WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

PROJECT TITLE: **Beauty Salon Service Management System For GlowCity**

Name	**ID**
Ahlam Nuredin	NSR/0174/13
Aziza Sied	NSR/0330/13
Alhamdu Bedewe	NSR/0203/13

Project Advisor: Miss Ergoye D.

May 10,2024

WOLKITE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATICS

DEPARTMENT OF COMPUTER SCIENCE

PROJECT TITLE

BEAUTY SALON SERVICE MANAGEMENT SYSTEM FOR GLOWCITY

SUBMITED TO DEPARTMENT OF COMPUTER SCIENCE IN PARTIAL

FULFILMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHLER OF SCIENCE IN COMPUTER SCIENCE.

Name	ID
Ahlam Nuredin	NSR/0174/13
Aziza Sied	NSR/0330/13
Alhamdu Bedewe	NSR/0203/13

Project Advisor: Miss Ergoye D.

Wolkite University, Wolkite, Ethiopia

May 10, 2024

Declaration

We, Ahlam Nuredin, Aziza Seid and Alhamdu Bedewe, declare that the project titled "Beauty Salon Service Management system for Glow City " was completed under the supervision of Ergoye D . We affirm that our contributions are original, and we have not engaged in any form of plagiarism. All referenced sources have been properly cited, and we take full responsibility for any consequences that may arise if this declaration is found to be violated.

Date _____

Group members name signature

1. Ahlam Nuredin _____

2. Aziza Seid _____

3. Alhamdu Bedewe _____

Acknowledgement

First and foremost, we would like to express our utmost gratitude to the Almighty God for His unwavering guidance, provision, and blessings throughout our lives, leading us to this point. We extend our sincere appreciation to the College of Computing for granting us the opportunity to undertake our project work on "Beauty Salon Service Management System for Glow City." We would like to convey our heartfelt thanks to our esteemed project Advisor, Miss Ergoye D., for her invaluable guidance and advice, steering us in the right direction from project initiation to completion. Furthermore, we would like to express our gratitude to Mr. Habtamu Alemayehu, the Head of the Department of Computer Science, for his keen interest in our progress and unwavering support.

Approval

This is to confirm that the project proposal entitled Beauty Salon Management System

For Glow City submitted to Wolkite University, College of Computing and Informatics Department of Computer Science by:

Name	ID	Signature
Ahlam Nuredin	NSR/0174/13	----------
Alhamdu Bedewe	NSR/0203/13	----------
Aziza Seid	NSR/0330/13	----------

Advisor Name	Signature	Date
Ergoye Destaw		
---	---	---
Department Head Name	Signature	Date
---	---	---
Examiner 1 Name	Signature	Date
---	---	---
Examiner 2 Name	Signature	Date
---	---	---
Examiner 3 Name	Signature	Date
---	---	---

Catalog

List of Tables

List of Figures

List of Acronyms

CSS: Cascading Style Sheets

HTML: Hypertext Markup Language

MYSQL:My Structure Query Language

UML**:**Unified Modeling Language

CHAPTER ONE

1. Introduction

Technology refers to the application of scientific knowledge and tools to create practical solutions and improve various aspects of human life. It encompasses a wide range of innovations and advancements that have transformed the way we communicate, work, learn, and interact with the world around us.

One of the fundamental aspects of technology is information and communication technology (ICT). ICT includes devices, systems, and networks that enable the processing, storage, retrieval, and transmission of information. This includes computers, smartphones, the internet, software applications, and telecommunications infrastructure.

Applying Information and Communication Technology (ICT) to beauty salon management systems offers several advantages over manual systems. It enables efficient appointment scheduling, helping to avoid double bookings and manage time effectively.In addition we also offer exceptional home salon services. With digital records, salon owners can maintain a centralized customer database, allowing for personalized services and improved customer satisfaction.

A beauty salon management system is a system that helps streamline operations in a salon, Including appointment scheduling, customer management and order product.

In a beauty salon operating with a manual system, customers encounter challenges such as long waiting times, limited appointment availability, inconsistent service quality, limited privacy and difficulty in maintaining personalized preferences. Administrators face difficulties in managing schedules, coordinating staff, tracking inventory, and maintaining customer records manually. These challenges highlight the limitations and inefficiencies of a manual system, impacting both customers and administrators in the beauty salon setting.

In addition to providing appointment scheduling and home services, our beauty salon management system also offers a convenient feature that allows customers to order products

directly through our platform. We strive to provide a comprehensive solution that encompasses all your salon needs, from scheduling appointments to accessing top-quality products with ease.

1.1 Background of the organization

Glow City is a beauty salon service provider located in Bole ,Addis Ababa, Ethiopia, that currently operates with a manual system. The salon's mission is to provide high-quality beauty services, enhancing customers' natural beauty and boosting their self-confidence. The vision of Glow City is to become a leading beauty destination in Addis Ababa, known for exceptional service quality and a customer-centric approach. The salon aims to streamline operations, improve customer experience, enhance customer retention, and increase overall efficiency through the implementation of a new system.

1.2 Statement of Problems

The current problem in the beauty salon industry is lacks an organized and efficient method for scheduling appointments and have professional come home to provide those services.

Existing methods of booking appointment often involve phone calls, which can be time-consuming and inconvenient for both customers and salon professionals.

There is no way for customers to check the availability of specific professionals, resulting in potential scheduling conflicts.

Lacks a robust feedback mechanism for customers to share their experiences and provide ratings and reviews.

In addition to the challenges faced by customers, salon owners and employees in the beauty salon industry also encounter many problems. Owners struggle with time-consuming appointment scheduling, limited visibility into business data, inefficient inventory management, and tedious record-keeping. Employees face communication barriers, scheduling conflicts, and limited access to relevant information. Transitioning to an

automated system streamlines scheduling, improves data visibility, optimizes inventory management, enhances communication, reduces conflicts, and provides easy information access for employees.

The lack of comprehensive online platform that integrates services selection, professional availability, appointment scheduling hinders the growth and efficiency of beauty salon.

1.2 Objectives of the project

1.3.1 General Objective

The general objective of our project is to design and develop online beauty salon service management system for Glow City .

1.3.2 Specific Objective

In order to achieve the general objective of the project stated above, the following specific objectives were formulated:

- Conduct a thorough analysis of existing beauty salon service management systems to identify major functions and problems.
- Determine both functional and non-functional requirements for the online beauty salon service management system.
- Design a database based on the analyzed data.
- Implement the system design.
- Conduct testing and debugging procedures to ensure the system's functionality and reliability.
- Deploy the online beauty salon service management system for Glow City after successful testing and debugging.

1.3 Feasibility Analysis

1.4.1 Technical Feasibility:

The system involves various functionalities, including user registration, service selection, cart management, worker availability checking, appointment scheduling, and payment processing. These functionalities can be implemented using established web development technologies such as HTML, CSS, React.js, and server-side languages like Node.js. MySQL is a suitable choice for the system's database management system because it supports relational data modeling and SQL queries. It is compatible with popular web development technologies, offers efficient performance and scalability, enjoys extensive community support, and incorporates robust security features. The system require a team of programmers, testers, and debuggers to develop and ensure its quality. Standard development tools, including IDE like Visual Studio are readily available. Security measures, such as secure authentication, data encryption, and protection against common web vulnerabilities, can be implemented using established best practices and libraries.

1.4.2 Economic Feasibility:

Assessing the economic feasibility involves analyzing the costs associated with the development, maintenance, and hosting of the website. We also need to determine potential revenue streams, such as service fees and calculating Return on Investment (ROI) to determine the project's financial viability. Also determine the cancellation fee process. By implementing the online management system, the beauty salons can attract a wider customer base and potentially increase revenue. Customers can benefit from the convenience and accessibility of the service. The revenue generated through service fees or commissions from successful bookings can contribute to the economic feasibility of the project.

1.4.3 Operational Feasibility:

Focuses on evaluating the practicality and ease of use of the system for both customers and beauty salon administrator. Strong support from management is crucial in providing necessary resources and removing potential hurdles during implementation. Addressing

industry pain points, we actively seek to identify and rectify existing inefficiencies within the beauty salon industry, thus making our system more appealing to users. The system's potential to significantly reduce operational time is a compelling factor for its adoption.

1.4.4 Legal and Ethical Feasibility:

Legal and ethical feasibility requires compliance with applicable laws and regulations. This includes data protection regulations, privacy policies, and payment security standards. We must ensure the protection of customer information and maintain confidentiality to address ethical considerations

1.5 Scope and limitations of project

1.5.1 Scope of the Project:

- **Customers Registration and Login:** Customers can create accounts or login using their credentials to access the appointment system.
- **Service Selection:** Customers can browse through the available services offered.
- **Cart Management:** A cart feature allows customers to add multiple services, adjust quantities to add more than one person, and manage their appointment.
- **Worker Availability:** Customers can view the schedules of service professionals and select a preferred worker based on availability.
- **Appointment Scheduling:** Customers can select the date and time for their appointment, considering the availability of the chosen worker.
- **Cancellation:** Implement clear and fair cancellation policies that specify time frames. Inform customers of these policies through privacy and security of the web-application.
- **Confirmation and Notifications:** After successful payment, customers receive confirmation of their appointments via SMS or email.

1.6 Significance of the project

- Convenience for busy individuals to book beauty salon services online and purchase beauty products in a single platform, saving time and effort.
- Increased accessibility for individuals who may have mobility issues or limited salon access, allowing them to access a wide range of services and products from the comfort of their homes.
- Enhanced customer experience through transparency and personalized choices.
- Business opportunities for salons to expand their reach and offer home services.

1.7 The beneficiaries of the project

- Customers seeking beauty salon services with convenience and personalized choices.
- Individuals with limited mobility or restricted access to physical salons, who can now access a wide range of beauty services and products from their homes.
- Salon administrators who can efficiently manage schedules, track appointments for services and product purchases.
- Professionals who can showcase their availability and receive appointments for their services.

1.8. Methodology of the Project

1.8.1. Data Collection Tools/Technique

1. Interviews:

Conduct interviews with the salon owner or manager to understand their needs, expectations, and pain points in managing appointments and services.

2. Brainstorming or Discussion Sessions:

Facilitate open conversations with salon staff and potential customers to gather insights on their experiences and gather ideas for improving appointment and service management.

3. Observation:

Spend time observing how the salon currently manages appointments and services. This can help you identify specific workflow and scheduling challenges.

4. Documentation Review:

Review any existing documents, forms, or systems used by the salon to manage appointments and services.

1.8.2 System analysis and design

1. Object Oriented Analysis (OOA)

During this phase we uses to model the function of the system (use case modelling), find and identify the business objects, organize the objects and identify the relationship between them and finally we will model the behaviour of the objects in detail.

2. Object Oriented Design (OOD)

During this phase we will refine the use case model and rational rose for designing the sequence, collaboration, activity diagrams and to model object interactions and behaviour that support the use case scenario.

1.8.3. System Development Model

1. Agile Methodology: The Agile methodology is well-suited for projects with evolving requirements, a focus on customer satisfaction, and the need for flexibility and responsiveness. It promotes iterative and incremental development, collaboration, and continuous feedback.

Scrum Framework: a popular Agile framework, is particularly suitable for our project. It divides the development process into short iterations called sprints. Each sprint involves planning, development, testing, and review activities.

We select Agile development model because this model offers adaptability for evolving requirements, fosters customer satisfaction through continuous delivery, promotes flexibility, collaboration, and feedback, and enables incremental value delivery, aligning with your online beauty salon booking system project goals.

1.8.4. System Testing Methodology

- **Unit Testing:** Unit testing focuses on testing individual components or units of code in isolation. It ensures that each component functions correctly and meets its specific requirements.
- **Integration Testing:** Integration testing verifies the proper interaction and data flow between different components or modules of the system.
- **System Testing:** System testing evaluates the behavior and functionality of the entire system as a whole. It involves testing the system's end-to-end workflows, user interfaces, business logic, and all the integrated components.
- **Acceptance Testing:** Acceptance testing validates whether the beauty salon booking system meets the expectations and requirements of the stakeholders, including customers and salon.

1.8.5 Development Tools and Technologies

To develop our system, we will use different tools.

Hardware tools

- Personal computer (PC): For every activity of the project.
- Flash disc: to store files.
- Printer: to print the documents.

Software tools

We will use the following tools to develop the web-based system:

Front-End Technologies:

- **CSS:** to specify the layout of web pages.
- **HTML:** to define the content of web pages.
- **React.js:** to program the behavior of web pages.

Back-end technologies:

- **Express.js:** fast, and minimalist web application framework for Node.js
- **MYSQL:** for reliable and salable efficient relational database management
- **Node.js:** for server-side scripting language.

Other software tools:

- Draw.io:- to draw diagram.
- EdawMax:- to draw diagram.
- Microsoft office word 2016 and Microsoft power point 2016 to edit the document and to prepare the presentation.
- Web browser:- to search reference and to execute the implementation.

CHAPTER TWO

2.DESCRIPTION OF THE EXISTING SYSTEM

The existing system of Glow City Salon uses manually approach that is document in order book the service and to record this task. If the user want to book a service the user must have to go to the place where the beauty salon located or it can use his phone in order to book the service. the beauty salon owner or administrator use a document in order to register the user and schedule the booking.

2.1. Introduction of Existing System

The existing system for Glow City Salon involves a step-by-step process :

1. **Appointment Booking**: Customers typically visit the salon to schedule an appointment for the desired service. They provide details such as the type of service, preferred date and time, and any specific requirements.

2. **Reception and Check-in:** The receptionist checks them in by recording their details, confirming the appointment, and assigning them to a stylist or technician.

3. **Consultation:** The customer meets with the stylist or technician to discuss their desired look or treatment. The stylist may ask questions to understand the customer's preferences, hair type, skin condition, or any allergies or sensitivities. This consultation helps determine the best approach and products to use.

4. **Service Execution:** The stylist or technician proceeds with the service based on the customer's requirements. This could include haircuts, hair coloring, styling, manicures, pedicures, facials, waxing, or other beauty treatments. They follow standard procedures and use appropriate tools, equipment, and products.

5. **Service Completion:** Once the service is finished, the stylist or technician ensures that the customer is satisfied with the result. They may show the customer their new look, provide styling tips, or answer any questions. The customer's feedback is important for the salon to improve its services.

6. **Payment and Checkout:** After the service, the customer proceeds to the checkout counter. The salon staff calculates the total cost based on the services rendered, additional products used, and any applicable taxes. The customer makes the payment, either in cash or through electronic methods like credit/debit cards or mobile wallets.

7. **Feedback and Follow-up**: Some salons may request feedback from customers to evaluate their experience. This feedback helps in identifying areas for improvement and maintaining customer satisfaction. The salon may also schedule follow-up appointments or offer loyalty programs to encourage repeat business.

2.2. Users of Existing System

1. Salon Owner:

- The owner is responsible for managing the salon and making strategic decisions regarding its operations, growth, and profitability.
- They may handle administrative tasks such as financial management, marketing, staff management, and overall business development.
- The owner may also perform services themselves, depending on their expertise and the needs of the salon.

2. Salon Staff:

- **Receptionist**: The receptionist assists with managing appointments, answering phone calls, greeting clients, and handling administrative tasks at the front desk.
- **Stylists/Hairdressers**: These professionals provide various hair services such as cutting, styling, coloring, and treatments.

- **Estheticians**: They specialize in skincare and perform services like facials, waxing, and other beauty treatments.
- **Nail Technicians:** These professionals focus on nail care, including manicures, pedicures, and nail enhancements.
- **Makeup Artists**: They provide makeup application services for special occasions, events, or bridal makeup.
- **Other Support Staff**: Depending on the size of the salon, there may be additional staff members such as assistants, shampoo technicians, or cleaning personnel.

3. Clients:

- Clients provide information about themselves when booking appointments or arriving at the salon.
- They may provide personal details such as their name, preferred services, and any specific requests or concerns they have.
- Clients may also provide information about their hair or skin type, allergies, or any previous treatments they have received to help the salon staff customize services accordingly.

2.3. Major Functions of the Existing System

- **Appointment Scheduling**: Organizing appointments for specific services, preferred dates, and times.
- **Service Management**: Manages services offered, including descriptions, durations, and costs for effective communication.
- **Staff Management**: Maintains records of staff members, qualifications and performance.
- **Billing and Payment**: Handles billing, discounts, payment calculations, and generates invoices or receipts with various payment options.

2.4. Drawbacks of the Existing System

A manual beauty salon booking system can be cumbersome and inefficient, leading to various drawbacks:

- The system is prone to errors due to the manual nature of the process.
- The manual nature of the current process in the system increases the risk of errors, leading to potential issues such as double bookings, missed appointments, and customer dissatisfaction.
- The system requires staff to spend time on administrative tasks such as scheduling appointments and managing customer data. This takes away from time that could be spent on other important tasks such as customer service and marketing.
- The system does not allow for easy tracking of customer information and preferences, making it difficult to personalize services or track customer history.
- The system is not easily accessible to customers, as they have to physically visit the salon or call to make an appointment.

2.5. Business Rules of the Existing System

- All employees must arrive on time and maintain a professional appearance.
- All clients' information must be kept confidential and secure.
- No appointments can be scheduled during another appointment the same time.
- All employees must attend regular training sessions to maintain their skills and knowledge up-to-date.
- Refunds will only be issued if the service received was not completed as requested.
- Payment must be made before leaving the salon.

