

Book My Barber App

Project report submitted in partial fulfilment of the requirement for
the award of the degree in

BACHELOR OF COMPUTER APPLICATION



Submitted by

MUHAMMED ASHKAR TK BIAWBCA024

Under the guidance of
Ms. NUSRATH BHANU

DEPARTMENT OF COMPUTER APPLICATIONS



(Affiliated to university of Calicut)

BAITHUL IZZA ARTS AND SCIENCE COLLEGE
NARIKKUNI
(2022-2025)

BAITHUL IZZA ARTS AND SCIENCE COLLEGE
NARIKKUNI

(Affiliated to university of Calicut)



CERTIFICATE

This is to certify that the project/thesis entitled "**Book My Barber App**" submitted here with is an authentic record of the project work carried out by **MUHAMMED ASHKAR TK (BIAWBCA024)** under the supervision and guidance of **Ms.Nusrath Bhanu**. This work has been completed in partial fulfillment of the requirements for the **Bachelor of Computer Applications** during the academic year 2024–2025, under the University of Calicut, at Baithul Izza Arts and Science College, Calicut.

Submitted to viva-voce examination held on.....

.....

Internal examiner

.....

External examiner



BAITHUL IZZA ARTS AND SCIENCE COLLEGE

(Affiliated to university of Calicut)

NARIKKUNI, KOZHIKODE - 673585

**DEPARTMENT OF COMPUTER APPLICATIONS
CERTIFICATION**

This is to certify that the project/thesis entitled "**Book My Barber App**" submitted herewith is an authentic record of the project work done by **MUHAMMED ASHKAR TK (BIAWBCA024)**. It was submitted under the supervision and guidance of **Ms. Nusrath Bhanu** in partial fulfillment of the requirements for the **Bachelor of Computer Applications** during the academic year 2024-2025. This work was undertaken under the University of Calicut at the College of Baithul Izza Arts and Science, Calicut.

Ms. NUSRATH BHANU
(H.O.D Dept. of computer applications)

ACKNOWLEDGEMENT

Let us take the opportunity to thank all those who have been directly or indirectly involved in making our project a success. First of all, we are grateful to God Almighty for helping us select this topic and giving us the interest to pursue it. Without His blessings, we could not have been able to complete this project.

We are also grateful to Prof. ABDURAHIMAN.N, Principal of Baithul Izza Arts and Science College. It is with great pleasure that we place on record our indebtedness and deepest senses of gratitude to Ms. NUSRATH BHANU, Head of the Department of Computer Applications, for her sound counsel and endless encouragement from time to time.

We are extremely grateful to Ms. NUSRATH BHANU, our project in charge, for providing valuable support. We express our sincere thanks to our family members and friends who helped us in overcoming all the hurdles, thus gearing us towards the fulfillment of the task that we had undertaken.

We also express our profound gratitude to all others who, in some way, helped us with this project till the end.

MUHAMMED ASHKAR TK

DECLARATION

We hereby declare that this project work entitled **Book My Barber App** is a bonafide project work done by me in partial fulfilment of the requirement for the award of **BACHELOR OF COMPUTER APPLICATION** under the guidance of Ms. NUSRATH BHANU, BAITHUL IZZA ARTS AND SCIENCE COLLEGE NARIKKUNI (Affiliated to University of Calicut)

We also declared that this report has not been submitted fully or partially for the award of any other degree Diploma Title or recognition before.

MUHAMMED ASHKAR TK

BIAWBCA024

Date:

Place:

Book My Barber App

ABSTRACT

The Barbershop Management System (BOOK MY BARBER) is a robust platform designed to streamline barbershop operations while delivering a superior user experience to customers. The system comprises four key modules: Admin, Barbershop, User, and Staff, each meticulously crafted to address distinct functional needs.

The Admin Module offers comprehensive control over the platform, enabling administrators to manage barbershop verifications, monitor services, handle user feedback, and efficiently address complaints to maintain smooth platform operations.

The Barbershop Module empowers shop owners with tools to manage profiles, services, staff assignments, booking slots, payment processes, and design elements. Barbershops can seamlessly interact with customers through reviews and feedback, fostering a strong client relationship.

The User Module enhances the customer journey by offering hassle-free registration, easy navigation through approved barbershops and services, quick appointment booking, secure payment options, and feedback submission. It also includes unique features like viewing staff profiles, checking queue status, and sending complaints with a reply mechanism.

The Staff Module supports staff members by providing access to assigned work, profile management, status updates, and efficient task handling.

By automating manual processes and offering a modern, user-friendly interface, the system significantly boosts operational efficiency and accuracy. The Barbershop Management System not only optimizes business workflows but also strengthens the connection between barbershops and their customers, ensuring enhanced satisfaction and streamlined service delivery.

CONTENT

1. INTRODUCTION	10
2. SYSTEM ANALYSIS	11
2.1. IDENTIFICATION OF NEED	12
2.2. PRELIMINARY INVESTIGATION	12
2.2.1. FACT FINDING TECHNIQUE	13
2.2.2. EXISTING SYSTEM	15
2.2.3. PROPOSED SYSTEM	15
2.3. FEASIBILITY OF STUDY	16
2.3.1. TECHNICAL FEASIBILITY	16
2.3.2. ECONOMIC FEASIBILITY	17
3. SOFTWARE REQUIREMENT SPECIFICATION	18
3.1. FUNCTIONAL REQUIREMENT	19
3.2. NON - FUNCTIONAL REQUIREMENT	20
3.3: SYSTEM CONFIGURATION	21
3.3.1. HARDWARE CONFIGURATION	21
3.3.2. SOFTWARE CONFIGURATION	21
3.4. CONCEPTUAL MODELS	22
3.4.1. ER DIAGRAM	22
3.4.2. USE CASE DIAGRAM	25
4.4.3. DATA FLOW DIAGRAM	29

4. SYSTEM DESIGN	34
4.1. MODULE DESCRIPTION	35
4.2. DATABASE DESIGN	37
4.3. DATA INTEGRITY AND CONSTRAINTS	37
4.4. NORMALIZATION	39
4.5. TABLE DESIGN	40
5. TESTING	44
5.1. UNIT TESTING	46
5.2. INTEGRATION TESTING	46
5.3. SYSTEM TESTING	46
5.4. ACCEPTANCE TESTING	47
5.5. VALIDATION TESTING	47
5.6. OUTPUT TESTING	48
6. ERROR HANDLING	48
7. FUTURE ENHANCEMENT	48
8. CONCLUSION	49
9. APPENDIX	50
9.1. SAMPLE SCREENSHOT	50

1. INTRODUCTION

The Barbershop Management System is a comprehensive platform designed to optimize barbershop operations while delivering a seamless experience for both business owners and customers. The system integrates web-based modules for Admin, Barbershop, and Staff, along with a dedicated User application.

The platform addresses common challenges faced by barbershops, such as appointment management, staff coordination, payment processing, and effective customer communication. The Admin Module provides system administrators with the ability to verify barbershops, oversee services, manage user feedback, and handle complaints efficiently. The Barbershop Module offers shop owners the tools to manage profiles, services, booking slots, staff assignments, and financial transactions through an intuitive web interface. The Staff Module allows staff members to manage their profiles, view assigned tasks, and update work statuses seamlessly.

The User Module ensures a user-friendly experience for customers by offering features such as easy registration, browsing of approved barbershops, booking appointments, secure payments, viewing queue status, and submitting reviews and complaints.

By automating manual tasks and providing a structured and user-friendly interface, BOOK MY BARBER enhances operational efficiency and improves communication between barbershops and their clients. The system contributes to a well-organized and efficient business environment, ultimately boosting customer satisfaction and business productivity.

2. SYSTEM ANALYSIS

2.1 IDENTIFICATION OF NEED

The Barbershop Management System is developed to address the challenges faced by barbershops in managing appointments, staff, and customer interactions. Traditional manual methods are often inefficient, time-consuming, and prone to errors. There is a need for a digital solution that enhances operational efficiency and provides a seamless experience for both business owners and customers.

The system offers distinct modules for Admin, Barbershop, User, and Staff, allowing for efficient management of services, bookings, staff assignments, and payments. It also improves communication between barbershops and customers, enabling easy registration, appointment booking, secure payments, and feedback submission.

By automating routine tasks and providing a structured, user-friendly interface, BOOK MY BARBER reduces manual effort, minimizes errors, and fosters a professional and well-organized business environment.

2.2 PRELIMINARY INVESTIGATION

Preliminary investigation is the first phase. In this phase, the system is investigated. The objective of this phase is to conduct an initial analysis and findings of the system. First, need for the new or the enhanced application is established. Only after the recognition of need, for the proposed system is done then further analysis is possible. Once the initial investigation is done and the need for new or improved system is established, all possible alternate solutions are chalked out. All these systems are known as "**candidate systems**". All the candidate systems are then weighed and the best alternative of all these is selected as the solution system, which is termed as the "**proposed system**". The proposed system is evaluated for its feasibility. Feasibility for a system means whether it is practical and beneficial to build that system.

The Barbershop Management System was identified as the ideal solution

during this phase. The system offers a comprehensive platform with web-based modules for Admin, Barbershop, and Staff, and an Android application for Users. It aims to streamline appointment management, staff assignments, payments, and customer interactions, providing a practical and efficient approach to modern barbershop management.

2.2.1. FACT FINDING TECHNIQUES

Fact finding is process of collection of data and information based on techniques which contain sampling of existing documents, research, observation, questionnaires, interviews etc. There are several methods for gathering the sort of information. We can use all of these methods for gathering information from the user of the existing system.

1. Sampling

Sampling is the process of collecting a representative sample of documents, forms and records. Because it would be impractical to study every occurrence of every form or record in a file or database, system analyst normally use sampling techniques to get a large enough cross section to determine what can happen in the system. The system analyst seeks to sample enough forms to represent the full nature and complexity of the data. First collected sample is a Sample receipt and conducted a study to know how these data can be converted to a digital method.

2. Research and Visit Sites

Research and site visits is the process of examining the problems which had previously solved by other sources that can be either human or documents. To solve the requirements of problem, the analyst visits to other organization that had previously experienced for similar problems. In addition, the analyst can also find the information from database, reference books, case studies and internet.

3. Observation

Another fact-finding technique is observation. In this technique, system analyst participates in the organization, studies the flow of documents,

applies the existing system, and interacts with the users. Observation can be a useful technique when the system analyst have user point of view. Sampling technique called work sampling is useful for observation. By using this technique, system analyst can know how employees spend their days. This is an effective data collection technique for obtaining an understanding of a system.

4. Interviews

Interview is the most commonly used technique to collect information from the face-to-face interaction. The purpose of interview is to find, verify, clarify facts, motivate end-users involved, identify requirements and gather ideas and opinions.

5. Questionnaires

Questionnaires are also one of useful fact-finding technique to collect information from large number of users. These are special-purpose documents that allow the analyst to collect the information and opinions from respondents. Users fill up the questions which are given by the system analyst and then give the answers back to the system analyst.

Questionnaires can save time because system analyst does not need to interview each of users. The documents can be mass produced and distributed to respondent, who can then complete on their own day. This allows analyst to collect facts from a large number of people while maintaining uniform responses. People can fill the forms and give answers freely to the analyst. This technique is inexpensive.

6. Prototyping

Another fact-finding technique is known as prototyping which collects the requirement facts of the system. Prototyping is sampling a small working model and it is more related to pre-design of the information system. The implementation of prototyping can be developed in earlier stage of system development life cycle when analyzing the facts. The process of prototyping facts in order to specify the user's requirements is also known as discovery prototyping.

In our system we mainly used 3 methods to find relevant data for the requirement of system those are research and site visiting, conducting questionnaire, observation. Research and site visiting was helped a lot us to get through the overall idea of the project. Documentations that are showcase the android and web related contents was so help full. Then conducting questionnaire between team members was really brought down major problems in project and as well as introducing new ideas to the system. Then observation of each member was leading us to introducing the different functionalities to the project prior to existing system.

2.2.2 Existing System

In traditional barbershop management, operations such as appointment scheduling, staff management, and payment processing are often handled manually or through basic systems. This approach is time-consuming, prone to human errors, and lacks the efficiency required to handle large volumes of bookings and staff coordination. Customers face inconvenience in booking appointments, checking availability, and providing feedback. Barbershops struggle with managing services, handling payments, and maintaining organized communication with both staff and clients. There is also limited transparency and accessibility for users to view barbershop details, services, and staff information.

2.2.3 Proposed System

The Barbershop Management System offers a digital solution to these challenges by integrating dedicated modules for Admin, Barbershop, Staff, and User functionalities. The system provides:

- **Admin Module:** Manages barbershop verification, monitors reviews and complaints, and ensures smooth platform functionality.
- **Barbershop Module:** Allows barbershops to manage profiles, services, booking slots, staff assignments, payments, and customer interactions.
- **Staff Module:** Empowers staff members to manage profiles, view assigned tasks, and update work status.
- **User Module (Android Application):** Enables users to register, view barbershops, book appointments, make payments, view queue status, and share feedback.

The proposed system automates manual processes, reduces operational inefficiencies, enhances customer convenience, and provides a structured, user-friendly interface for all stakeholders.

2.3. FEASIBILITY STUDY

A feasibility analysis is performed to choose the system that will meet the performance requirement at least cost. The study will decide if the proposed system will be cost effective from the business point of view and if it can be developed in the given existing budgetary constraints the feasibility study should be relatively cheap and quick. The feasibility study is done in these phase :

2.3.1. Technical Feasibility

Technical feasibility concentrates on the existing computer system and what tends it can support the proposed system. Technical feasibility is highly Computer Oriented process. The existing technology can support the proposal system to an extent and hence the system is technically feasible. Using a computerized system, time and effort is reduced to a great extent.

