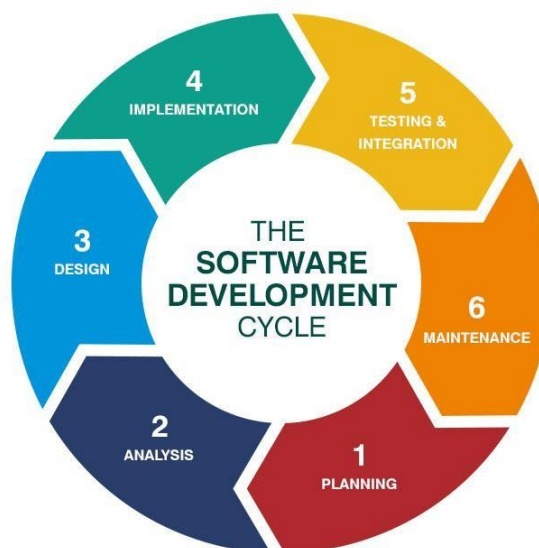


Figure 3.5.1.1

Software and Hardware Requirements

Software Requirements

1. **Operating System:** Windows 7 or above 64-bit
2. **Code Editor:** Visual Studio Code
3. **Prototyping Tools:** FIGMA
4. **Front-end Languages:** HTML 5, Bootstrap 5, CSS,JavaScript.
5. **Back-End Languages:** PHP

Hardware Requirements

1. **RAM:** 4GB
2. **SSD:** 256 GB
3. **Processor:** Intel i3 GEN 11

CHAPTER IV

DESIGN

Architecture

The architecture of the QuickStyle Fashion E-Commerce Website encompasses the following key components:

1. User Interface:

- **Homepage:** Features a curated assortment of trending fashion items, search functionality, and options for user authentication or registration.
- **Login and Registration Pages:** Collect and validate user credentials for secure access.
- **Product Pages:** Present detailed product information, including images, descriptions, and user reviews.
- **Cart and Checkout Pages:** Enable users to add items to their cart, review their selections, and complete secure transactions.
- **Payment Gateway:** Facilitates secure payment processing and confirmation of purchases.
- **Style Advisor Chat:** Provides a chat interface for users to interact with the virtual style advisor.
- **Feedback Portal:** Allows users to share feedback and ratings regarding their shopping experience.
- **Order Confirmation Emails:** Sends automated confirmation emails to users after successful purchases.

2. Backend Components:

- **User Management:** Manages user authentication, registration, and profile management.
- **Product Management:** Stores and retrieves fashion product details from a database.
- **Cart and Order Management:** Tracks and handles user cart items and order processing, including inventory management.
- **Payment Gateway Integration:** Integrates with a trusted payment gateway to ensure secure payment transactions.
- **Style Advisor Integration:** Implements an AI-powered style advisor that understands user preferences, offers fashion recommendations, and assists with queries.
- **Feedback Management:** Collects and stores user feedback, enabling administrators to review and respond.
- **Email Service:** Sends automated order confirmation and updates to users.

3. Data Storage:

- **User Database:** Stores user credentials, contact information, and purchase history.
- **Product Database:** Contains detailed information about fashion items, including images, descriptions, and pricing.
- **Order Database:** Stores order details, shipping information, and transaction records.
- **Feedback Database:** Archives user feedback, ratings, and comments for analysis and improvement.
- **Style Advisor Knowledge Base:** Employs an AI model to comprehend user fashion preferences and provide tailored recommendations.

4. Security Measures:

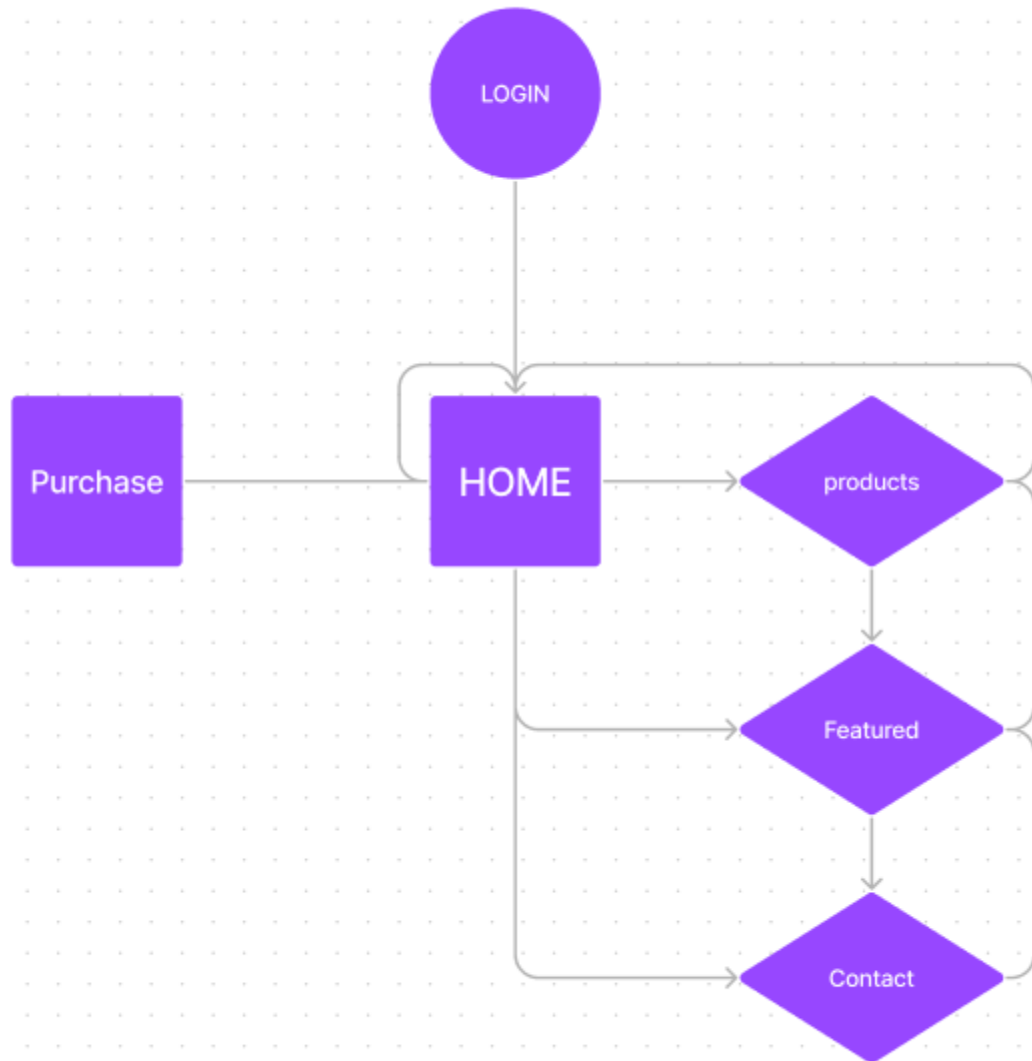
- **Encryption:** Utilizes encryption techniques to safeguard sensitive user data, including personal information and payment details.