CHAPTER THREE

3. PROPOSED SYSTEM

3.1 Functional Requirements

3.1.1.User Registration and Login

- The system should provide user registration functionality for salon customers to create an account.
- Users should be able to log in to their accounts using their credentials.

3.1.2.Service Selection

- Customers should be able to browse and view the list of services offered by the salon.
- The system should display detailed information about each service, including the description, duration, and price.
- Customers should be able to select the desired service and add it to their cart.

3.1.3.Cart Management

- The system should allow customers to view and modify the services and products in their cart.
- Customers should have the option to customize their appointment, such as specifying the number of people or adding additional services.
- Customers should be able to review the cart and proceed to the appointment confirmation or product purchase.

3.1.4. Professional/Worker Availability

- The system should display the availability schedule of beauty professionals/workers associated with the salon.
- Customers should be able to view the schedule of a specific professional and check their availability for a desired date and time.

3.1.5. Appointment Scheduling

- Customers should be able to select a date and time for their appointment based on the availability of the chosen professional.
- If the chosen professional is not available, the system should suggest alternative professionals who are free on the desired date.

3.1.6. Appointment Confirmation

- After successful payment, the system should send a confirmation notification to the customer via SMS or email.
- The confirmation should include the details of the scheduled appointment, such as date, time, and service, as well as the details of the product purchase.

3.1.8.Administrator Dashboard

▪ Salon administrators should have access to a dashboard to manage services, professionals, and customer appointments.

▪ Administrators should be able to add or update service offerings, manage professional schedules, view and manage product inventory, and view customer appointments and product orders.

3.1.9.Payment

▪ The system simulate the payment process using CHAPA simulation for appointment bookings and product purchases.

▪ Customers should be able to proceed to the payment step during appointment confirmation or product purchase.

▪The payment simulation is for testing purposes only and does not involve real transactions at this stage.

3.2. Non-functional Requirements

3.2.1. User Interface and Human Factors

- The system provides a user-friendly interface that is intuitive and easy to navigate.
- The interface accommodates users with varying levels of expertise, catering to both novice and experienced users.
- Clear instructions, informative labels, and appropriate feedback is incorporated to assist users in their interactions with the system.

3.2.2. Hardware Consideration

- The system is compatible with commonly used devices such as desktop computers, laptops, tablets, and smartphones.
- Compatibility with specific operating systems and web browsers should be ensured.

3.2.3. Security Issues

- The system prioritizes security measures to protect against internal and external intrusions.

- User authentication and authorization mechanism is implemented to ensure that only authorized individuals can access sensitive information or perform specific actions.

3.2.4. Performance Consideration

- The system is highly responsive to provide a seamless user experience.
- It is designed to handle concurrent users efficiently, supporting a specific number of simultaneous users without significant performance degradation.
- The system is able to handle a typical load of concurrent users, and provisions should be made to scale up the system to handle extreme loads during peak usage periods.

3.2.5. Error Handling and Validation

- The system is robust error handling mechanisms in place to handle exceptions and provide appropriate error messages to users.
- User inputs is validated to prevent invalid or malicious data from causing disruptions.

3.2.6. Quality Issues

- The system should strive to be reliable, available, and robust.
- Fault-tolerant mechanisms is implemented to minimize the impact of failures and ensure high system availability.
- Regular quality assessments is conducted, with client involvement in evaluating the system's quality and development process.
- Feedback from users and stakeholders is gathered to continuously improve the system's quality.

3.2.7. Resource Issues

- The system optimizes resource consumption, such as CPU, memory, and storage, to ensure optimal performance and scalability.

CHAPTER FOUR

4. SYSTEM ANALYSIS

Here in this chapter we are going to discussed about use case model, object model and dynamic models.

4.1. System Model

The beauty salon management system model encompasses interactions and components that facilitate seamless service appointment and delivery. At its core, the customer accesses the user-friendly website to explore salons and services, selecting preferred options that are added to the cart. This cart functionality allows for customization, additional services. Service professionals are integrated, with customers able to select based on availability displayed in their schedules. Overall, the system model enhances convenience for customers, empowering them to access salon services while providing salons an effective means of managing appointments and their workforce.

4.1.1. Use case model

The use case model for beauty salon management system consists of a use case diagram and accompanying documentation that outlines the various use cases, actors, and their relationships. The use case diagram visually represents the interactions between the system and its external environment, providing an overview of the system's intended functionalities within its operating context. It showcases the different actions and behaviors expected from the system, as well as the roles played by actors interacting with the system. The use case model serves as a valuable tool for understanding and documenting the functionality and interactions of the beauty salon online appointment system in a clear and concise manner.

Actor and Use Case Identification:

Use case: A use case is the process of identifying and representing a series of actions that actors (users) undertake within a system to achieve a specific goal or target. Use cases are typically depicted as ellipses with descriptive names that explain their purpose.

Actors: User (Tourists): Represents users who use the proposed web application to view information and access services. Admin Represents the system administrator responsible for managing the system and maintaining the database environment.

Use case:

The initial step in analyzing a new system is to identify the key activities that are primarily performed within the proposed system. From the system specification, the following use cases have been identified as crucial components of the system.

- ✓ Registration
- ✓ Appointment
- ✓ Add to Cart
- ✓ Rating Professional
- ✓ Payment
- ✓ Manage Product
- ✓ Notification
- ✓ View Feedback
- ✓ Professional Feedback
- ✓ Manage Service
- ✓ View Order
- ✓ Manage Billing

- ✓ General Feedback
- ✓ Manage Account

4.1.1.1 Use Case Model

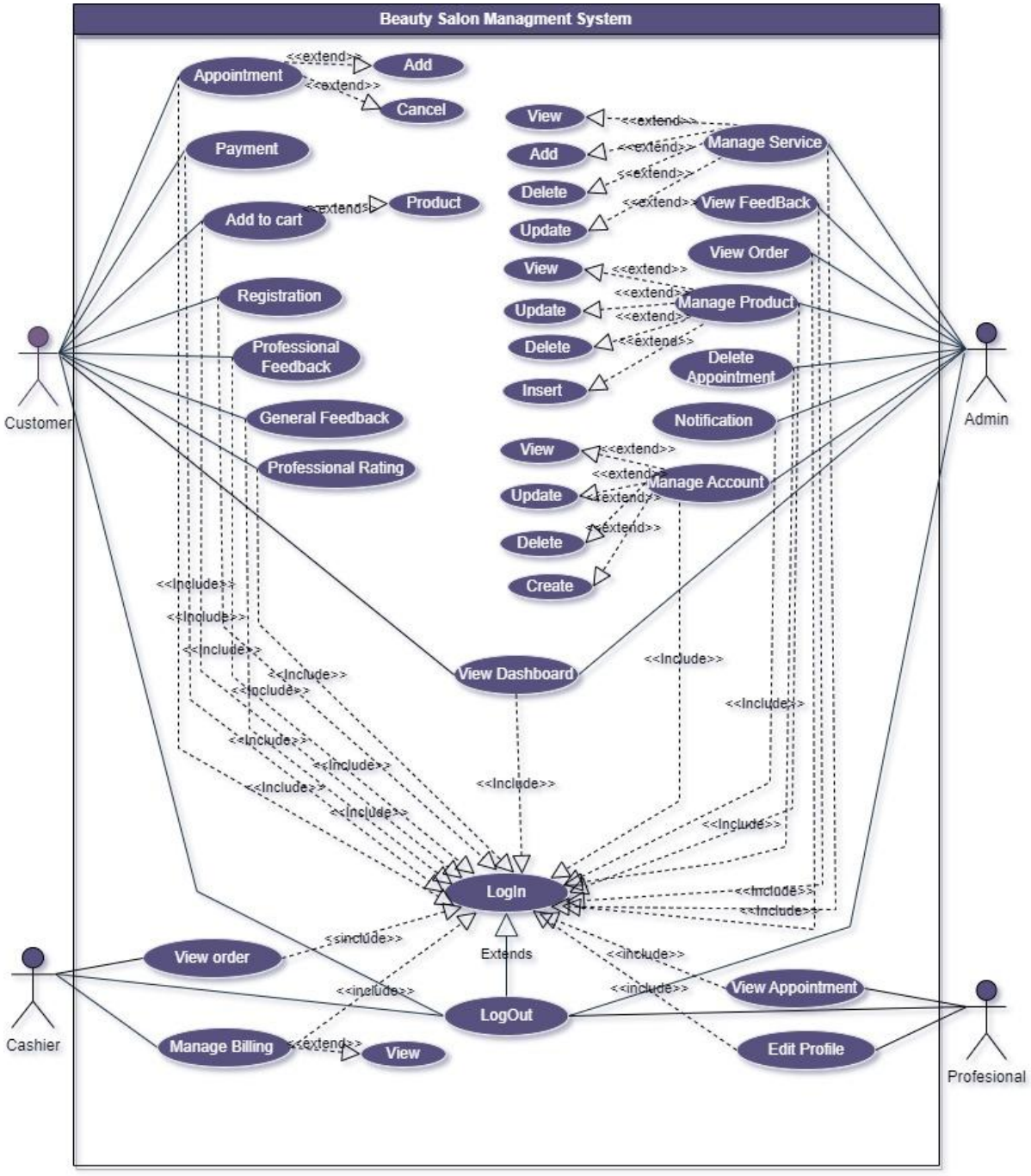


Figure 1 Use Case Model

4.1.1.2. Use Case Description

A use case description is a document that provides concise information about a specific use case. It includes essential components such as the use case name, use case ID, participating actors, description, basic course of action, alternative course of action, preconditions, and post-conditions. This description outlines how the actors involved should perform the use case.

Table 1 Use case description for login

Use Case Name	**Login**
Actors	Customer,employee(Professional, Cashier),Admin
Description	Registered user can login into the system
Pre-Condition	User must be registered in the system,user should have username and password
Post-Condition	User given access to system and redirected to their homepage
Main Flow	1. User got homepage 2. User click link that display login form 3. System display login form 4. User enter username and password 5. User click login button 6. System validate data 7. If the data is valid, system display the correct page 8. The use case ends.
Alternative	7.If the data is invalid,system display error message 8.Step 3 repeated

Table 2 Use case description for Make Appointment

Use Case Name	**Make Appointment**
Actors	Customer
Description	Customer can schedule to make an appointment for service
Pre-Condition	1.User must be logged into the system 2.The user has access to the appointment scheduling feature
Post-Condition	1.Appointment data stored in database 2.Customer receive confirmation email that contain appointment details 3.The appointment added to customer personnel schedule 4.User can view and manage appointment
Normal Flow	1.Customer go to appointment scheduling page 2.The system display the page 3.Customer enter the required information 4.User click appoint button 5.The system validate the entered information 6.The system conform the appointment 7.The use case ends.

Alternative	7.If the data is invalid,system display error message 8.Step 3 repeated

Table 3 Use case description for Cancel Appointment

Use Case Name	**Cancel Appointment**
Actors	Customer
Description	Customer can edit an appointment
Pre-Condition	1.Customer must be logged into the system 2.Customer should have an appointment .to edit
Post-Condition	1. Update data stored in database 2.Customer receive confirmation 3.CU can view and manage appointment

Normal Flow	1.Customer goto their personnel scheduling 2.The system display the page 3.Customer select the appointment 4.Customer click edit button 5.Customer enter necessary information 6.The system update the database 7.The use case ends.
Alternative	

Table 4 Use case description for make Payment

Use Case Name	**Make Payment**
Actors	Customer
Description	Customer pay in order to get service and product
Pre-Condition	1. Customer must be logged into the system 2. Customer should have credit card
Post-Condition	1. Payment is successfully processed and authorized. 2. Customer receive confirmation

Normal Flow	1. Customer select service, product 2. The system present total of selected services, product 3. Customer click checkout button 4. System display the payment form 5. System validates the entered payment details 6. If the payment details are valid, the system proceeds to the next step. 7. System updates the transaction status as "Paid" 8. The use case ends.
Alternative	-If the user cancels the payment before completing the transaction, the system will return to the previous screen -If any validation errors occur during the input of payment details in step 6, The system will display error messages to the use

Table 6 Use case description for Rating

Use Case Name	**Rate**
Actors	Customer
Description	Customer can rate professional

Pre-Condition	1.Customer must be logged into the system 2.The customer should interact with the professional that can be rated 3.System has to allow user to rate the professional
Post-Condition	1.The user successfully rated the professional 2.System record rating
Normal Flow	1.Customer go to rating page 2.Customer select specific professional they want to rate 3.The system present rating system 4. System record rating 5. The use case ends.
Alternative	If the user has already submitted a rating for the item or experience, the system may display a message indicating that only one rating per user is allowed, preventing duplicate submissions.

Table 7 Use case description for Add to cart

Use Case Name	**Add to Cart**
Actors	Customer
Description	Customer can select product and add them to their cart

Pre-Condition	1.Customer must be logged into the system 2. Customer has to access the product
Post-Condition	1.Customer added product to cart 2.Customer can proceed to checkout 3.User can modify cart
Main Flow	1.User go to product page 2.View product 3.Select specific product 4.Click add to cart button 5.The product added to cart 6.User can modify quantity 7.Use case ends
Alternative	If the user try to add product that are already in cart,the system display message .

Table 8 Use case description for view cart

Use Case Name	View Cart

Actors	Customer
Description	Customer can view their cart
Pre-Condition	1. Customer must be logged into the system 2. Customer has should have product that added to cart
Post-Condition	1. Customer viewed the product in their cart 2. Customer can proceed to checkout 3. User can modify quantity or remove product
Main Flow	1. User go to cart page 2. View product 3. Customer can remove product 4. Customer can modify quantity 5.Use case ends
Alternative	

Table 9 Use case description for viewing schedule

Use Case Name	**View Schedule**
Actors	Employee

Description	Employee can view personnel schedule
Pre-Condition	1.Employee must be logged into the system 2. Employee should have schedule
Post-Condition	.Employee viewed their schedule
Normal Flow	1.Employee go to their schedule page 2.View schedule 3.Use case ends
Alternative	

Table 10 Use case description for Managing service

Use Case Name	**Manage service**
Actors	Admin
Description	Admin can update,add and delete service

Pre-Condition	Admin must be logged into the system
Post-Condition	Admin successfully managed service

Normal Flow	1. Admin go to service management page 2. System present list of service 3. Admin select specific service to manage 4. For Adding service A. Admin provide necessary information to add new service B. System validate the information C. If the information is valid,system add new service D. System provide confirmation message 5. For deleting service A. Admin select service they want to delete B. Admin click delete button C. System delete service from service catalog D. System provide confirmation message 6. For updating service A. Admin select service they want to update B.system present form with selected service information C. Admin modify the necessary information of service D. If the information is valid,system save change and update service information E. System provide confirmation message 7. The use case ends
Alternative	

Table 11 Use case description for managing Employee

Use Case Name	**Manage employee**
Actors	Admin
Description	Admin can update,add and delete employee
Pre-Condition	Admin must be logged into the system
Post-Condition	Admin successfully managed employee
Normal Flow	1. Admin go to employee management page 2. System present list of employee if there is 3. Admin select specific employee to manage 4. For Adding employee A. Admin provide necessary information to add new employee B. System validate the information C. If the information is valid,system add new employee D. System provide confirmation message 5. For deleting employee A. Admin select employee he want to delete B. Admin click delete button C. System delete employee from employee catalog

	D. System provide confirmation message 6. For updating employee A. Admin select employee he want to update B. System present form with selected employee information C. Admin modify the necessary information of service E. If the information is valid,system save change and update service information D. System provide confirmation message 7. For Viewing employee A. Admin select employee he want to view details B. System displays the employee's details 8.The use case ends
Alternative	

Table 12 Use case description for managing product

Use Case Name	**Manage Product**
Actors	Admin
Description	Admin can update,add and delete Product
Pre-Condition	Admin must be logged into the system

Post-Condition	Admin successfully managed Product
Normal Flow	1. Admin go to Product management page 2. System present list of Product if there is 3. Admin select specific Product to manage 4. For Adding Product A. Admin provide necessary information to add new Product B. System validate the information B. If the information is valid,system add new Product C. System provide confirmation message 5. For deleting Product A. Admin select Product he want to delete B. Admin click delete button C. System delete Product from employee catalog D. System provide confirmation message 6. For updating Product A. Admin select Product he want to update B. System present form with selected Product information C. Admin modify the necessary information of service E. If the information is valid,system save change and update service information D. System provide confirmation message 7. For Viewing Product

	A. Admin select Product he want to view details B. System displays the Product details 8.The use case ends
Alternative	

4.1.1.3. Use Case Scenario

1.Scenario Name: Login

Participant actor:Ms.Ahlam

Normal Flow of Event

- ✓ Ahlam Should have account to Login
- ✓ System should display login page
- ✓ Ahlam insert her username and password
- ✓ Ahlam click login button
- ✓ System check the ahlam information is correct
- ✓ If the information is correct,ahlam redirected to homepage
- ✓ Use case end

If the information entered are invalid or incorrect.system display messages that indicate the username and password are incorrect

2.Scenario Name: Make appointment

Participant actor:Ms.Aziza

Normal Flow of Event

- ✓ Aziza login into the system
- ✓ Aziza select service she want to make appointment
- ✓ System display appointment page

- ✓ Aziza enter necessary information
- ✓ System validate the information
- ✓ If the information is correct ,system save the information and display confirmation message
- ✓ Case end

3.Scenario Name: Payment

Participant actor:Ms.Sarah

Normal Flow of Event

- ✓ Sarah login into the system
- ✓ Select service she want to use
- ✓ System display checkout page
- ✓ Sarah enter the required information for online payment such as credit number
- ✓ System validate the information
- ✓ If the information is valid,system display confirmation message
- ✓ Use case end

If the information entered is invalid the system display error message that indicating the data is incorrect

4.Scenario Name: Manage Service

Participant actor:Mr.Alhamdu

Normal Flow of Event

- ✓ Alhamdu should have admin account to manage service
- ✓ Alhamdu go to service management page
- ✓ System present list of service
- ✓ Alhamdu select specific service to manage
- ➢ If Alhamdu want to Add service

A. Alhamdu provide necessary information to add new service

B. System validate the information

C. If the information is valid,system add new service

D. System provide confirmation message

- If Alhamdu want to Delete service

A. Alhamdu select service they want to delete

B. Alhamdu click delete button

C. System delete service from service catalog

D. System provide confirmation message

- If Alhamdu want to Delete service

A. Alhamdu select service they want to update

B. System present form with selected service information

C. Alhamdu modify the necessary information of service

D. If the information is valid,system save change and update service information
E. System provide confirmation message

- The use case ends

What if Temesgen provides invalid information when managing a service?