Our system's technical requirements include:

- **Frontend Technologies:** HTML, CSS, JavaScript for creating responsive and user-friendly web interfaces.
- **Backend Development:** Python (Django) in PyCharm for robust server-side logic, API development, and database management.
- **User Application:** Developed using Android Studio with Flutter, providing a smooth and native experience for mobile users.
- **Database Management:** SQL Yog with MySQL for efficient data storage, retrieval, and management.
- **Operating System:** Windows 10 or higher for development and deployment environments.

2.3.2. Economic Feasibility

Economic feasibility is the most frequently used method for evaluating the effectiveness of a system. The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. Our system is economically feasible since it provides benefits and protection to the resources and Lives of People. It is cost effective.

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1. FUNCTIONAL REQUIREMENTS

A functional requirement describes what a software system should do. Functional requirements specifically define functionalities of the system, behavior of the system and the goals to achieve it. This system has the following functional requirements:

Functional Requirements	Importance	Description
Security	Essential	User login is required
Web interface	Essential	Provides an intuitive and interactive experience for Admin, Barbershop, and Staff modules.
Database	Essential	Stores user details, booking information, payment records, and service data.
Booking System	Essential	Allows users to book services and view available slots in real-time.
Staff Management	Important	Enables barbershops to manage staff profiles and assign staff to bookings.
Review & Rating System	Important	Allows users to provide feedback on barbershop services.
Payment Integration	Important	Supports secure payment processing for service bookings.

3.2. NON FUNCTIONAL REQUIREMENTS

Non-functional requirements are also known as quality of a system. Hence it provides us knowledge regarding the operations instead of behavior or functionalities, contradicting with the functional requirements in the manner. Non-functional requirements described in the system architecture helping us to achieve the quality goals and improve the functionalities of the system. Non-functional requirements are explained in the bellow given table:

Non-Functional Requirement	Importance	Description
Performance	Performance	Performance of system should be adequate and fast.
User friendly environment	Essential	System interface Should be simple and user friendly
Application maintenance	Essential	Maintenance and documentation should be done thoroughly
Application scalability	Essential	System should be flexible for future use
Platform independence	Essential	System should be capable to work in any environment

3.3. SYSTEM CONFIGURATION

The hardware and software configurations required for the installation and smooth functioning of this project are based on the specific needs of the operating environment's components. The front-end system, in particular, necessitates a minimum configuration for both hardware and software components. Ensuring compatibility with the system environment is crucial for optimal performance.

It includes two phases:

3.3.1. HARDWARE CONFIGURATION

The selection of hardware is very important in the existence and proper working of any software. Then selection hardware, the size and capacity requirements are also important.

Processor	:	Intel Pentium Core i3 and above
Primary Memory	:	8GB RAM and above
Storage	:	320 GB hard disk and above
Display	:	VGA Color Monitor
Key Board	:	Windows compatible
Mouse	:	Windows compatible

3.3.2. SOFTWARE CONFIGURATION

One of the most difficult task is selecting software for the system, once the system requirements is found out then we have to determine whether a particular software package fits for those system requirements. The application requirement:

Operating System	:	Windows 10 (32-bit or 64-bit)
Front End	:	HTML, CSS, JavaScript
Back End	:	MySQL (Database) using SQL Yog
Framework	:	Django
Script	:	Python 3.6 or above
User Application	:	Dart (Flutter) in Android Studio

3.4. CONCEPTUAL MODELS

3.4.1. ER DIAGRAM

An Entity Relationship Diagram (ERD) shows the relationship of entity sets stored in an entity in this context is a component of data. In other words, ER diagram illustrate the logical structure of databases. At first glance an entity relationship diagram looks very much like a flowchart. It is the specialized symbols, and the meanings of those symbols, that make it unique.

Entity

Entities are represented by means of rectangles. Rectangles are named with the entity set they represent.

Attributes

Attributes are the properties of entity. Attributes are represented by means of eclipses every ellipse represents one attribute and is directly connected to its entity. If the attributes are composite, they are further divided in a tree like structure. Every node is then connected to its attributes. That is, composite attributes are represented by ellipses that are connected with an ellipse. Multivalued attributes are depicted by double ellipses. Derived attributes are depicted by dashed ellipses.

Relationship

Relationships are represented by diamond-shaped box name of the relationship is written inside the diamond box. All the entities participating in a relationship, are connected to it by a line.

Binary relationship and Cardinality

A relationship where two entities are participating is called a binary relationship. Cardinality is the number of instances of an entity from a relation that can be associated with the relation.

One-to-one - when only one instance of an entity is associated with the relationship. It is marked as "1:1". The following image reflects that only one instance of each entity should be associated with the relationship. It depicts one-to-one relationship.

One-to-many -when more than one instance of an entity is associated with a relationship, it is marked as „1:N“. The following image reflects that only one instance of entity on the left and more than one instance of an entity on the right can be associated with the relationship.

Many-to-one -when more than one instance of an entity is associated with the relationship, it is marked instance of an entity on the left and only one-one instance of an entity on the right can be associated with the relationship.

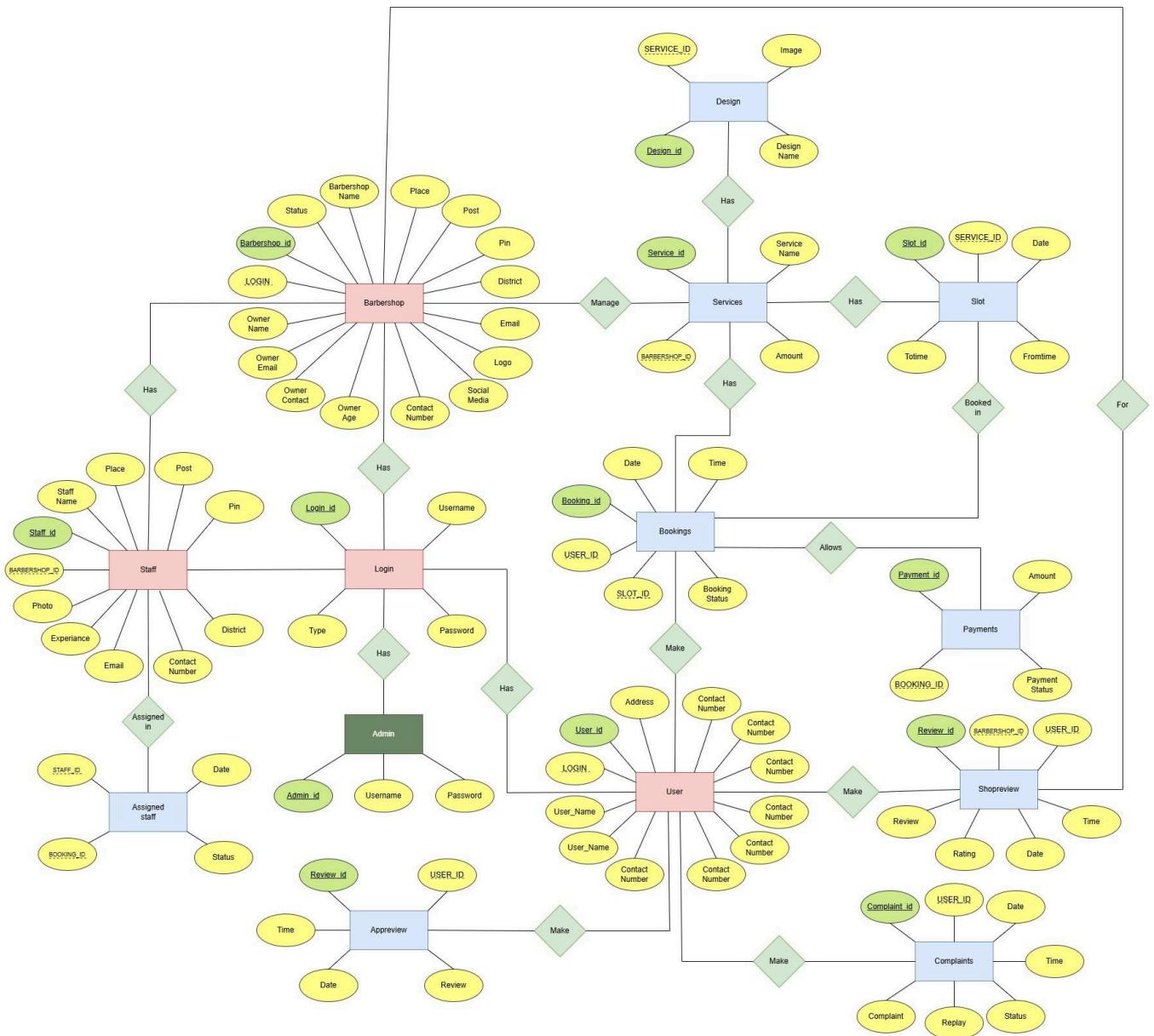
Many-to-many -The following image reflect that more than one instance of an entity on the left and more than one instance of an entity on the right can be associated with the relationship.

Participation constraints

Total participation-Each entity is involved in a relationship. Total prod by double lines.

Partial participation- Not all entities are involved in a relationship. Partial participation is represented in single lines.

Entity Relationship diagram:



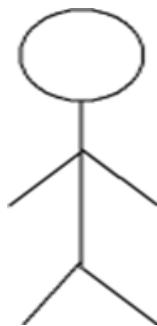
3.4.2. USE CASE DIAGRAM

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements.

In this context, the term "system" refers to something being developed or operated. Such as a mail-order product sale and service web site. Use case diagrams are employed UML (Unified Modeling Language), standard notation for the modeling of real world objects and system.

Actor

An actor represents a role that an outsider takes on when interacting with a business system. For instance, an actor can be a customer, a business partner or another business system. Every actor has a name:



Use case

A use case represents a user goal that can be achieved by accessing the software application. In visual paradigm, you can make use of the sub-diagram feature to describe the interaction between user and system within a use case by creating a sub-sequence diagram under a use case. You can also describe the use case scenario using the flow of Events editor.

Relationships

A line between actor and use case represents communication. Typically, bidirectional (e.g. actor initiates the use case, a use case sends information to an actor)

