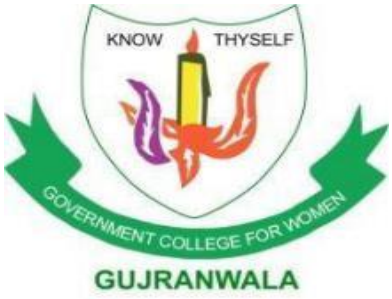


Submitted By

Aqsa Shehzadi	**042302**
Ayesha M Yaqoob	**042308**
Hafiza Attika Imran	**042338**

SESSION 2020-2024

Govt. Graduate College for Women Satellite Town, Gujranwala

Govt. Graduate College (W) Satellite Town, Gujranwala

Final Documentation

ResideMe

STATEMENT OF SUBMISSION

This is to certify that the following students have successfully completed the final project named as: **ResideMe** at The University of the Punjab, Gujranwala Campus, and to fulfill the partial requirement of the degree of **BSIT.**

Sr. No.	Student's Roll No	Student's Name
1	042302	Aqsa Shehzadi
2	042308	Ayesha Muhammad Yaqoob
3	042338	Hafiza Attika Imran

Internal Examiner

Name: Ms. Sadaf Sikandar

Head of the Department

Govt. Graduate College (W),

Satellite town, Gujranwala

Head of Department

Name: Ms. Sadaf Sikandar

Head of the Department

Govt. Graduate College (W),

Satellite town, Gujranwala

Acknowledgements

This project is the direct result of teamwork. We sincerely thank the instructors and students who have shared their suggestions with us over the last few weeks.

Many thanks go to our project coordinator and supervisor, whose management helped us to get this project produced in a timely and efficient manner.

And finally, we thank to our parents and all the other people who weren't directly involved in this project but who have stood by us the whole way, we appreciate your encouragement and support. We are also thankful to our friends and families whose silent support led us to complete our project.

Write to us

We welcome your response to this project. If there is anything you want to mention about the improvement of this project, please let us know:

Aqsa Shehzadi	aqshaqsh1234@gmail.com
Ayesha Muhammad Yaqoob	aisha.yaqoob095@gmail.com
Hafiza Attika Imran	attikamughal2003@gmail.com

For Our Beloved Parents and Teachers

"Dedicated to our parents and teachers without whose wholehearted support, encouragement and guidance it would have been impossible for us to make this project."

Table of Contents

.....1
Chapter 1.....1
Project Feasibility Report 1
1. Introduction.....2
1.1 Project Feasibility Report.....2
1.1.1 Technical Feasibility.....2
1.1.2. Operational Feasibility.....2
1.1.3. Economic Feasibility.....3
1.1.4. Schedule Feasibility 3
1.1.5. Specification Feasibility 3
1.1.6. Information Feasibility.....3
1.1.7. Motivational Feasibility 3
1.1.8. Legal & Ethical Feasibility 3
Chapter 2.....4
Project scope 4
2.1 Project Scope 5
.....6
Chapter 3.....6
Project Costing.....6
3. Project Costing:.....7
3.1 Project Cost Estimation By Function Point Analysis: 8
3.2 Project Cost Estimation By Function Point Analysis: 9
Chapter 4.....11
Project Planning.....11
4. Project Planning.....12
4.1 Project Scope 12
4.2 CMP Critical Path.....12
4.2.1 Estimate Activity Completion.....12
4.2.2 Estimate Activity Completion Time.....13
4.3 Network Diagram.....14
4.4 Identify the Critical Path.....14
4.5 Gantt Chart.....14
4.6 Introduction To Team Members and Their Skill Set 15

4.7 Tools and Technology with Reasoning 15
4.8 Risk List 16
..... 17
Chapter 5 17
Requirement Engineering 17
5. Requirement Engineering 18
5.1 Introduction 18
5.2 Existing System 18
5.3 Scope of the system 19
5.4 Summary of Requirements 19
5.5 Identifying External Entities 19
5.6 Context Level Data Flow Diagram 20
5.7 Capture "shall" Statement 20
5.8 Allocate Requirements 21
5.9 Prioritize Requirements 21
5.10 Requirements Trace Ability Matrix 22
5.11 High-Level Case Diagram 23
5.11.1 Admin Use Case 23
5.11.2 User Use Case 24
Chapter 6 25
Use Case Description 25
6. Introduction 26
6.1 Use Case Description 26
UC _ 1: Connect to Wi-Fi 26
6.2. Use Case Diagram 31
Chapter 7 32
Sequence Diagram 32
7. Sequence Diagram 33
7.1 Login 33
7.2 Room Booking 34
7.3 View Facilities 35
7.4 Profile Updation 36
7.5 Contact Us 37
7.6 Generating Report 38
7.7 Logout 39

Chapter 8 40
Collaboration Diagram 40
8. Collaboration Diagram 41
8.1 Login 41
8.2 Room Booking 42
8.3 View Facilities 42
8.4 Profile Updation 43
8.5 Contact Us 43
8.6 Generating Report 44
8.7 Logout 44
Chapter 9 45
Domain Model 45
9.1 Domain Model 46
..... 47
Chapter 10 47
Class Diagram 47
10.1 Design Class Diagram 48
Chapter 11 49
Data Model 49
11.1 Data Model 50
Chapter 12 51
Interfaces 51
12. Project Interfaces 52
12.1 Admin 52
12.1.1 Login 52
12.1.2 Admin Dashboard 53
12.1.3 Admin Profile 54
12.1.4 Manage Residents 55
12.1.5 Manage Rooms 56
12.1.6 Manage Facilities 57
12.1.7 Manage Booking Request 58
12.1.8 Manage Feedbacks 59
12.1.9 Live Streaming 60
12.1.10 Admin Notice Board 61
12.2 User 62

12.2.1 Login.....62
12.2.2 User Profile.....63
12.2.3 Home Page.....64
12.2.4 Rooms Page.....65
12.2.5 Facilities Page.....66
12.2.6 Booking Request Details.....67
12.2.7 Approved Booking Request Detail.....68
12.2.8 User Report.....69
12.2.9 User Feedbacks.....70
12.2.10 Contact Us Page.....71
12.2.11 About Page.....72

Chapter 1
Project Feasibility Report

1. Introduction

In today's fast paced environment, ResideMe provides a streamlined solution for managing hostel administration. This platform allows admins to efficiently handle user records, room and bed assignments and booking requests all from a single dashboard. Admins can track room availability, and respond to booking requests with ease. Meanwhile residents can log in to view room options, book rooms, submit feedback and manage their profiles anytime, anywhere. With ResideMe management becomes simple, organized, and accessible, ensuring a smooth experience for both admins and residents.

a. Project Feasiblity
b. Project Scope
c. Project Costing
d. Critical Path Method Analysis (CMP Analysis)
e. Gantt Chart
f. Introduction to team members
g. Tools and Technologies
h. Risk List

1.1 Project Feasibility Report

After careful analysis, we've determined that the ResideMe project is feasible. All necessary resources, including software, skilled personnel, and data, are available to achieve the project's goals. The planning for each phase ensures timely completion within the available resources, meeting user requirements. The system is designed to be flexible for future updates, making it a practical and beneficial solution for hostel management. There are many types of feasibilities.

- Technical
- Operational
- Economic
- Schedule
- Specification
- Information
- Motivational
- Legal and Ethical

1.1.1 Technical Feasibility

The ResideMe project has been thoroughly examined from all technical aspects. It successfully meets all the technical requirements because we have the necessary tools, technologies, and expertise to develop it. Our team members possess the required technical skills to implement technologies and build a robust, user-friendly platform. With resources and capabilities available, we are confident that the project can be developed smoothly and effectively.

1.1.2. Operational Feasibility

ResideMe will make hostel management easier. The platform will help admins manage rooms, and bookings in a simple way. Residents can easily book rooms, send feedback, and manage their profiles. The system is user-friendly, so both admins and residents can use it without problems. The ResideMe enhances overall efficiency and operational effectiveness.

1.1.3. Economic Feasibility

The project is affordable. It will need some initial investment in development, but once set up, the platform will save time and reduce manual work for admins. This means it will save money in the long run by making hostel management more efficient.

1.1.4. Schedule Feasibility

The project can be completed within a reasonable time frame (about 5-6 months). It is divided into phases, so the core features can be developed first, and additional features can be added after. The timeline is realistic and achievable.

1.1.5. Specification Feasibility

The requirements are clear and achievable. Features like user registration, room booking, are straightforward. The project can be built with the available resources, and all features can be delivered as per the plan.

1.1.6. Information Feasibility

The system will securely store all the necessary data, like user information and booking records. Only authorized users (admins) will be able to access this information, ensuring data privacy and security.

1.1.7. Motivational Feasibility

ResideMe will solve many problems faced by hostel admins, like managing bookings. The platform will make these tasks easier, which will motivate both admins and residents to use it regularly.

1.1.8. Legal & Ethical Feasibility

The ResideMe project will adhere to all legal requirements related to data protection and privacy, ensuring that user information is handled securely. We will implement clear terms and conditions to inform users about data usage. Ethically, the system will respect user privacy, provide transparent processes, and ensure fair handling of all data, building trust with both admins and residents.

Chapter 2
Project scope

2.1 Project Scope

The scope of the system is defined on the basis of various functionalities provided by the system, The scope can be explained as:

The ResideMe project is designed to provide an efficient and easy-to-use system for managing hostel operations, benefiting both admins and residents. For admins, the system offers features like managing user records, room bookings, and bed assignments, along with the ability to track and generate reports on room bookings. Admins will also have control over booking requests, allowing them to approve or reject resident bookings, and they will be able to issue unique IDs and passwords to users for login purposes. On the resident side, the system allows users to easily register, log in using credentials provided by the admin, and browse available rooms and facilities. Once a resident finds a suitable room, they can submit a booking request and wait for admin approval. Residents will also have the ability to send feedback and submit feedback, as well as manage their personal profiles within the system. In terms of security, ResideMe ensures the protection of all user data, making sure that only authorized users can access the system. The platform is designed to be flexible, allowing for future updates and changes to meet evolving user needs. Overall, the ResideMe project aims to streamline management by making the booking more transparent and efficient, improving the overall experience for both admins and residents.

.

Chapter 3
Project Costing

3. Project Costing:

Function-oriented software matrices use a measure of the functionality delivered by the application as a normalization value. Since "Functionality cannot be measured directly, it must be derived indirectly using other direct measures". Function-oriented metrics were first proposed by Albrecht, who suggested a measure called the function point. Function Point are derived using an empirical relationship based on countable (direct) measure of software's information domain and assessment of software complexity.

For Admin:

We have chosen Function Point Analysis for our Project Cost Estimation:
Number of external inputs: 6
Number of external outputs: 4
Number of external inquiries: 2
Number of internal logic files: 7
Number of external interface files: 2

External Inputs:

- Admin login
- Adding residents, rooms, or facilities
- Approving / rejecting booking requests
- Update feedback status
- Posting Notices
- Updating the admin profile

External Outputs:

- List of residents, rooms, facilities, and booking requests
- Live streaming
- Feedback reports
- Notices displayed to users

External Inquiries:

- Searching for residents, rooms, or facilities
- Viewing booking details or feedbacks

Internal Logic Files:

- Resident records
- Facility records
- Room records
- Booking request record
- Bed records
- Feedback records
- Notices

External Interface Files:

- Video streaming service
- Email / notification service

3.1 Project Cost Estimation By Function Point Analysis:

Type of Component	Complexity of Components			
	Low	Average	High	Total
External Inputs	2*2=4	2*4=8	2*6=12	24
External Outputs	0*2=0	2*4=8	2*6=12	20
External Inquiries	1*2=2	1*4=4	0*6=0	6
Internal Logic Files	1*2=2	6*4=24	0*6=0	26
External Interface Files	0*2=0	0*4=0	2*6=12	12
Total Number of Unadjusted Function Points for Admin				**88**

For User:

We have chosen Function Point Analysis for our Project Cost Estimation:
Number of external inputs: 4
Number of external outputs: 3
Number of external inquiries: 2
Number of internal logic files: 3
Number of external interface files: 1

External Inputs

- User login
- Booking requests
- Feedback submission
- User profile updates

External Outputs

- Approved / rejected booking statuses
- User reports
- Notifications about room assignments, updates or feedback status

External Inquiries

- Checking room availability
- Viewing approved bookings and updated feedback status

Internal Logic Files

- User profiles
- Approved booking records
- Feedbacks

External Interface Files

- Email / notification service

3.2 Project Cost Estimation By Function Point Analysis:

Type of Component	Complexity of Components			
	Low	Average	High	Total
External Inputs	4*2=8	0*4=0	0*6=0	8
External Outputs	2*2=4	1*4=4	0*6=0	8
External Inquiries	2*2=4	0*4=0	0*6=6	4
Internal Logic Files	0*2=0	3*4=12	0*6=0	12
External Interface Files	1*2=2	0*4=0	0*6=0	2
Total Number of Unadjusted Function Points for User				**34**

Total Number of Unadjusted Function Points = Total Number of Unadjusted Function Points for Admin + Total Number of Unadjusted Function Point for User
Total Number of Unadjusted Function Points = 88+34 = 122

Calculation of (Fi):