- Secure Socket Layer (SSL) Certificate: Implements SSL/TLS encryption to ensure secure data exchange between the website and users.
- Input Validation: Enforces rigorous input validation to thwart potential security vulnerabilities like SQL injection and cross-site scripting attacks.

In essence, the architecture of the QuickStyle Fashion E-Commerce Website ensures a seamless, secure, and personalized fashion exploration and shopping journey for users.

Flow chart

Figure 4.2.1



Algorithm

1. Start
2. Display home page
3. Prompt user to log in or sign up
4. If the user chooses to sign up:
 - Collect user details (name, email, password, etc.)
 - Create a new user account
 - Go to step 6
5. If the user chooses to log in:
 - Prompt user to enter login credentials
 - Verify the credentials
 - If login is successful, go to step 6
 - If login fails, display error message and go back to step 3
6. Display the main page with options to search or browse Products
7. If the user chooses to search for products:
 - Prompt user to enter search criteria
 - Retrieve destinations from the database based on the search criteria
 - Display search results
 - Go to step 9
8. If the user chooses to browse featured products:
 - Retrieve featured products from the database
 - Display featured products
 - Go to step 9
9. If the user wants to purchase an item:
 - Prompt user to select the item
 - Retrieve its details from the database
 - Display product details
 - Prompt user to enter delivery details
 - Collect booking details

- Go to step 10
10. Prompt user to proceed to the payment page
 11. If the user confirms the booking:
 - Collect payment details
 - Process the payment using a payment gateway
 - If payment is successful, go to step 12
 - If payment fails, display an error message and go back to step 10
 12. Update the booking database with the user's booking details
 13. If the user submits feedback:
 - Prompt user to provide feedback (ratings, comments, etc.)
 - Store the feedback in the feedback database
 - Display a confirmation message
 14. Send an email confirmation to the user's provided email address
 15. End

UML Diagram

Unified Modelling Language (UML):

1. Model

- Real life is simplified into a model.
- The schematics for a system are provided by a model.
- A model could be behavioral, emphasizing the dynamics of the system, or structural, emphasizing the way the system is organized.
- We create models in order to better comprehend the system we are creating.
- We create models of complex systems because we are unable to fully understand them. Four objectives are attained through modelling.
- With the aid of models, we can see a system as it is or as we want it to be. We can specify a system's structure or behaviour using models.
- Models provide us with a framework that directs our construction of a system. Models serve as a record of our choices.

2. Principles of Modelling

- The models that are created have a significant impact on how a problem is approached and how a solution is developed.
- Different precision levels can be expressed for every model. The most accurate models are grounded in reality.
- No one model works well enough. The best way to tackle any nontrivial system is with a condensed set of essentially independent models.

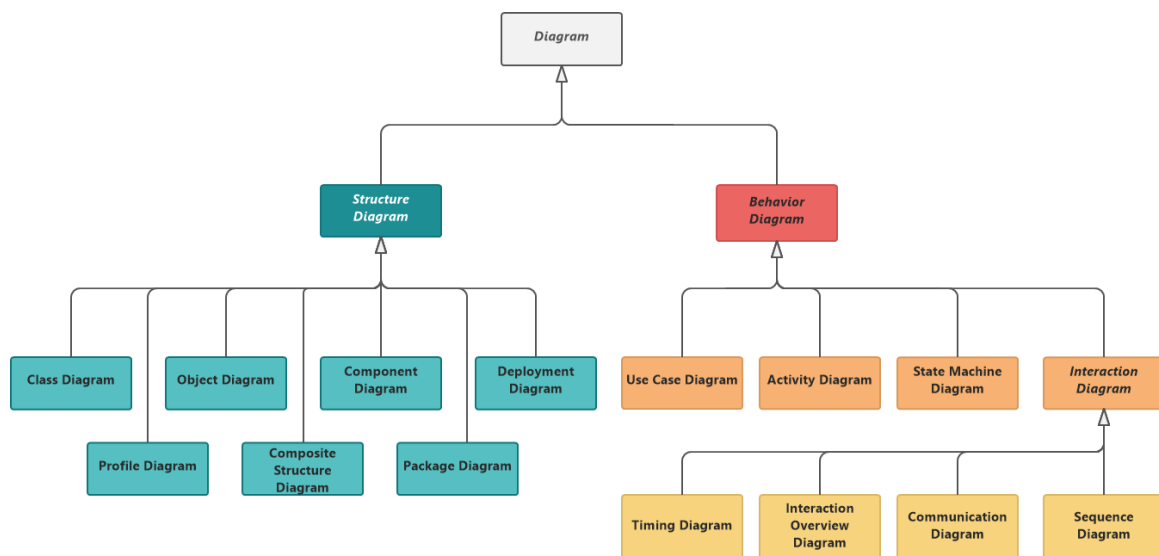


Figure 4.4.1.1

UML is a graphical notation that is used to visualize, define, build, and document software-intensive artefacts. Enterprise information systems, distributed web-based applications, and even hard, real-time embedded systems can all be modelled using UML. In order for UML to work effectively, the language must first be conceptually modelled.