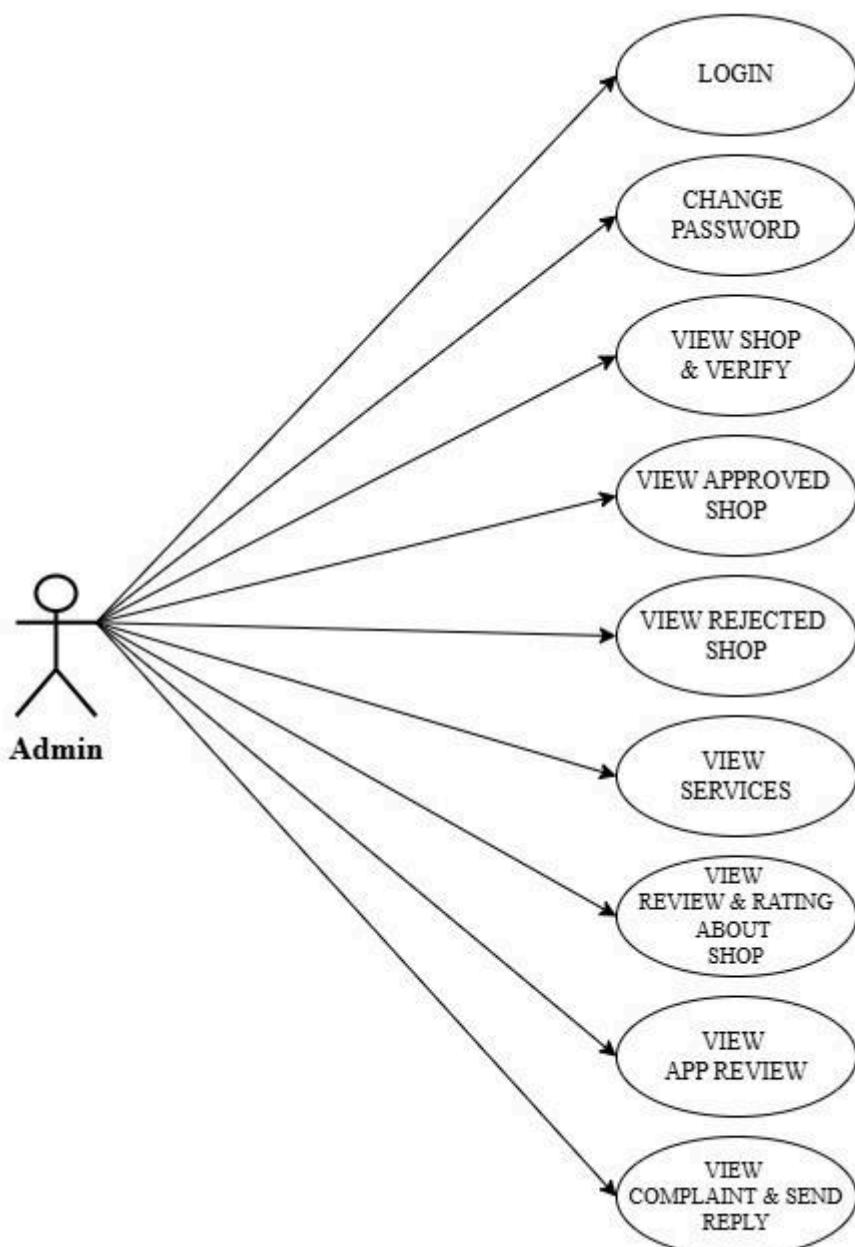
<include>>

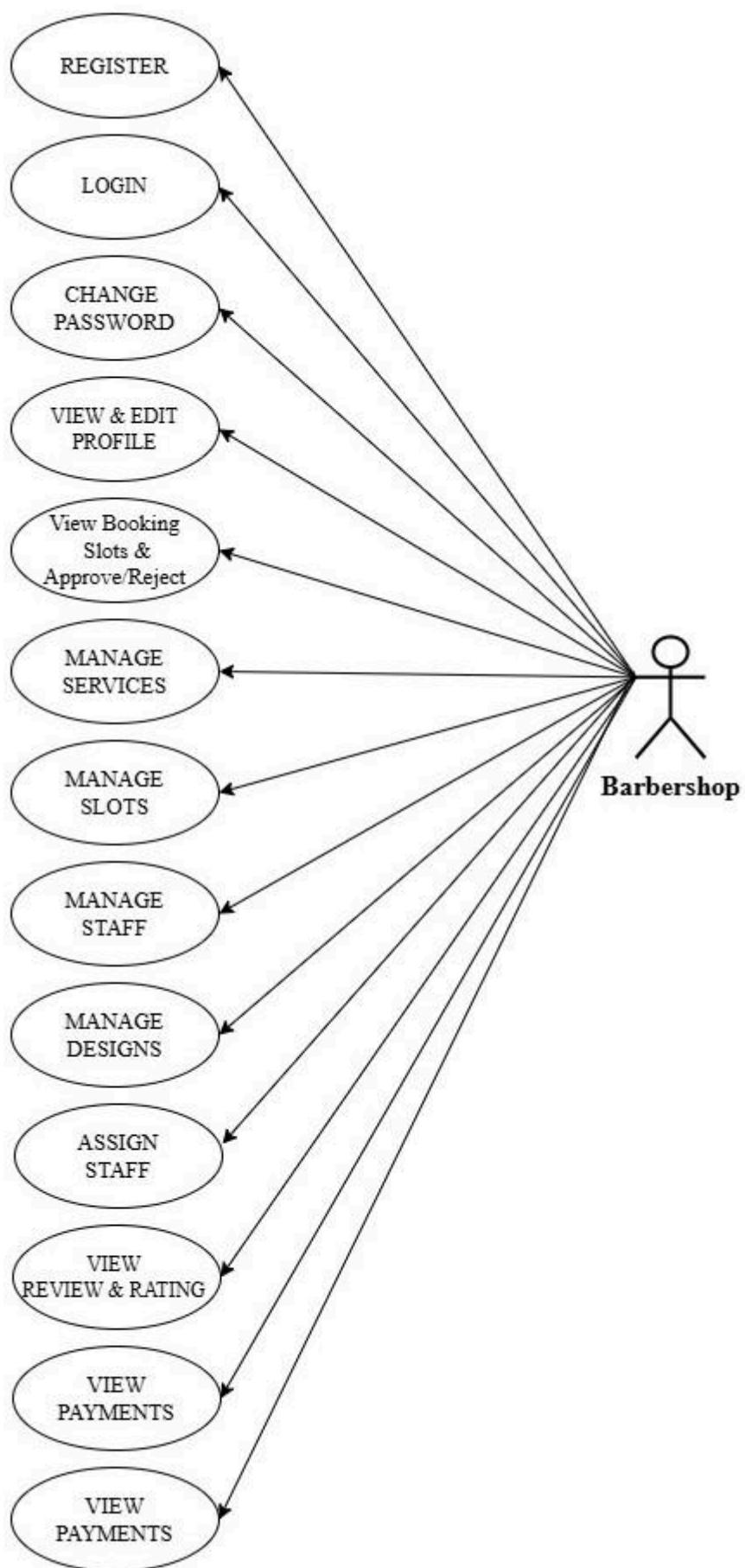


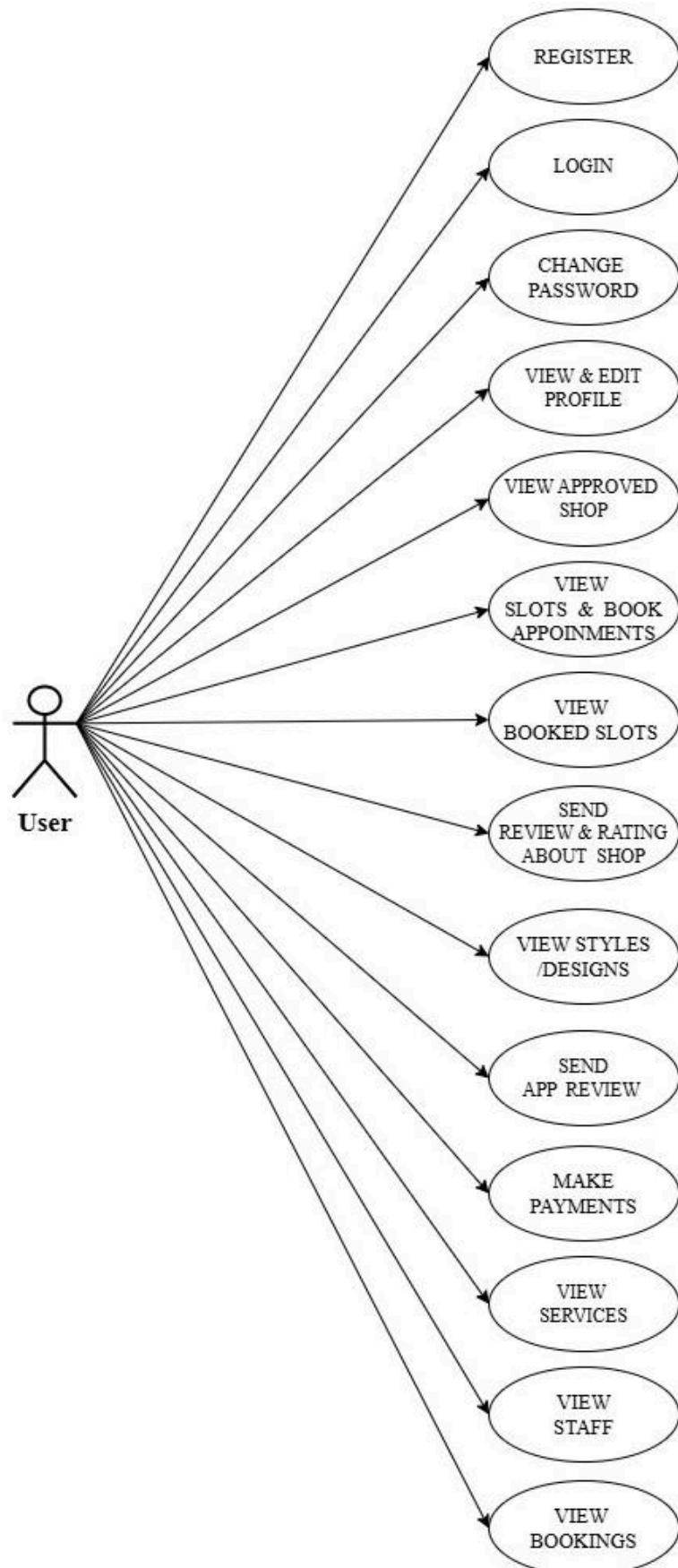
<<extend>>

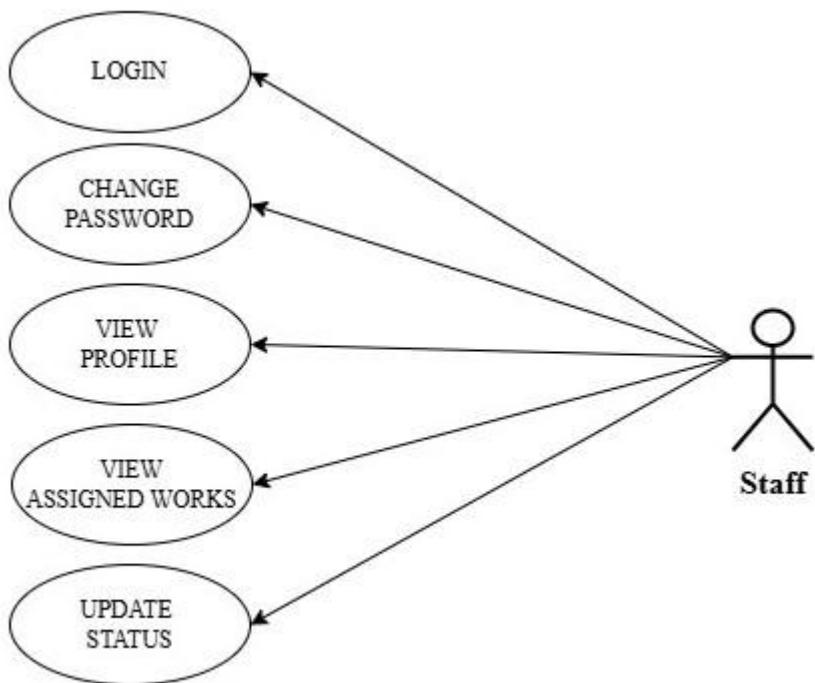


Use case Diagram:



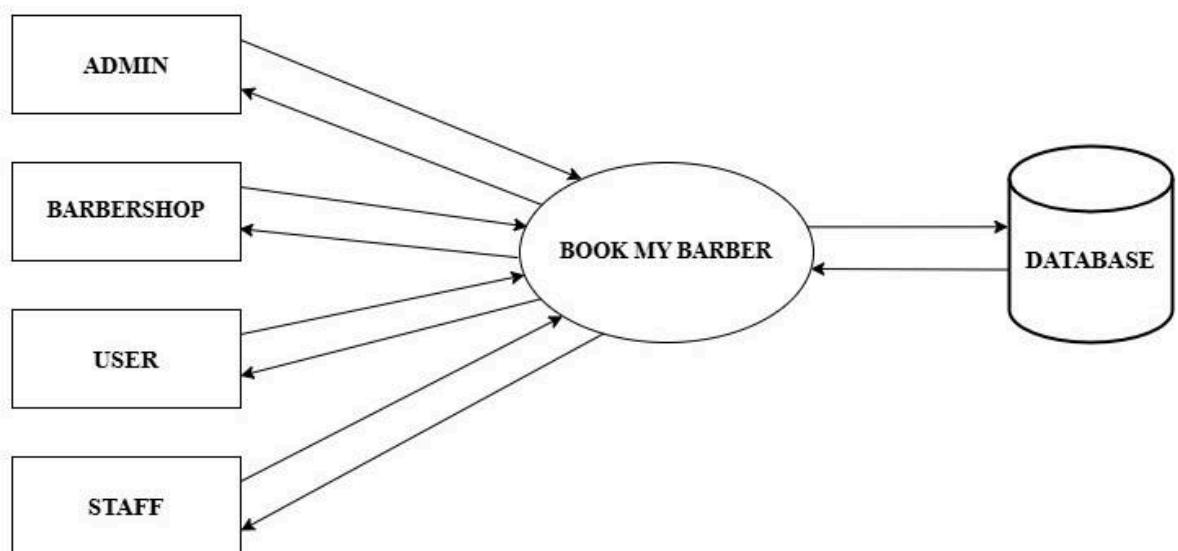




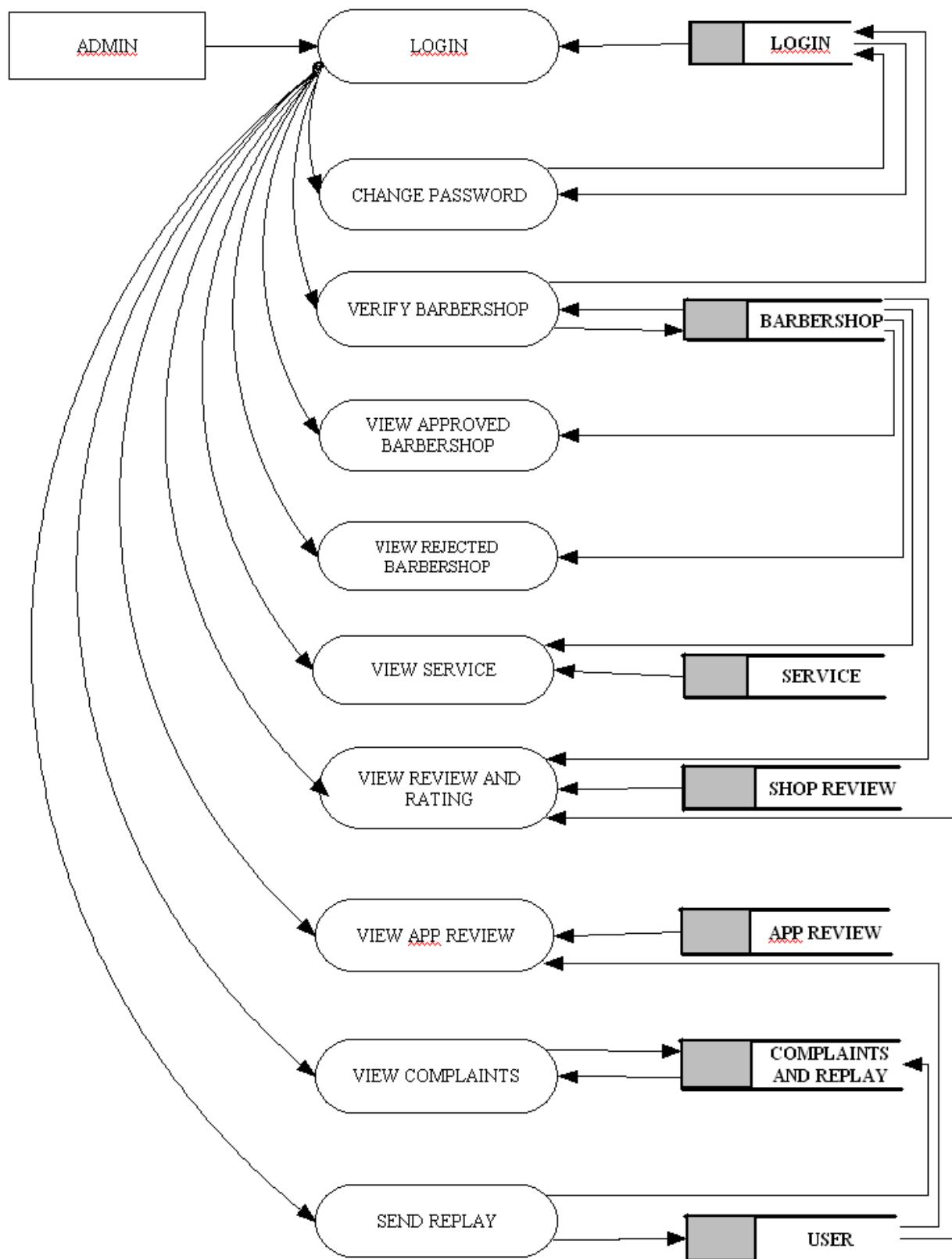


3.4.3. DATA FLOW DIAGRAM

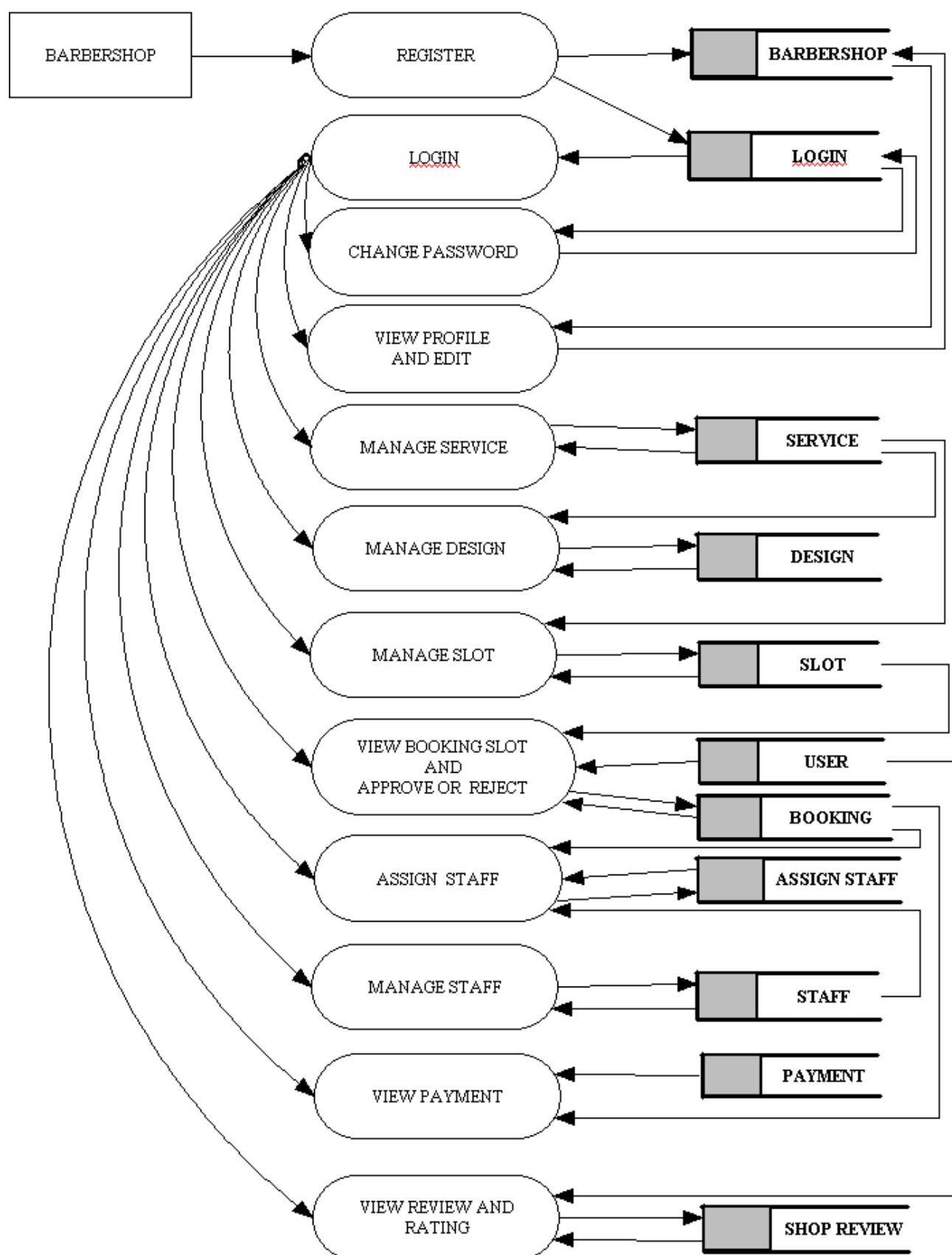
LEVEL 0



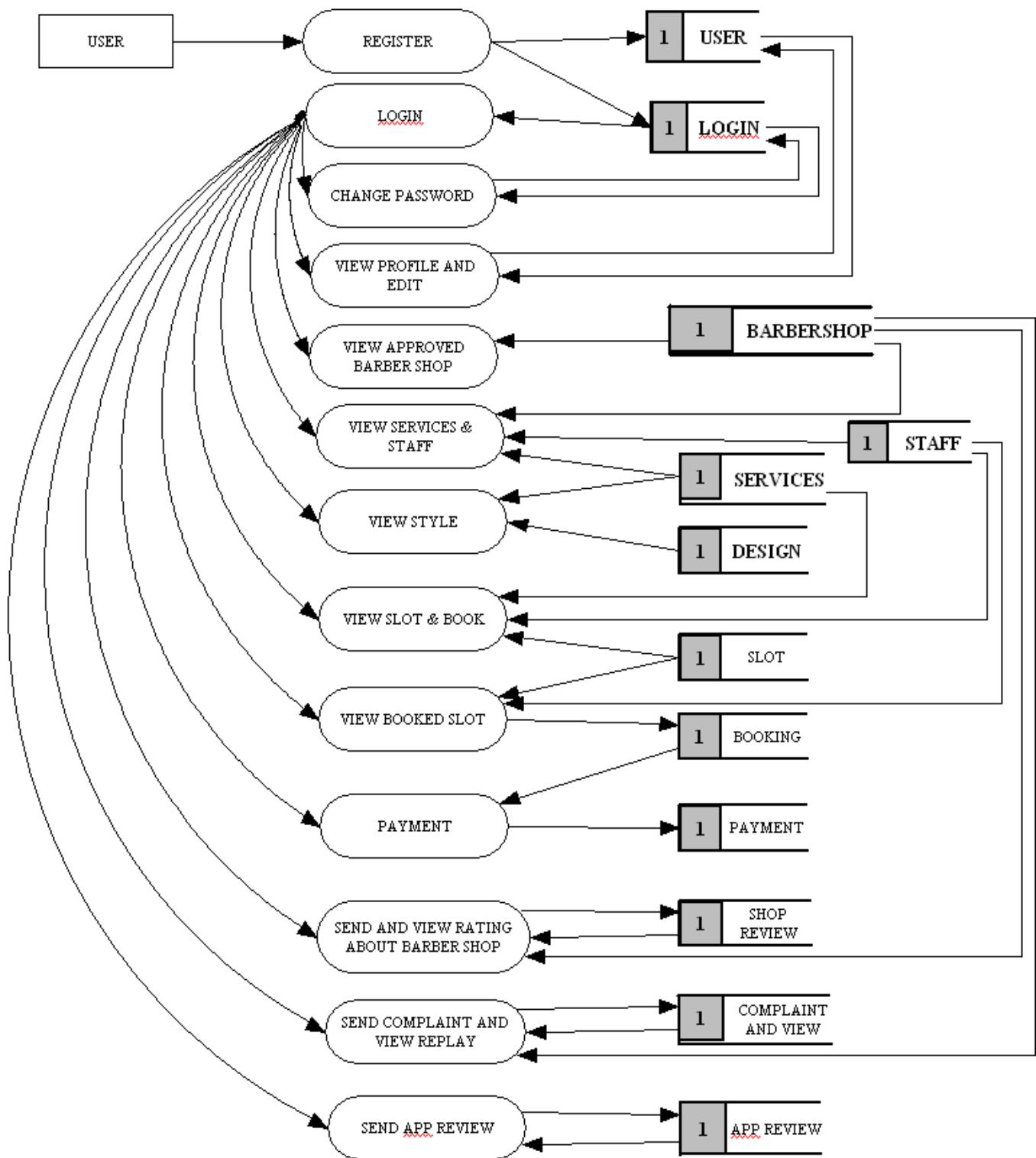
LEVEL 1



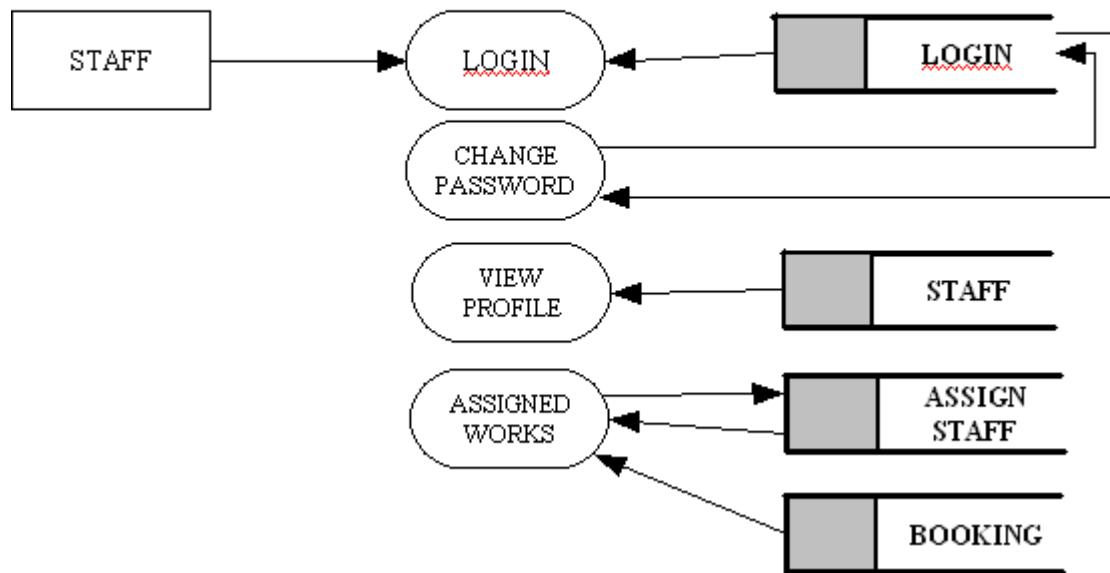
LEVEL 2



LEVEL 3



LEVEL 4



4. SYSTEM DESIGN

4.1. MODULE DESCRIPTION

The system after careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

Admin :

The Admin is the central authority responsible for managing the BOOK MY BARBER system through a web-based platform. The Admin oversees barbershop approvals, user interactions, and system integrity. Key responsibilities include:

- Login
- Change password
- Verify barbershop
- View approved barbershop
- View rejected barbershop
- View services
- View review rating about shop
- View app review
- View complaint
- send reply

Barbershop :

The Barbershop module enables barbershop owners to manage their services, bookings, and staff through a web-based platform. It provides essential tools to streamline operations and enhance customer experience.

Key functionalities include:

- Register
- Login
- Change password
- View profile & edit profile
- Manage services
- Manage slot
- View booking slot & approve/reject
- View payment

- Manage design
- View review rating
- Manage staff
- Assign staff

Staff :

The **Staff** module is a web-based system that allows barbers and employees to manage their assigned tasks efficiently. It provides essential tools for tracking work status and updating job progress.

Key functionalities include:

- Login.
- Change Password.
- View Profile
- View Assigned Works
- Update Status:

User :

The **User** module is an Android application that allows customers to find and book appointments with barbershops conveniently. It provides a seamless experience for browsing services, making payments, and sharing feedback.

Key functionalities include:

- Register & Login
- Change Password
- View & Edit Profile
- View Approved Barbershops
- View Services
- View Staff
- View Slots & Book Slot
- View Bookings
- View Booked Slot
- Payment
- View Styles/Designs
- Send Review & Rating About Barbershop
- Send App Review
- Send Complaint & View Reply

4.2. DATABASE DESIGN

A database is a collection of related records. The main objective of database design is to provide effective auxiliary storage without any applications and to contribute overall efficiency of the computer program components of the whole system.

The organization of data in the database aims to achieve the following objectives:

- Controlled redundancy
- Ease of learning in use
- Data independence
- More information in low cost
- Accuracy and integrity
- Recovery from failures
- Privacy and security
- Performance

The design should be done in a way the information stored in the database can be retrieved quickly whenever necessary. The general theme behind a database is to handle information as an interrelated whole. A database is a collection of interrelated data stored with minimum redundancy to serve users quickly and efficiently. Database design runs parallel without application design. As we collect information about what is to be done, we will obviously collect information about data need to be entered, stored messages and printed reports. The designing of database is done with outmost care and security during the designing phase of the system. Special care was taken to develop minimum number of databases for the maximum efficiency of the System.