Sr.#	Questions	Scale
1	Does the system require heavy configuration?	2
2	Is communication required?	5
3	Are there distributed processing functions?	0
4	Is performance critical?	4
5	Will the system run in an existing, heavily utilized operational environment?	2
6	Does the system require online data entry?	4
7	Does the online data entry require the input transaction to be built over operations?	3
8	Are the files updated online?	4
9	Are the inputs, output, files, or inquiries complex?	3
10	Is the internal processing complex?	2
11	Is the code designed to be reusable?	3
12	Are conversion/installation included in the design?	1
13	Is the system designed for multiple installations in different organizations?	2
14	Is the application designed to facilitate change?	4
Value Adjusted Factors (Fi)		**39**

Calculation of Function Point (FP):

FP $_{estimated}$ = Count Total * (0.65 + 0.01 * (Fi))
= 122 * (0.65 + 0.01 * 39)
= 122 * 1.04
= 127

Final Function Point Calculation is:

Final Adjusted FP =UFC*VAF
=122 * 127
= 15494

Cost per Function Point:

Labor Rate = Rs. 20000
Cost per FP = Labor Rate / Productivity parameter
= 20000 / 10
= 2000

Total Project Cost:

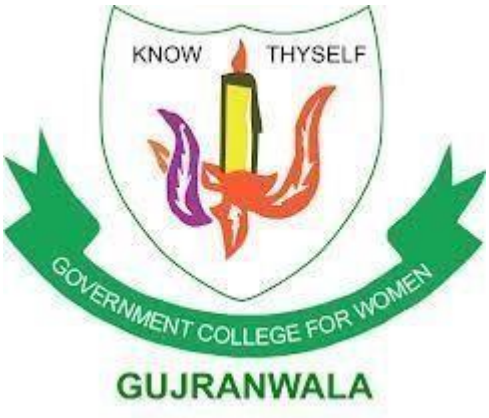
Total Project Cost = FP $_{estimated}$ * Cost Per FP
= 127 * 2000
=254000

Total Estimated Effort:

Total Estimated Effort = FP est. / productivity parameter
= 127 / 10
= 12.7

Duration of Project:

Duration of Project = Effort months / No. of persons
= 18/3
= 6 months



Chapter 4
Project Planning

4. Project Planning

4.1 Project Scope

The ResideMe project is designed to provide an efficient and easy-to-use system for managing hostel operations, benefiting both admins and residents. For admins, the system offers features like managing user records, room bookings, and bed assignments, along with the ability to track and generate reports on room bookings. Admins will also have control over booking requests, allowing them to approve or reject resident bookings, and they will be able to issue unique IDs and passwords to users for login purposes. On the resident side, the system allows users to easily register, log in using credentials provided by the admin, and browse available rooms and facilities. Once a resident finds a suitable room, they can submit a booking request and wait for admin approval. Residents will also have the ability to provide feedback and submit messages, as well as manage their personal profiles within the system. In terms of security, ResideMe ensures the protection of all user data, making sure that only authorized users can access the system. The platform is designed to be flexible, allowing for future updates and changes to meet evolving user needs. Overall, the ResideMe project aims to streamline hostel management by making the booking more transparent and efficient, improving the overall experience for both admins and residents.

4.2 CMP Critical Path

4.2.1 Estimate Activity Completion

Project Scope Definition	**A**
Requirement Gathering	**B**
Analysis	**C**
Design	**D**
Coding (Development)	**E**
Testing & Debugging	**F**
Implementation (Deployment)	**G**
Documentation & User Manual	**H**

Determine the Sequence of the Activities

Some activities are dependent on the completion of others. There are many activities that are dependent on each other.
Following activities are dependent on one another:

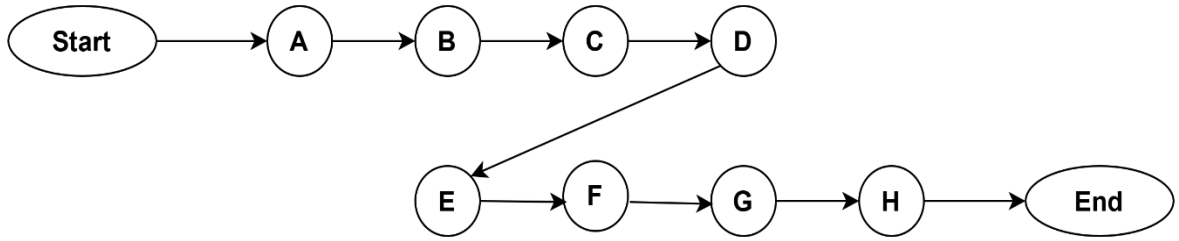
- Analysis ⇨ Project Scope Definition, Requirement Gathering
- Logical Desingn ⇨ Analysis
- Coding ⇨ Logical Design
- Testing and Debugging ⇨ Coding
- Implementation ⇨ Testing and Debugging, Coding

4.2.2 Estimate Activity Completion Time

Activity ID	Predecessors	Duration (Days)
A	None	7
B	A	17
C	B	14
D	C	22
E	D	74
F	E	22
G	F	15
H	G	15

4.3 Network Diagram

Once the activities and their sequence have been defined, the CPM diagram can be drawn.

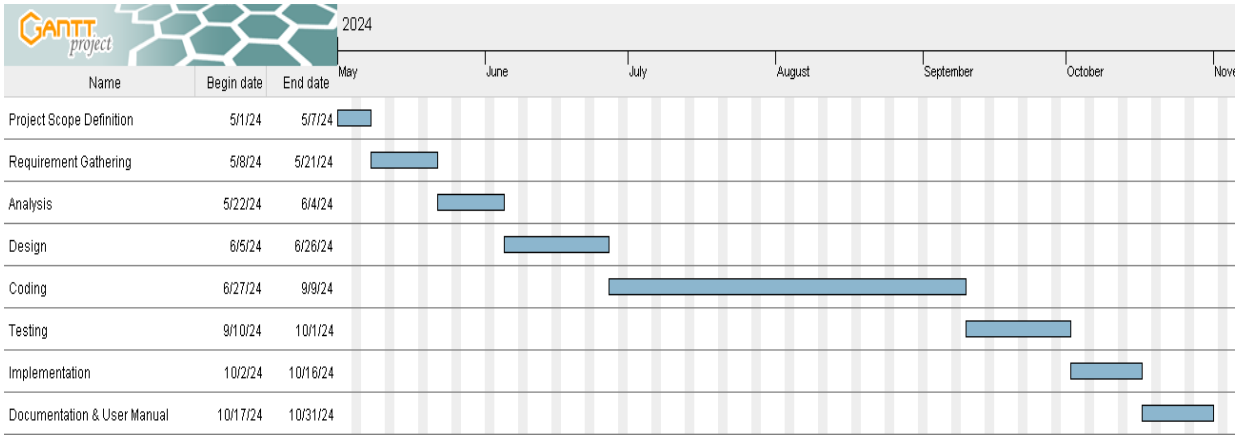


Path: A → B → C → D → E → F → G → H 7+17+14+22+74+22+15+15=**186**

4.4 Identify the Critical Path

Activity	Duration	ES	EF	LS	LF	TS
A	7	0	7	0	7	0
B	17	7	24	7	24	0
C	14	24	38	24	38	0
D	22	38	60	38	60	0
E	74	60	134	60	134	0
F	22	134	156	134	156	0
G	15	156	171	156	171	0
H	15	171	186	171	186	0

4.5 Gantt Chart



4.6 Introduction To Team Members and Their Skill Set

Student's Roll No	Student's Name
042308	Ayesha Muhammad Yaqoob
042338	Hafiza Attika Imran
042302	Aqsa Shehzadi

Ayesha M Yaqoob

Ayesha Muhammad Yaqoob is leader of the group and backend programmer for this project. She is responsible for implementing and maintaining backend infrastructure, ensuring seamless functionality.

Hafiza Attika Imran

Hafiza Attika Imran contributions include writing comprehensive project documentation in Microsoft Word and creating detailed UML diagrams to visualize the system's structure and interactions. She is also responsible for the frontend development, focusing on creating user friendly and attractive designs.

Aqsa Shehzadi

Aqsa Shehzadi as the Full-Stack Developer for the ResideMe project, responsible for both frontend and backend development, creating user-friendly and attractive designs and strong backend system.

4.7 Tools and Technology with Reasoning

Visual Studio

A development environment where you'll write, test, and debug your code efficiently for the entire project.

JavaScript

JavaScript, often abbreviated as JS, is a programming language and core technology of the Web. It will adds interactive features likes form validation & dynamic content updates, improving the user experience on ResideMe.

Bootstrap

A front-end framework that will help make your site responsive, meaning it looks good on all screen sizes and devices.

HTML

HTML is a markup language used by the browser to manipulate text, images, and other content, in order to display it in the required format. It will structure the content of your web pages, defining the layout and elements on each page of ResideMe.

CSS
Styles HTML elements, controlling visual aspects like colors, fonts, & layouts to make ResideMe visually appealing.

jQuery
jQuery is a JavaScript library, simplifies tasks like animations, form validations, and event handling, creating a smooth & responsive interface for ResideMe.

Laravel
A PHP framework that manages backend logic, database interactions, & routing, handling the server-side functionality of ResideMe.

Microsoft Word 2013
Microsoft Office 2013 is used to prepare the documentation of the Project in a professional manner.

GanttProject
GanttProject is a free tool for creating Gantt charts to plan and manage projects. It helps you schedule tasks, assign resources, task progress, and visualize time easily.

MySQL DB
A relational database system used to securely store & manage data, including user information, bookings, rooms, and transactions.

OBS Studio
OBS Studio is a free tool for recording and streaming videos. For ResideMe, it can be used to live stream via a virtual camera, enabling the admin to monitor live streaming in real-time. This feature allows admins to oversee activities or interactions on the platform efficiently.

Draw.io
A diagramming tool used to create flowcharts & organizational charts, helping visualize the structure & workflow.

4.8 Risk List
- Protecting user data from unauthorized access.
- Preventing unplanned system downtime.
- Ensuring system can handle growth in users and data.
- Maintaining accurate records for users, rooms, and bookings.
- Avoiding double booking or room availability errors.
- Securing login and registration functions.
- Gathering and responding to user feedback regularly.

Chapter 5
Requirement Engineering

5. Requirement Engineering

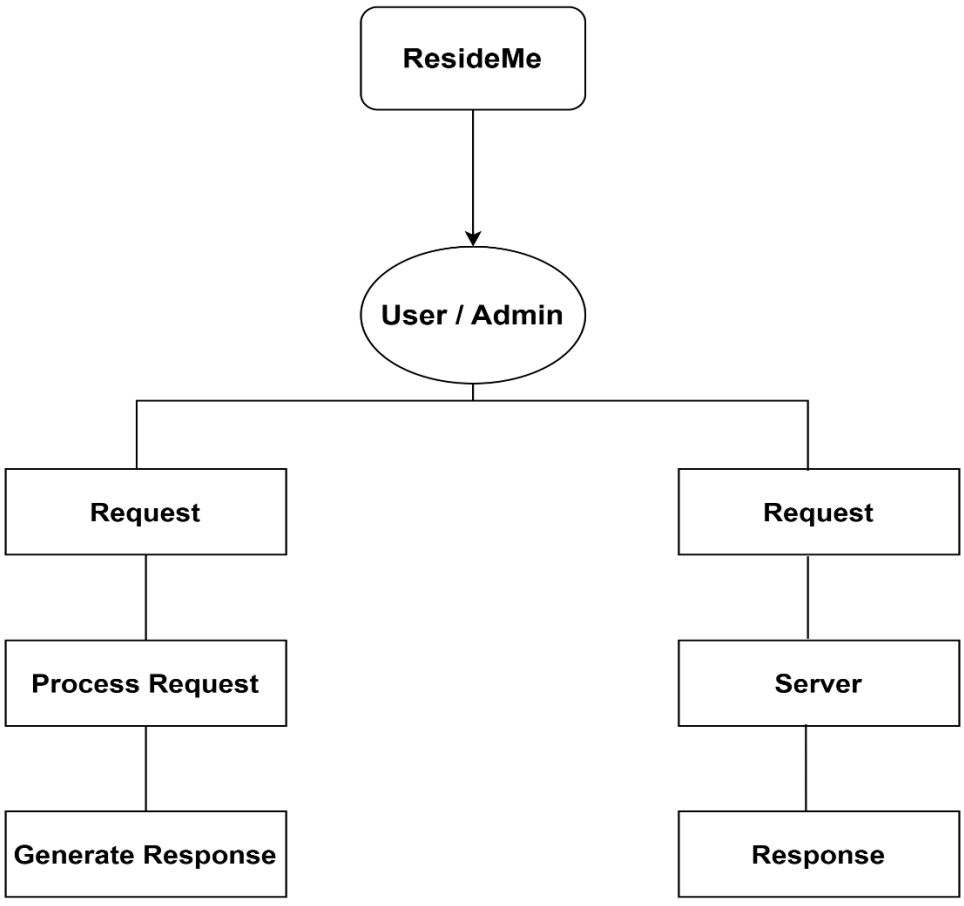
5.1 Introduction

ResideMe is an advanced System designed to address the inefficiencies and challenges of traditional manual systems. This web application provides a comprehensive solution for managing hostel-related information, including rooms, students, facilities, and registrations. By modern technologies, ResideMe aims to deliver a user-friendly, error-free, secure, and efficient management system customized to the unique needs of hostel operations.