Diagrams come in two varieties: They are

1. Static Diagrams
 - a) Use case diagrams
 - b) Class diagrams
 - c) Object diagrams
 - d) Component diagrams
 - e) Deployment diagrams
2. Dynamic diagrams
 - a) Interaction diagrams
 - b) Sequence diagrams
 - c) Collaboration diagrams
 - d) State machine diagrams
 - e) Activity diagrams

Applications of UML:

Software-intensive systems are the primary target audience for UML. It has been utilised successfully in areas like

1. Enterprise Information Systems
2. Banking and Financial Services
3. Telecommunications
4. Transportation
5. Defence and Aerospace
6. Retail
7. Medical Electronics
8. Scientific
9. Distributed Web-based Services

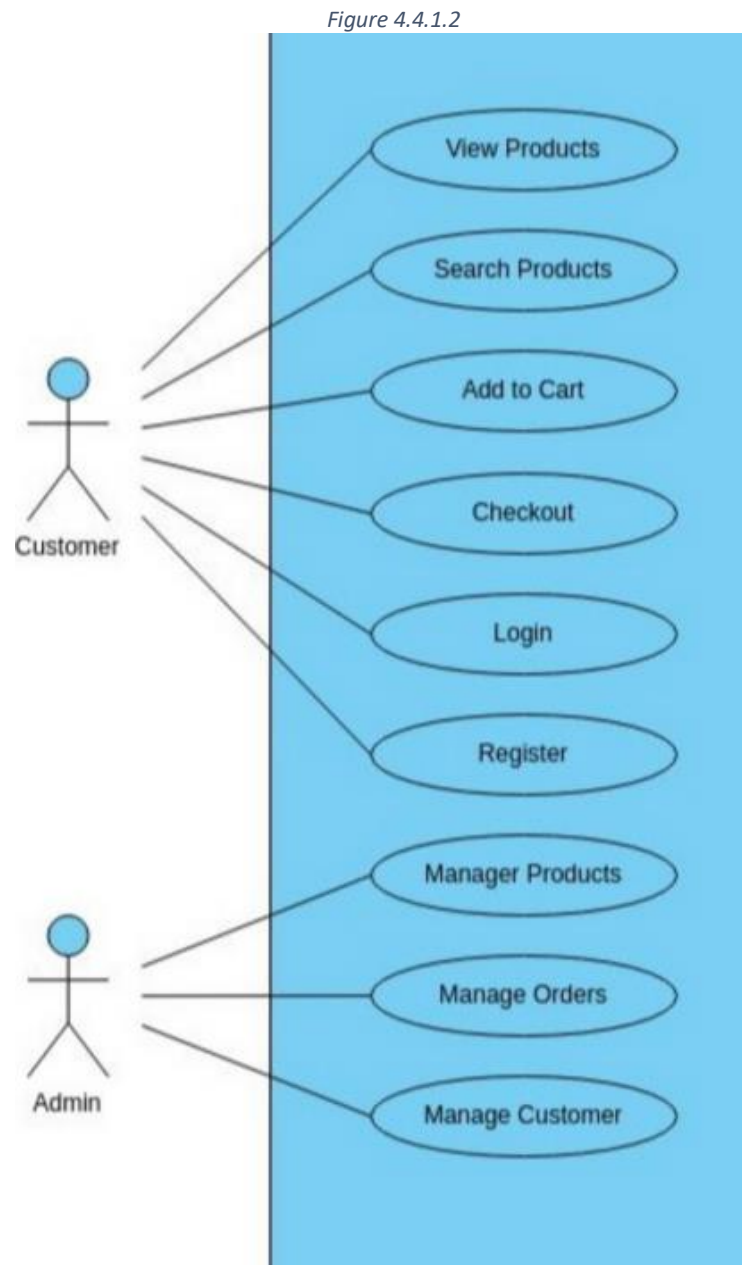
Basic building blocks of UML:

The components of UML can be divided into two categories:

1. Things
2. Relationships and
3. Diagrams
4. Things: - The most crucial UML building block is a thing. Things could
 - a) Structural
 - b) Behavioural
 - c) Grouping
 - d) Annotation

1. Use Case Diagram

A use case diagram is a visual representation of the interactions between actors (users or external systems) and the system under consideration. In the context of a Ecommerce Fashion website here's a description of some potential use cases and the actors involved:



2. Activity Diagram

An activity diagram is a graphical representation used to model the flow of activities or actions within a system or process. It allows you to visualize the sequential and parallel activities, decisions, and the overall workflow involved in a particular process.

Activity diagrams are commonly used in software development and business process modelling to depict the steps and interactions involved in a system's operation. They provide a clear and structured overview of how different activities relate to each other, allowing stakeholders to understand the process flow and identify potential bottlenecks or improvements.

In an activity diagram, activities are represented as nodes, and the flow of control between activities is depicted through arrows or transitions. Decision points, represented as diamonds, determine which path to follow based on certain conditions. Forks and joins indicate parallel or concurrent activities, allowing for synchronization or merging of control flow.

The primary elements of an activity diagram include initial and final nodes, which indicate the start and end of the process respectively. Actions represent individual steps or tasks, and control flows indicate the logical sequence in which activities are performed. Additionally, swim lanes or partitions can be used to depict different actors or organizational units responsible for specific activities.

Activity diagrams can be used to model various scenarios, such as system behaviour, business processes, software workflows, and user interactions. They provide a visual representation that facilitates communication and collaboration between stakeholders, developers, and designers, enabling a shared understanding of the system's operation.

By creating activity diagrams, you can gain insights into the structure, dependencies, and logic of a process, aiding in the identification of potential issues and facilitating the design of efficient and optimized systems. Overall, activity diagrams serve as valuable tools for visualizing and analyzing complex processes, enhancing the clarity and effectiveness of system development and process improvement efforts.