4.3. DATA INTEGRITY AND CONSTRAINTS

In this project data security, data validation checking methods are applied using a password authentication. Generally the access to the whole system can be provided only if given user id and password as authorized otherwise, an error message box is displayed once user gives correct password he is authenticated and hence permitted to get in to the master form.

Data integrity refers to the process of ensuring that a database remains an accurate reflection of the universe of discourse it is modeling or representing in other words there is a close correspondence between the facts stored in the design should be done in a way the information stored in the database can retrieved quickly whenever necessary. The general theme behind a database is to handle information as an interferred whole. A database is a collection of interrelated data stored with minimum redundancy to serve users quickly and efficiently. Database design runs parallel without application design. As we collect information about what is to be done, we will obviously collect information about data need to entered, stored messages and printed reports.

The designing of database is done outmost care and security during the designing phase of the system. Special care was taken to develop minimum number of databases for the maximum efficiency of the system. the database and the real world it models data integrity is normally enforced in a database system by a series of integrity constraints or rules .Three types of integrity constraints are an inherent part of the relational data model, entity integrity, referential integrity and domain integrity. Entity integrity is same as the concept of a primary key. Entity integrity is integrity rule which status that every table must have primary key and that the column or columns chosen to be the primary key should he unique and not null. Referential integrity concerns the concept of a foreign key.

The referential integrity rules states that any foreign key value can only being one of two states Domain integrity specifies that all columns in relational database must be declared upon a defined domain. A constraint is a property assigned to a column or the set of columns in a table that prevents certain type of inconsistent data values from being placed in the column Constraints are used to enforce the data integrity. This ensures the accuracy and reliability of the database.

- A PRIMARY KEY constraint is a unique identifier for a row within database. A table should have a constraint to uniquely identify each row and only one primary key constraint can be created for each table.

- A UNIQUEconstraint enforces the uniqueness of the value in a set of columns, so no duplicate values are entered. The unique key constraints are used to integrity as the primary key constraint.
 - A FOREIGNKEYconstraint prevents any action that would destroy link between tables with the corresponding data values
 - A CHECKconstraint is used to limit the values that can be placed in a column. The check constraint is used to enforce domain integrity
- A NOT NULL constraint enforces that column will not accept null value. The not null constraint is used to enforce domain integrity, as the check constraints.

4.4. NORMALIZATION

The process of normalization is concerned with the transformation of the conceptual schema to a computer representable form. Normalization reduces the redundancies and anomalies. Redundant data wastes disk space and creates maintenance space. If data that exists in more than one place must be changed, the data must be changed, the data must be changed in exactly the same way in all locations. There are a few rules for database normalization. It is process that helps analysts or database designers to design table structures for an application.

The focus of normalization is to reduce table data be very minimum. By this process RDBMSschema designers try their best to reduce table data to the very minimum. It is essential to remember that redundant data cannot be reduced to zero in any database management system. Each rule is called a "Normal Form".

The First Normal Form:

First normal form does not allow multivalued and composite valued attributes. It states that the domain of an attribute must include only atomic values and that value of any attribute in a tuple must be single value from the domain of that attribute.

The Second Normal Form:

In second normal form, for relations where primary key contains multiple attributes, non-key attributes should not be functionally dependent on a part of the primary key.

Third Normal Form:

In third normal form satisfies the second normal form and no non-key attributes of relation transitively dependent on primary key.

In our project BOOK MY BARBER, we are using the Third Normal Form (3NF) for database normalization.

4.5. TABLE DESIGN

LOGIN

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
username	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
password	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
type	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

BARBERSHOP

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
name	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
place	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
post	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
pin	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
district	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
email	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
contact	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
socialmeadia	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
logo	varchar	250		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
status	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ownername	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
owneremail	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ownercontact	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ownerage	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
LOGIN_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

BOOKING

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
time	time	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
status	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SLOT_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
USER_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

USER

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
username	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
place	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
pin	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
photo	varchar	300		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
post	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
district	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
contact	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
email	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
dateofbirth	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
gender	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
LOGIN_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

STAFF

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
staffname	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
contact	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
email	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
photo	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
experience	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
place	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
post	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
pin	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
district	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BARBERSHOP_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
LOGIN_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

SLOT

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
fromtime	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
totime	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SERVICE_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

SERVICE

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
servicename	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BARBERSHOP_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
amount	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

SHOPREVIEW

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
time	time	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
review	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
rating	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BARBERSHOP_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
USER_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

PAYMENT

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
time	time	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
amount	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BOOKING_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

DESIGN

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
designname	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
image	varchar	250		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
SERVICE_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

COMPLAINT

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
complaint	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
status	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
replay	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
time	time	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
USER_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

ASSIGN STAFF

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
status	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
BOOKING_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Staff_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

APP REVIEW

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
id	int			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
review	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
time	time	6		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
date	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
USER_id	int			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. TESTING

In general, testing is finding out how well software or a product works. Testing is a set of activity that can be planned and conducted systematically. Testing begins at the module level and work towards the integration of entire computers-based system. Nothing is complete without testing, as it is vital success of the system. Software testing is a process of executing a program or application with the intent of finding the software bugs. Testing is for evaluating a system or its components to find whether it satisfies the specified requirements or not. Testing cannot show the absence of defects, it can only show that errors are present in the software. It can also be stated as the process of validating and verifying that a software program or software or product.

Testing Objectives:

There are several rules that can serve as testing objectives, they are:

1. Testing is a process of executing a program with the intent of finding an error.
2. A good test case is one that has high probability of finding an undiscovered error.
3. A successful test is one that uncovers an undiscovered error.

If testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrates that software functions appear to be working according to the specification, that performance requirements appear to have been met.

There are three ways to test a program:

1. For correctness.
2. For implementation efficiency.
3. For computational complexity.

Tests for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

Test for implementation efficiency attempt to find ways to make a correct program faster or use less storage. It is a code refining process, which reexamines the implementation phase of algorithm development.

Tests for computational complexity amount to an experimental analysis of the complexity of an algorithm or an experimental comparison of two or more algorithms, which solve the same problem

5.1 UNIT TESTING

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit has implemented is producing expected output against given input. In this the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. Unit testing can be done manually but is often automated. In our system, unit testing has been successfully handled.

5.2 INTEGRATION TESTING

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

5.3. SYSTEM TESTING

System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black-box testing, and as such, should require no knowledge of the inner design of the code or logic. As a rule, system testing takes, as its input, of the "integrated" software components that have passed integration testing and also the software system itself integrated with any applicable hardware system. The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together (called assemblages) or between any of the assemblages and the hardware. System testing is a more limited type of testing it seeks to detect defects both within the "inter-assemblages" and also within the system as a whole.

5.4. ACCEPTANCE TESTING

In engineering and its various sub disciplines, acceptance testing is a test conducted to determine if the requirements of a specification or contract are met. It may involve chemical tests, physical test, or performance tests. In systems engineering it may involve black-box testing performed on a system (for example: a piece of software, lots of manufactured mechanical parts. or batches of chemical products) prior to its delivery.

In software testing the ISTQB defines acceptance as: formal testing with respect to user needs, requirements, and business processes conducted to determine whether a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system.

Acceptance testing is also known as user acceptance testing (UAT), end-user testing, and operational acceptance testing (OAT) or field (acceptance) testing. A smoke test may be used as an acceptance test prior to introducing a build of software to the main testing process.

Various forms of acceptance testing are as follows:

1) Alpha and beta testing

Alpha testing takes place at developer's sites, and involves testing of the operational system by internal staff, before it is released to external customers. Beta testing takes place at customers' sites, and involves testing by a group of customers who use the system at their own locations and provide feedback, before the system is released to other customers. The latter is often called "field testing".

5.5. VALIDATION TESTING

In this, all the code modules were tested individually one after the other. The following were tested in all the modules.

1. Loop testing
2. Boundary Value Analysis
3. Equivalence Partitioning Testing

In our case all the modules were combined and given the test data. The combined module works successfully without any side effect on other programs. Everything was found fine working.

5.6. OUTPUT TESTING

Output testing popularly known as Black Box testing. This is the final step of testing. Black box testing is a software testing technique in which functionality of the software under test is tested without looking at the internal code structure. Implementation details and knowledge of internal paths of the software. It just focuses on inputs and output of the software system without bothering about internal knowledge of the software program.

6. ERROR HANDLING

Error handling in software development manages unexpected issues like invalid input, hardware failures, or logical errors. It involves detecting errors via exception handling or status flags, reporting them through logs or messages, diagnosing causes through debugging, and applying solutions like retries, rollbacks, or default values. If recovery is possible, the program continues; otherwise, it terminates gracefully, releasing resources. Transparent feedback ensures users and administrators stay informed. Effective error handling improves software reliability and user experience, requiring careful planning, implementation, and testing.

7. FUTURE ENHANCEMENT

- 1) Automated Appointment Reminders – Send SMS or push notifications to users and barbers for upcoming bookings to reduce no-shows.
- 2) Loyalty & Referral Program – Reward users with discounts and free services for frequent bookings and referring new customers.
- 3) Dynamic Pricing & Discounts – Enable barbershops to set special prices based on demand, time slots, or customer loyalty.
- 4) AI-Based Hairstyle Recommendations (*Future AI Integration*) – Suggest hairstyles based on face shape and trends using machine learning.

- 5) Barber Portfolio & Ratings – Let users view barbers' profiles, past work, and ratings before booking an appointment.
- 6) Multi-Branch Support – Expand the system for barbershops with multiple branches to manage bookings centrally.
- 7) Voice Command Booking (*Future AI Integration*) – Allow users to book appointments through voice assistants.
- 8) Barbershop Inventory Management – Help barbershops track stock levels of grooming products and automate restocking.
- 9) In-App Product Sales – Enable barbershops to sell hair care and grooming products directly through the app.
- 10) Video Consultation for Styling Advice – Provide virtual consultations where users can discuss hairstyle preferences before booking.

8. CONCLUSION

The BOOK MY BARBER system is a comprehensive Barbershop Management System designed to streamline appointment scheduling, staff management, and customer engagement. By integrating web-based functionalities for barbershops, staff, and administrators, along with an Android application for users, the system ensures a seamless and efficient booking experience.

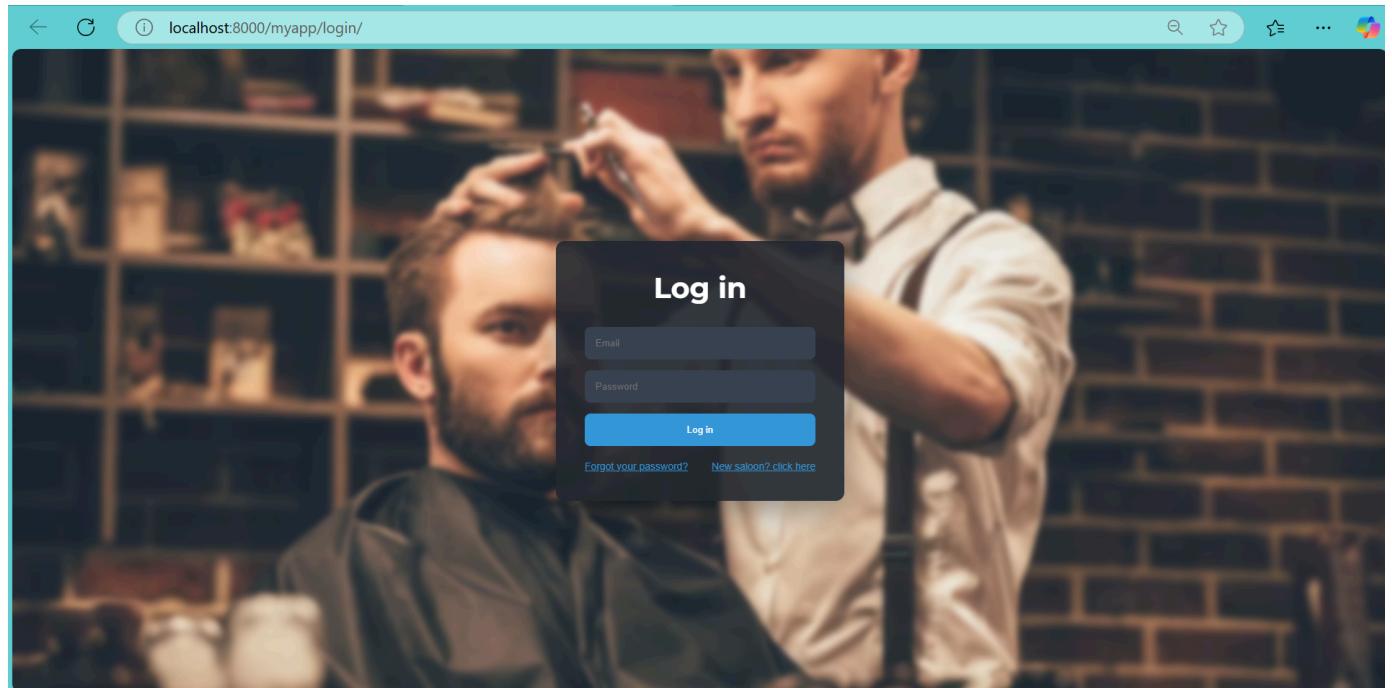
With essential features such as queue management, payment integration, staff assignment, and review systems, the platform enhances customer satisfaction and operational efficiency. The modern, responsive design and structured database management provide a reliable and scalable solution for barbershops of all sizes.