5.2 Existing System

In many hostels, management work is done manually using paper records or simple spreadsheets. This method takes a lot of time, often leads to mistakes, and causes data problems. Finding information quickly is difficult, and there is no option to access it from a distance, which makes it hard to make quick decisions. Tracking and managing resources is not efficient, and physical records can easily be lost or damaged. Creating accurate reports is also a slow process, making it harder to make timely and effective decisions.

Business Organization Chart



5.3 Scope of the system

ResideMe aims to automate and streamline hostel tasks with a user-friendly interface, robust security measures, and scalability to handle growth. It provides real-time accessibility, advanced reporting capabilities, seamless integration with existing systems, and ongoing support to ensure efficient and reliable hostel operations.

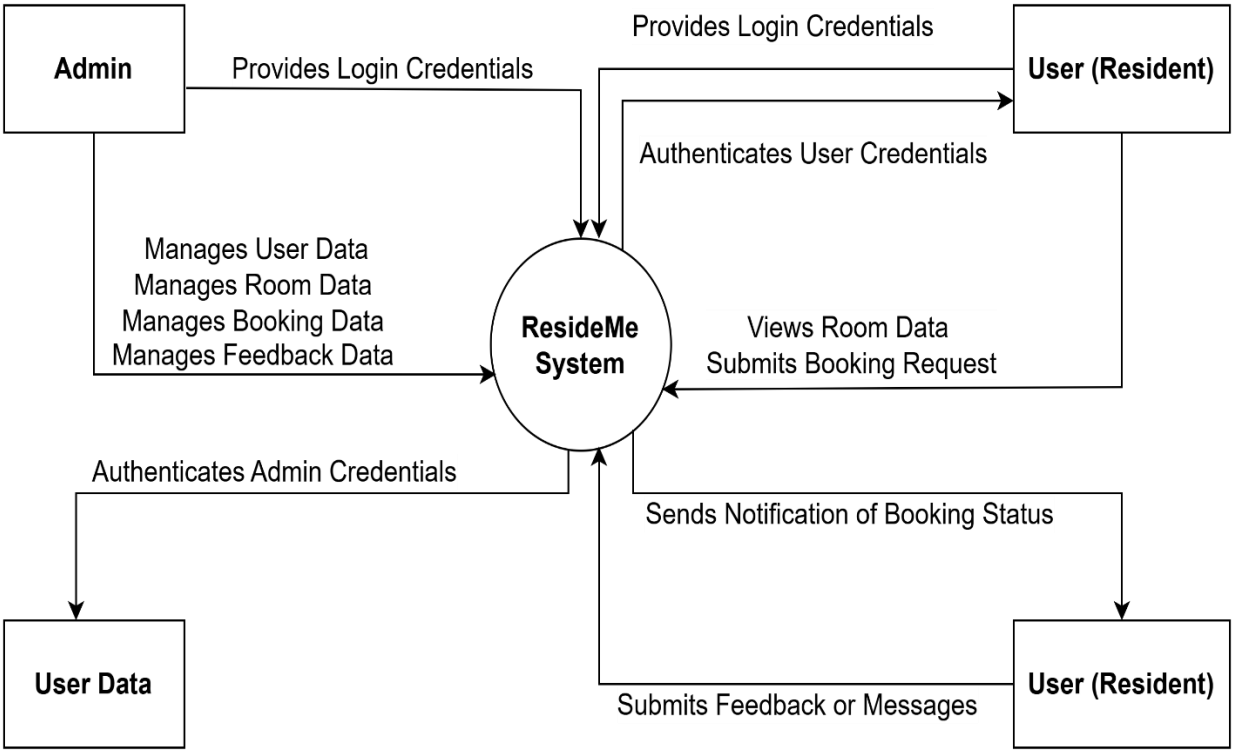
5.4 Summary of Requirements

1. **Admin Features**
   - Manage user records, room assignments.
   - Approve/reject booking requests.
   - Track room availability.
2. **User Features**
   - Login and manage personal profiles.
   - View available rooms and facilities.
   - Book rooms and track booking status.
   - Provide feedback.
   - Access and download booking reports.
3. **System Features**
   - Secure authentication for admin and users.
   - Generate detailed reports.
   - Real-time room and resource tracking.
   - User friendly interface with responsive design.

5.5 Identifying External Entities

1. **Admin**
   Manages system operations like bookings, and reports.
2. **User**
   Views rooms, books rooms, gives feedback, and downloads reports.
3. **Wi-Fi Network**
   Ensures connectivity for web application access.
4. **Database**
   Stores user, room, and transaction data securely.

5.6 Context Level Data Flow Diagram



5.7 Capture "shall" Statement

Para #	External Entity	Initial Requirement
1.0	User	User shall log in to use the web application.
1.0	System	The system shall maintain the record of the user.
1.0	System	The system shall authenticate the user.
1.0	System	The system shall connect to Wi-Fi.
1.0	User	User shall be able to send the notifications to devices.
1.0	User	User shall be able to check recent notifications.
1.0	User	User shall be able to edit their profile.
1.0	System	The system shall maintain the user profile record.
1.0	System	The system shall allocate rooms to residents.
1.0	User	User shall register and manage resident profiles.
1.0	System	The system shall generate reports for management review.
1.0	System	The system shall ensure data security through encryption and access controls.
1.0	System	The system shall provide backup and recovery features.

5.8 Allocate Requirements

Para #	Initial Requirements	Use Case Name
1.0	A user shall login to the web application.	Login
1.0	User shall view and edit their profile in the web application.	User profile
1.0	User shall view the current status of their room and facilities.	View room status
1.0	User shall be able to book a room using ID.	Room Booking
1.0	User shall receive notifications from the system.	Notifications
1.0	User shall be logout from the web application.	Logout

5.9 Prioritize Requirements

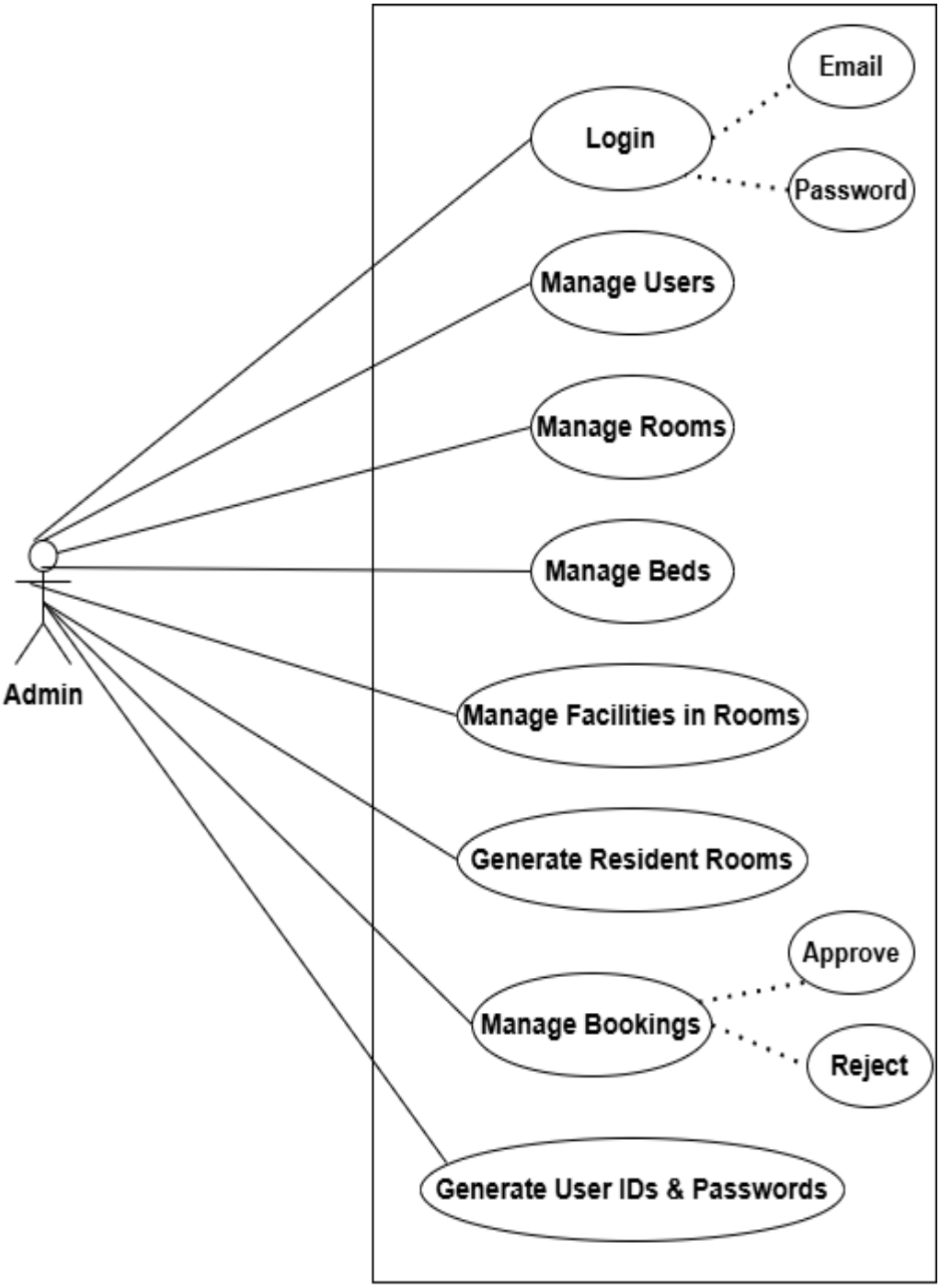
Para #	Initial Requirements	Use Case Name	Use Case ID	Rank
1.0	A user shall login to the web application.	Login	UC 01	High
1.0	Shall view and edit their profile in the web application.	User profile	UC 02	High
1.0	User shall view the current status of their room and facilities.	View current status	UC 03	High
1.0	User shall provide feedback through the system.	Feedback	UC 04	Medium
1.0	User shall book a room using their ID.	Room booking	UC 05	Medium
1.0	User shall receive notifications from the system.	Notifications	UC 06	Medium
1.0	User shall be logout from the web application.	Logout	UC 07	Low

5.10 Requirements Trace Ability Matrix

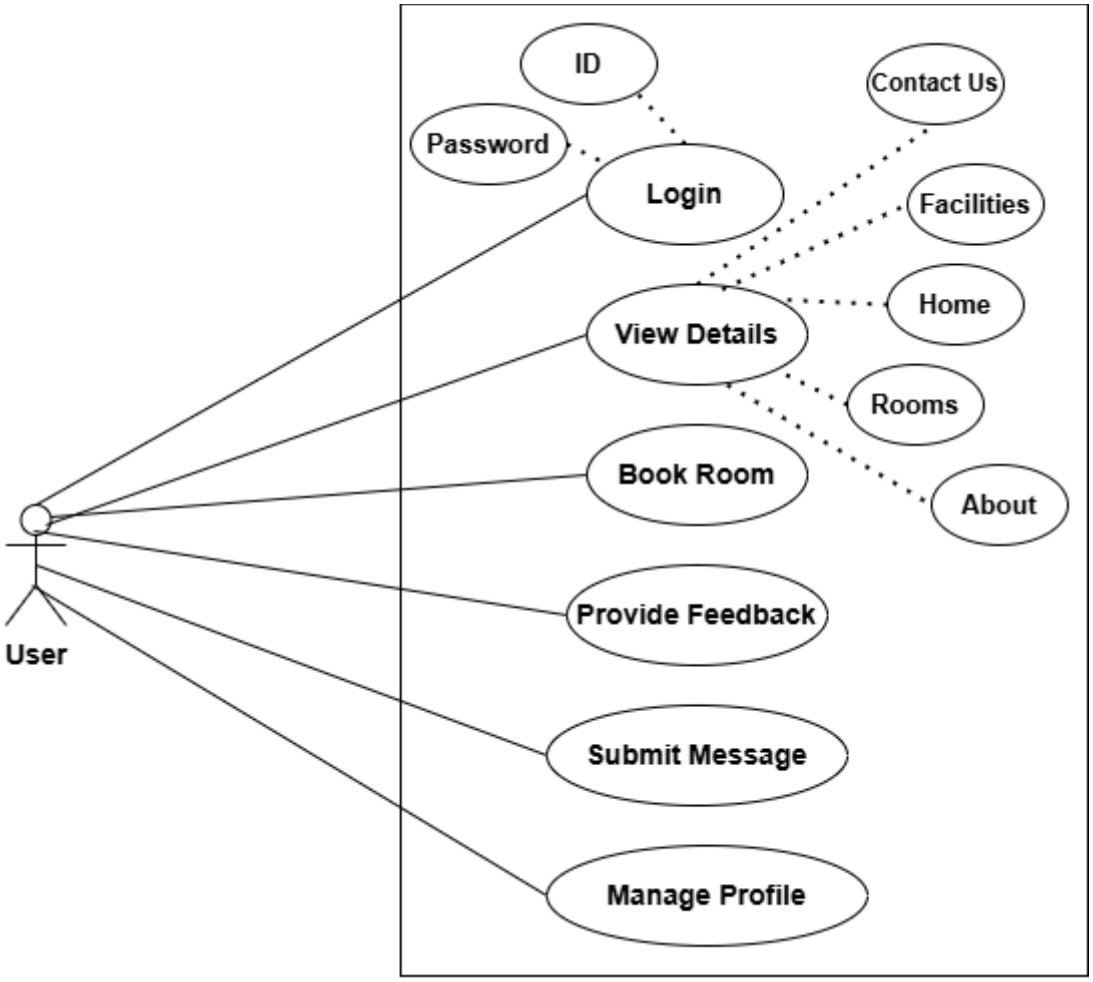
Para #	Initial Requirements	Use Case Name	Build	Category
1.0	Admin shall be able to manage rooms.	Manage rooms	Admin	Admin
1.0	Admin shall be able to register residents into the system.	Residents registration	Admin	Admin
1.0	Admin shall be able to manage facilities.	Manage facilities	Admin	Admin
1.0	User shall register for a room using ID.	Room booking	User	User
1.0	User shall be able to provide feedback through the system.	Feedback system	User	User

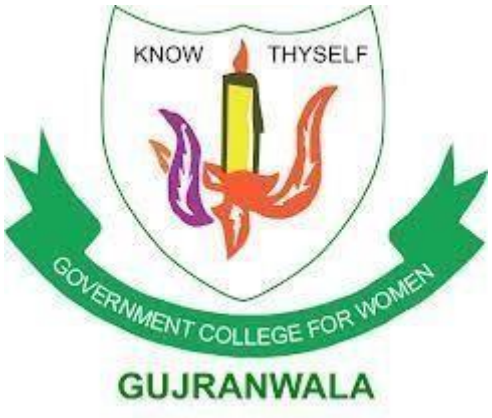
5.11 High-Level Case Diagram

5.11.1 Admin Use Case



5.11.2 User Use Case





Chapter 6
Use Case Description

6. Introduction

We have completed analysis of the system. So we understand the current situation of the problem domain. Now we are ready to strive for a solution for the problem domain by using object-oriented approach. Following artifacts are discussed:

1. Use case description
2. Use case diagram refined
3. Sequence Diagram
4. Collaboration Diagram
5. Domain Model
6. Design Class Diagram
7. Data Model

6.1 Use Case Description

Following are the use cases that we have used in our project.