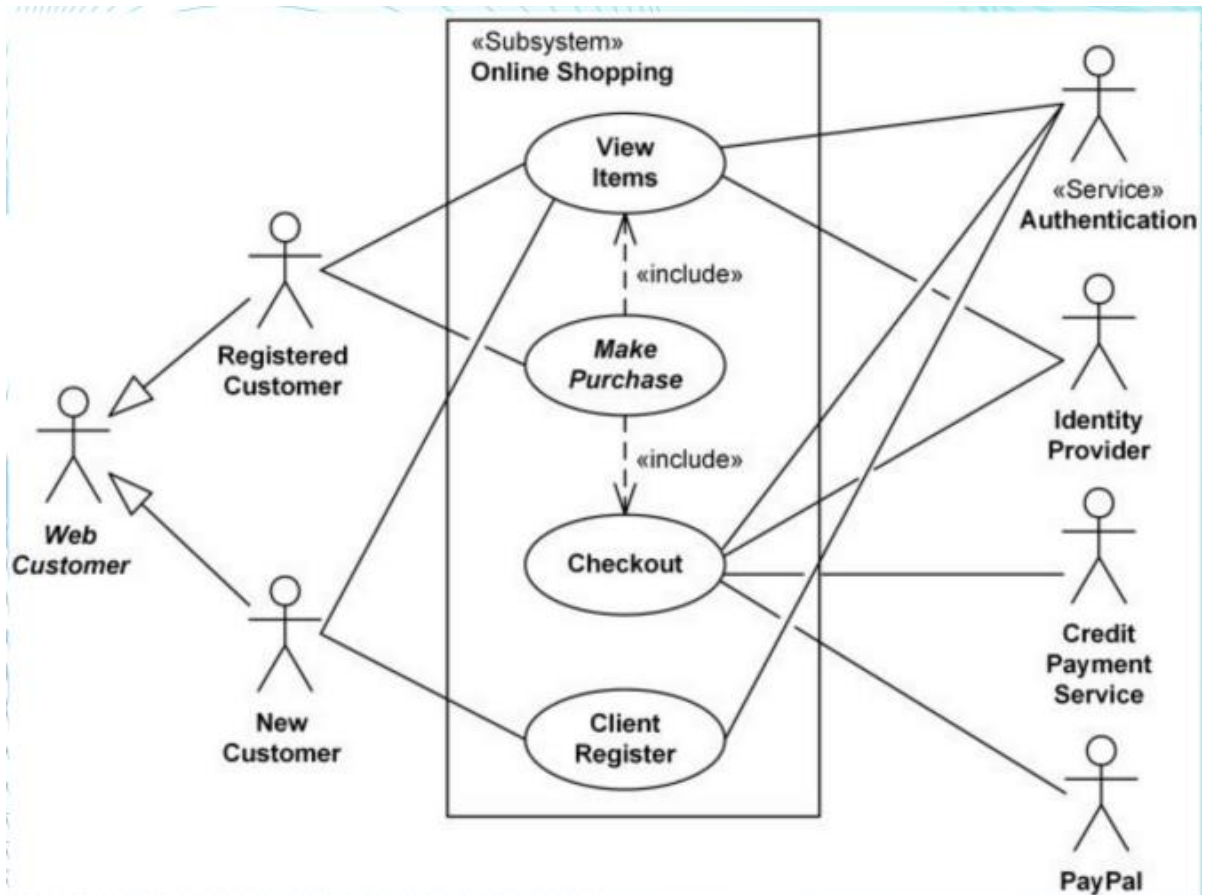


Figure 4.4.1.3

3. Class Diagram

A class diagram is a type of UML (Unified Modelling Language) diagram that represents the structure and relationships of the classes within a system or software application. It provides a visual representation of the classes, their attributes, methods, and the associations between them. Here's a detailed explanation of a class diagram:

In a class diagram, each class is represented as a rectangle containing three compartments: the class name, attributes, and methods. The class name is usually written in bold or underlined. The attributes represent the properties or characteristics of the class, while the methods represent the behaviours or operations that the class can perform.

class diagrams may include other elements such as interfaces, abstract classes, and dependencies between classes. Interfaces define a contract of methods that implementing classes must adhere to. Abstract classes provide common characteristics and behaviours that subclasses can inherit. Dependencies represent the relationship between classes when one class relies on another class for some functionality.

Class diagrams help in understanding the structure of a system, identifying the relationships between classes and visualizing the organization and hierarchy of the classes. They serve as a blueprint for software design and facilitate communication between stakeholders, designers, and developers.

4. Interaction Diagram

An interaction diagram is a type of UML (Unified Modelling Language) diagram that depicts the dynamic behaviour of a system or software application. It focuses on illustrating the interactions and message exchanges among objects or components within a specific scenario or use case. There are two main types of interaction diagrams: sequence diagrams and collaboration diagrams.

1. Sequence Diagram:

A sequence diagram represents the sequence of interactions between objects or components over time. It shows the order of messages exchanged between objects and the lifeline of each object, which represents the object's existence during the interaction.

2. Collaboration Diagram:

A collaboration diagram, also known as a communication diagram, emphasizes objects' structural organization and interactions. It focuses on illustrating the relationships and connections between objects, rather than the sequence of events.

In a collaboration diagram, objects are represented as rectangles, and the connections between objects are depicted as lines with arrowheads. Messages are shown as labelled arrows that indicate the direction of communication.

5. Component Diagram

A component diagram is a type of UML (Unified Modeling Language) diagram that depicts the organization and dependencies of software components within a system. It provides a high-level view of the system's architecture, illustrating how various components interact and collaborate to achieve specific functionalities.

Components represent modular units of the system that encapsulate a set of related functionalities and behaviors. They can be physical components (e.g., hardware devices) or software components (e.g., classes, modules, libraries). In a component diagram, components are represented as rectangles with their names and, optionally, stereotypes.

6. DFD Level 0

The context level DFD provides an overview of the entire system, showing the system as a single process or a black box. It represents the system's boundaries and the interactions between the system and external entities, without going into detailed processes within the system. It focuses on the flow of data into and out of the system, depicting external entities and the data exchanged with them. The context level DFD is often the starting point of the DFD modeling process.