System display error message the help alhamdu to provide valid information

5.Scenario Name : Manage Product

Participant actor:Mr.Temesgen
Normal Flow of Event

- ✓ Temesgen should have admin account to manage service
- ✓ Temesgen go to Product management page
- ✓ System present list of Product if Available
- ✓ Temesgen select specific service to manage
- ✓ If Temesgen want to Add Product

A. Temesgen provide necessary information to add new product

B. System validate the information

C. If the information is valid,system add new product

D. System provide confirmation message

- If Temesgen want to Delete product

A.Temesgen select product they want to delete

B.Temesgen click delete button

C.System delete product from service catalog

D.System provide confirmation message

- If Temesgen want to Delete product

A.Temesgen select product they want to update

B.System present form with selected product information

C.Temesgen modify the necessary information of service

D.If the information is valid,system save change and update product information

E.System provide confirmation message

- If Temesgen want to view Product

A.Temesgen select Product he want to view details

B.System displays the Product details

- The use case ends

What if temesgen provides invalid information when managing a product?

System display error message the help temesgen to provide valid information

4.2. Object Model

An object model serves as a logical representation of the system using object-oriented techniques. It helps to create a software or system model, enabling architectural planning before the development or programming phase. In this system, we utilize object models, including class diagrams and data dictionaries, to effectively depict and organize the different components and relationships specific to your project. These models provide a visual representation of the structure and behavior of the system, aiding in the understanding and design of the software or system

4.2.5. Class Diagram

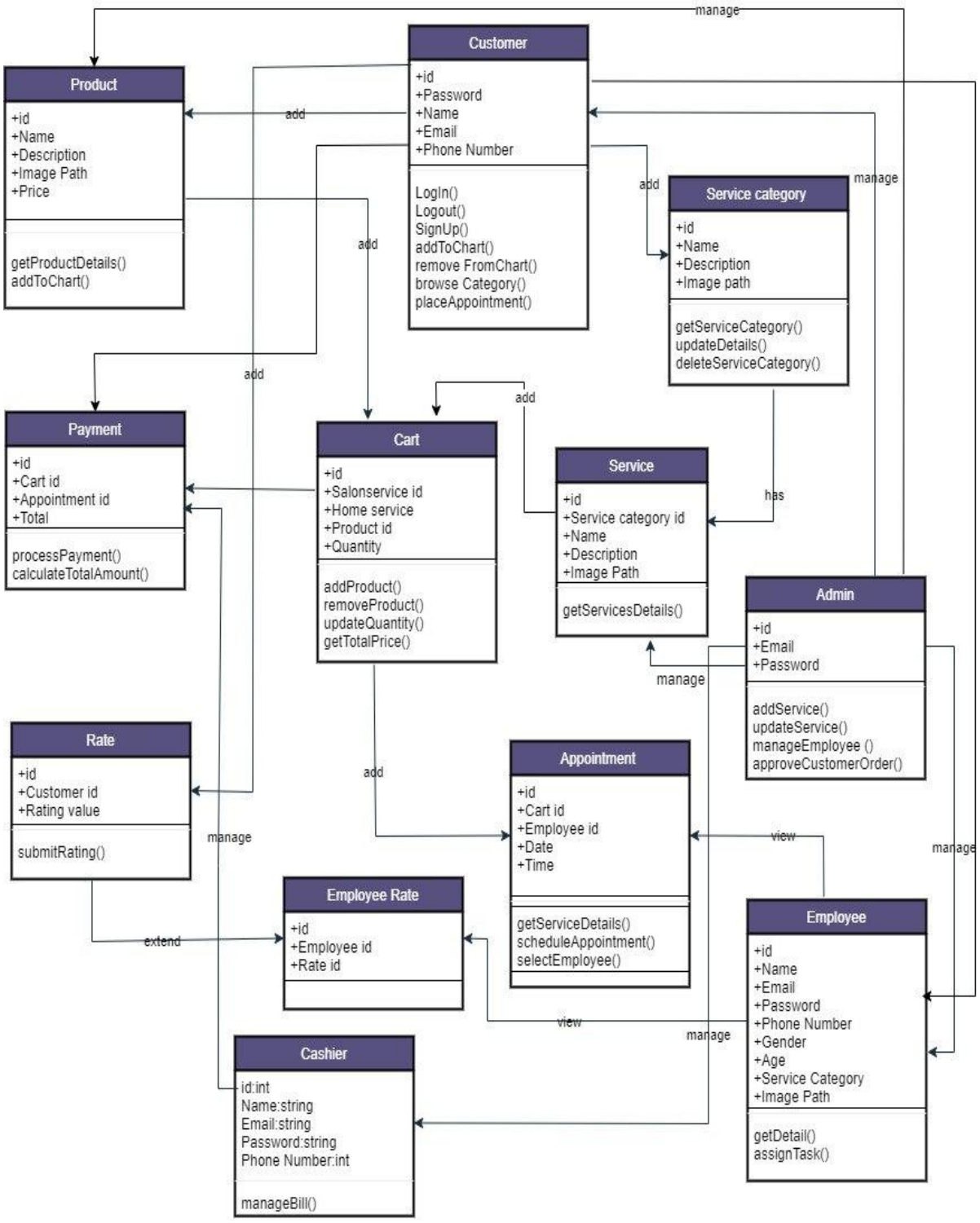


Figure 2 Class diagram

4.2.2. Data Dictionary

In our project, a data dictionary refers to a collection of files that contain metadata specific to our system. It provides a comprehensive record of various objects and their attributes within the system's database. The data dictionary includes information such as data ownership, data relationships between objects, and other relevant data elements. It serves as a valuable resource for understanding the structure and organization of data within the system, facilitating effective database management and ensuring accurate data retrieval and manipulation**.**

Table 13 Data dictionary for customer

Attributes	**Data type**	**Data size**	**Key constraints**
ID	int	4	Primary Key
Name	String	50	Not Null
Email	Varchar	50	Not Null
Phone Number	int	10	Not Null
Password	Varchar	8	Not Null

Table 14 Data dictionary for Product

Attributes	**Data type**	**Data size**	**Key constraints**
id	int	4	Primary Key
Customer id	int	4	Foregin Key
Product Name	Varchar	50	Not Null
Description	Varchar	200	Not Null
Price	Double	17	Not Null
Image Path	String	256	Not Null

Table 15 Data dictionary for Service category

Attributes	**Data type**	**Data size**	**Key constraints**
id	int	4	Primary Key
Customer id	int	4	Foregin Key
Description	Varchar	200	Not Null

Image Path	String	256	Not Null

Table 16 Data dictionary for Service

Attributes	**Data type**	**Data size**	**Key constraints**
id	int	4	Primary Key
Category id	int	4	Foregin Key
Service Name	Varchar	50	Not Null
Description	Varchar	200	Not Null
Price	Double	17	Not Null
Image Path	String	256	Not Null

Table 17 Data dictionary for Employee

Attributes	**Data type**	**Data size**	**Key constraints**
ID	int	4	Primary Key

Name	String	50	Not Null
Email	Varchar	50	Not Null
Phone Number	int	10	Not Null
Password	Varchar	8	Not Null
Gender	String	10	Not Null
Age	int	2	Not Null

Table 18 Data dictionary for Admin

Attributes	**Data type**	**Data size**	**Key constraints**
ID	int	4	Primary Key
Email	Varchar	50	Not Null
Password	Varchar	8	Not Null

4.3.Dynamic Model

A dynamic model is utilized to represent the evolving behavior of objects or the system over time. This type of model effectively captures the sequence of states and actions that occur within the system. To depict and analyze this behavior, we employ various graphical tools such as sequence diagrams, activity diagrams, and state diagrams. These diagrams provide visual representations of the interactions, workflows, and states within the system.

4.3.1. Sequence diagram

Sequence diagrams are interaction diagrams that illustrate the ordering of messages according to time. In the context of our proposed system, we have created sequence diagrams to showcase the major functionalities, offering a comprehensive understanding of the sequential message exchanges between relevant entities.