UC _ 1: Connect to Wi-Fi
Brief description: User will connect to Wi-Fi.
Level: User
Primary actor: User
Precondition: • User must be logged into the web application. • Wi-Fi must be availabe for connection.
Main success scenario: • User opens the website. • User connects to the Wi-Fi by entering credentials (if required). • System connects to Wi-Fi and provides access.
Alternate flows: **1.1 Wi-Fi is unavailable or down** • User receives a notification of unavailable service. **1.2 Incorrect credentials for Wi-Fi** • User is prompted to re-enter correct details.
Postconditions: User is successfully connected to Wi-Fi.

Table 6.1

UC _ 2: View
Brief description: User view available rooms and features.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User opens the web application. • User views room availability and other information.
Alternate flows: **1.1 If the application fails to load room data** • User is prompted with a “try again later” message.
Post conditions: User successfully views room availability.

Table 6.2

UC _ 3: Login
Brief description: User logs into the system.
Level: User
Primary actor: User
Precondition: User must be registered with the web application.
Main success scenario: • User enters login credentials. • System authenticates the user. • User is granted access to the system.
Alternate flows: **1.1 Incorrect credentials enteted** • User is prompted to retry or recover password.
Post conditions: User is successfully logged into the web application.

Table 6.3

UC _ 4: User Profile
Brief description: User manages their personal profile information.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User opens the profile settings. • User edits and saves profile information.
Alternate flows: **1.1 Internet connection lost during profile update** • Profile information is saved locally and synced later.
Postconditions: User's profile is successfully updated.

Table 6.4

UC _ 5: Room Booking
Brief description: User sends a booking request to reserve a room in the hostel.
Level: User
Primary actor: User
Precondition: • User must be logged into the web application. • Rooms must be available for booking.
Main success scenario: • User navigates to the room booking section. • User selects a room and sends a booking request. • System records the request and notifies the user of the submission.
Alternate flows: **1.1 Room is no longer available** • User notified with "room unavailable" message & prompted to select another room. **1.2 Request submission fails due to connectivity issues** • System prompts the user to retry.
Postconditions: Booking request is successfully submitted and awaits admin approval.

Table 6.5

UC _ 6: Post Feedback
Brief description: User provides feedback about their experience in the hostel or web application.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User submits feedback. • Feedback is saved in the system.
Alternate flows: **1.1 Feedback submission fails due to connectivity** • User receives a “try again” message.
Post conditions: Feedback is successfully saved and sent.

Table 6.6

UC _ 7: Access Report
Brief description: User can access and view reports related to their room own booking hiostory.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User navigates to the reports section. • User selects a report (e.g., booking history) • System retrieves and displays the selected report.
Alternate flows: **1.1 Report data is unavailable** • System shows a “ no data available” message. **1.2 Network connection is lost during the operation** • User receives a “connection error. Please try again later” notification.
Post conditions: User successfully views the report.

Table 6.7

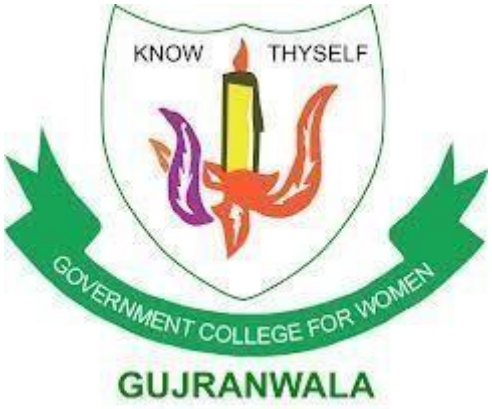
UC _ 8: Download Report
Brief description: User downloads their own booking for offline use if booking status is approved.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User navigates to the report section. • User selects a report to download. • System generates and downloads the report in the selected format.
Alternate flows: **1.1 Download fails due to netword issues** • User is prompted to retry the download. **1.2 Report file generation error** • System notifies the user with "Report could not be generated. Please try again later."
Post conditions: User successfully downloads the report for offline access.

Table 6.8

UC _ 9: Logout
Brief description: User logs out of the web application after completing tasks.
Level: User
Primary actor: User
Precondition: User must be logged into the web application.
Main success scenario: • User clicks on the logout option. • System logs the user out.
Alternate flows: **1.1 Logout fails due to system error** • User is notified and can try again.
Post conditions: User is successfully logged out of the web application.