7. DFD Level 1

The level 1 DFD provides a more detailed representation of the system than the context level DFD. It decomposes the single process of the context level DFD into subprocesses, illustrating how data flows between these subprocesses. It represents the major functional components of the system and their interconnections. Each subprocess represents a unique function or process within the system, and the data flows depict the movement of data between these subprocesses.

8. DFD Level

The level 2 DFD further decomposes the processes from the level 1 DFD into more detailed subprocesses. It provides a more granular view of the system, representing additional processes, data stores, and data flows. The level 2 DFD breaks down complex processes into smaller, manageable subprocesses, showing the detailed flow of data between them. This level of DFD allows for a more comprehensive understanding of the system's internal operations and the relationships between different subprocesses.

9. ER Diagram

An ER (Entity-Relationship) diagram is a visual representation of the relationships among entities in a database. It depicts the logical structure of a database and helps to visualize the entities, attributes, and relationships between them.

ER diagrams are widely used in database design and development. They serve as a visual tool to represent the structure of a database, including the entities, attributes, and relationships between them. ER diagrams help in understanding the data requirements, identifying key entities and their relationships, and designing the database schema.

By creating an ER diagram, developers and stakeholders can gain a clear understanding of the database structure, assist in database normalization, and facilitate communication and collaboration between stakeholders during the database design process.

CHAPTER V

IMPLEMENTATION

Partial Code

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Fashion</title>
  <link rel="stylesheet" href="css/style.css">
  <link rel="shortcut icon" href="img/favicon.ico"/>
  <!-- swiper-js cdn link -->
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/swiper@8/swiper-bundle.min.css"/>
  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
</head>
<body>

<!-- ===== Header Section ===== -->
<header class="header">
  <a href="#" class="logo"><i class="fas fa-shopping-cart"></i> QuickStyle </a>
  <nav class="navbar">
    <a href="#home">Home</a>
    <a href="#products">Products</a>
    <a href="#featured">Featured</a>
    <a href="#one">Contact</a>
  </nav>
  <div class="icons">
    <div id="menu-btn" class="fas fa-bars"></div>
    <div id="search-btn" class="fas fa-search"></div>
    <a href="#" class="fas fa-shopping-cart"></a>
    <a href="#" class="fas fa-heart"></a>
  </div>
  <form action="" class="search-form">
    <input type="search" name="" id="search-box">
    <label for="search-box" class="fas fa-search"></label>
  </form>
```

</header>

<!-- ===== Home Section ===== -->

<section class="home" id="home">

<div class="swiper home-slider">

<div class="swiper-wrapper">

<div class="swiper-slide slide" style="background:url(img/banner1.jpg) no-repeat">

<div class="content">

Upto 50% off

<h3>Women's Fashion</h3>

Buy Now

</div>

</div>

<div class="swiper-slide slide" style="background:url(img/banner2.jpg) no-repeat">

<div class="content">

Upto 20% off

<h3>Men's Fashion</h3>

Buy Now

</div>

</div>

<div class="swiper-slide slide" style="background:url(img/banner3.jpg) no-repeat">

<div class="content">

Upto 10% off

<h3>Children's Fashion</h3>

Buy Now

</div>

</div>

</div>

<div class="swiper-button-next"></div>

<div class="swiper-button-prev"></div>

</div>

</section>

<!-- ===== Banner Section ===== -->

<section class="banner-container">

<div class="banner">

<div class="content">

Special offer

<h3>Upto 50% off</h3>

Buy Now

</div>

</div>

<div class="banner">

<div class="content">

Special offer

```

    <h3>Upto 50% off</h3>
    <a href="/purchase.html" class="btn">Buy Now</a>
  </div>
</div>
</section>

```

```

<!-- ===== Products Section ===== -->

```

```

<section class="products" id="products">
  <h1 class="heading"> Product <span>exclusives</span></h1>

```

```

  <div class="filter-buttons">
    <div class="buttons active" data-filter="all">Everything</div>
    <div class="buttons" data-filter="arrivals">New Arrivals</div>
    <div class="buttons" data-filter="featured">Prominence</div>
    <div class="buttons" data-filter="special">Special</div>
    <div class="buttons" data-filter="seller">Vendor</div>
  </div>

```

```

  <div class="box-container">
    <div class="box" data-item="featured">

```

```

      <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
      </div>

```

```

      <div class="content">
        <h3>Product Name</h3>
        <div class="price">
          <div class="amount">$20.00</div>
          <div class="cut">$25.00</div>
          <div class="offer">20% off</div>
        </div>

```

```

        <div class="stars">
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="fas fa-star"></i>
          <i class="far fa-star"></i>
          <span>(50)</span>
        </div>

```

```

      </div>
    </div>

```

```

  <div class="box" data-item="special">

```

```

    <div class="image">
      <a href="/purchase.html" class="purchase1"></a>
    </div>

```

```

    <div class="content">
      <h3>Product Name</h3>
      <div class="price">
        <div class="amount">$20.00</div>
        <div class="cut">$25.00</div>
        <div class="offer">20% off</div>

```

```

</div>
<div class="stars">
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="far fa-star"></i>
  <span>(50)</span>
</div>
</div>
</div>

<div class="box" data-item="seller">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>
  </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>
</div>

<div class="box" data-item="special">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>
  </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>
</div>

```



```

    </div>
  </div>
</div>

<div class="box" data-item="arrivals">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>    </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>
</div>

<div class="box" data-item="arrivals">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>    </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>
</div>

<div class="box" data-item="seller">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>    </div>
  <div class="content">
    <h3>Product Name</h3>

```

```

<div class="price">
  <div class="amount">$20.00</div>
  <div class="cut">$25.00</div>
  <div class="offer">20% off</div>
</div>
<div class="stars">
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="fas fa-star"></i>
  <i class="far fa-star"></i>
  <span>(50)</span>
</div>
</div>
</div>

<div class="box" data-item="seller">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>    </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>
</div>
</div>

<div class="box" data-item="featured">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>    </div>
  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
    </div>
  </div>
</div>

```

```

        <i class="far fa-star"></i>
        <span>(50)</span>
    </div>
</div>
</div>

<div class="box" data-item="featured">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>
    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>