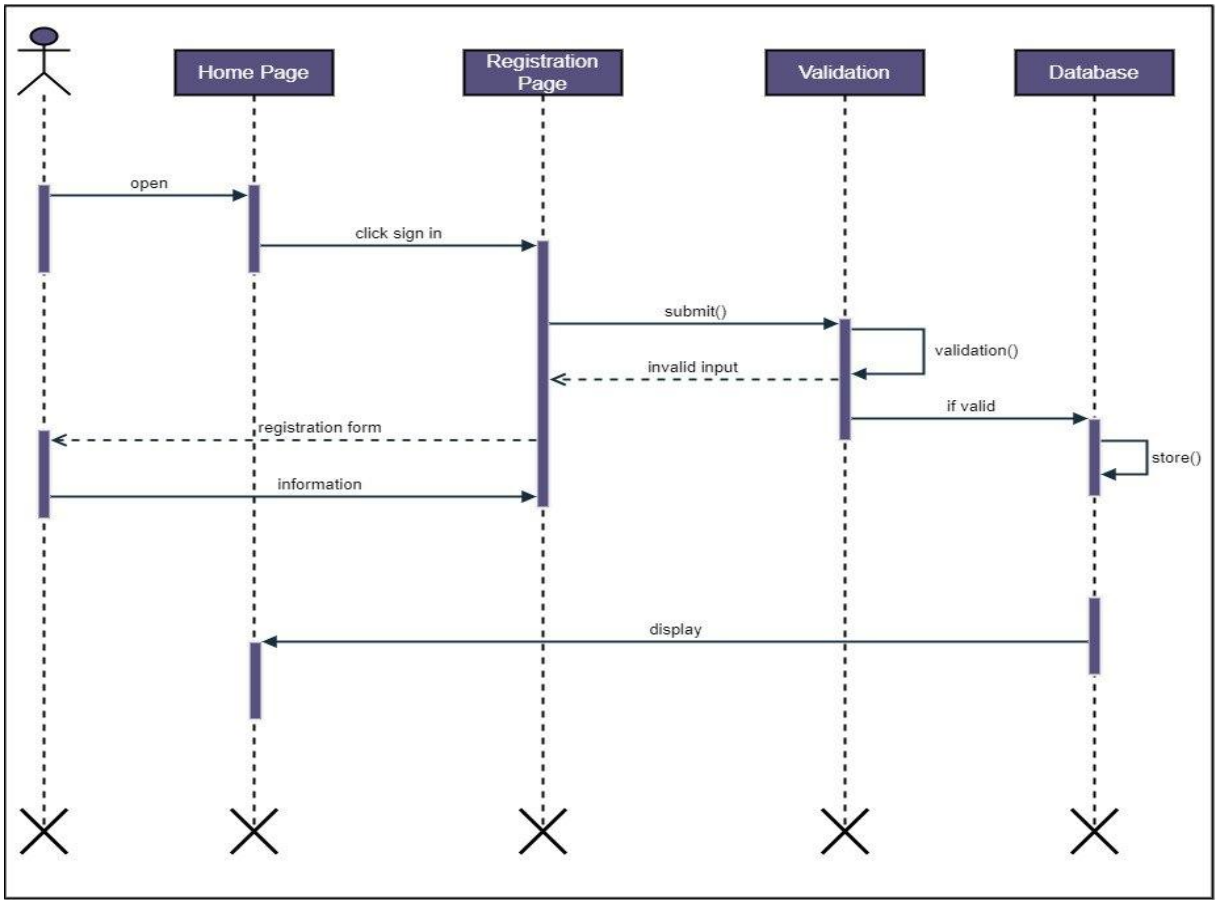


Figure 3 Register in Sequence Diagram

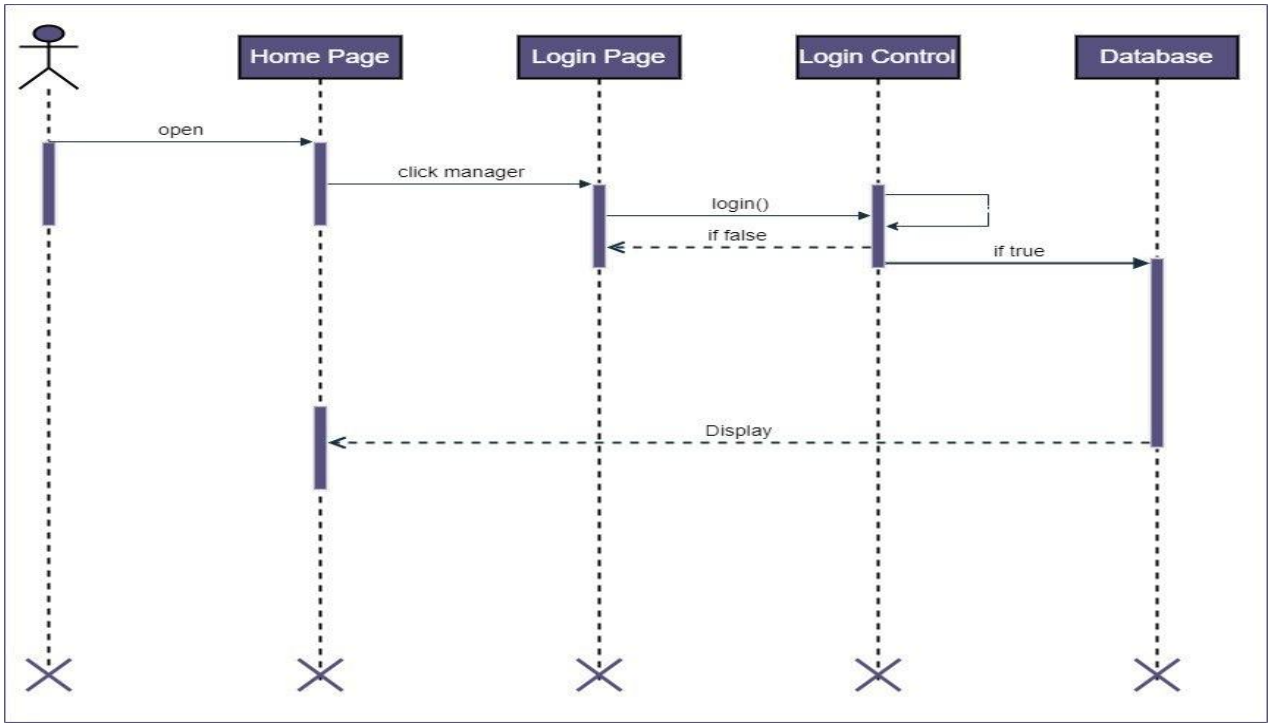


Figure 4 Login in Sequence Diagram

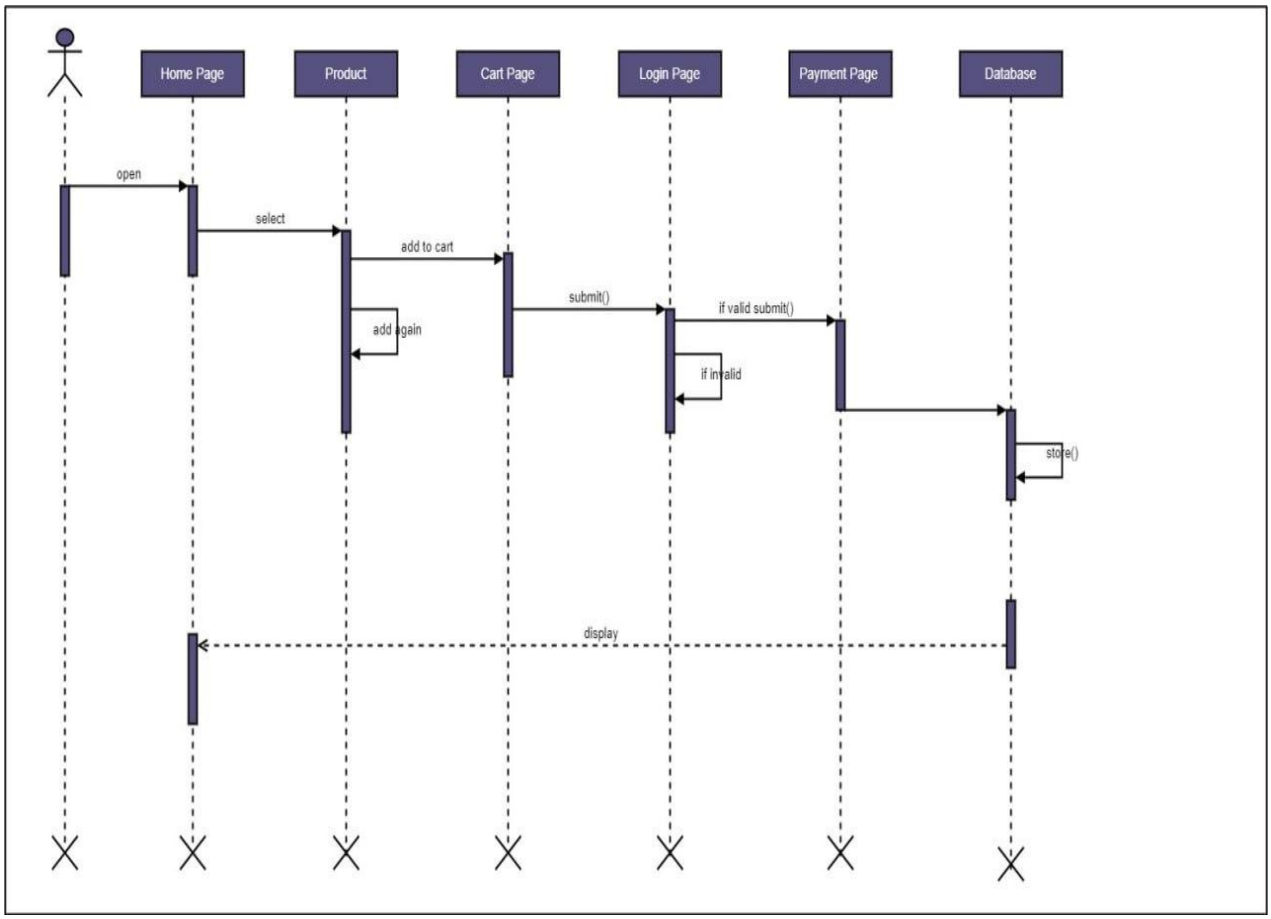


Figure 5 Sell Flow in Sequence Diagram

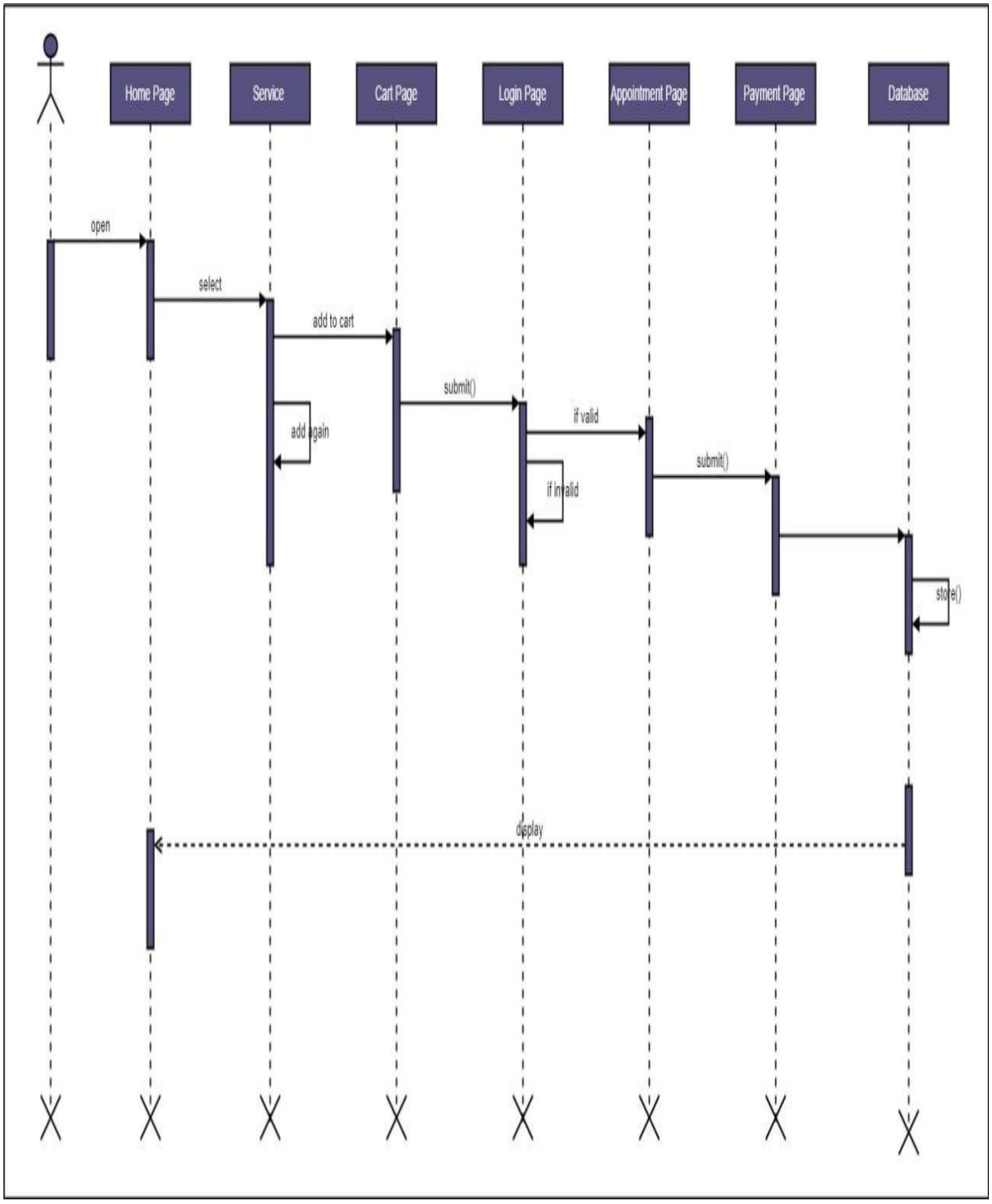


Figure 6 Appointment flow in Sequence Diagram

4.3.2. Activity diagram

Activity diagrams are dynamic models that depict the flow of control within a system, showcasing the progression from one activity to another.activity diagrams provide a concise and precise visual representation of the flow of control in a system, highlighting the activities performed and their relationships.