Moving forward, the system can be further enhanced with automation, loyalty programs, and advanced analytics, making it a future-proof solution for barbershop businesses.

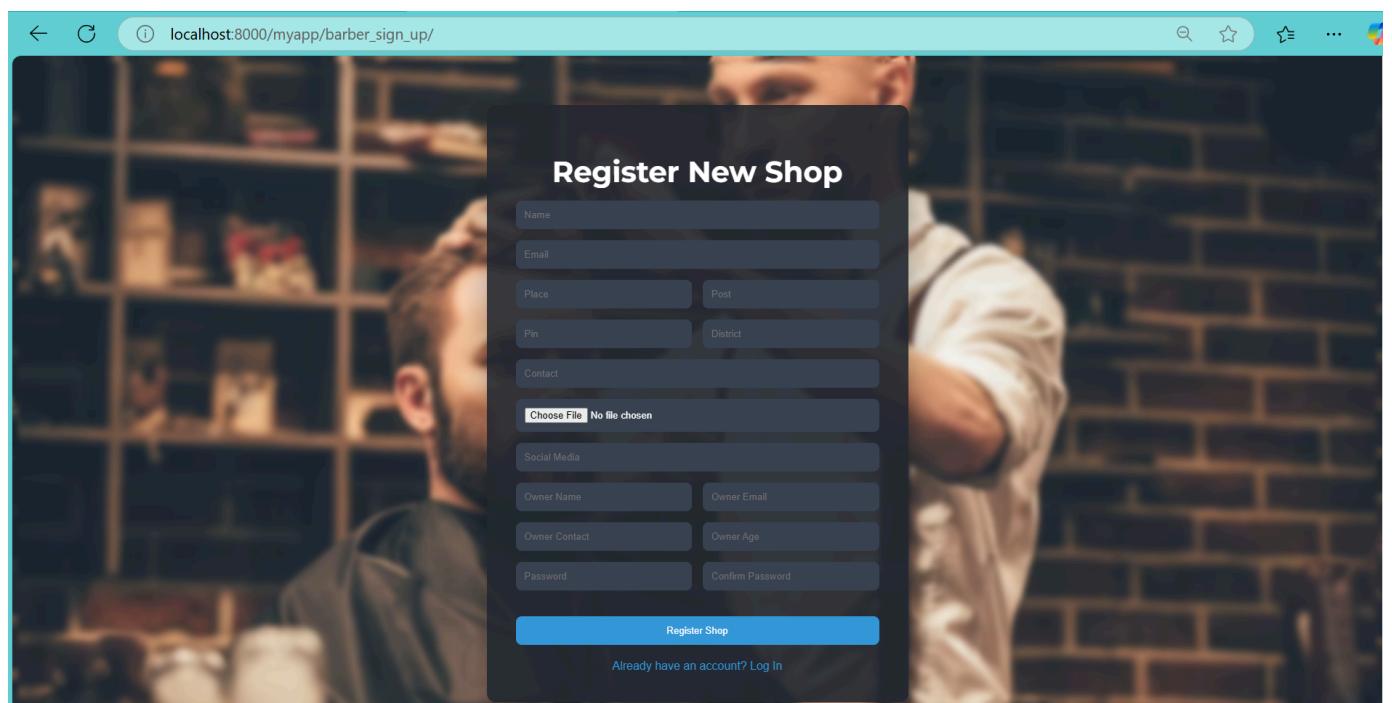
9. APPENDIX

9.1 SAMPLE SCREENSHOTS

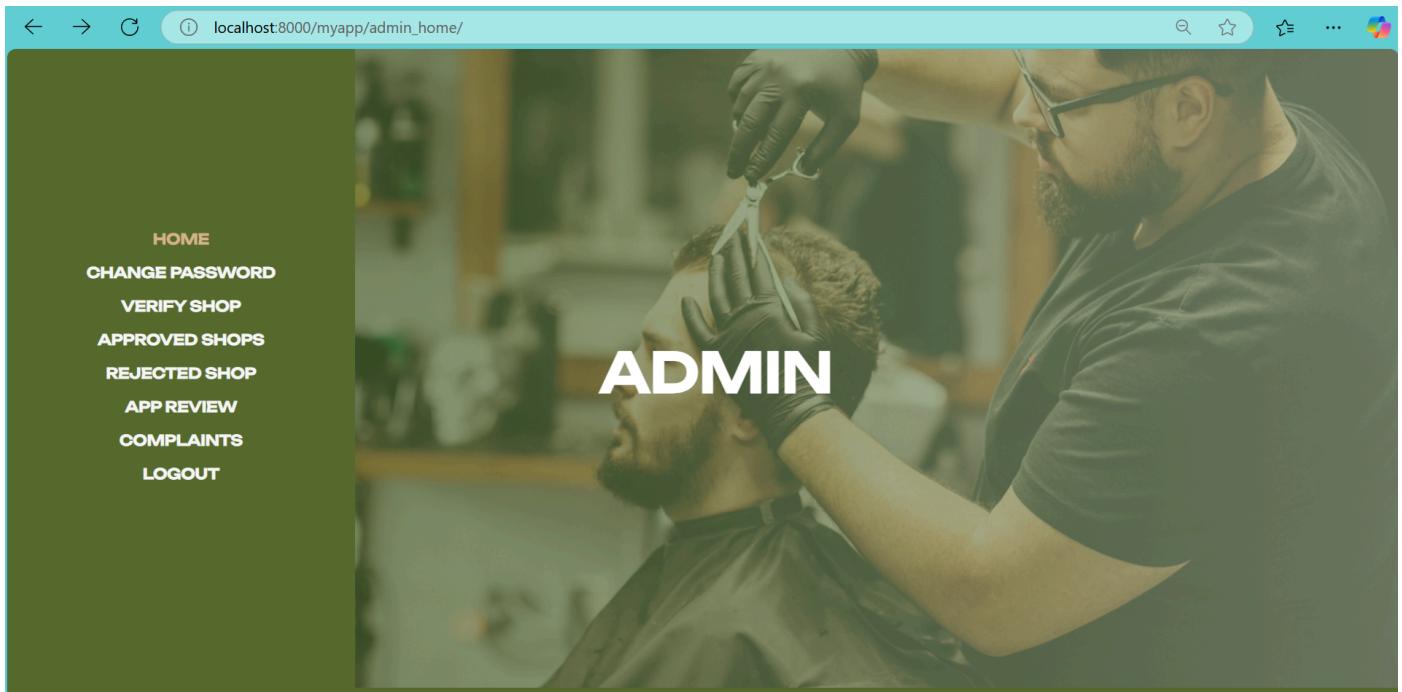
LOGIN PAGE



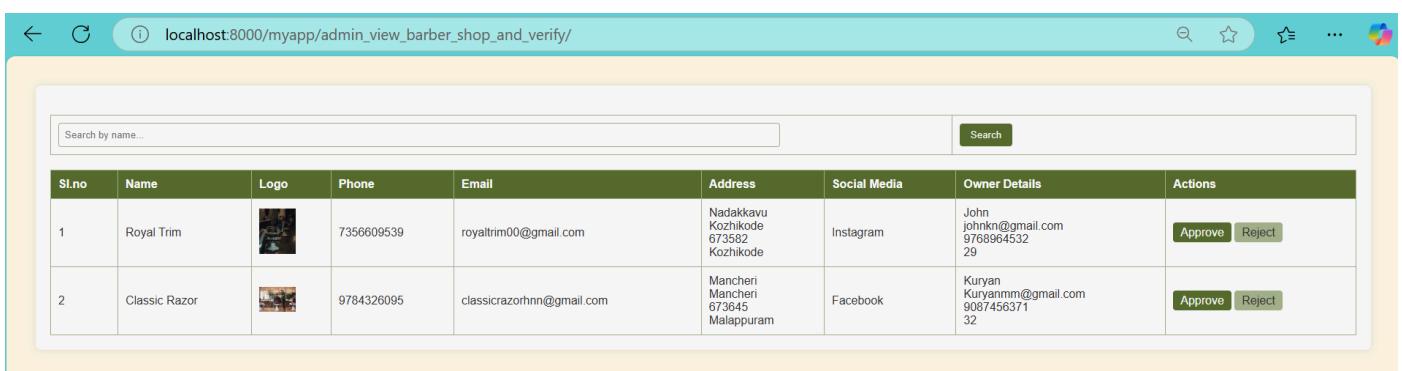
BARBERSHOP REGISTRATION PAGE



ADMIN HOMEPAGE



ADMIN-SHOP VERIFICATION PAGE



Sl.no	Name	Logo	Phone	Email	Address	Social Media	Owner Details	Actions
1	Royal Trim		7356609539	royaltrim00@gmail.com	Nadakkavu Kozhikode 673582 Kozhikode	Instagram	John johnkn@gmail.com 9768964532 29	<button>Approve</button> <button>Reject</button>
2	Classic Razor		9784326095	classicrazorhn@gmail.com	Mancheri Mancheri 673645 Malappuram	Facebook	Kuryan Kuryanmm@gmail.com 9087456371 32	<button>Approve</button> <button>Reject</button>

ADMIN-VIEW APP REVIEW PAGE

localhost:8000/myapp/admin_View_app_review/

From Date To Date

Sl.No	Date	Review	User	Time
1	March 4, 2025	Super easy to book appointments! No more waiting at the shop. Love it!	ram	11:55 a.m.
2	March 6, 2025	Saved me so much time! Booking slots in advance is a game-changer.	ayan	2:20 a.m.

BARBER HOMEPAGE

localhost:8000/myapp/barber_home/

BookMyBarber

Home Add Design Add Service Add Staff Logout

welcome to Book My Barber.com

Shop's Panel

Manage Your Shop

About Our Shop

[VIEW DESIGN](#) [VIEW PAYMENT](#) [VIEW PROFILE](#)

[VIEW REVIEW & RATING](#) [VIEW SERVICES](#) [VIEW STAFF](#)

localhost:8000/myapp/barber_home/

BookMyBarber

Home Add Design Add Service Add Staff Log Out

BookMyBarber

Meet Staffs



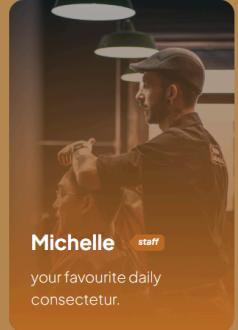
Sofia Boss
your favourite BMB lives tempor.



Sandra Manager
your favourite daily lives.



Jackson Senior
your favourite daily lives.



Michelle staff
your favourite daily consectetur.

ADD DESIGN PAGE

localhost:8000/myapp/barber_add_design/

BookMyBarber

Home Add Design Add Service Add Staff Log Out Add Staff

Design form

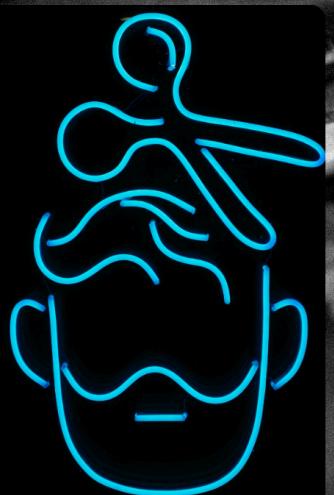
Add Design

Service: Cutting

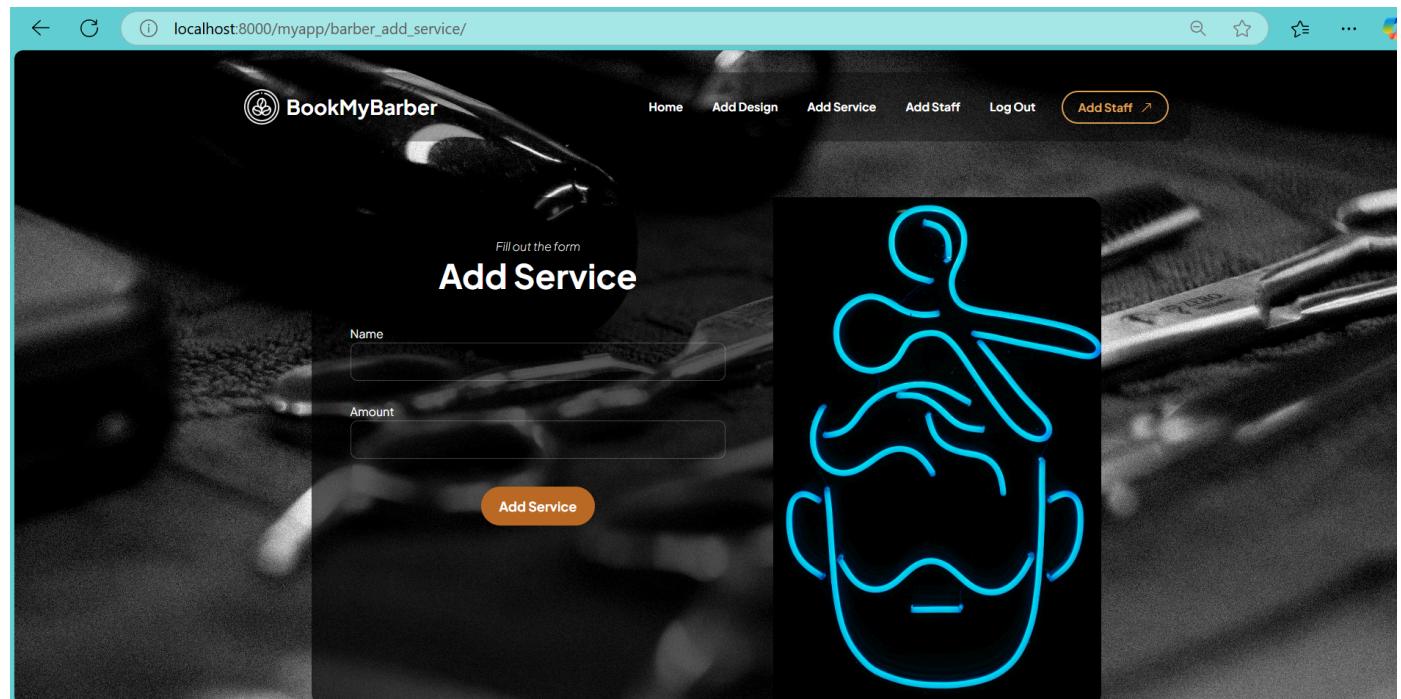
Design:

Image: Choose File No file chosen

Add Design



ADD SERVICE PAGE



localhost:8000/myapp/barber_add_service/

BookMyBarber

Home Add Design Add Service Add Staff Log Out Add Staff

Fill out the form

Add Service

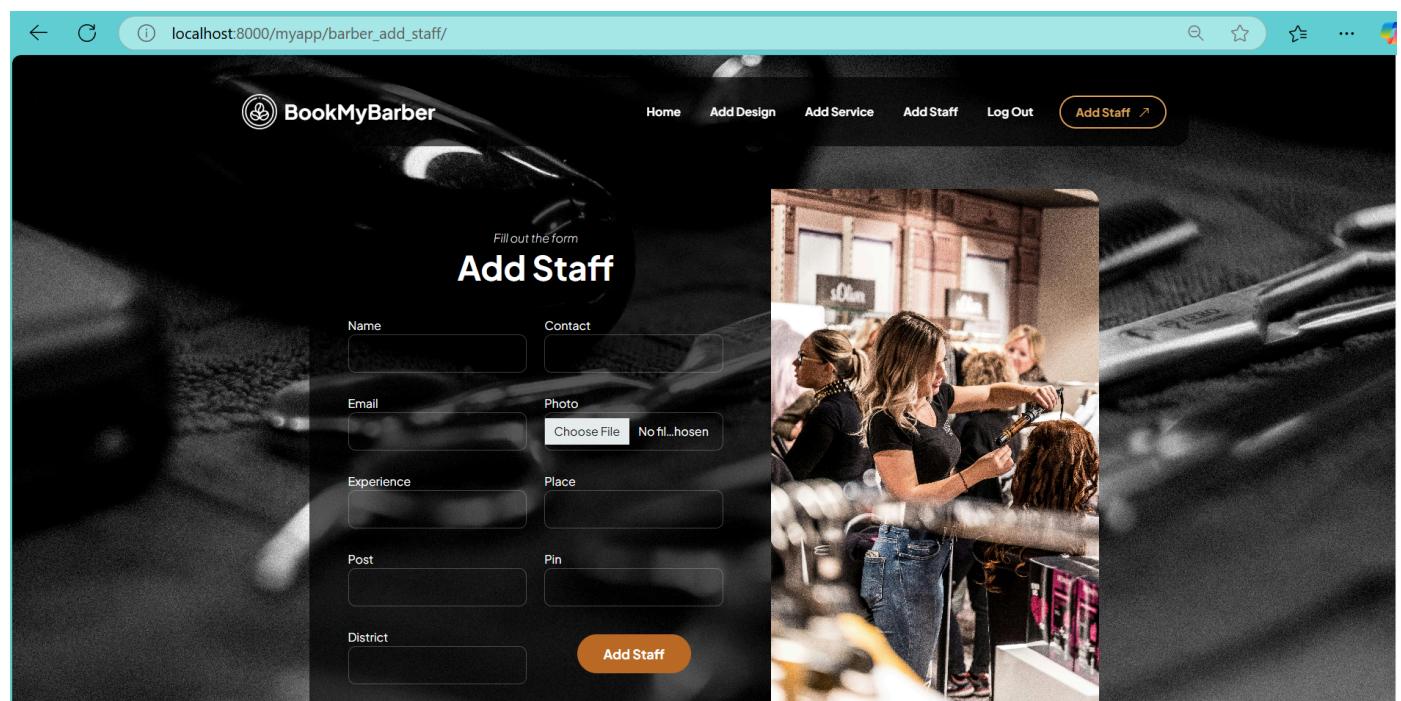
Name

Amount

Add Service

Blue neon-style graphic of a head and shoulders.

ADD STAFF PAGE



localhost:8000/myapp/barber_add_staff/

BookMyBarber

Home Add Design Add Service Add Staff Log Out Add Staff

Fill out the form

Add Staff

Name

Contact

Email

Photo

Choose File No fil...hosen

Experience

Place

Post

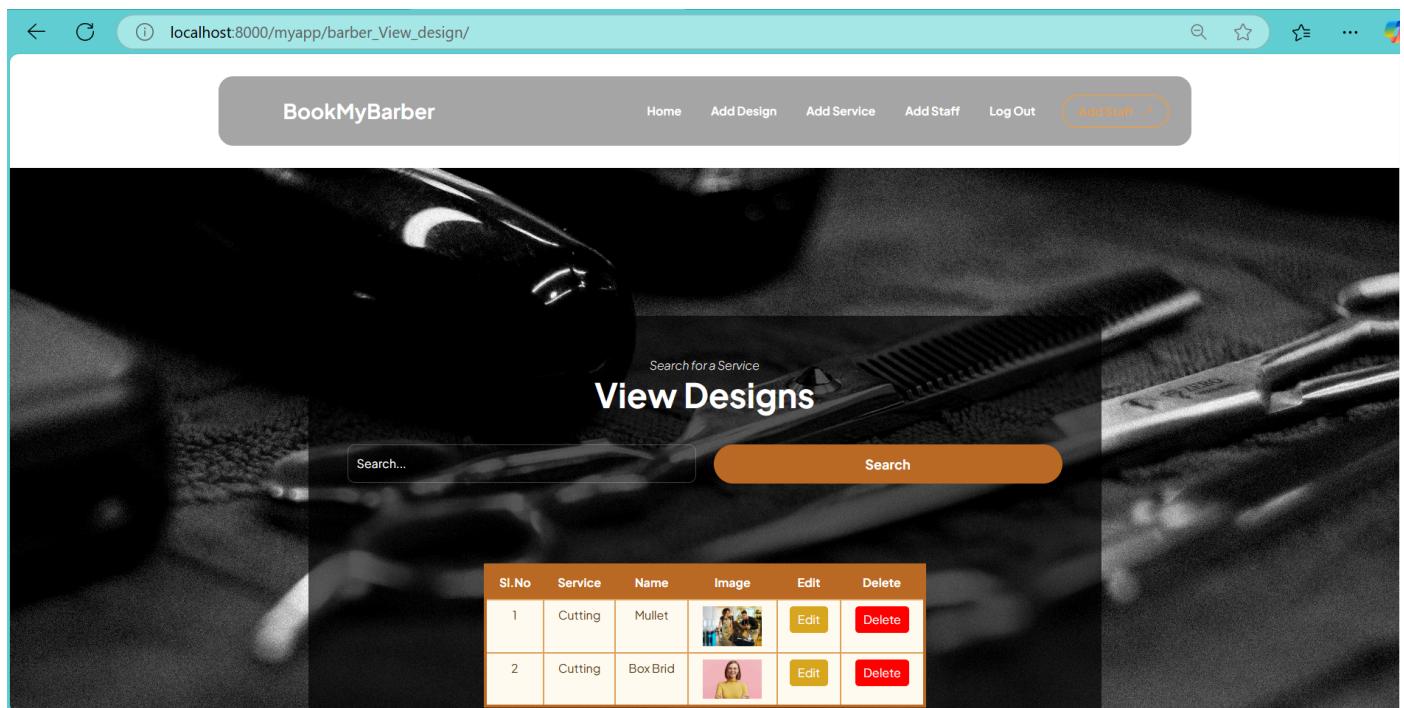
Pin

District

Add Staff

Photo of a woman working on a client's hair.

VIEW DESIGNS PAGE



localhost:8000/myapp/barber_View_design/

BookMyBarber

Home Add Design Add Service Add Staff Log Out Add Staff ↗

Search for a Service

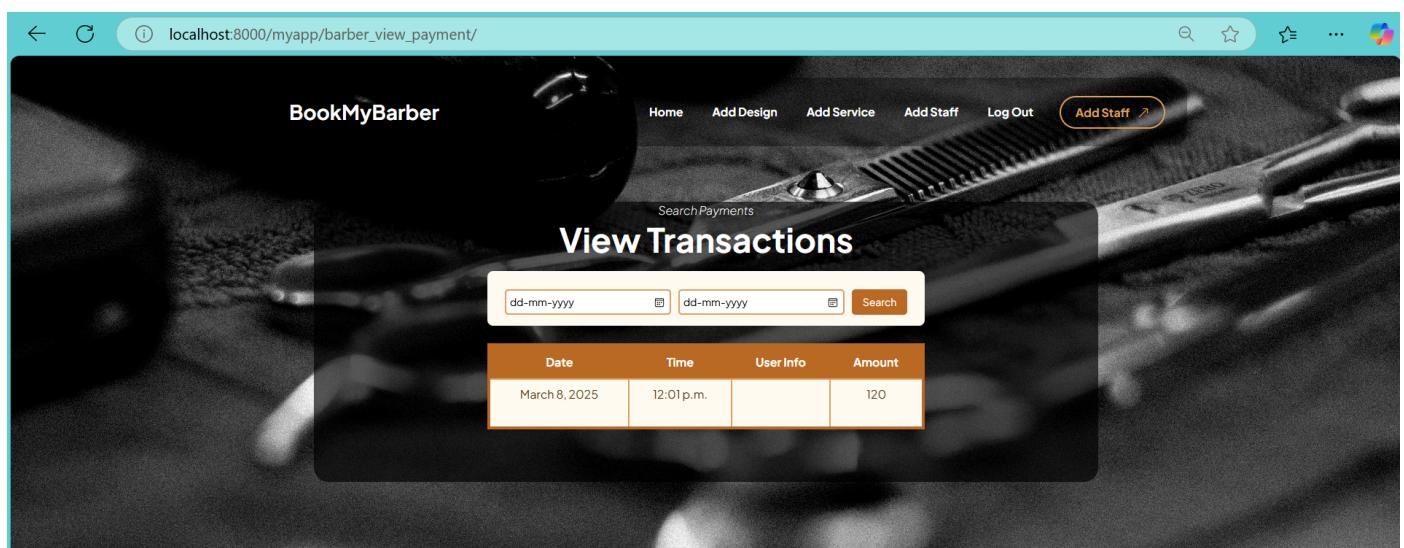
View Designs

Search...

Search

Si.No	Service	Name	Image	Edit	Delete
1	Cutting	Mullet		Edit	Delete
2	Cutting	Box Brid		Edit	Delete