<div class="box" data-item="special">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>
    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>

<div class="box" data-item="sspecial">

```

```

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>
    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>
</section>

```

```

<!-- ===== Deal Section ===== -->
<section class="deal">
    <div class="image">
        
    </div>

    <div class="content">
        <span>Trends Of The New Season</span>
        <h3>Best Summer Collection</h3>
        <p>Get Up To 50% Off</p>
        <a href="/purchase.html" class="btn">Buy Now</a>
    </div>
</section>

```

```

<!-- ===== Featured Section ===== -->
<section class="featured" id="featured">

    <h1 class="heading"> <span>Featured </span>Products</h1>

    <div class="swiper featured-slider">

        <div class="swiper-wrapper">

            <div class="swiper-slide slide">

                <div class="image">
                    <a href="/purchase.html" class="purchase1"></a>
                </div>
                <div class="content">

```

```

    <h3>Product Name</h3>
    <div class="price">
        <div class="amount">$20.00</div>
        <div class="cut">$25.00</div>
        <div class="offer">20% off</div>
    </div>
    <div class="stars">
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="far fa-star"></i>
        <span>(50)</span>
    </div>
</div>
<div class="swiper-slide slide">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>
    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>
<div class="swiper-slide slide">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>
    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>

```

```

        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="far fa-star"></i>
        <span>(50)</span>
    </div>
</div>
</div>

<div class="swiper-slide slide">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>

    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>

<div class="swiper-slide slide">

    <div class="image">
        <a href="/purchase.html" class="purchase1"></a>
    </div>

    <div class="content">
        <h3>Product Name</h3>
        <div class="price">
            <div class="amount">$20.00</div>
            <div class="cut">$25.00</div>
            <div class="offer">20% off</div>
        </div>
        <div class="stars">
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="fas fa-star"></i>
            <i class="far fa-star"></i>
            <span>(50)</span>
        </div>
    </div>
</div>

```

```

<div class="swiper-slide slide">

  <div class="image">
    <a href="/purchase.html" class="purchase1"></a>
  </div>

  <div class="content">
    <h3>Product Name</h3>
    <div class="price">
      <div class="amount">$20.00</div>
      <div class="cut">$25.00</div>
      <div class="offer">20% off</div>
    </div>
    <div class="stars">
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="fas fa-star"></i>
      <i class="far fa-star"></i>
      <span>(50)</span>
    </div>
  </div>

</div>

<div class="swiper-button-next"></div>
<div class="swiper-button-prev"></div>

</div>

</section>
<footer>
  <div class="foot" id="one">
    <h1></h1>
  </div>
</footer>