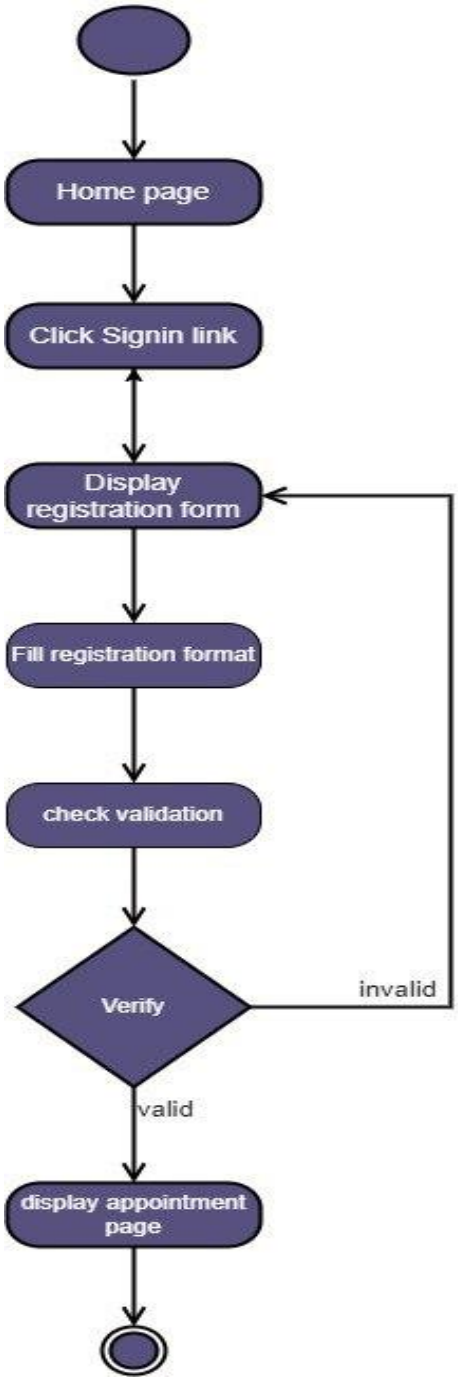


Figure 7 Activity diagram for Registration

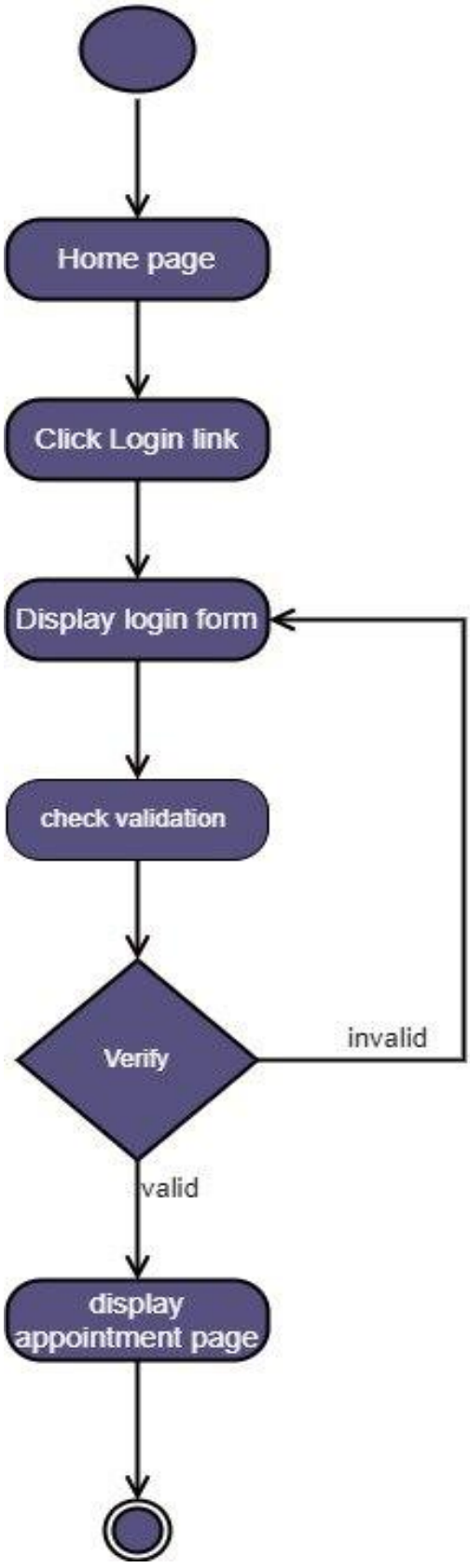


Figure 8 Activity diagram for Login

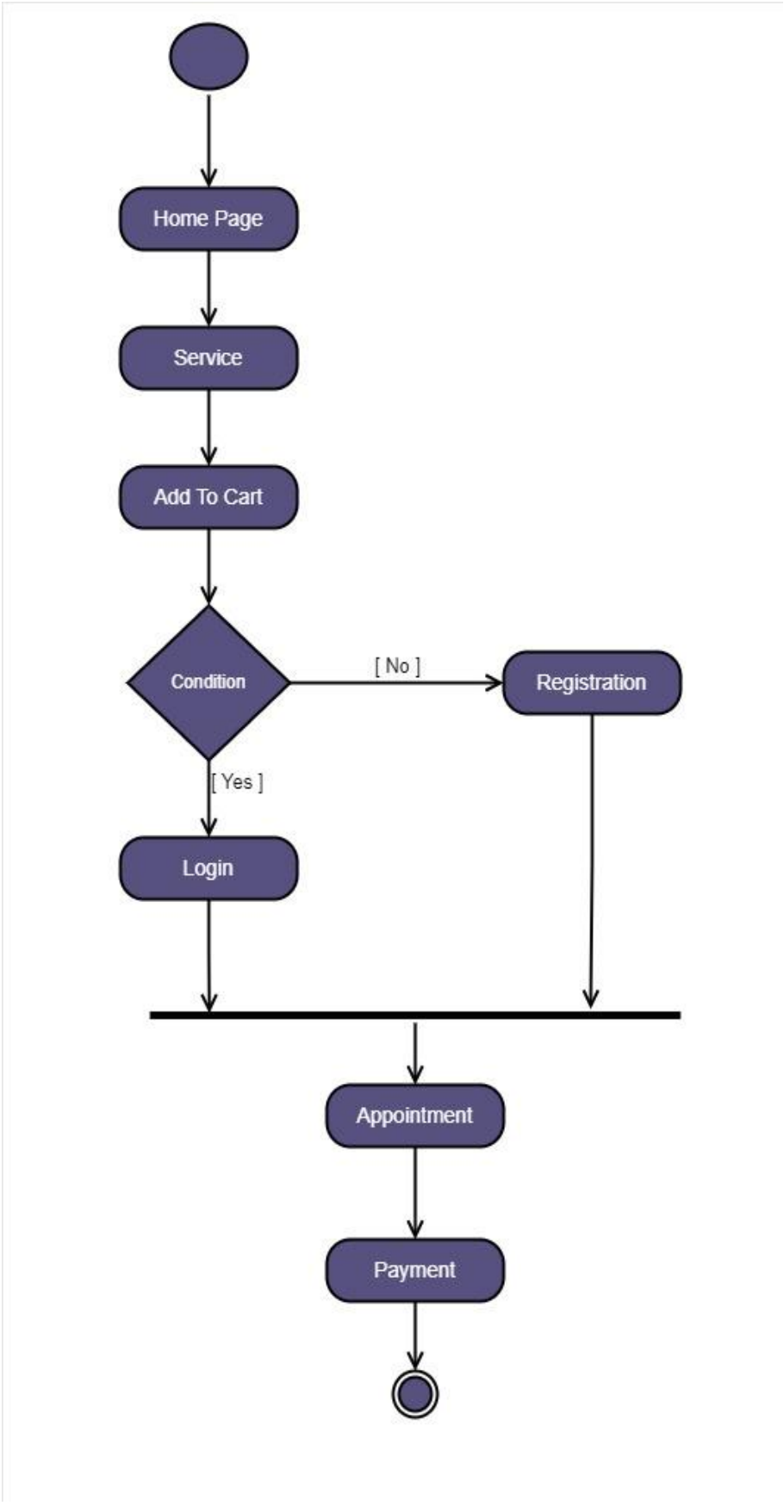


Figure 9 Activity diagram for Service

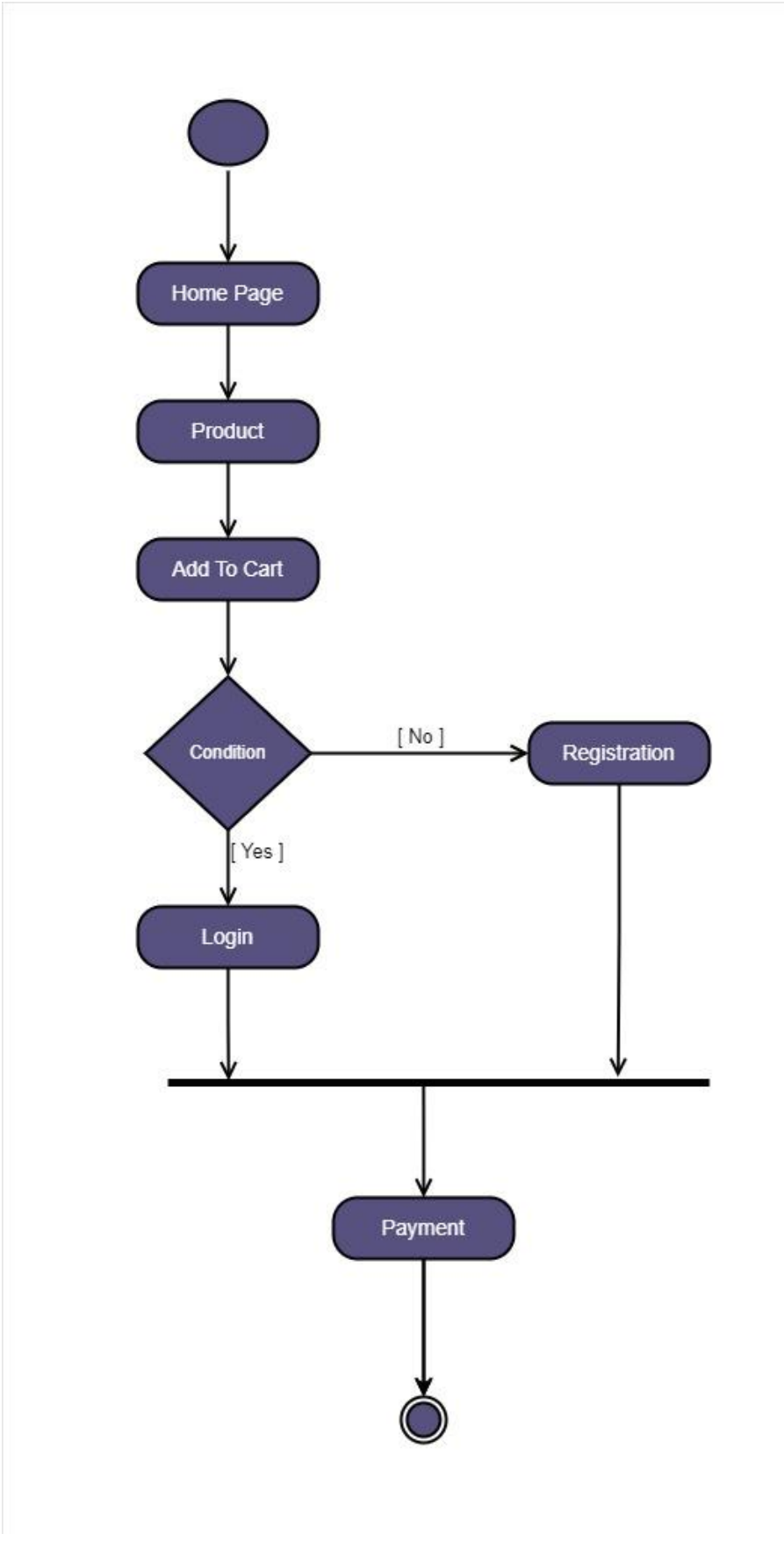


Figure 10 Activity diagram for Product

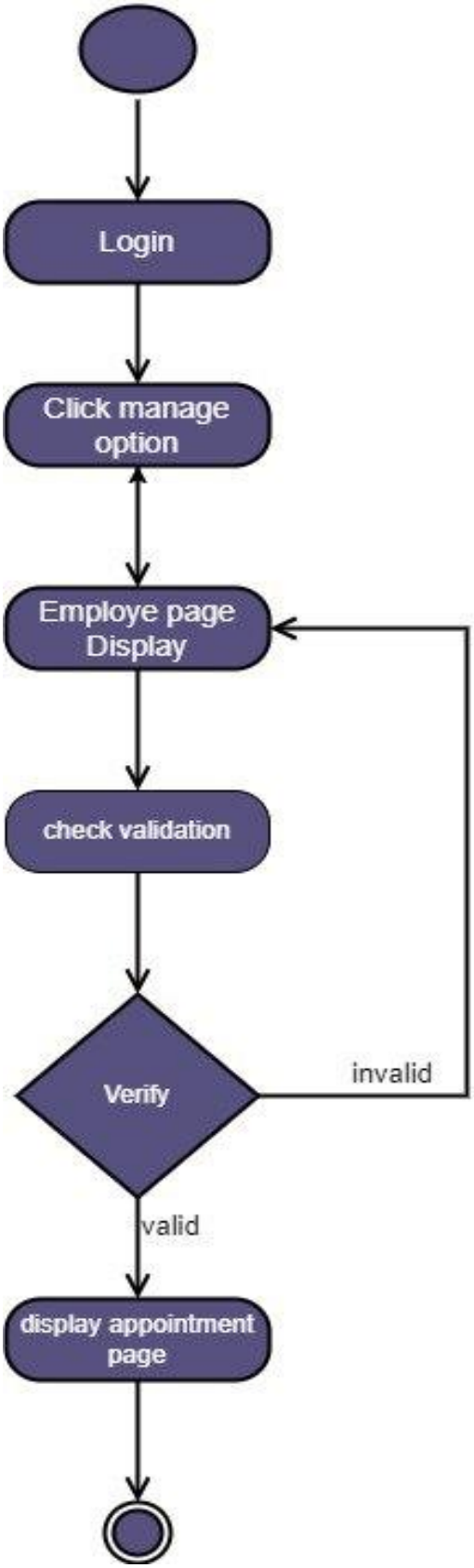


Figure 11 Activity diagram for Employee

4.3.2. State Diagram

A state-chart diagram shows a state machine that depicts the control flow of an object from one state to another. It is used for modeling objects which are reactive in nature. The following diagrams shows the reactive nature of reservation object when confirmation is made and reservation times' out.

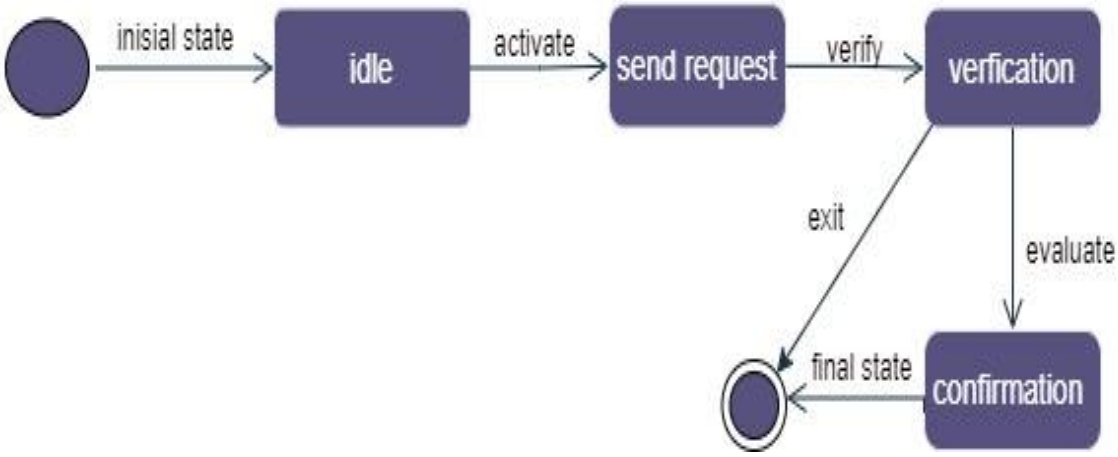


Figure 12 State chart diagram for Registration

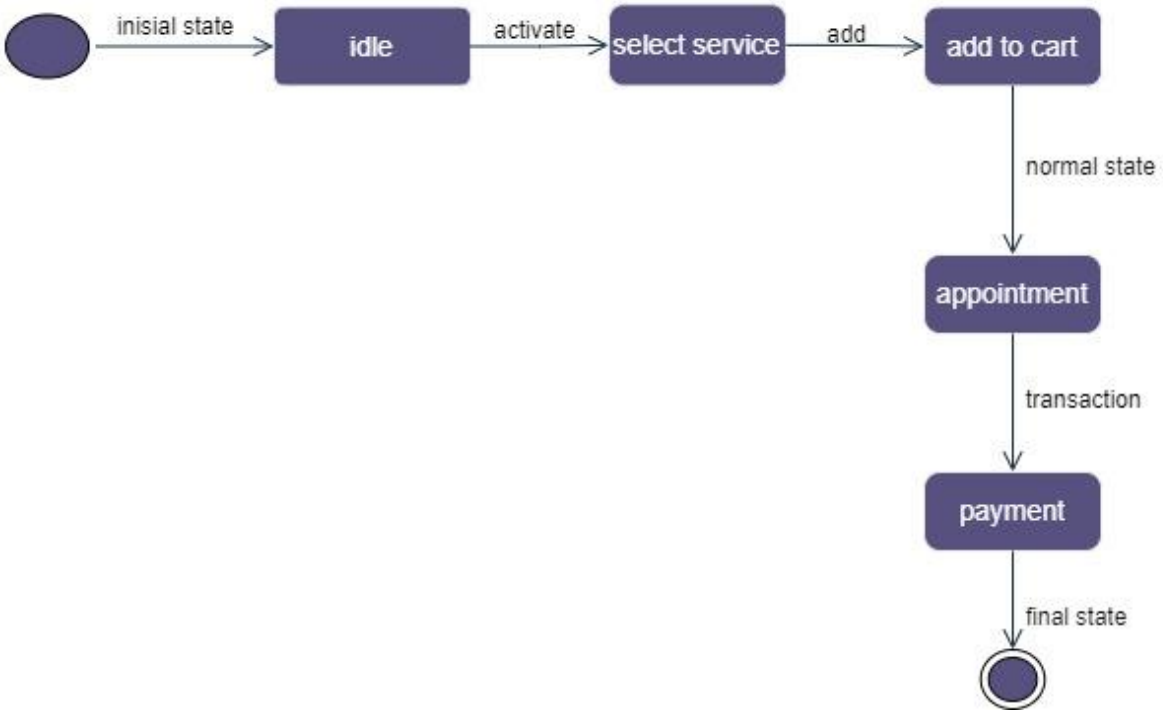


Figure 13 State chart diagram for Service

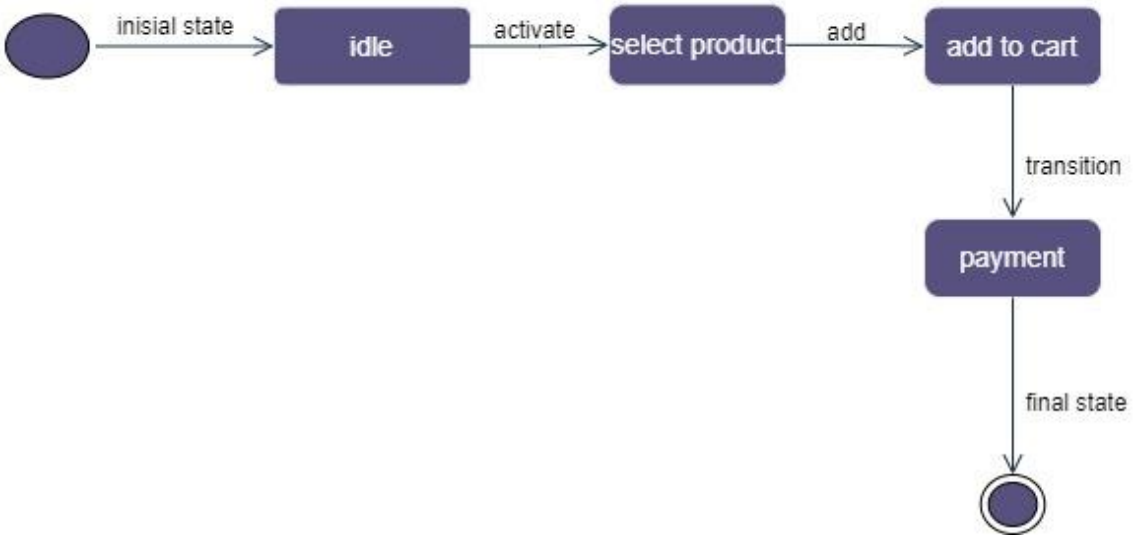


Figure 14 State chart diagram for Product

CHAPTER FIVE

5. SYSTEM DESIGN

This chapter provides a brief overview of the design goals, current and proposed software architecture, Hardware/software mapping, Persistent data management, and Access control and security.

5.1 Design Goal

Design goals outline the desired qualities and objectives that should be achieved and addressed during the design phase of a system. These goals serve as guidelines to ensure that the system meets the intended requirements and functions effectively.

1. User Interface and Human Factors:

▪ Develop a user-friendly and intuitive interface that is easy to navigate and understand.

▪ Consider usability principles to ensure a smooth and efficient user experience.

2. Hardware Consideration:

▪ Design the system to be compatible with different hardware configurations, such as desktop computers, laptops, tablets, and smartphones.

▪ Optimize resource usage to ensure efficient performance on various hardware platforms.

3. Security Issues:

▪ Implement authentication mechanisms to ensure secure access to the system.

▪ Use a combination of username and password for user authentication.

▪ Implement password complexity requirements, such as a minimum length.

4. Performance Consideration:

▪ Design the system to handle a significant number of concurrent users and process service bookings efficiently.

▪ Optimize database queries and system components to minimize response times and maximize system performance.

5. Error Handling and Validation:

▪ Perform input validation to prevent invalid or malicious data from compromising the system's integrity and security.

6. Quality Issues:

▪ Adhere to coding standards and best practices to ensure high-quality, maintainable code.

▪ Conduct thorough testing, including functional, integration, and performance testing, to identify and rectify any defects.

7. Documentation:

▪ Prepare comprehensive documentation, including system architecture, design decisions, and user manuals, to facilitate system understanding, maintenance, and future enhancements.

5.2. Current System Architecture

Current system architecture describes the architecture of the system being replaced

1. User Interaction:

- Users access the salon's website to browse services and products.

2. Booking Process:

- Users call the provided phone number to book an appointment with the salon.
- The salon staff manually records the booking details, such as the customer's name, requested service, date, and time.

3. Check-In Process:

- Customers arrive at the salon at their scheduled appointment time.
- They proceed to the reception desk or designated area to announce their arrival and check-in for their appointment.

4. Service Categories:

▪ The salon offers various service categories, including haircuts, hair dyeing, hair braiding, makeup, nail design, pedicure, manicure, and eyelash extensions.

5. Product Sales:

▪ The salon sells hair products, eyelashes, human hair, and jewelry on its website.

▪ The details regarding product specifications are not specified.

6. Payment and Checkout:

▪ After receiving the desired services, customers proceed to the checkout counter.

▪ The salon staff calculates the total cost based on the services received, any additional products purchased, and applicable taxes.

▪ The customer settles the bill by making payment.

7. Website:

▪ The salon has a website that showcases its services and product offerings.

▪ However, the booking and sales processes are conducted either through phone calls or in person at the salon.

5.3. Proposed System Architecture

This section presents a general view of your system architecture and briefly describes the assignment of functionality to each subsystem.\

5.3.1. Subsystem Decomposition and Description

Subsystem decomposition draws and describes the decomposition into subsystems and the responsibilities of each the main product of system design.

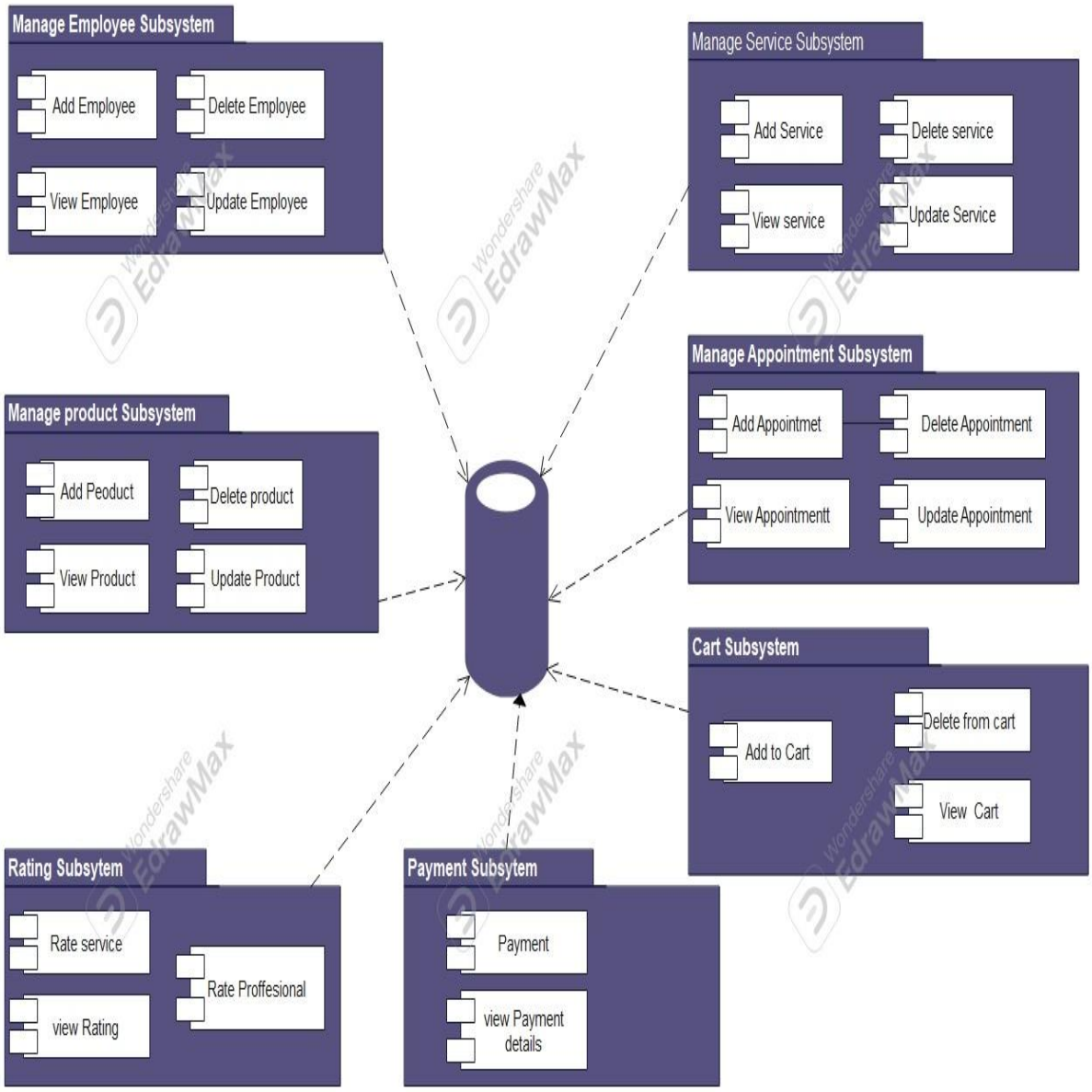


Figure 15 Subsystem decomposition and description

5.3.2. Hardware/Software Mapping

Hardware/software mapping describes how subsystems are assigned to hardware and customized components .