VIEW PAYMENTS PAGE



localhost:8000/myapp/barber_view_payment/

BookMyBarber

Home Add Design Add Service Add Staff Log Out Add Staff ↗

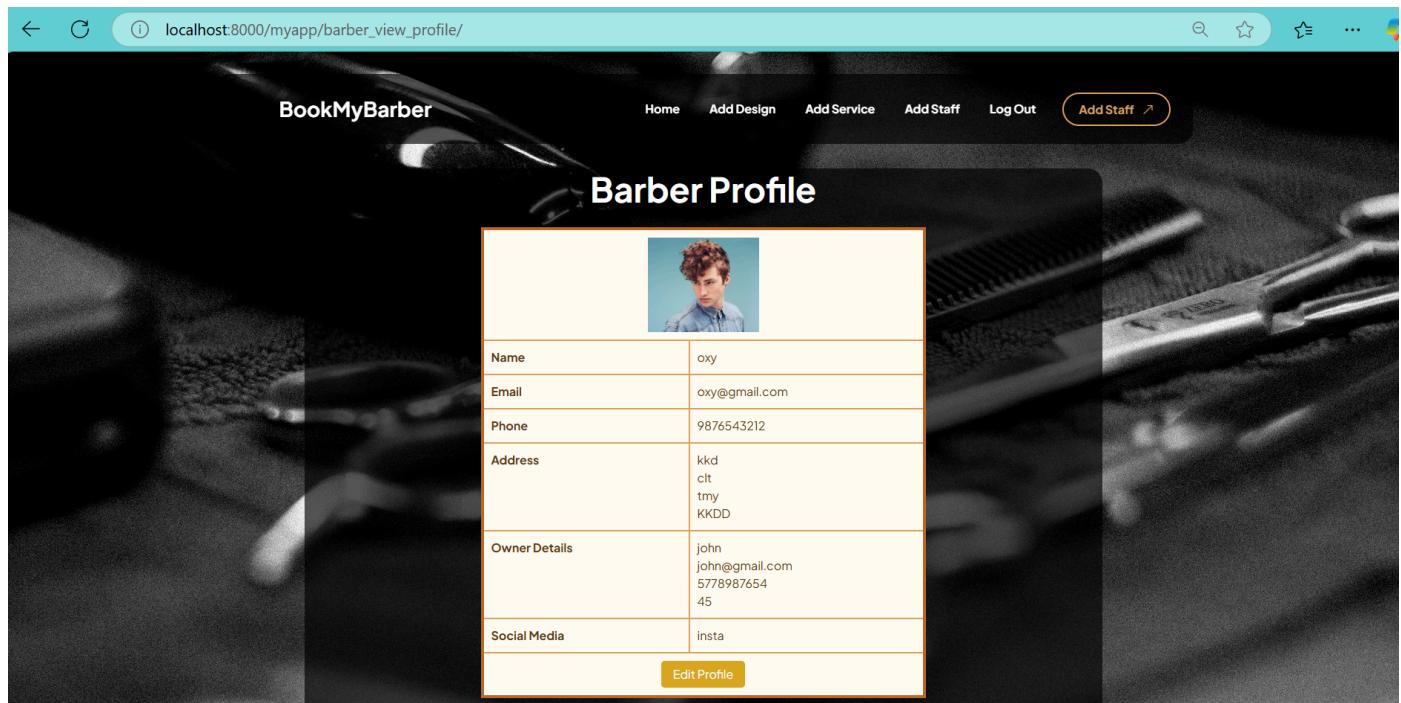
Search Payments

View Transactions

dd-mm-yyyy dd-mm-yyyy Search

Date	Time	User Info	Amount
March 8, 2025	12:01 p.m.		120

SHOP-VIEW PROFILE PAGE



localhost:8000/myapp/barber_view_profile/

BookMyBarber

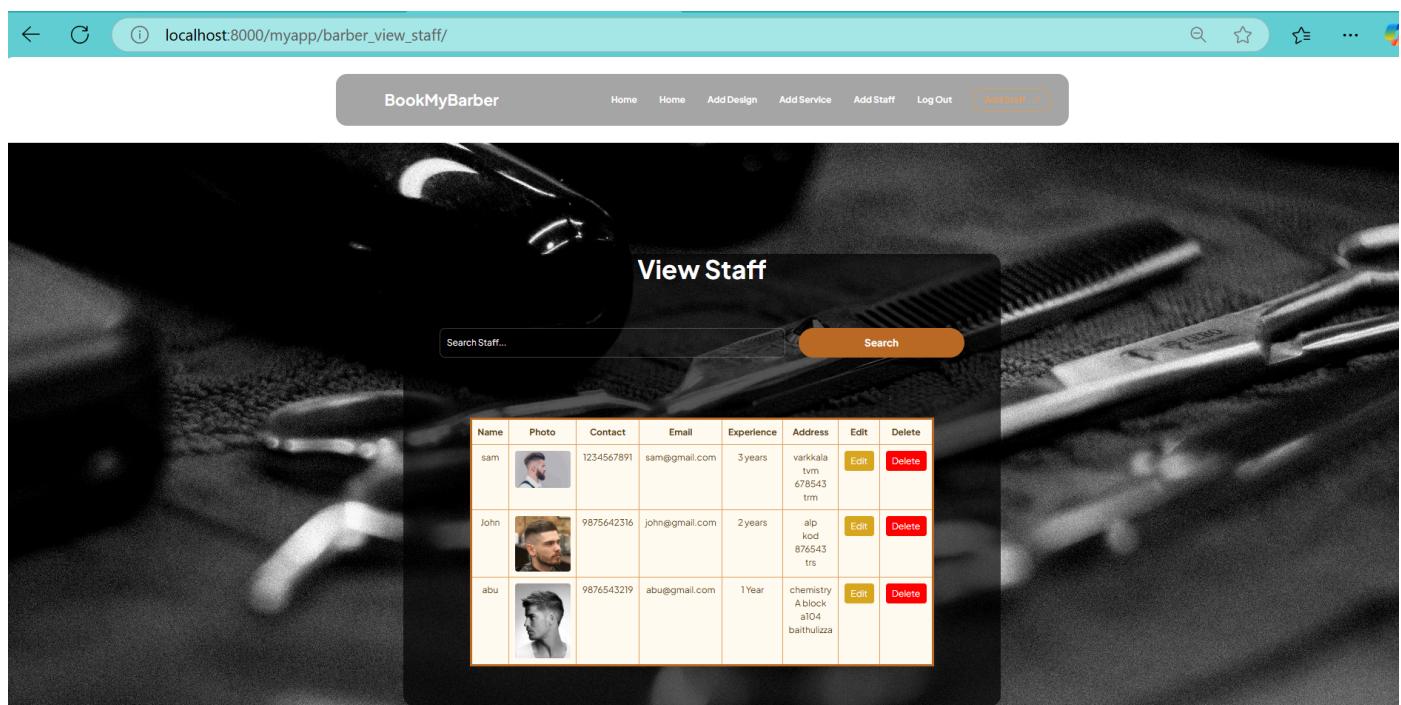
Home Add Design Add Service Add Staff Log Out Add Staff

Barber Profile

	
Name	oxy
Email	oxy@gmail.com
Phone	9876543212
Address	kkd clt tmy KKDD
Owner Details	john john@gmail.com 5778987654 45
Social Media	insta

Edit Profile

VIEW STAFF PAGE



localhost:8000/myapp/barber_view_staff/

BookMyBarber

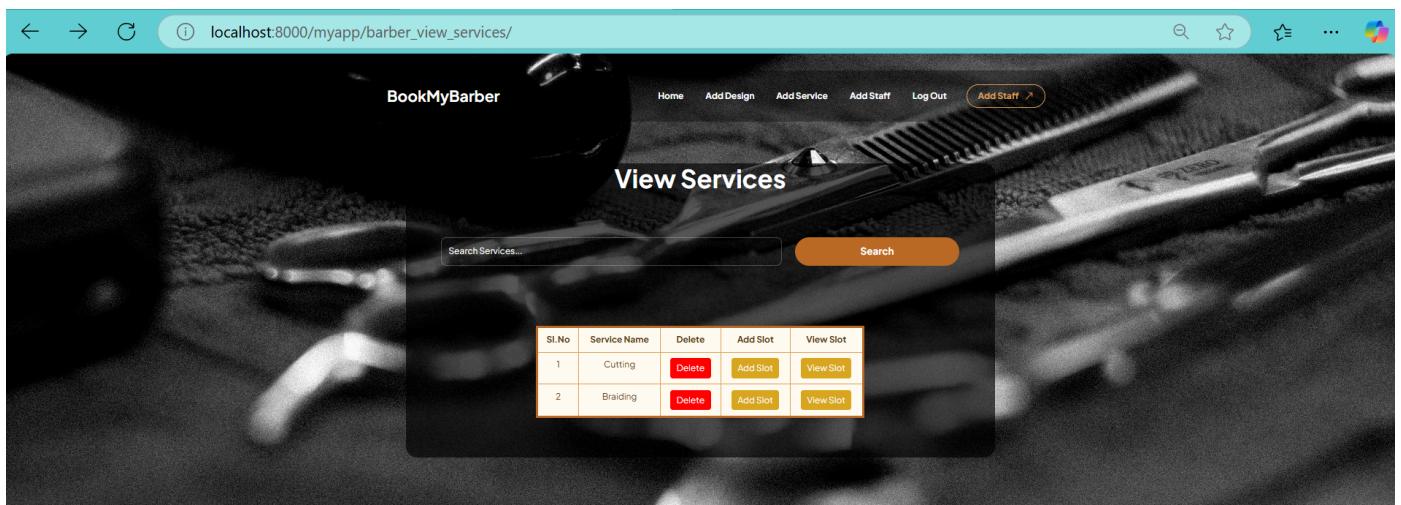
Home Add Design Add Service Add Staff Log Out Add Staff

View Staff

Search Staff... Search

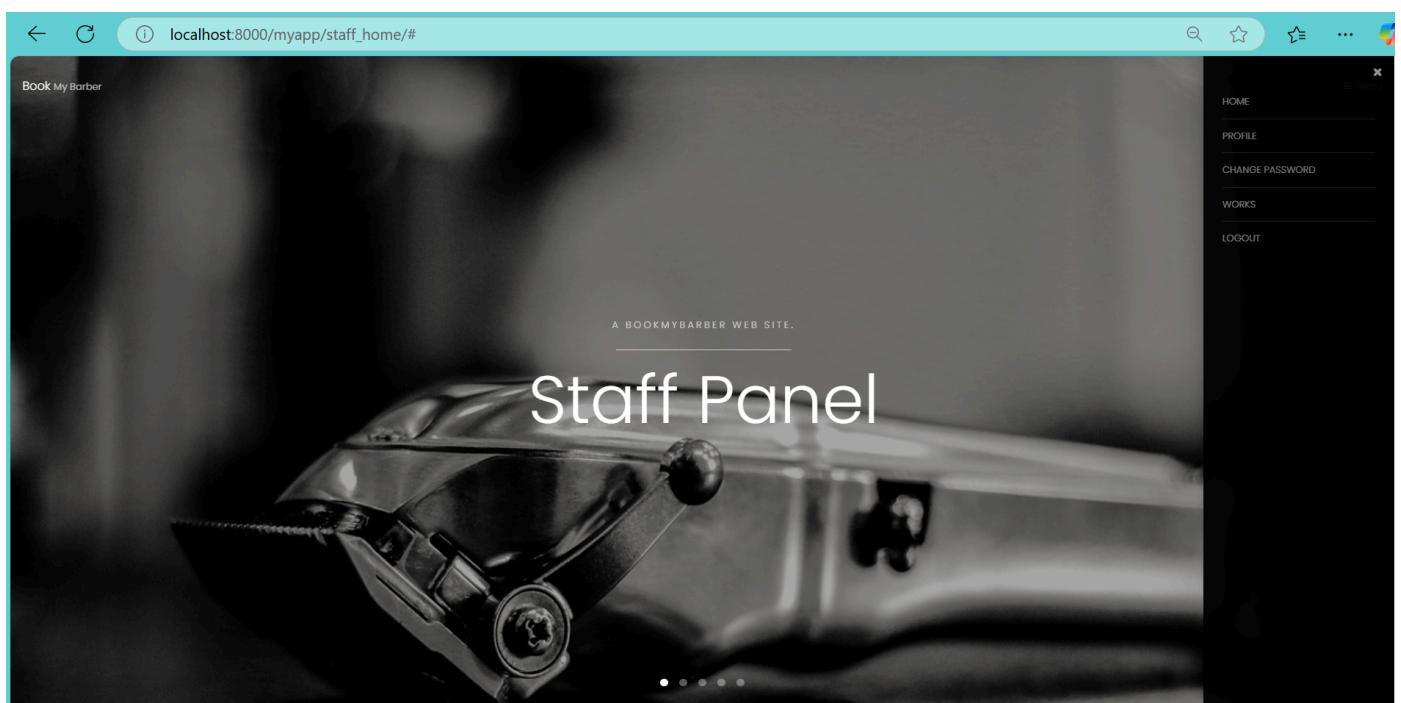
Name	Photo	Contact	Email	Experience	Address	Edit	Delete
sam		1234567891	sam@gmail.com	3 years	varikala tm 678543 tm	Edit	Delete
John		9875642316	john@gmail.com	2 years	alp kod 876543 trs	Edit	Delete
abu		9876543219	abu@gmail.com	1 Year	chemistry A block a104 baitulizza	Edit	Delete

VIEW SERVICES PAGE



A screenshot of a web browser showing the 'View Services' page of the BookMyBarber application. The page has a dark background with a black and white image of a barber's tools (scissors, comb, and brush) in the background. The title 'View Services' is centered above a search bar with the placeholder 'Search Services...'. Below the search bar is a table with two rows of service data. The table has columns for Sl.No, Service Name, Delete, Add Slot, and View Slot. The first row contains '1' and 'Cutting' with buttons for 'Delete', 'Add Slot', and 'View Slot'. The second row contains '2' and 'Braiding' with the same set of buttons. The top navigation bar includes links for Home, Add Design, Add Service, Add Staff, Log Out, and Add Staff (with a plus icon). The URL in the address bar is 'localhost:8000/myapp/barber_view_services/'.

STAFF HOMEPAGE



A screenshot of a web browser showing the 'Staff Panel' homepage of the BookMyBarber application. The page has a dark background with a black and white image of a vintage-style electric shaver in the foreground. The title 'Staff Panel' is prominently displayed in the center. At the bottom of the page, there is a horizontal navigation bar with five dots. The top navigation bar includes links for Home, Profile, Change Password, Works, and Logout. The URL in the address bar is 'localhost:8000/myapp/staff_home/#'.

localhost:8000/myapp/staff_home/#

HOME
PROFILE
CHANGE PASSWORD
WORKS
LOGOUT

CHECK YOUR ASSIGNED WORKS HERE
Assigned Works
WORKS

CHECK YOUR PROFILE HERE
View Profile
PROFILE

STAFF-VIEW PROFILE PAGE

localhost:8000/myapp/staff_view_profile/

Book My Barber
BOOKMYBARBER APPLICATION
staff panel
HOME
PROFILE
CHANGE PASSWORD
WORKS
LOGOUT

BOOKMYBARBER
View Profile

Name	sam
Email	sam@gmail.com
Phone	1234567890
Experience	3 years
Address	varkkala tvm 678543 tmm

CHANGE PASSWORD

The screenshot shows a web browser window with the URL `localhost:8000/myapp/staff_change_password/`. The page title is "Change Password". The interface includes fields for "Current Password", "New Password", and "Confirm Password", with a "CONFIRM" button. The background features a dark theme with the "staff" role indicator.

ASSIGNED WORKS

The screenshot shows a web browser window with the URL `localhost:8000/myapp/staff`. The page title is "View Assigned Works". It features a search bar with "From Date" and "To Date" fields and a "SEARCH" button. Below is a table with columns: "Sl.no", "Booking", "Date", and "Status". One row is shown: Sl.no 1, Booking 12:42 p.m., Date March 8, 2025, and Status finished. The background features a dark theme with the "staff panel" role indicator.