Table 6.9

6.2. Use Case Diagram



Chapter 7
Sequence Diagram

7. Sequence Diagram

7.1 Login

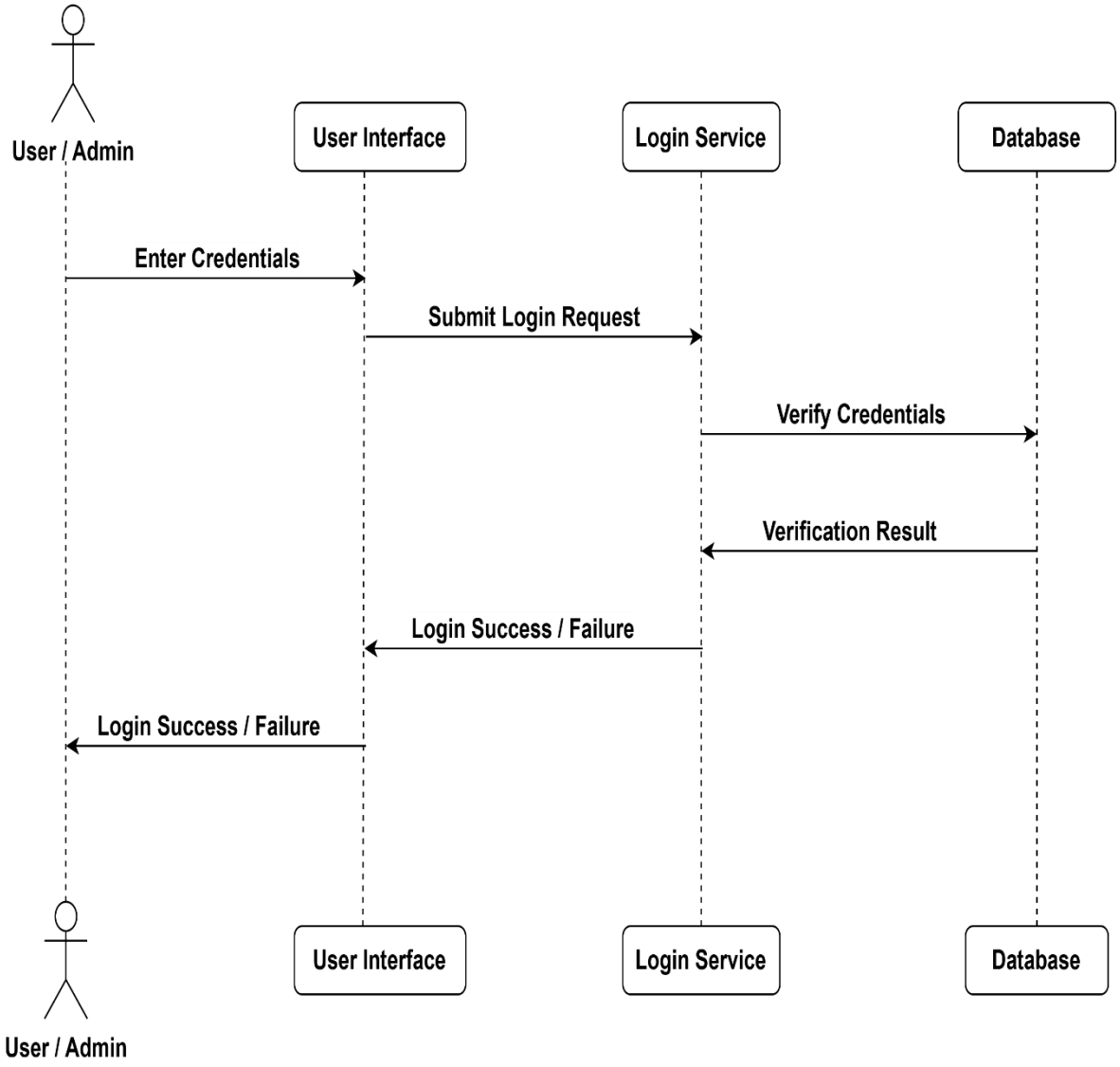


Figure 7.1

7.2 Room Booking

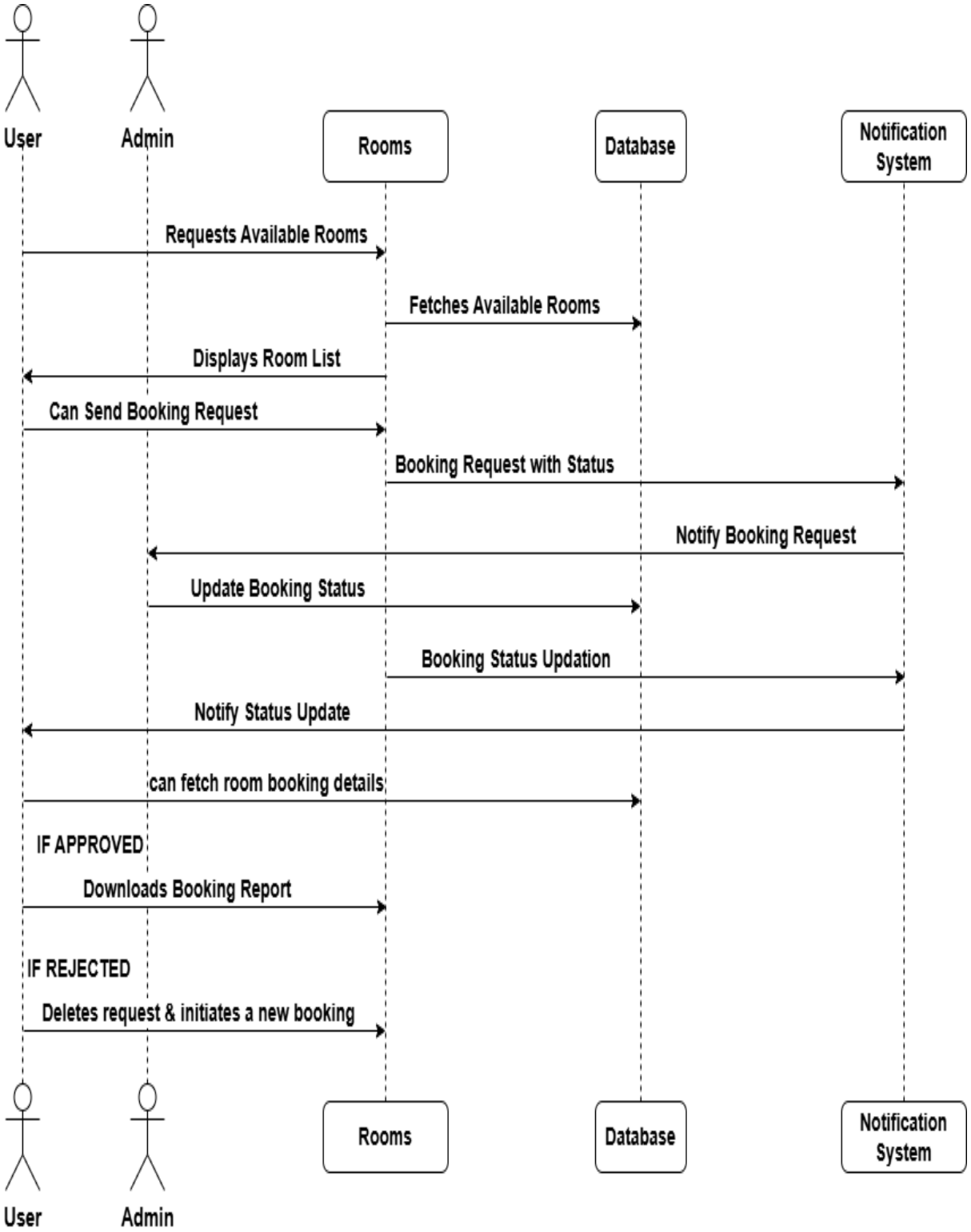


Figure 7.2

7.3 View Facilities

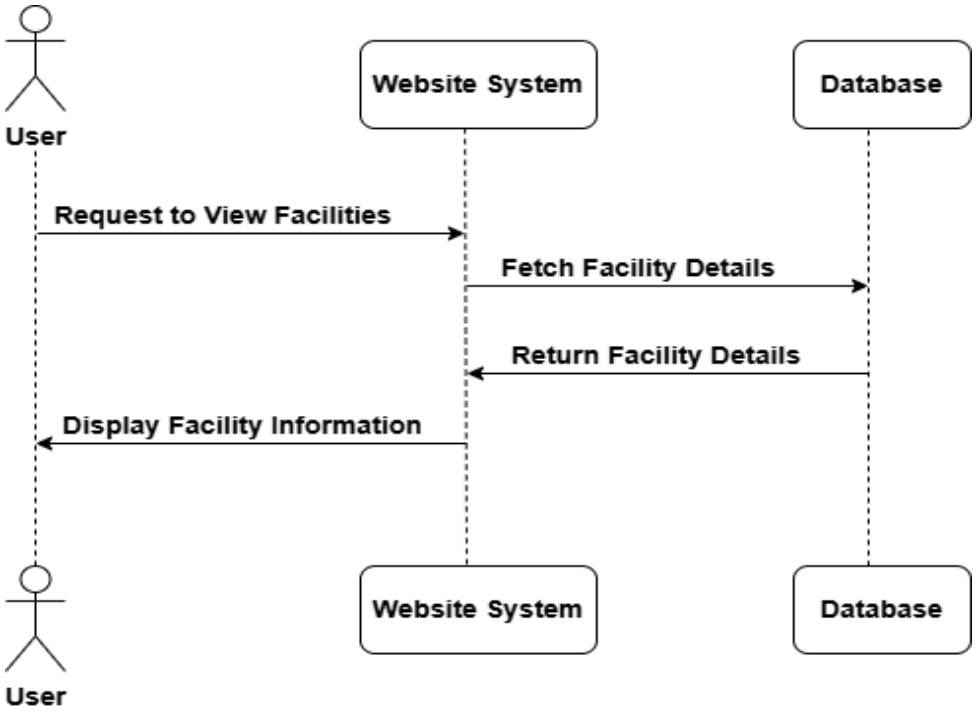


Figure 7.3

7.4 Profile Updation

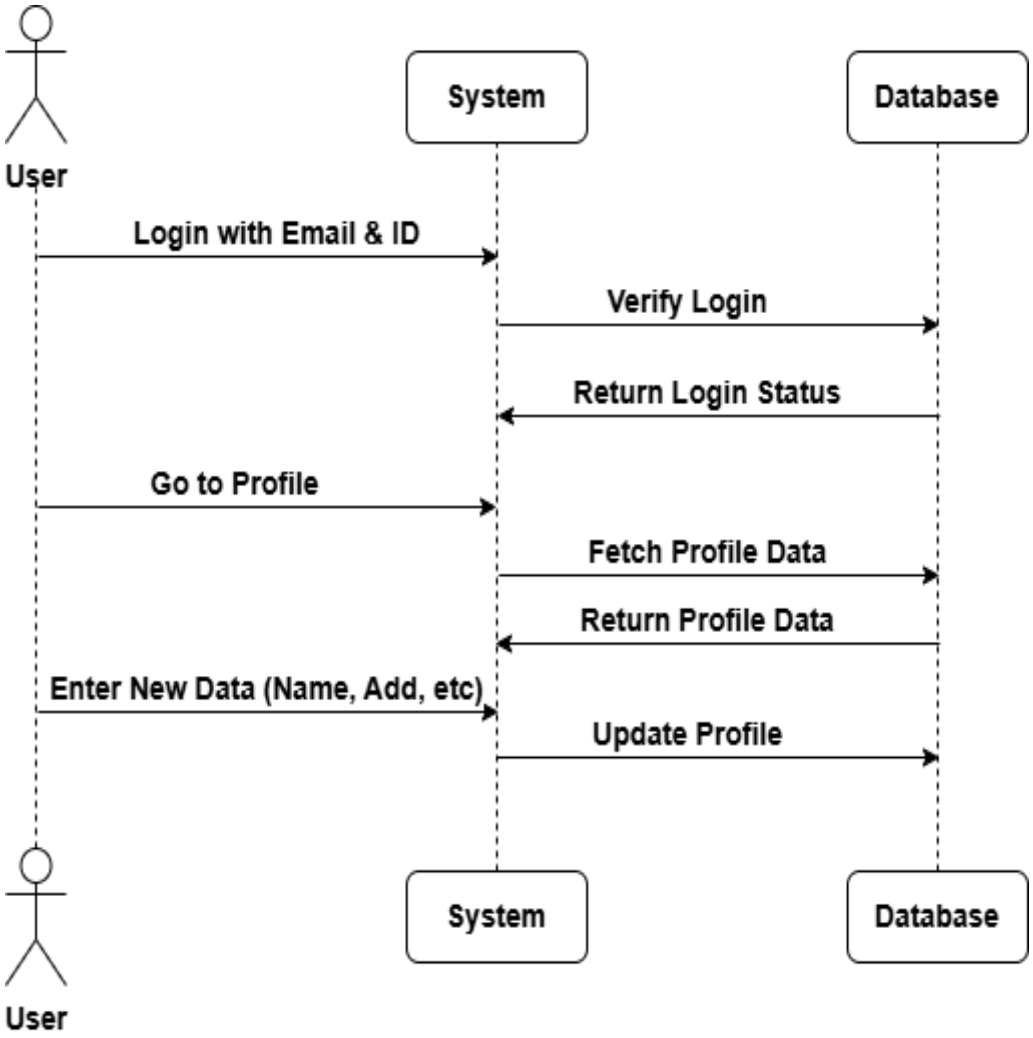


Figure 7.4

7.5 Contact Us