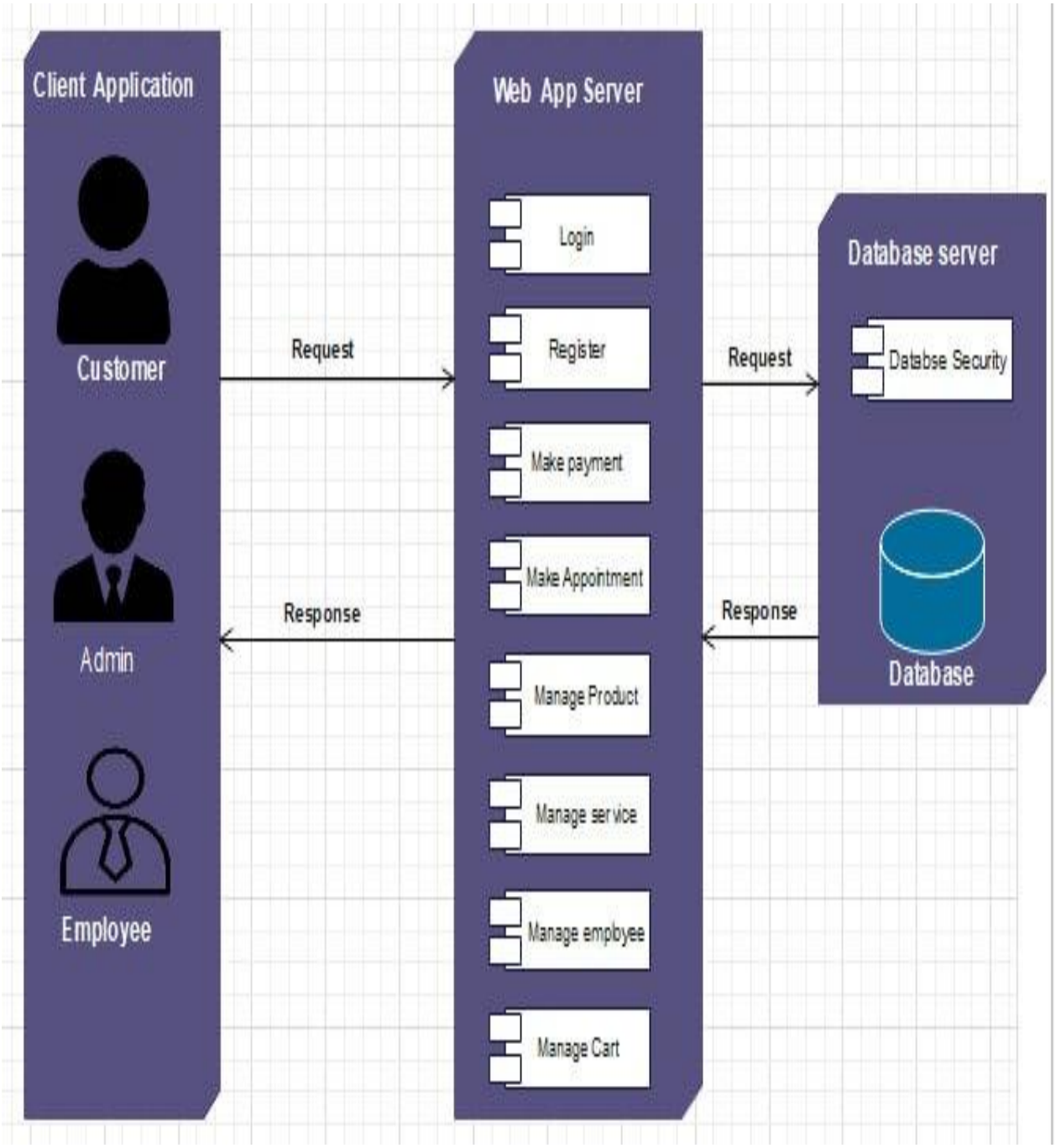


Figure 16 Hardware/Software mapping diagram

5.3.4. Detailed Class Diagram

In this section show, classes, attributes, methods, attribute data types, visibility ((Public (+), Private (-), Protected (#)) of attributes and methods), inheritance, association, aggregation, composition, dependency, and municipality (cardinality and optimality).

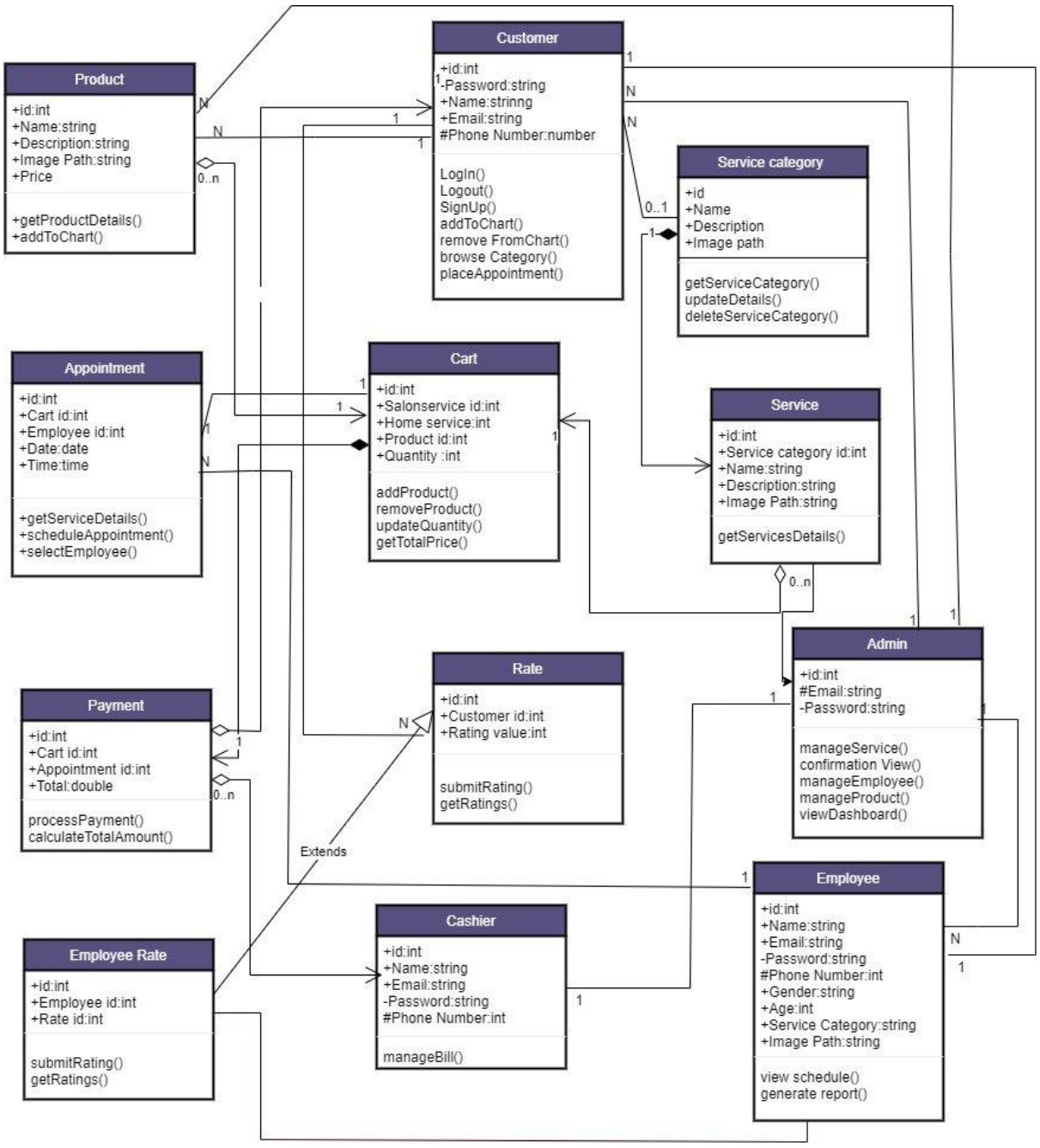


Figure 17 Detailed class diagram

5.3.5. Persistent Data Management

Persistent data management describes the persistent data stored by the system and the data management infrastructure required for it. This section typically includes the description of data schemes, the selection of database, and the description of the encapsulation of the

database.

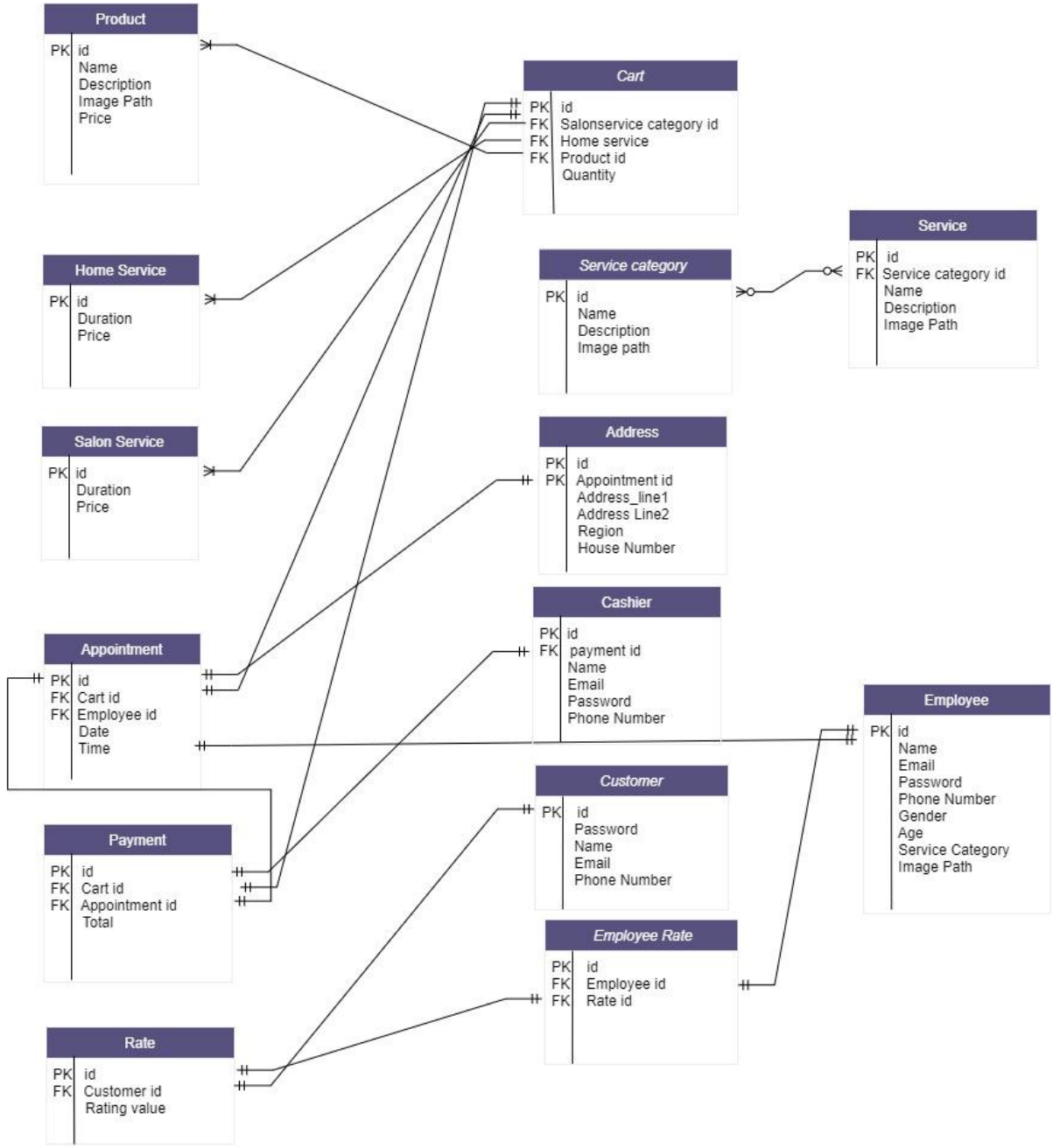


Figure 18 Persistent data management diagram

5.3.6. Access Control and Security

Access control and security describes the user model of the system in terms of an access privilege. Utilize global access table, describing the access relation between the actors, objects and operations in the system.

- **Access Control Model**:

Actors: Customers, Service Professionals, Salon Administrators.

Objects: Customer Information, Service Listings, Appointment Schedule, Payment Data.

Operations: View Services, Add to Cart, Select Service Professional, Schedule Appointment, Make Payment, View Appointments.

The access control model can be represented using a table:

Table 19 Access Control Model

Actor	Object	Operations
Customers	Service Listings	View
Customers	Cart	Add to Cart
Customers	Service Professionals	View, Select
Customers	Appointment Schedule	View, Schedule
Customers	Payment Data	Make Payment
Service Professionals	Appointment Schedule	View

Salon Administrators	Customer Information	View
Salon Administrators	Service Professionals	View, Add, Update, Delete
Salon Administrators	Appointment Schedule	View, Update
Salon Administrators	Payment Data	View

5.4. Packages

This section describes the decomposition of subsystems into packages and the file organization of the code. This includes an overview of each package, its dependencies with other packages, and its expected usage.

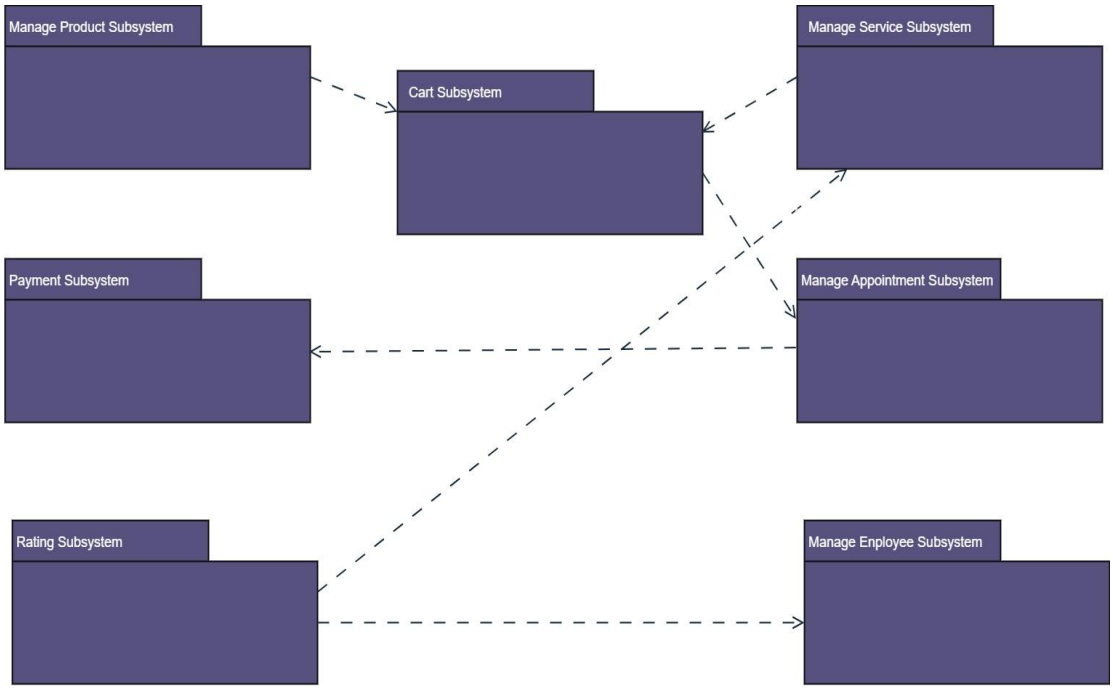


Figure 19 Package diagram

5.5. Algorithm Design

For the elements found in architectural design define the algorithm required for each element to accomplish its tasks.

Cart Subsystem:

1. Initialize an empty cart

function addToCart(item):

if item is available:

add item to cart

display "Item added to cart successfully."

else:

display "Item is not available."

function viewCart():

if cart is not empty:

display items in cart

else:

display "Cart is empty."

function deleteCart():

empty the cart

display "Cart deleted successfully.

Manage Product Subsystem:

1. Initialize an empty list to store products

function addProduct(product):

add product to the list of products

display "Product added successfully."

function deleteProduct(productId):

find the product with the specified productId in the list of products

if product is found:

remove the product from the list of products

display "Product deleted successfully."

else:

display "Product not found."

function updateProduct(productId, newDetails):

find the product with the specified productId in the list of products

if product is found:

update the product with the new details provided

display "Product updated successfully."

else:

display "Product not found."

function viewProduct(productId):

find the product with the specified productId in the list of products

if product is found:

display the details of the product

else:

display "Product not found."

function viewAllProducts():

if list of products is not empty:

display all the products with their details

else:

display "No products available."

Manage Service Subsystem:

1. Initialize an empty list to store services

function addService(service):

add service to the list of services

display "Service added successfully."

function deleteService(serviceId):

find the service with the specified serviceId in the list of services

if service is found:

remove the service from the list of services

display "Service deleted successfully."

else:

display "Service not found."

function updateService(serviceId, newDetails):

find the service with the specified serviceId in the list of services

if service is found:

update the service with the new details provided

display "Service updated successfully."

else:

display "Service not found."

function viewService(serviceId):

find the service with the specified serviceId in the list of services

if service is found:

display the details of the service

else:

display "Service not found."

function viewAllServices():

if list of services is not empty:

display all the services with their details

else:

display "No services available."

Managing Employee Subsystem:

1. Initialize an empty list to store employee records.

Function: AddEmployee(employeeDetails)

Input: employeeDetails (employeeID, name, position, salary, etc.)

Append employeeDetails to the list of employee records.

Display "Employee added successfully."

Function: DeleteEmployee(employeeID)

Input: employeeID

Find the employee with the specified employeeID in the list of employee records.

If employee is found:

Remove the employee from the list of employee records.

Display "Employee deleted successfully."

Else:

Display "Employee not found."

Function: UpdateEmployee(employeeID, updatedDetails)

Input: employeeID, updatedDetails (name, position, salary, etc.)