<script src="https://cdn.jsdelivr.net/npm/swiper@8/swiper-bundle.min.js"></script>
<script src="js/script.js"></script>
</body>
</html>

```

Screenshots

Figure 5.2.1

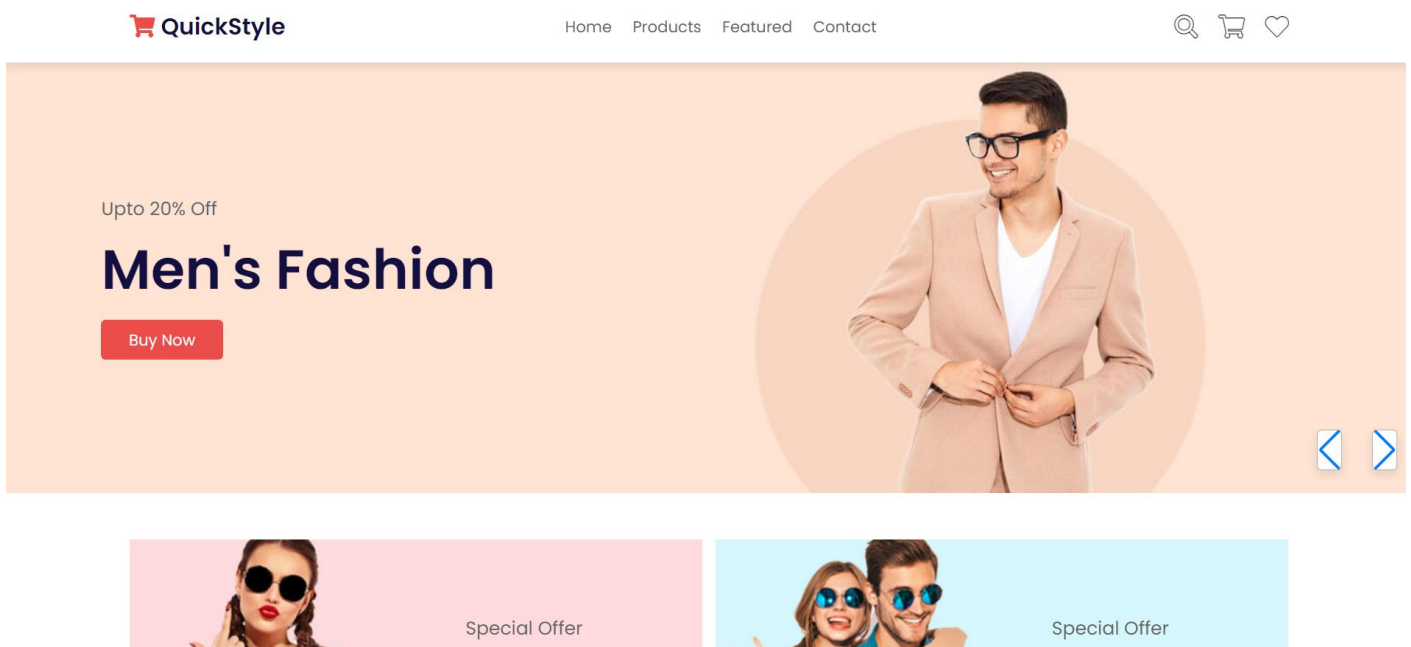


Figure 5.2.3

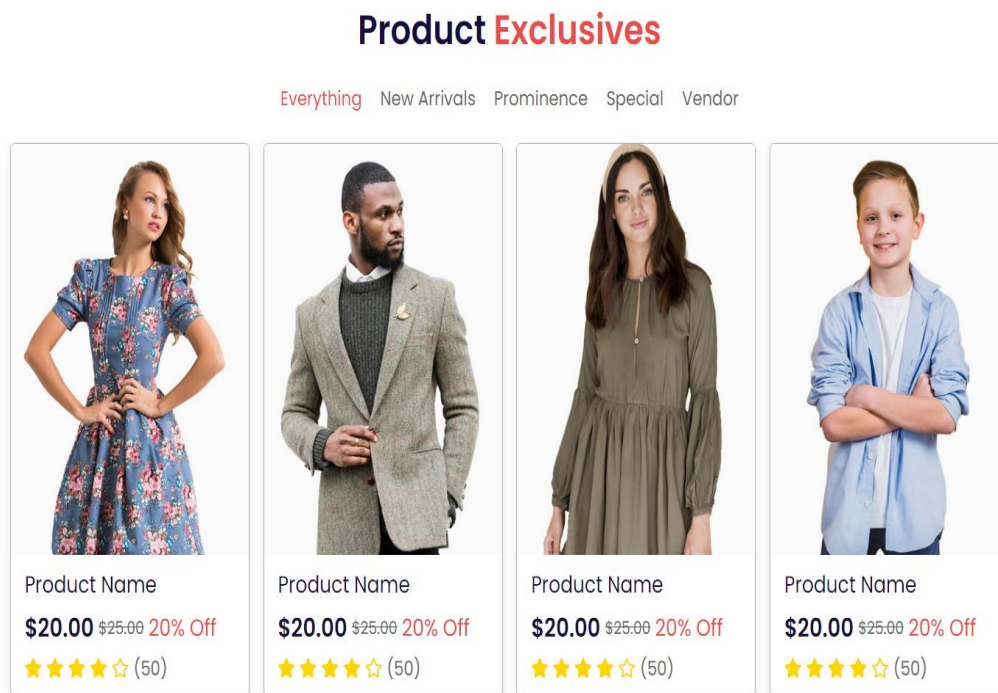


Figure 5.2.9

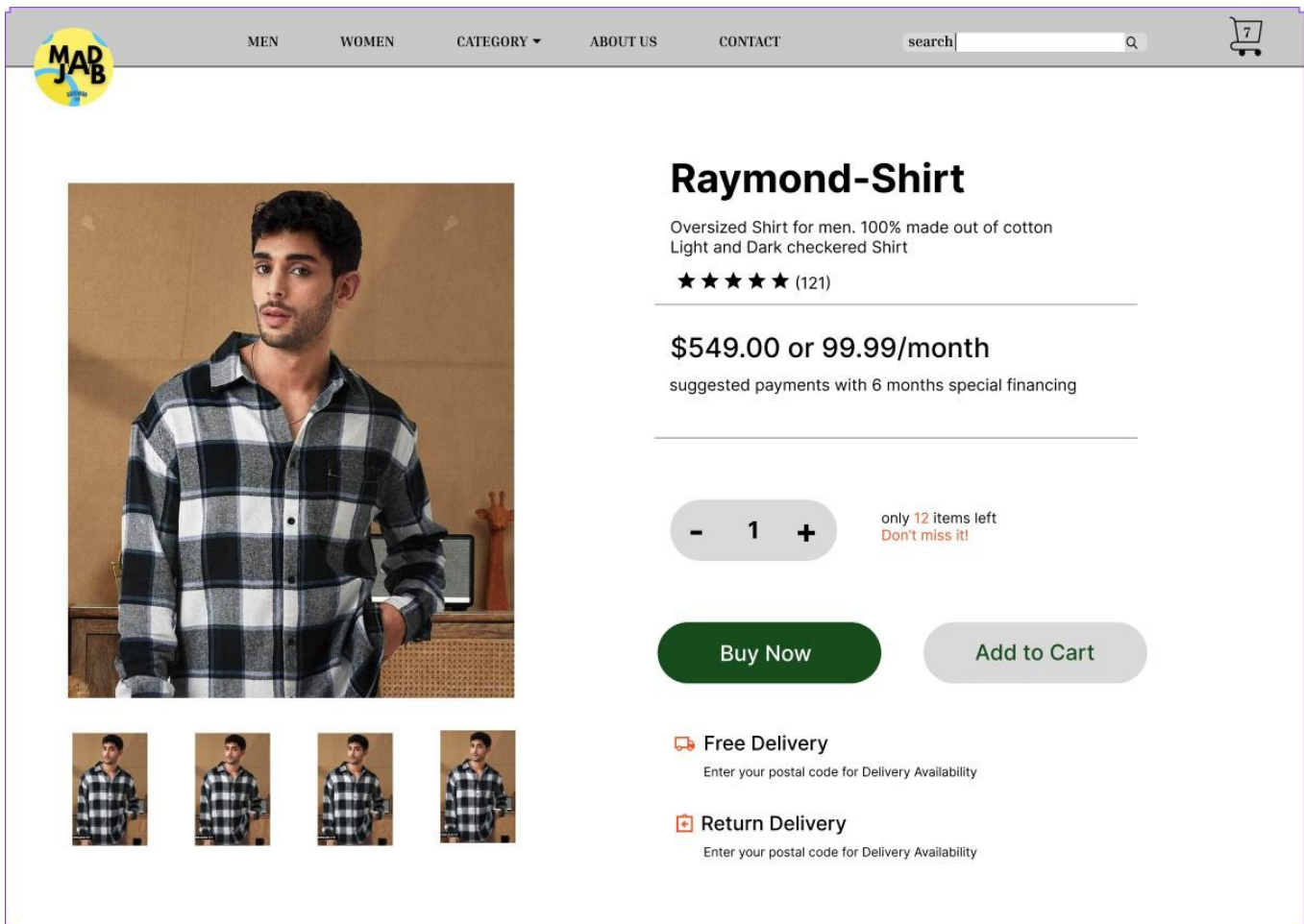
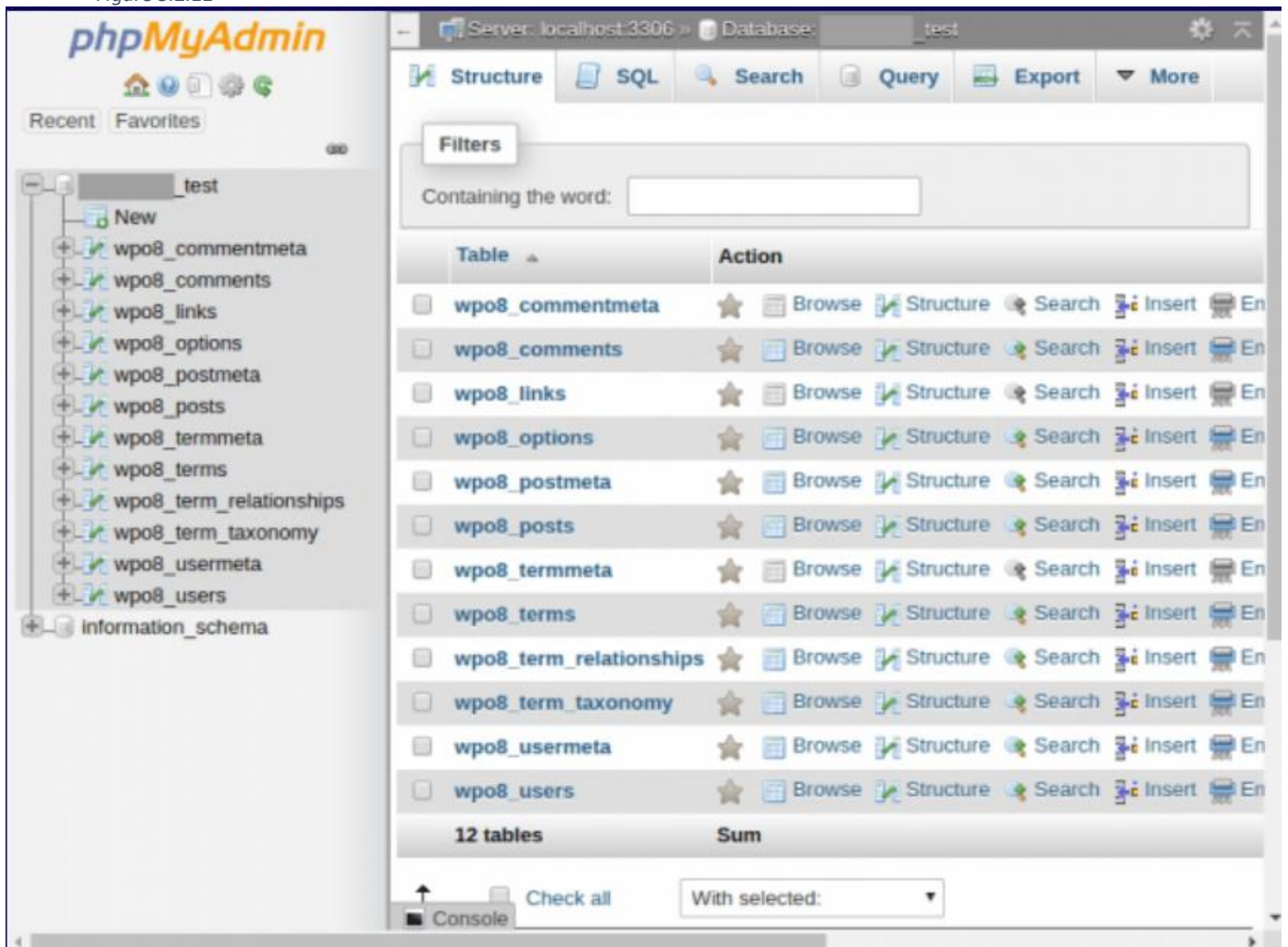


Figure 5.2.11



Chapter VI

Conclusion

In conclusion, our ecommerce fashion website 'QuickStyle' offers a comprehensive set of features to enhance the user experience. With functionalities such as user login and sign up, product browsing, shopping cart, checkout, and order tracking, we provide a seamless platform for users to shop for fashion items online.

The user login and sign up feature allows users to create accounts, securely log in, and manage their personal information. This ensures a personalized experience and easy access to their purchase history and preferences.

The product browsing feature allows users to search and select products from our vast catalog. Users can filter products by category, price, size, and other criteria.

The shopping cart feature allows users to add products to their cart and checkout. Users can review their cart before checkout and apply coupons or promo codes.

The checkout feature allows users to enter their shipping and billing information and pay for their order. Users can choose from a variety of payment methods, including credit cards, debit cards, and PayPal.