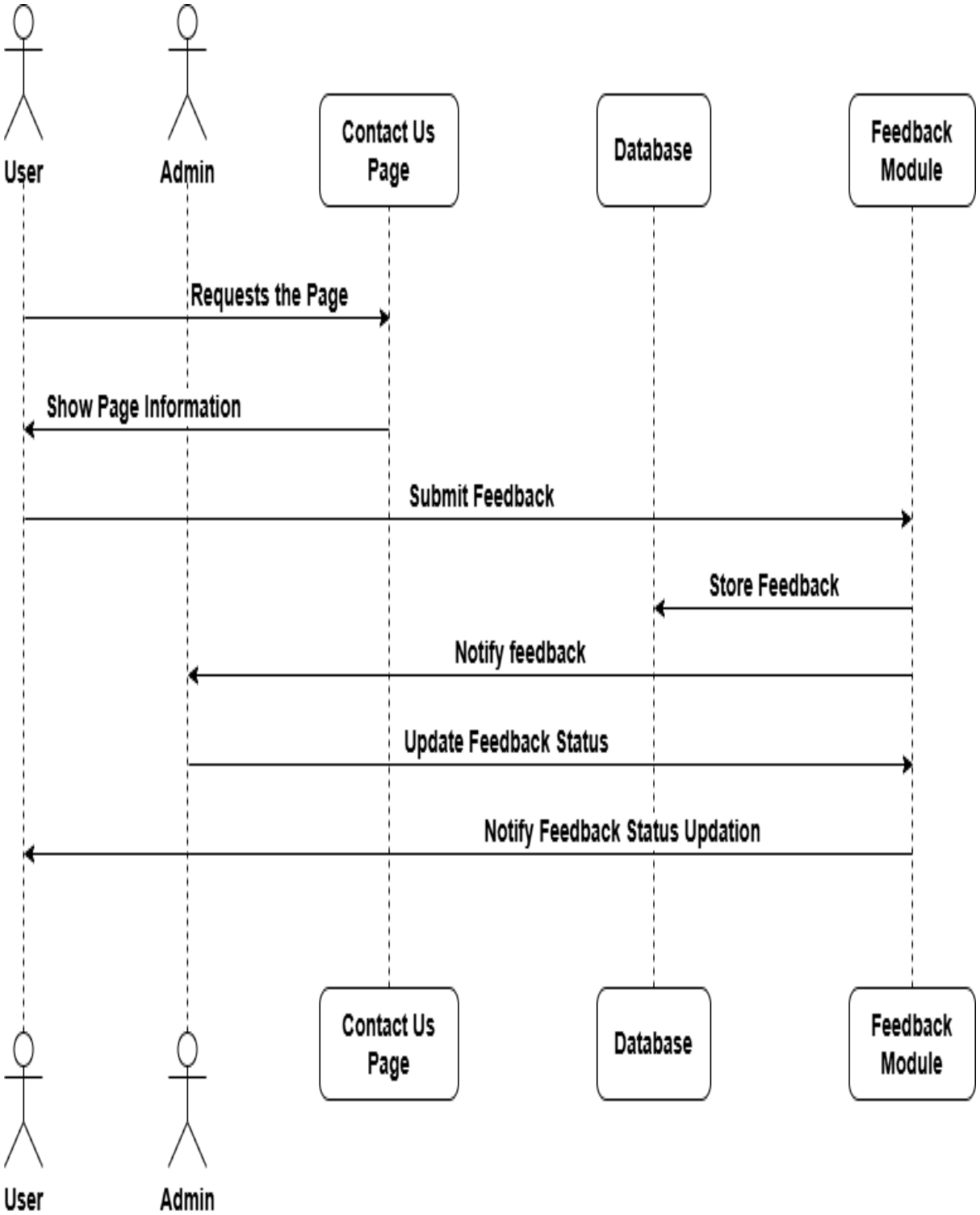


Figure 7.5

7.6 Generating Report

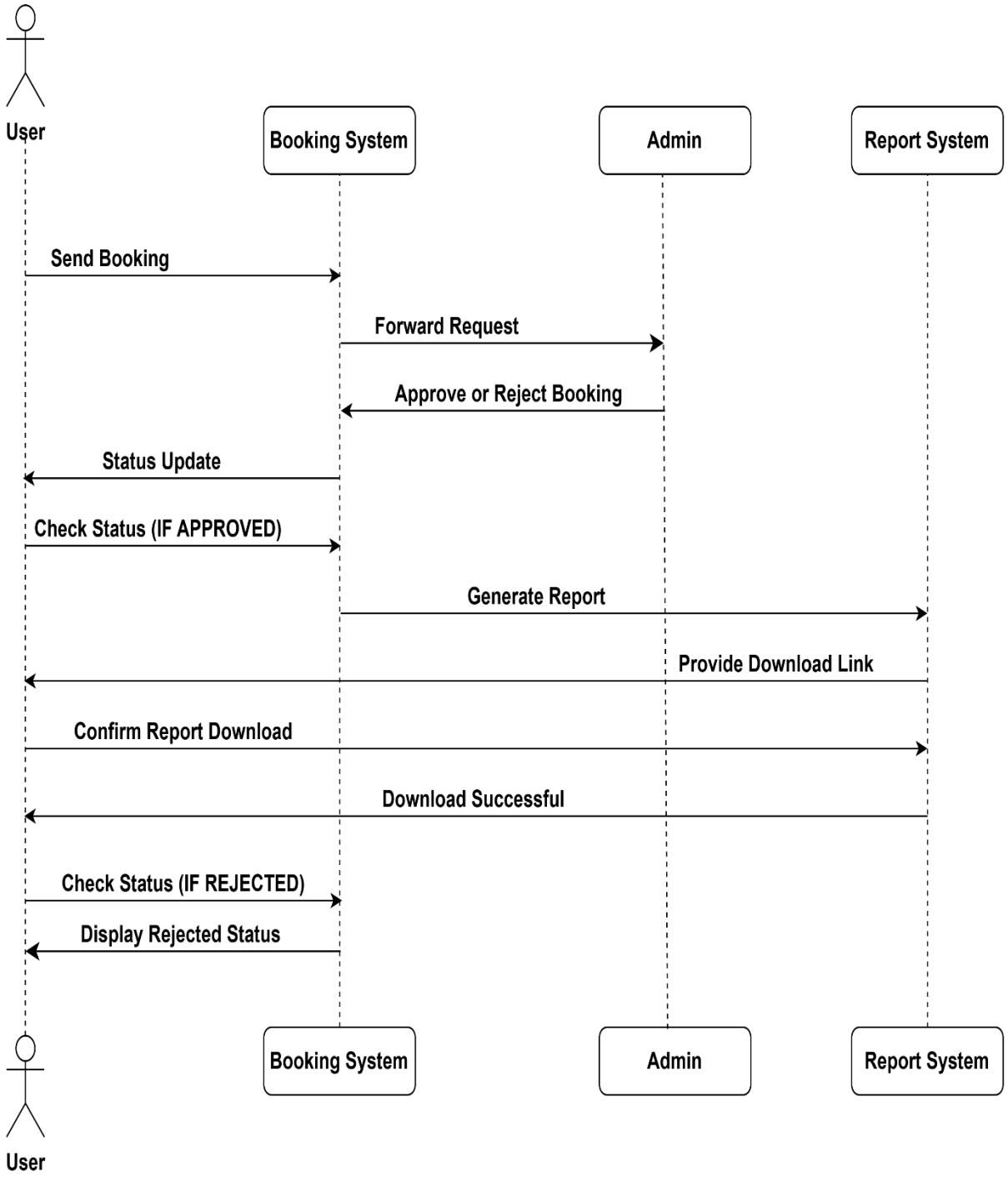


Figure 7.6

7.7 Logout

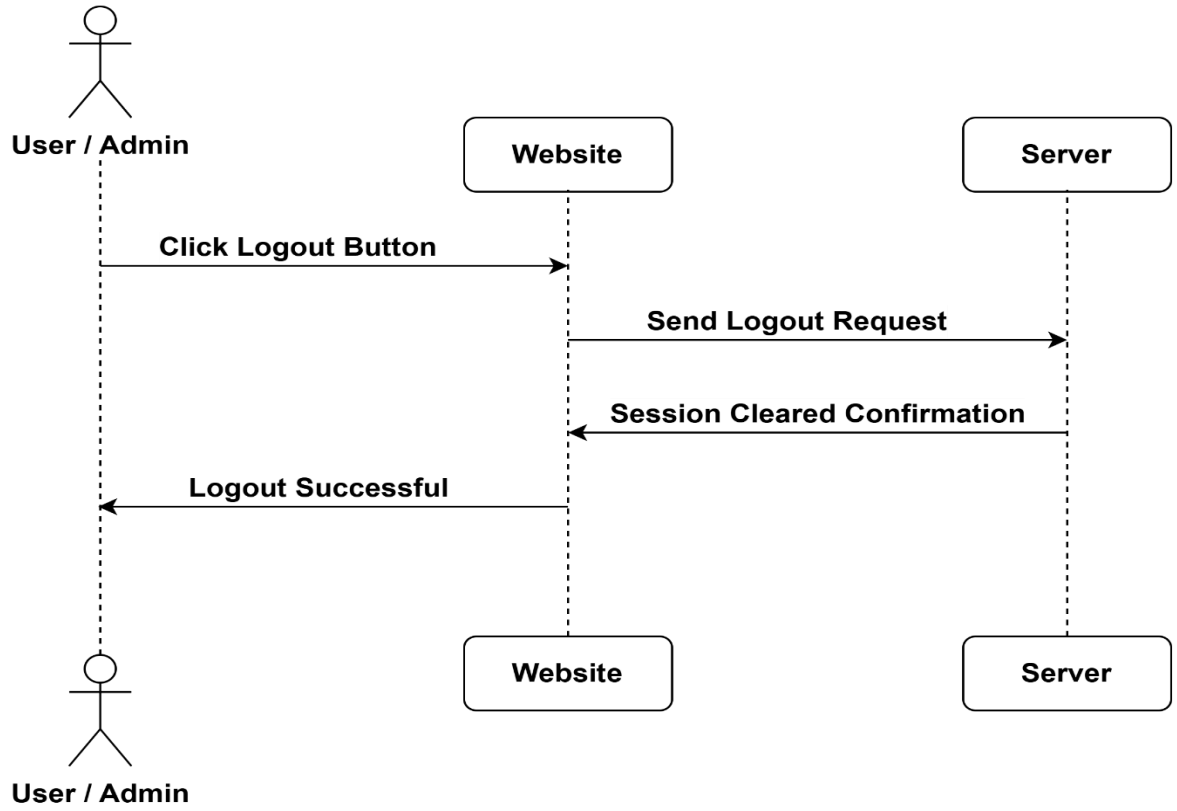


Figure 7.7

Chapter 8
Collaboration Diagram

8. Collaboration Diagram

8.1 Login

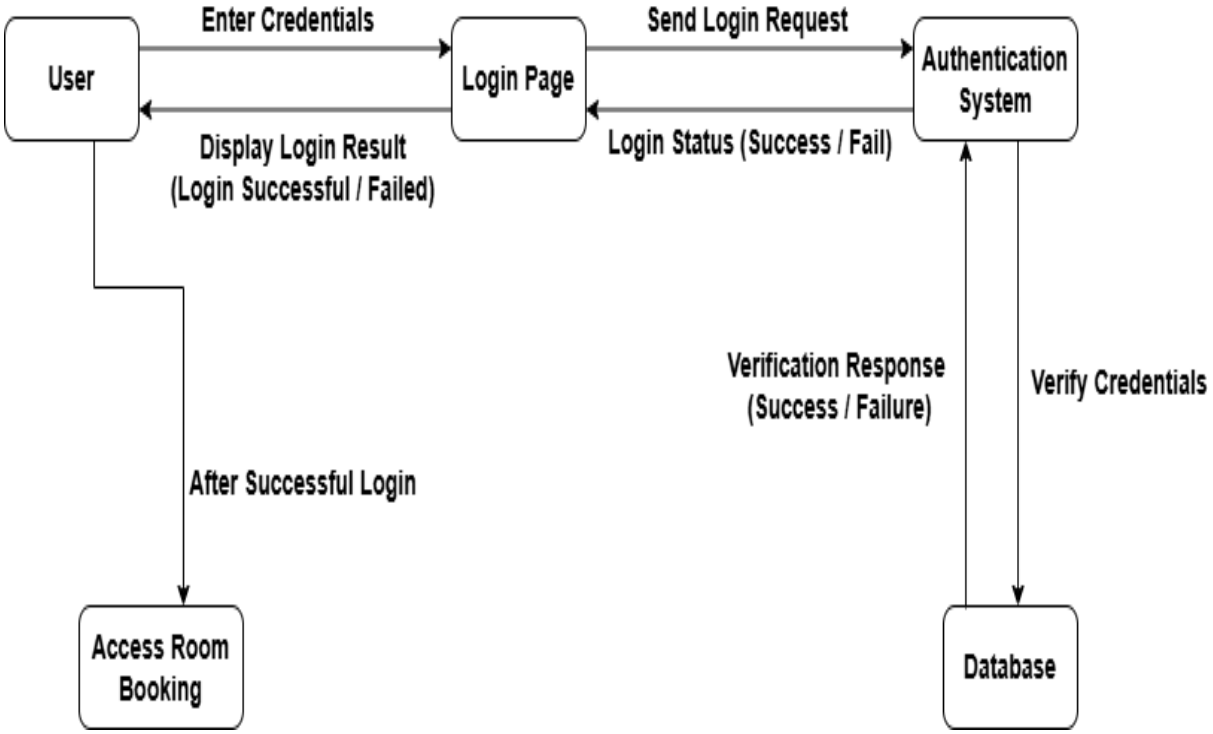


Figure 8.1

8.2 Room Booking

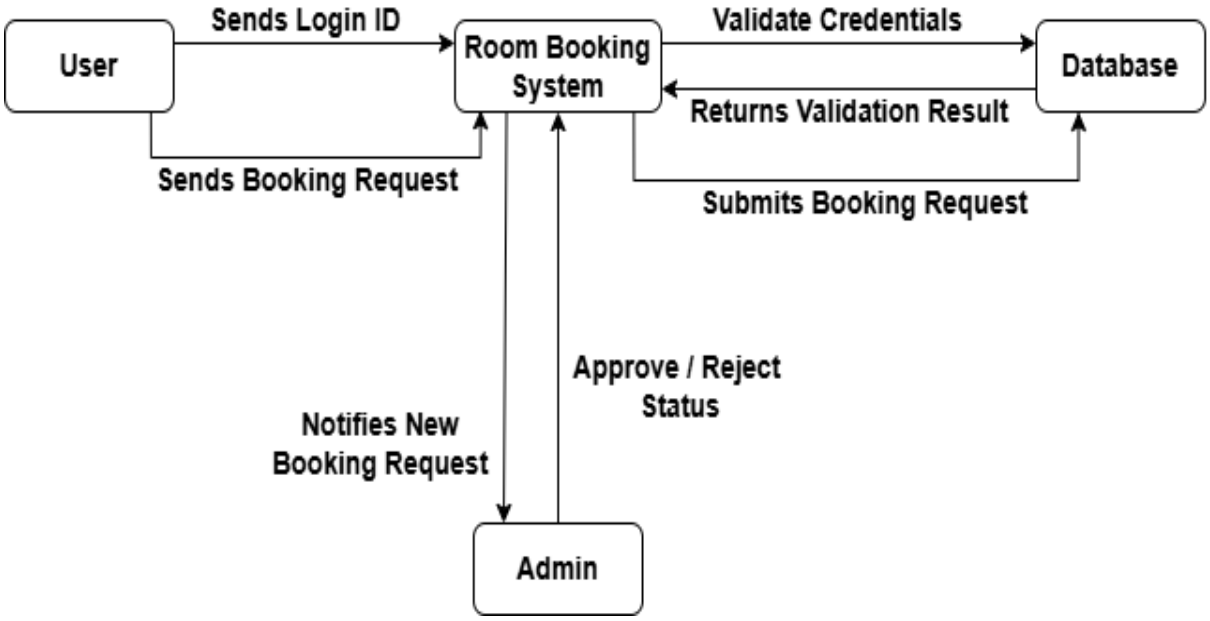


Figure 8.2

8.3 View Facilities