Find the employee with the specified employeeID in the list of employee records.

If employee is found:

Update the employee with the new details provided in updatedDetails.

Display "Employee updated successfully."

Else:

Display "Employee not found."

Function: ViewEmployee(employeeID)

Input: employeeID

Find the employee with the specified employeeID in the list of employee records.

If employee is found:

Display the details of the employee.

Else:

Display "Employee not found."\

Function: ViewAllEmployees()

If the list of employee records is not empty:

Display all the employee records with their details.

Else:

Display "No employees available."

Rating Subsystem:

1. Initialize an empty list or data structure to store ratings.

Function: RateService(serviceId, rating)

Input: serviceId, rating

Create a new rating object with the provided serviceId and rating.

Add the rating object to the list of ratings.

Display "Service rated successfully."

Function: ViewServiceRating(serviceId)

Input: serviceId

Find all ratings associated with the specified serviceId in the list of ratings.

If ratings are found:

Display the average rating for the service.

Else:

Display "No ratings available for this service."

Function: RateProfessional(professionalId, rating)

Input: professionalId, rating

Create a new rating object with the provided professionalId and rating.

Add the rating object to the list of ratings.

Display "Professional rated successfully."

Function: ViewProfessionalRating(professionalId)

Input: professionalId

Find all ratings associated with the specified professionalId in the list of ratings.

If ratings are found:

Display the average rating for the professional.

Else:

Display "No ratings available for this professional."

Payment Subsystem:

1. Initialize an empty list or data structure to store payment details.

Function: MakePayment(paymentDetails)

Input: paymentDetails (amount, paymentMethod, etc.)

Validate the paymentDetails to ensure it meets the required criteria.

If the paymentDetails are valid:

Add the paymentDetails to the list of payment details.

Display "Payment successful."

Else:

Display "Invalid payment details."

Function: ViewPaymentDetails(paymentId)

Input: paymentId

Find the payment details with the specified paymentId in the list of payment details.

If payment details are found:

Display the details of the payment (amount, paymentMethod, etc.).

Else:

Display "Payment details not found."

Managing Appointment Subsystem:

1.Initialize an empty list or data structure to store appointments.

Function: AddAppointment(appointmentDetails)

Input: appointmentDetails (appointmentID, date, time, customerID, professionalID, etc.)

Validate the appointmentDetails to ensure it meets the required criteria.

If the appointmentDetails are valid:

Create a new appointment object with the provided details.

Add the appointment object to the list of appointments.

Display "Appointment added successfully."

Else:

Display "Invalid appointment details."

Function: DeleteAppointment(appointmentID)

Input: appointmentID

Find the appointment with the specified appointmentID in the list of appointments.

If appointment is found:

Remove the appointment from the list of appointments.

Display "Appointment deleted successfully."

Else:

Display "Appointment not found."

Function: UpdateAppointment(appointmentID, updatedDetails)

Input: appointmentID, updatedDetails (date, time, customerID, professionalID, etc.)

Find the appointment with the specified appointmentID in the list of appointments.

If appointment is found:

Validate the updatedDetails to ensure it meets the required criteria.

If the updatedDetails are valid:

Update the appointment's information with the provided details.

Display "Appointment updated successfully."

Else:

Display "Invalid updated details."

Else:

Display "Appointment not found."

Function: ViewAppointment(appointmentID)

Input: appointmentID

Find the appointment with the specified appointmentID in the list of appointments.

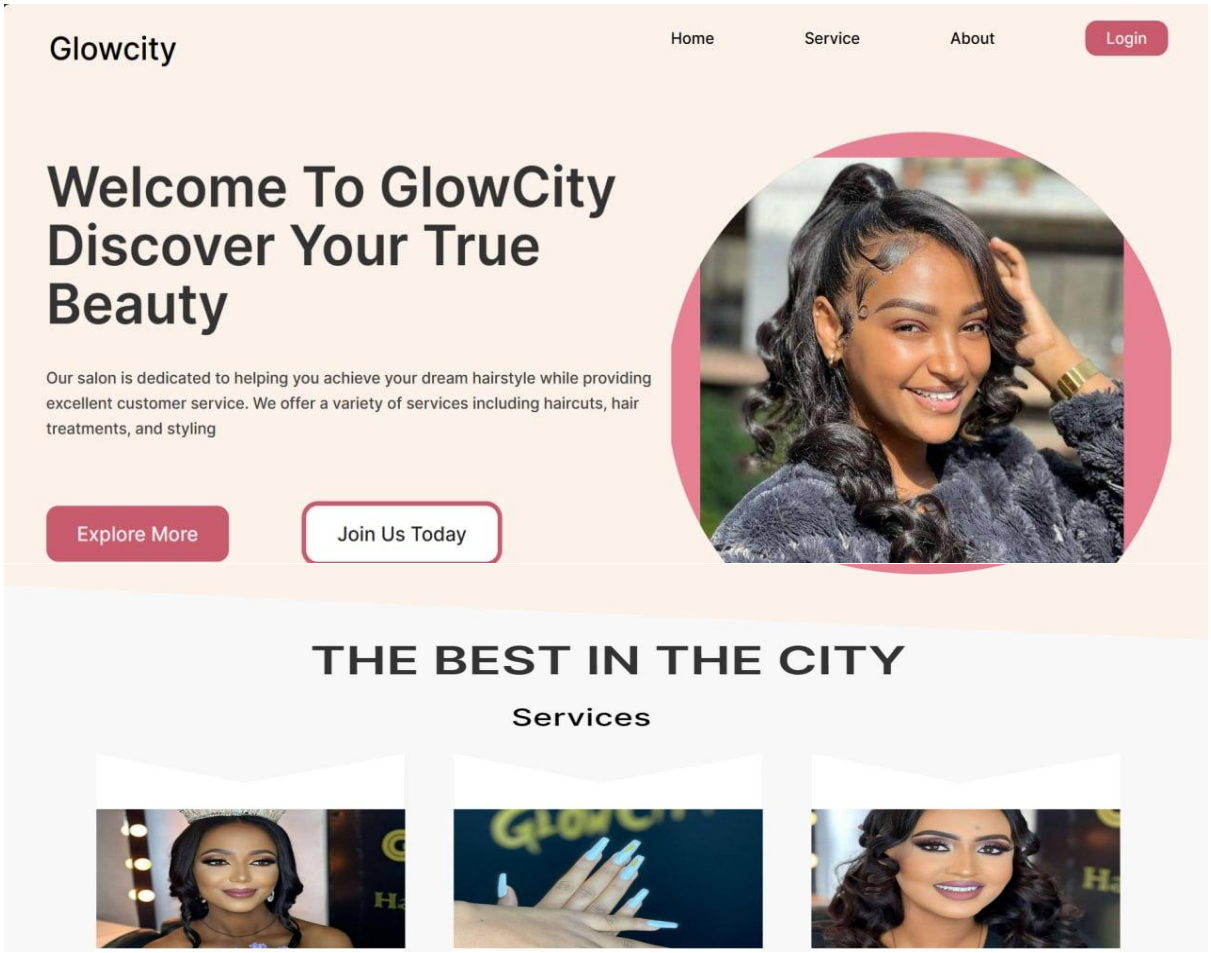
If appointment is found:

Display the details of the appointment.

Else:

Display "Appointment not found."

5.6. User Interface Design



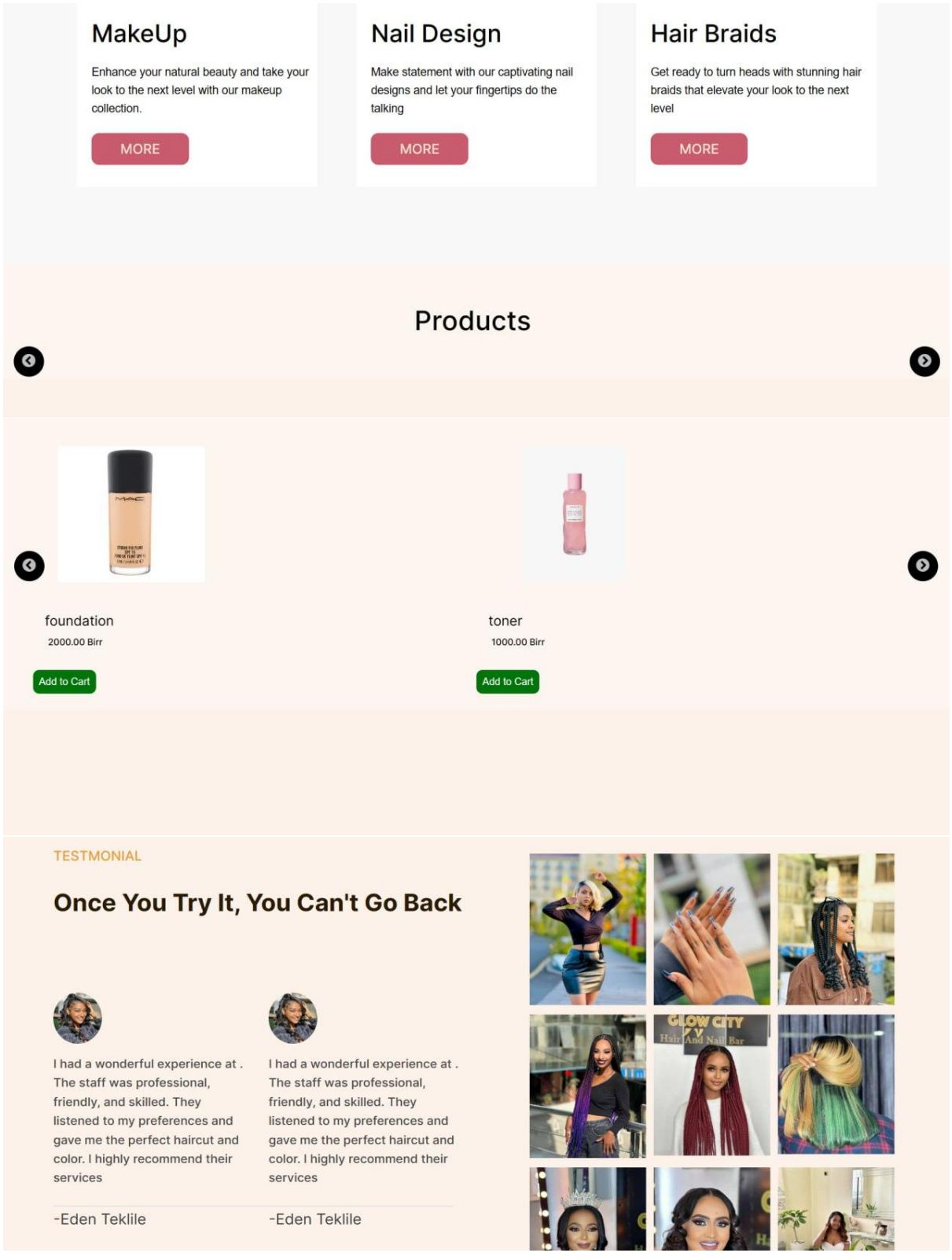


Figure 20 Home Page

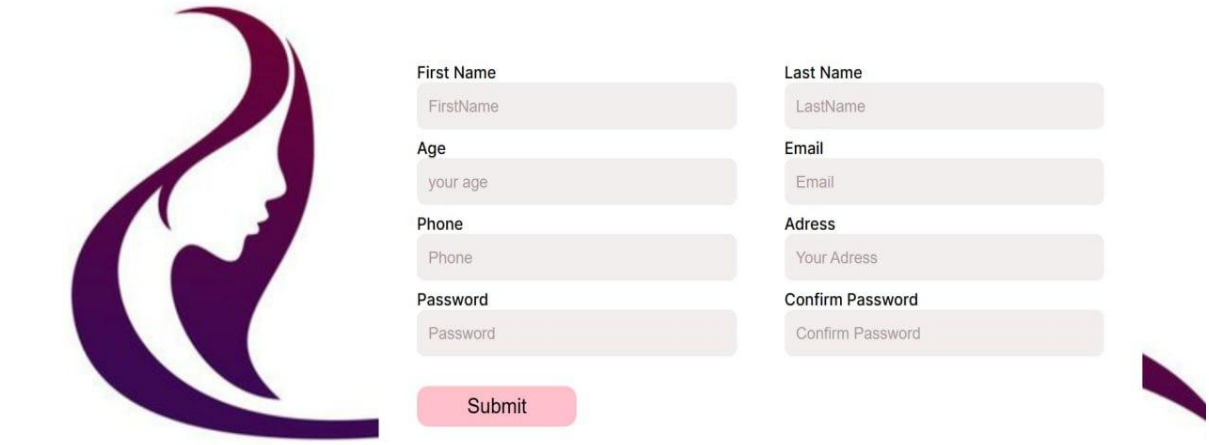


Figure 21 Sign up page

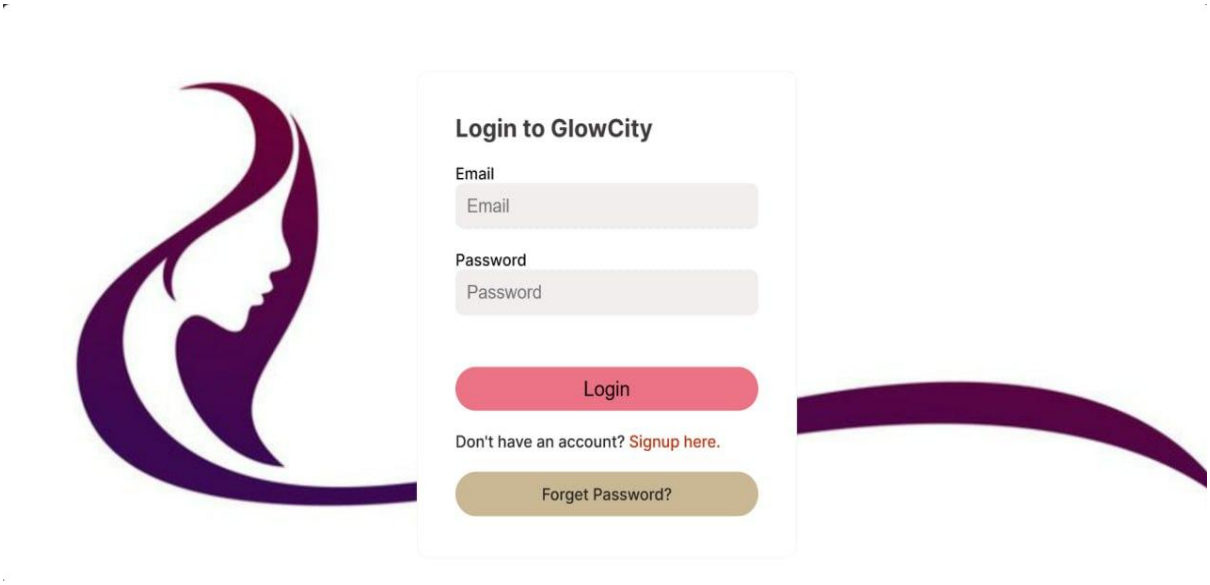


Figure 22 Login page

CHAPTER SIX

IMPLEMENTATION AND TESTING

This chapter basically highlights the issues deal with the implementation phases. Implementation is the phase where objectives of physical operations of the system turned into reality i.e. real working model. In this phase the coding convention has made it possible as it's the real phase of objectivity to reality. Then the code is tested until most of the errors have been detected and corrected. The goal of implementation is to introduce our system for the users in real sense that how they use this new system which is developed for their intended objectives.

6.1. Implementation of the Database

In order to implement the main application language is MYSQL that used to store database values and used to store it for long time. We have used MYSQL database because that can run any operating or browser application,Installation Is Streamlined,Security Features Are Better,Enhanced Performance,It's Important To Maintain An Environment That's Standardized. MYSQL is easy fast and can used for any type of database weather it is relational or simple database, large or small database.so MYSQL server for implementation our system or used to store our data.

6.2. Implementation of the class diagram

In this class diagram implementation we have implement the class diagram structure. That implements the view of an application, visualizing, describing and documenting different aspect of the system. And also implements the attribute and operation of the class.

6.3 Configuration of the Application Server

Configuration of the application server for Node.js involves tasks such as installing Node.js, managing dependencies, and organizing the application's structure. These steps lay the

foundation for the server's optimal operation and meeting the requirements of the Node.js application.

Further configuration tasks include creating server files, configuring middleware, establishing database connections, setting up logging and error handling, optimizing performance, choosing a deployment strategy, and monitoring server health and scalability. These tasks collectively aim to create a stable and efficient environment for running Node.js applications, ensuring their security, performance, and scalability

6.4 Configuration of application security

Our system called online beauty salon management system is an reservation issue validates all the inputs by returning error message and suggesting to try again when invalid input occurred. The system has its security principles that control unauthorized authentication.