The order tracking feature allows users to track the status of their orders. Users can see when their orders are shipped and when they are expected to arrive.

By leveraging the latest ecommerce technologies, we have developed a secure and reliable website. Our website is PCI compliant and uses SSL encryption to protect user data.

Chapter VII

Reference

1. HTML, CSS, Bootstrap, and JavaScript:

- Mozilla Developer Network (MDN): MDN (<https://developer.mozilla.org/>) is an authoritative resource for web development technologies. Reference it for HTML, CSS, and JavaScript documentation.
- Bootstrap Documentation: If you used Bootstrap (<https://getbootstrap.com/>), cite its official documentation as a reference for the Bootstrap framework.

2. SQLite Database:

- SQLite Documentation: Include a reference to the official SQLite documentation (e.g., <https://www.sqlite.org/docs.html>) for information on using SQLite as your database.

3. NLP Libraries:

- NLTK (Natural Language Toolkit) Documentation: Reference the NLTK documentation (e.g., <https://www.nltk.org/>) to acknowledge the usage of NLTK for NLP tasks.
- PyTorch Documentation: If you employed PyTorch (<https://pytorch.org/>), include a reference to its official documentation as a resource for deep learning and machine learning.

4. OpenAI:

- OpenAI Documentation: If you utilized OpenAI resources or models, provide a reference to the OpenAI documentation (e.g., <https://docs.openai.com/>) for information about their tools and services.