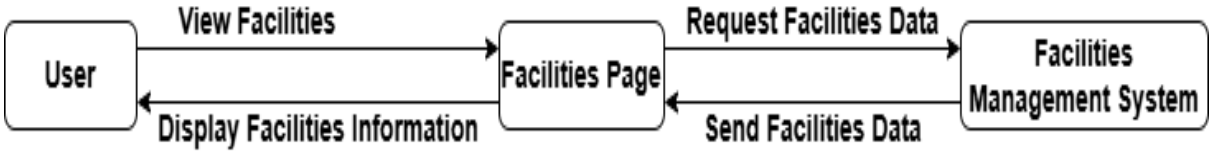


Figure 8.3

8.4 Profile Updation

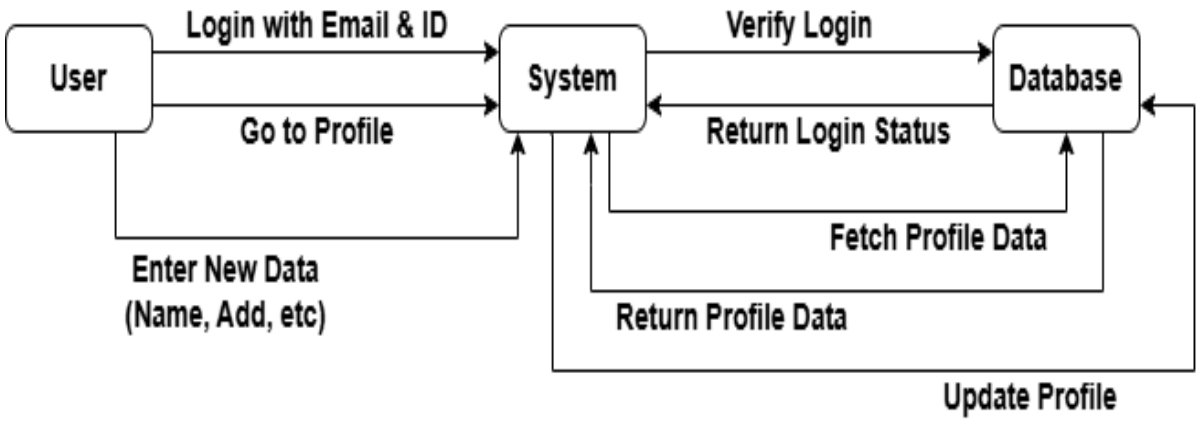


Figure 8.4

8.5 Contact Us

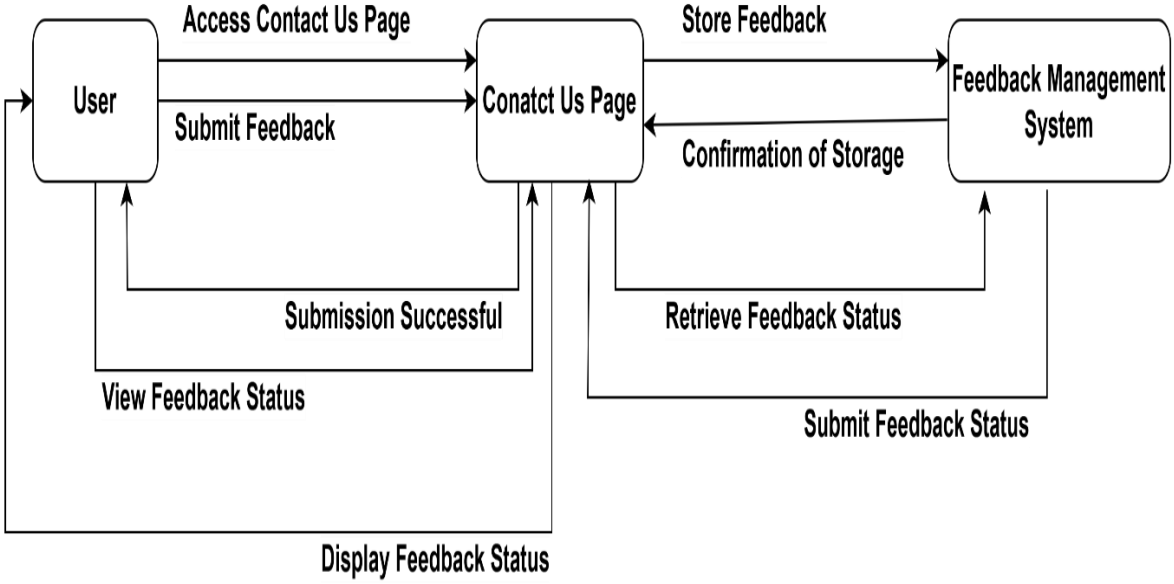


Figure 8.5

8.6 Generating Report

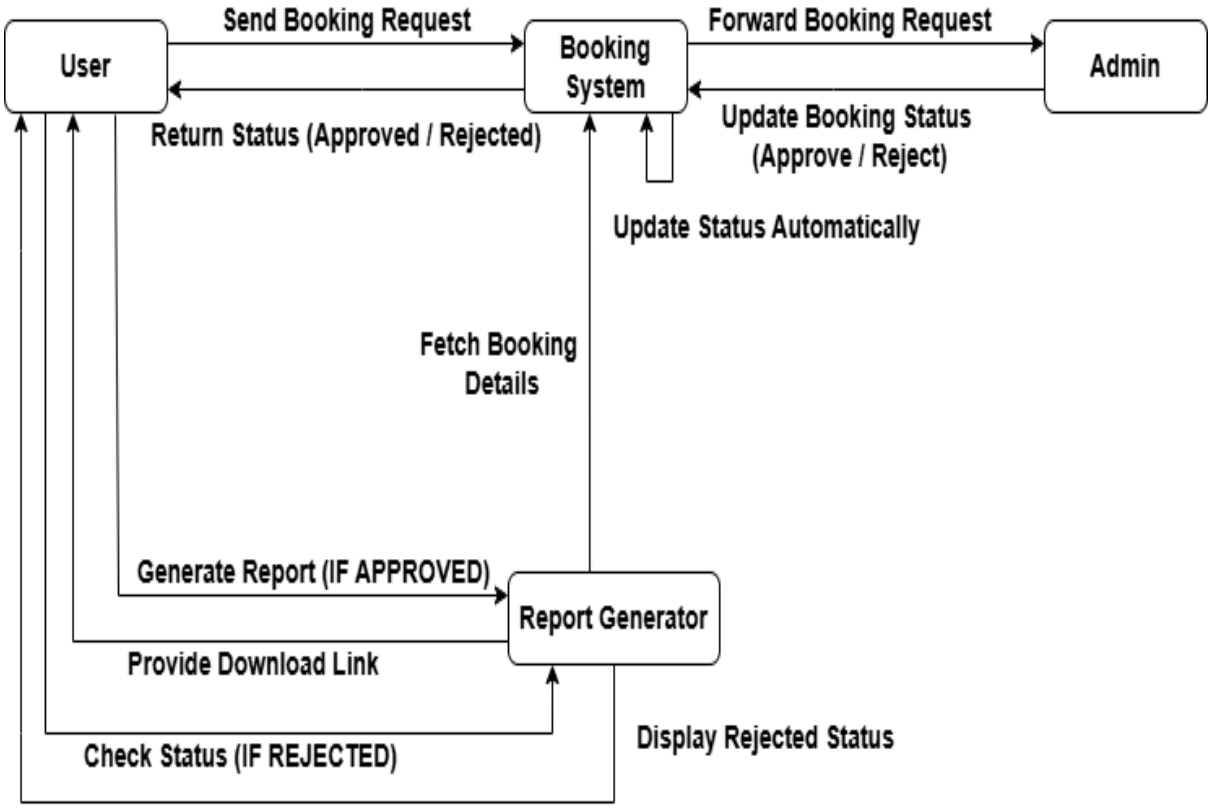


Figure 8.6

8.7 Logout

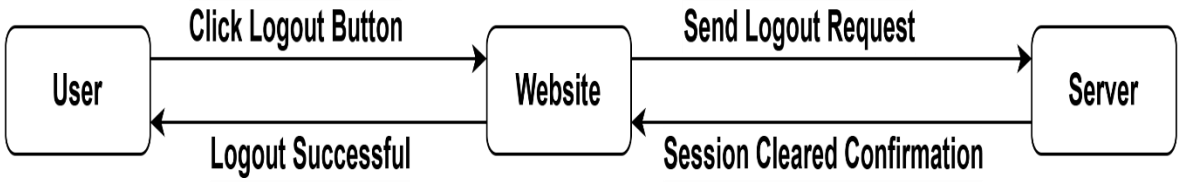
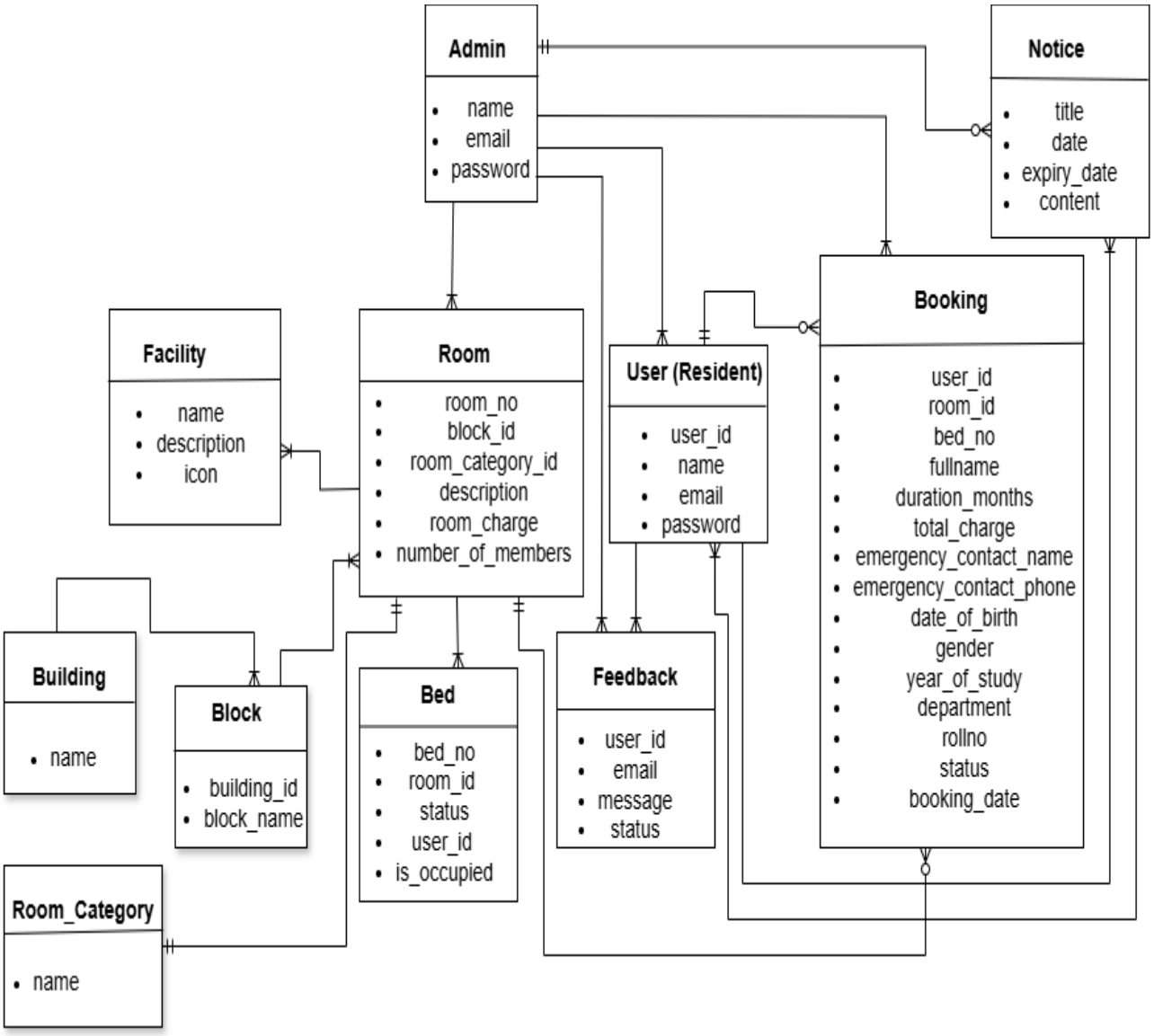
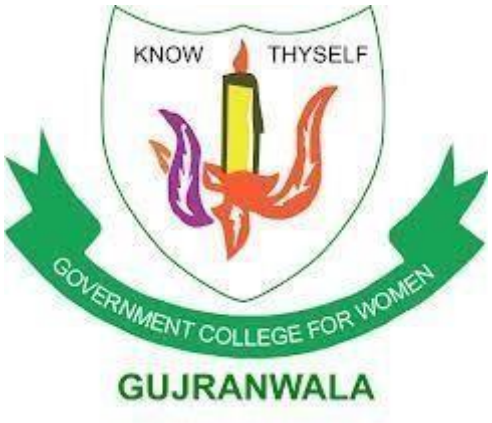