Form validation of our system

```
const validateForm = () => {
 let errors = {};
 if (!fname) {
  errors.fname = "First Name is required";
 } else if (!/^[a-zA-Z]+$/.test(fname)) {
  errors.fname = "First Name should only contain letters";
 }
 if (!lname) {
  errors.lname = "Last Name is required";
 } else if (!/^[a-zA-Z]+$/.test(lname)) {
  errors.lname = "Last Name should only contain letters ";
 }
```

```
  if (!email) {
    errors.email = "Email is required";
  } else if (!/^[\w-.]+@([\w-]+\.)+[\w-]{2,4}$/.test(email)) {
    errors.email = "Email is invalid";
  }
  if (!phone) {
    errors.phone = "Phone is required";
  } else if (!phone.match(/^(09|07)\d{8}$/)) {
    errors.phone = "Invalid phone number format";
  }
  if (!age) {
    errors.age = "Age is required";
  } else if (age < 20 || age > 45) {
    errors.age = "Age should be between 20 and 45";
  }
  if (!gender) {
    errors.gender = "Gender is required";
  }
  if (!profession) {
    errors.profession = "Profession is required";
  }
  if (!image) {
```

```
  errors.image = "Employee Image is required";

 }

 setErrors(errors);

 return Object.keys(errors).length === 0;

};
```

Figure 23 form validation

6.5 Implementation of user interface

The user interface of this system is implemented based on the design. It is clear and interactive; as much as possible complexity is handled in this user interface.

- ✓ It reduces the memory loads of the users.
- ✓ It is consistent and interactive.
- ✓ It is easy and clear to use.

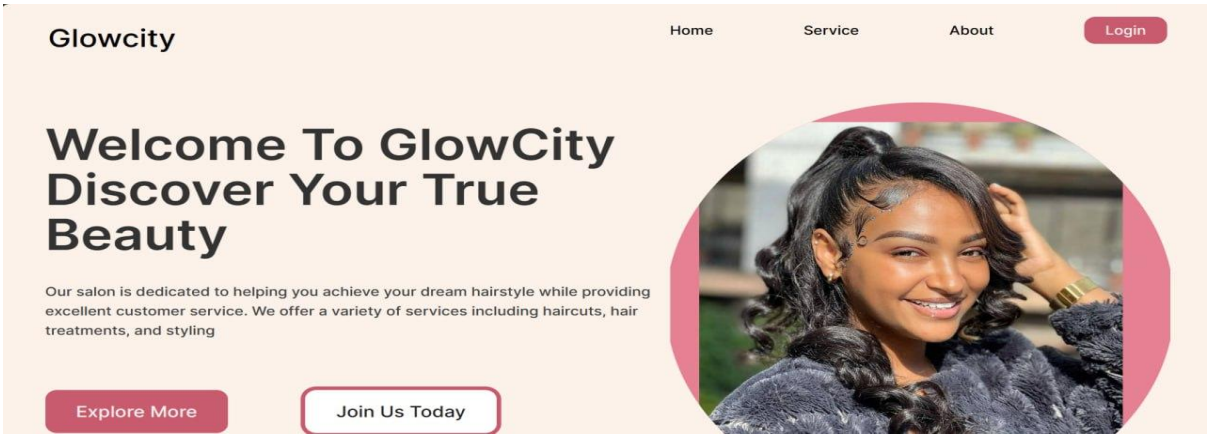


Figure 24 Home page

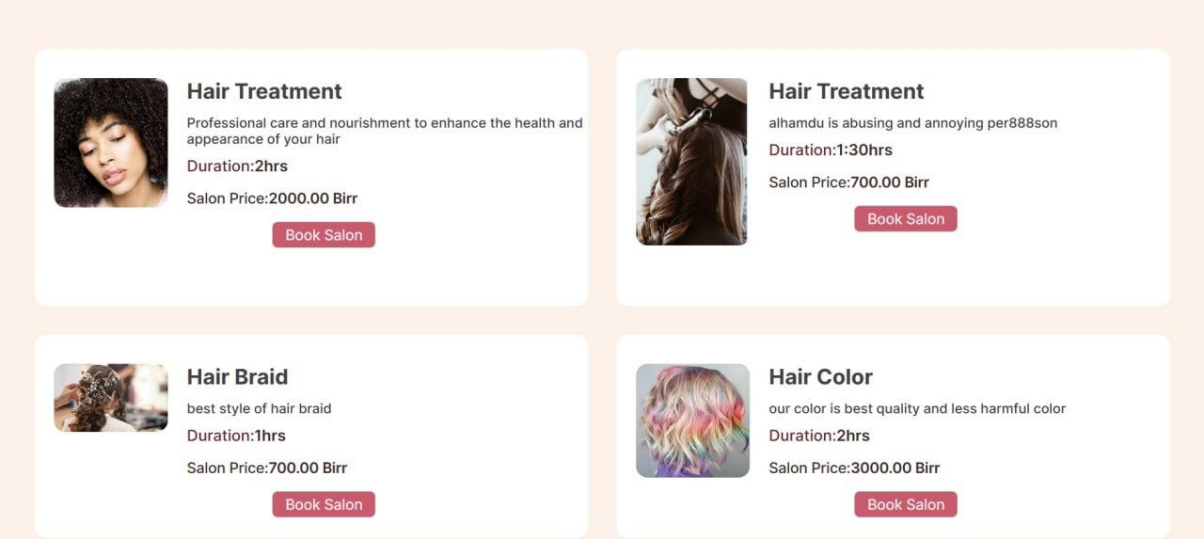


Figure 25 Hair Service page

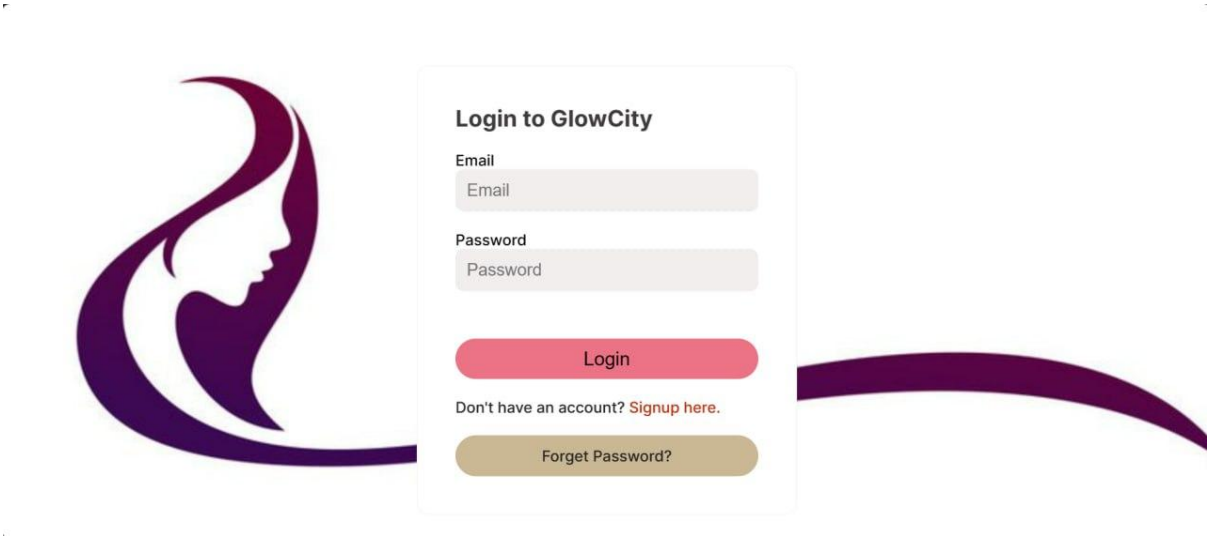


Figure 26 Login page

6.6 Testing

Testing evaluates a software product to ensure that it satisfies its planned purpose. A test that is modified to and consistent with development methodologies provides an an observable and structured approach to verifying requirements and quantifiable performance.

6.6.1 Testing criteria

Criteria are standards by which we evaluate our systems that help us determine whether a test case passes or fails.

Fail Criteria: when the system does not meet the all specific requirements of the system and if the test case is said to fail the expected result is not satisfied by the system that relates to its functionality.

Pass Criteria: when the system meets the all specific requirements of the system and if the test case is said to pass the expected result is satisfied by the system that relates with its functionality.

6.6.2 Test case

Is a set of actions executed to verify a particular feature or functionality of your software applications.

Test Case 1: Check the result on entering Valid User Email & Password

Test Case 2: Check the result on entering Invalid User Email & Password

Test Case 3: Check the response when a User Email is empty & Login Button is pressed, and many more

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.

Table 20 Login page

Input	Expected Result	Actual Outcome	Pass/Fail
Valid Email and	The user logs to the	The user logs to the	Pass

Valid Password	system successfully	system successfully	
If inserted only Email , Without Password	The system displays an error message “please enter correct password”	The system displays an error message “please enter correct password”	Fail
If the Email or password Text Box or both are not filled	The system displays an error message “Email is required” or “password is required”.	The system displays an error message “Email is required” or “password is required”	Fail

6.6.3. Unit testing

Unit testing is a crucial aspect of software development, including online beauty salon services. It involves testing individual units or components of the system to ensure their correctness and functionality. The primary objective of unit testing is to verify that each unit works correctly and as intended. Typically, unit tests are performed by software developers themselves.

When conducting unit testing for an online beauty salon service, you can employ various techniques such as black box testing and white box testing.

- ✓ Check return type correctness.
- ✓ Verify correct sub procedure or function calls.
- ✓ Validate correct output for different inputs.

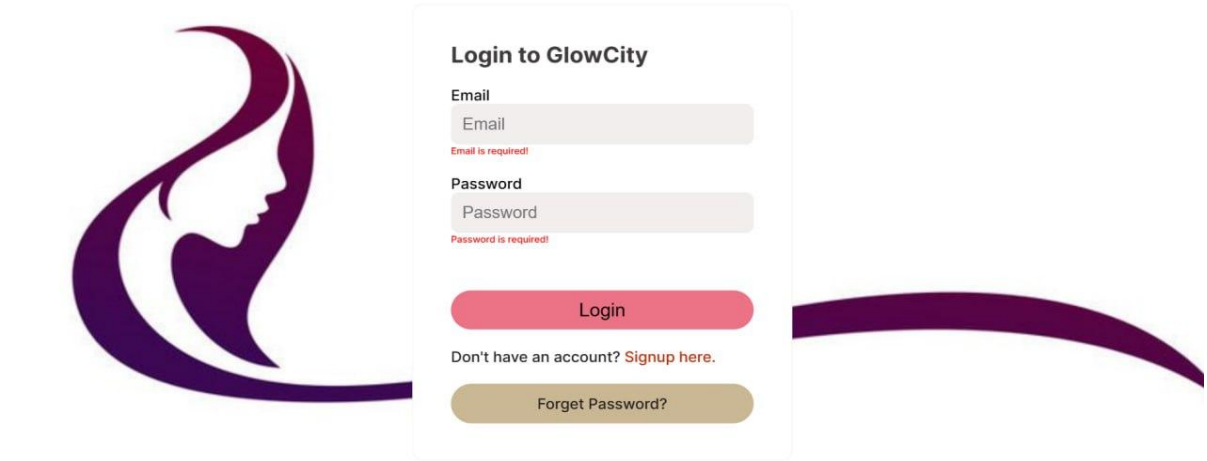


Figure 27 Login validation

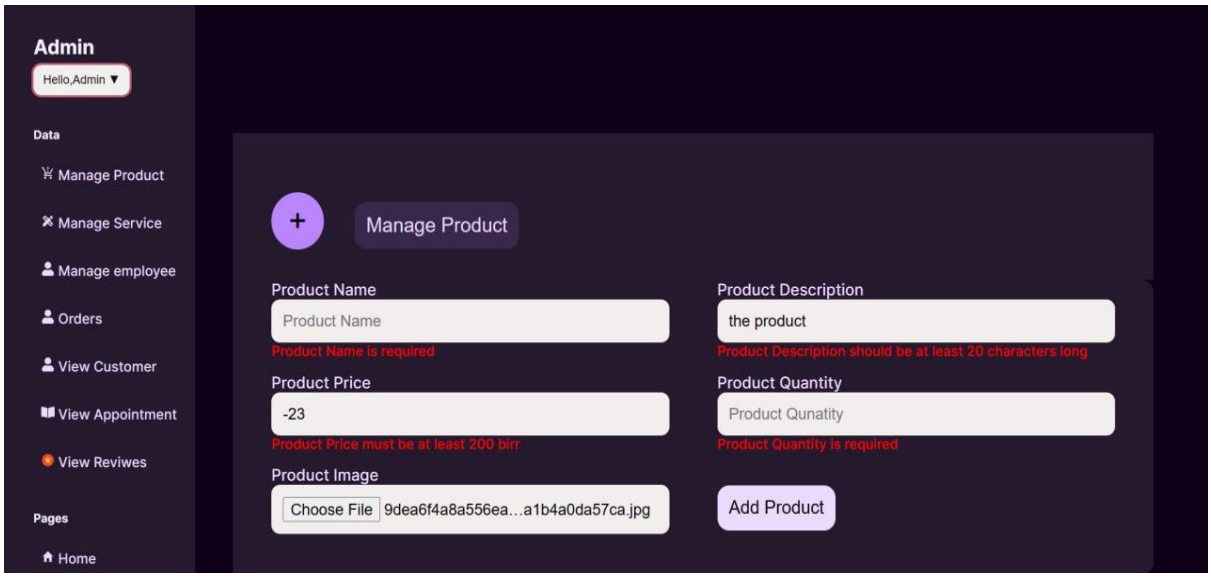


Figure 28 Admin product validation error

6.6.4. System testing

System testing for your beauty salon service involves testing the entire system as a whole to ensure that it meets the desired functional and non-functional requirements. It focuses on assessing the system's behavior, performance, and reliability. Key points for system testing in your beauty salon service:

- ✓ User Interface Testing
- ✓ Functional Testing
- ✓ Error Handling and Recovery Testing

6.6.5. Integration testing

Integration testing for the Beauty Salon System refers to the process of verifying the seamless integration and interaction between different modules and components within the system. It focuses on ensuring that the online product purchase, appointment scheduling, professional feedback and rating, website feedback, and payment service functionalities work together harmoniously.

- ✓ Online Product Purchase and Appointment Scheduling Integration
- ✓ Professional Feedback and Rating Integration
- ✓ Website Feedback Integration
- ✓ Payment Service Integration

CHAPTER SEVEN

Conclusion and Recommendation

Matching of the conclusions with the objectives framed and fulfillment of the objectives are taking into consideration in this part. Further scope and further enhancement of the work done also indicated here.

7.1.Conclusion

In conclusion, the development of an online beauty salon booking system offers significant advantages for customers and salon administrators. The system allows customers to conveniently browse services, select their desired service, and book appointments online. The system enhances the booking experience and provides flexibility for customers.

Furthermore, the system enables customers to select their preferred professional based on availability, view the professional's schedule, and schedule appointments accordingly. This feature optimizes the allocation of resources and ensures that customers can book appointments with their preferred professionals.

The integration of a secure payment system allows customers to make payments during the appointment scheduling process, ensuring a seamless and efficient transaction experience. Additionally, the approval process, where customers receive confirmation via email from the system, adds an extra layer of reliability and professionalism to the system.

By implementing this online beauty salon booking system, the project aims to streamline the appointment scheduling process, enhance customer satisfaction, and improve operational efficiency for the salon.By implementing this online beauty salon booking system, the project aims to streamline the appointment scheduling process, enhance customer satisfaction, and improve operational efficiency for the salon.

7.2.Recommendation

Based on the final project proposal for the online beauty salon booking system, the following recommendations are suggested for further enhancements and future development:

- ✧ Mobile Application: Consider developing a mobile application for the beauty salon booking system in addition to the website. A mobile app would provide customers with the convenience of booking services on the go, accessing their appointments, and receiving notifications directly on their mobile devices. This would expand the reach of the system and cater to a wider range of users.

- ✧ Loyalty Program: Introduce a loyalty program to incentivize repeat customers and encourage customer retention. Offer rewards, discounts, or exclusive promotions to customers who frequently book appointments through the system. This would promote customer loyalty and provide an additional incentive for customers to use the online booking system.

- ✧ Real-Time Availability Updates: Enhance the system by implementing real-time updates for professional availability. Instead of relying solely on pre-set schedules, incorporate a feature that allows professionals to update their availability in real-time, ensuring accurate and up-to-date information for customers. This would minimize scheduling conflicts and improve the overall booking experience.

- ✧ Automated Reminders: Integrate an automated reminder system that sends appointment reminders to customers via email or SMS. This would help reduce no-shows and improve efficiency by ensuring customers are well-informed and adequately prepared for their appointments.

REFERENCES

[1]. E. Birhanu, W. Muluye, and Z. Mulugeta, Final_WKU_Reviewed_Industrial_Project-1 Guidline, 2nd ed

[2]. “PoE - Fast, helpful AI Chat.” htktps://poe.com/

[3]. Lucid Software, “UML class diagrams,” YouTube. Aug. 10, 2023. [Online]. Available: https://www.youtube.com/watch?v=6XrL5jXmTwM

[4]. “Technical Feasibility: Definition, (Process & Best Practices).” https://www.resurgentindia.com/technical-feasibility-definition-process-best-practices

[5]. “Economic feasibility,” CEOpedia | Management Online. https://ceopedia.org/index.php/Economic_feasibility

[6]. “System Analysis and Design - Overview.” https://www.tutorialspoint.com/system_analysis_and_design/system_analysis_and_design_overview.htm

[7]. GeeksforGeeks, “Functional vs Non Functional Requirements,” GeeksforGeeks, Jan. 17, 2024. https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/

[8]. Nico, “What Is Legal Feasibility and Why Is It Important - Colectivo Ley de Segunda Oportunidad,” Colectivo Ley De Segunda Oportunidad, Dec. 08, 2022. https://www.colectivoleysegundaoportunidad.es/what-is-legal-feasibility-and-why-is-it-important/