Figure 8.7

Chapter 9
Domain Model

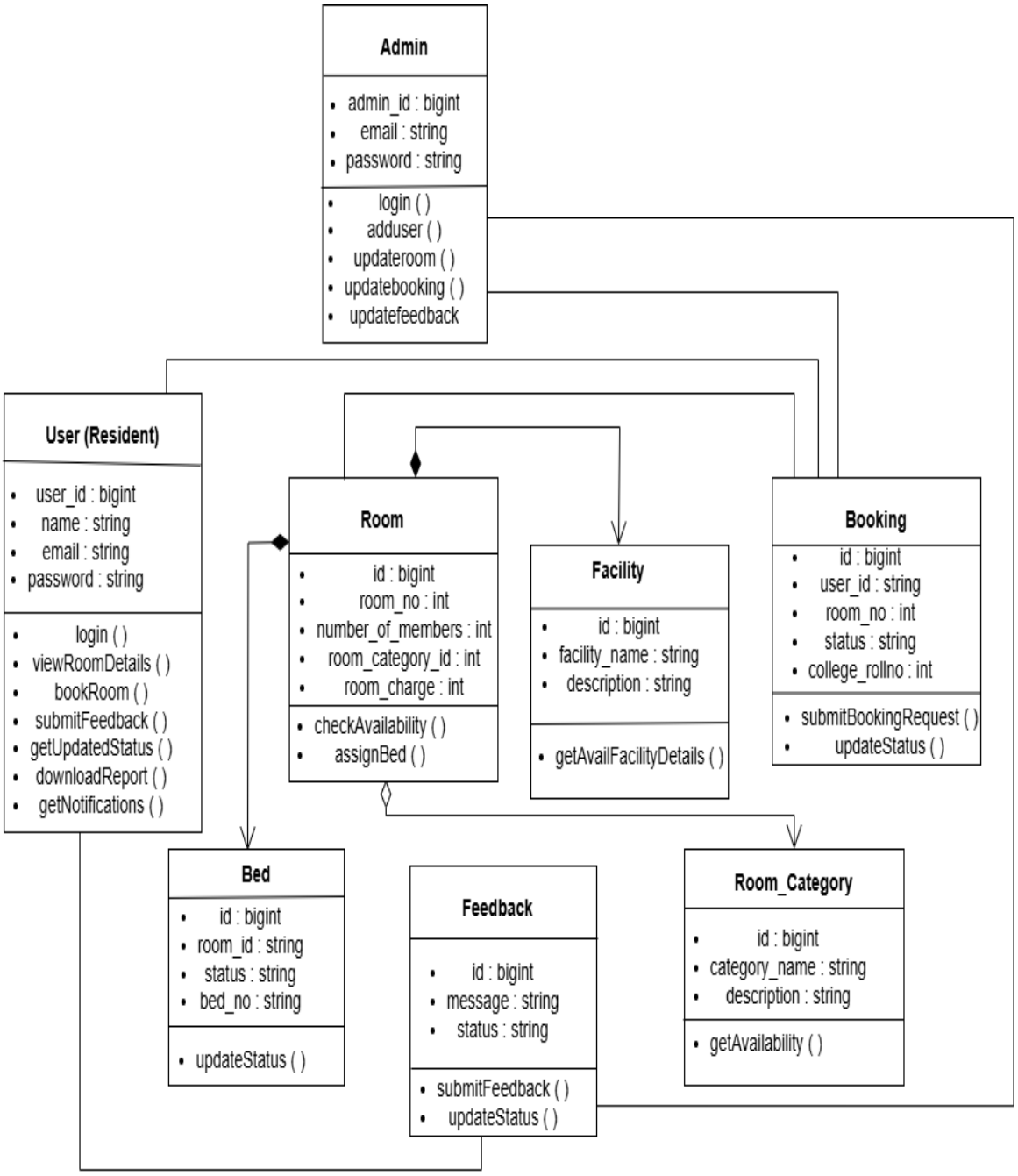
9.1 Domain Model

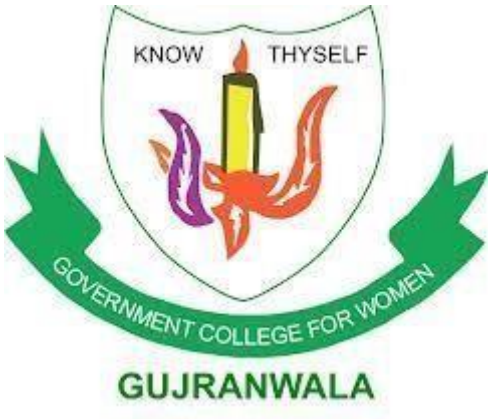




Chapter 10
Class Diagram

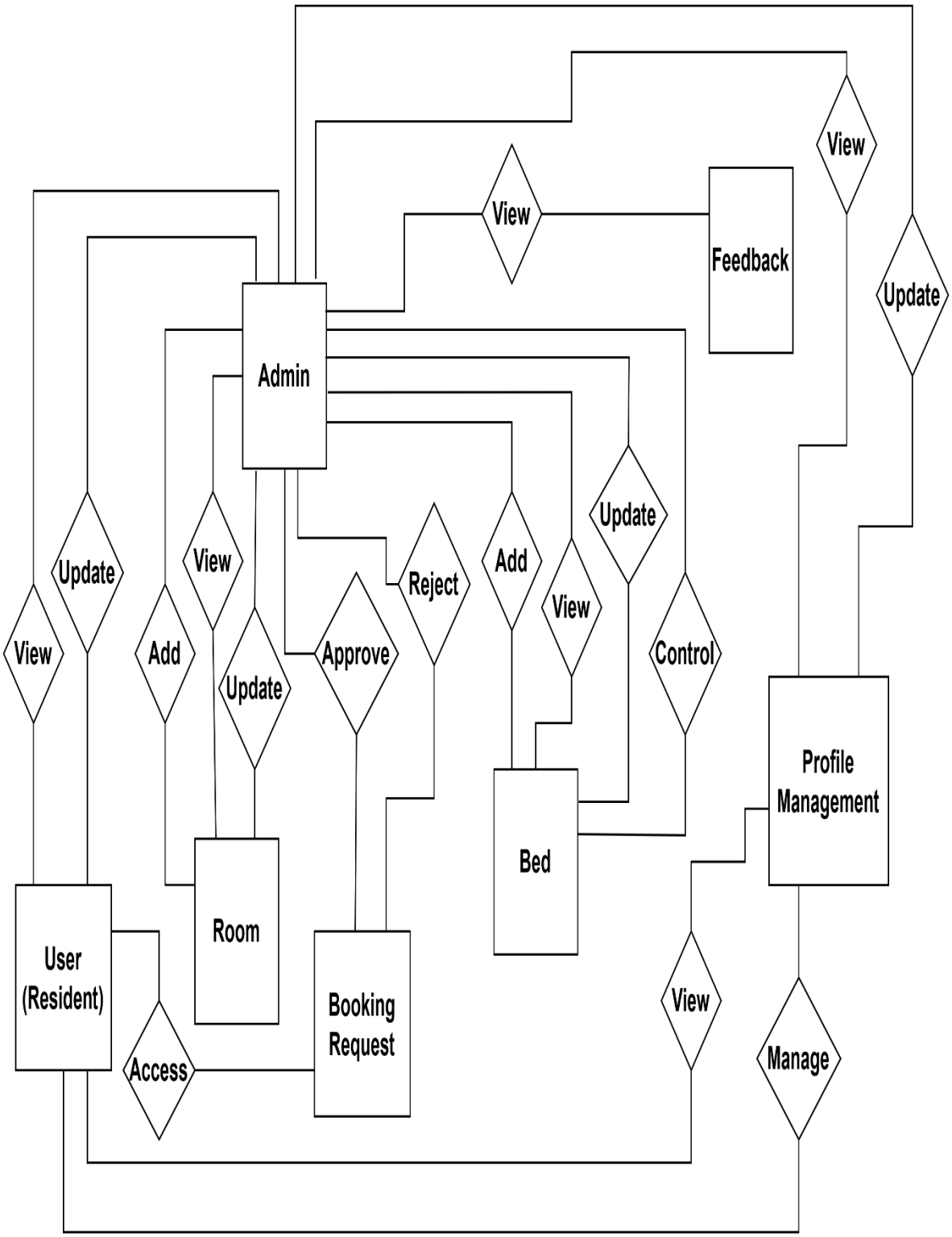
10.1 Design Class Diagram





Chapter 11
Data Model

11.1 Data Model

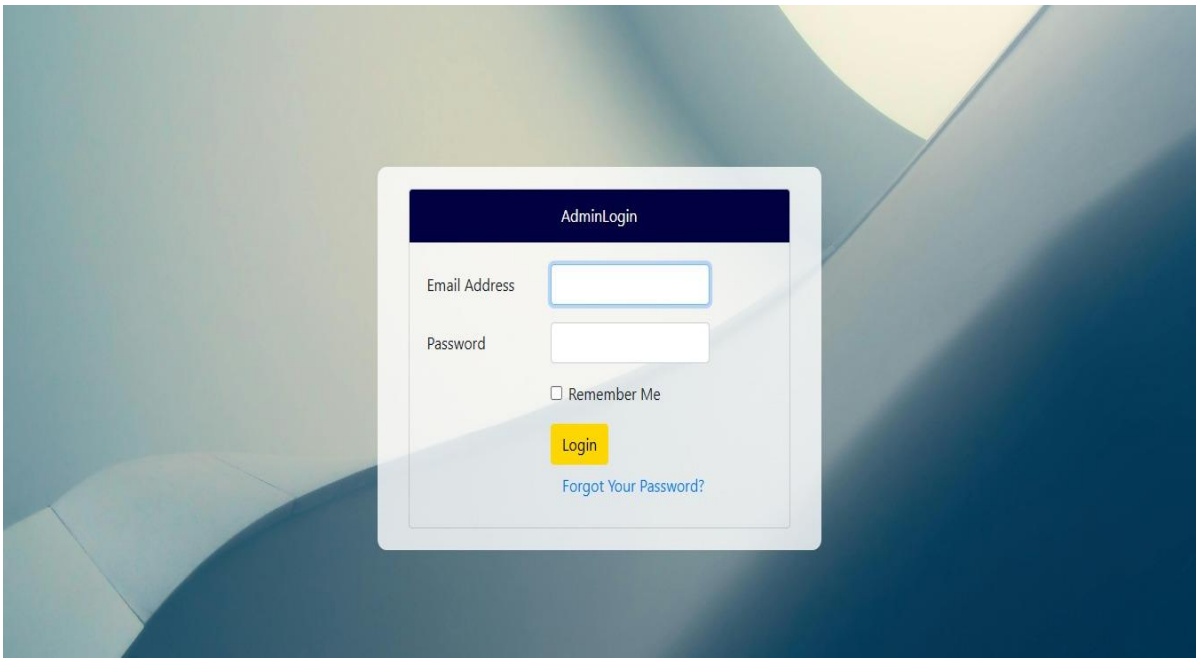


Chapter 12
Interfaces

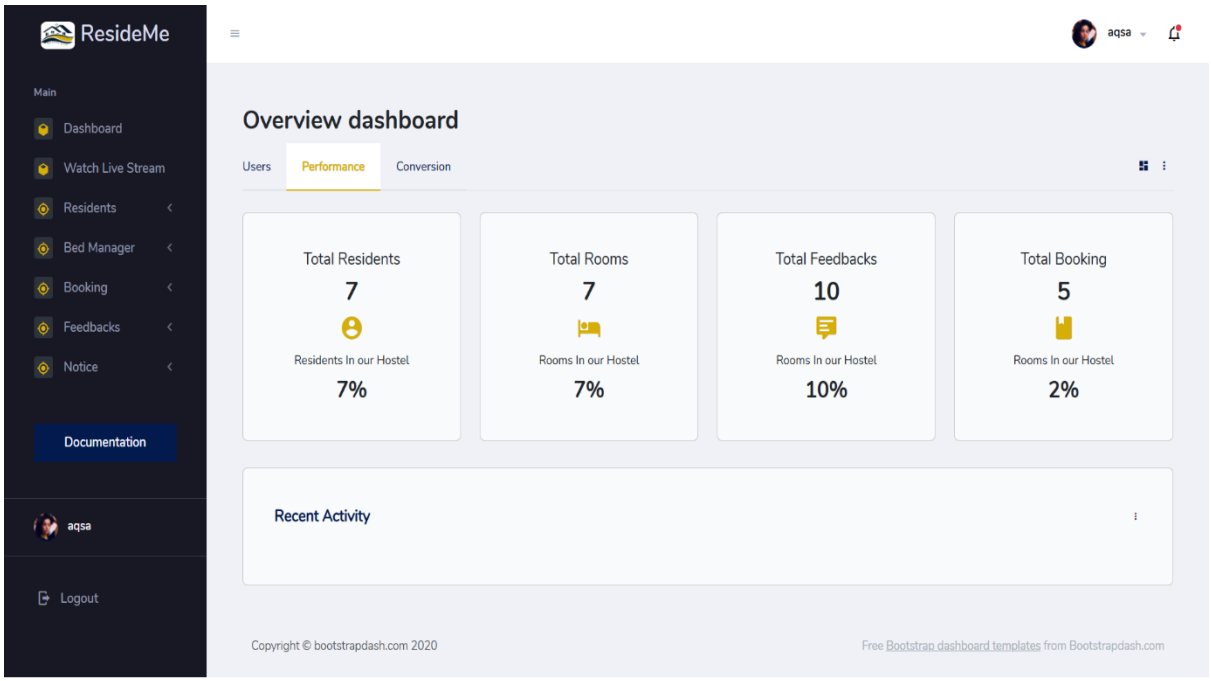
12. Project Interfaces

12.1 Admin

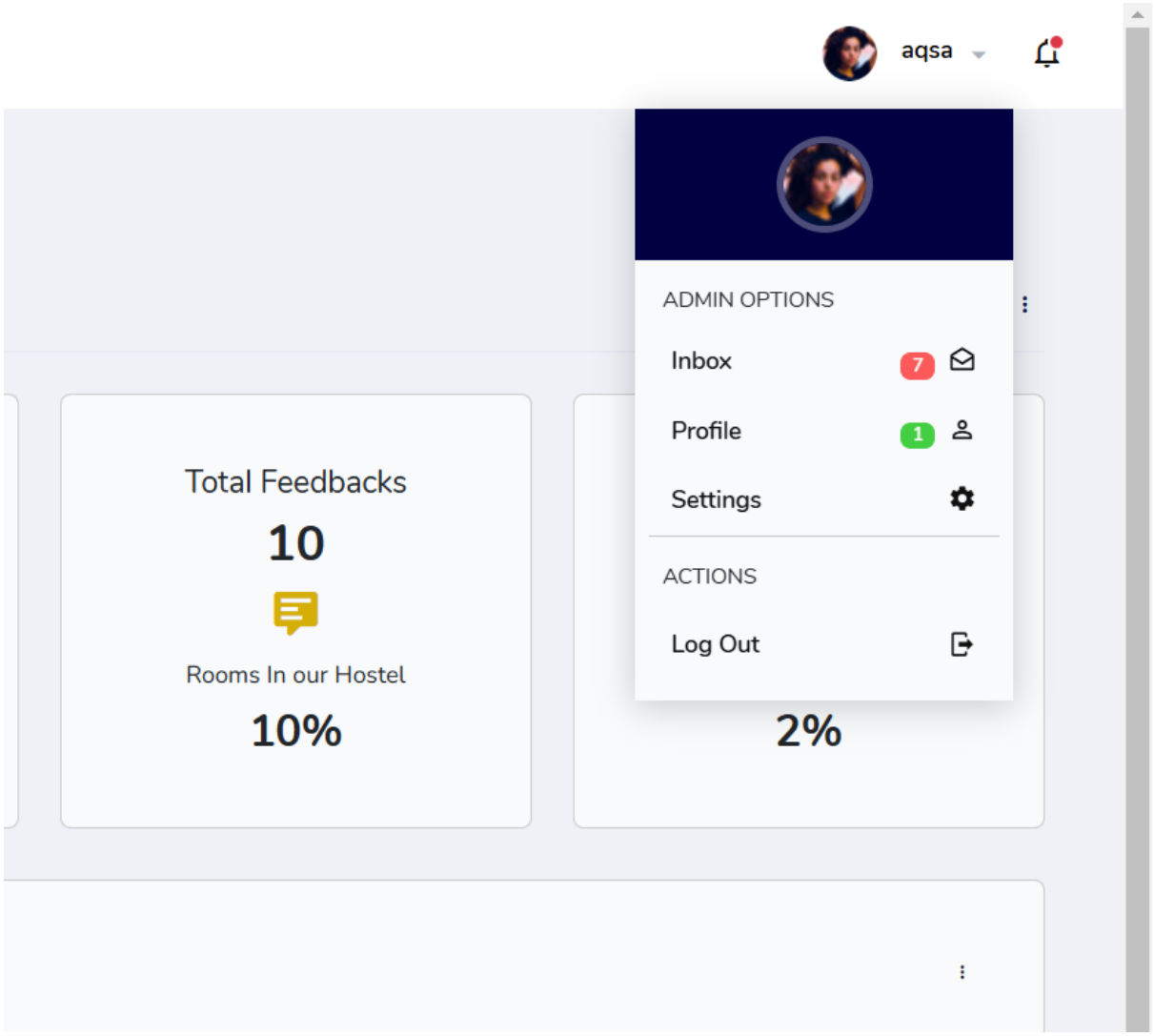
12.1.1 Login



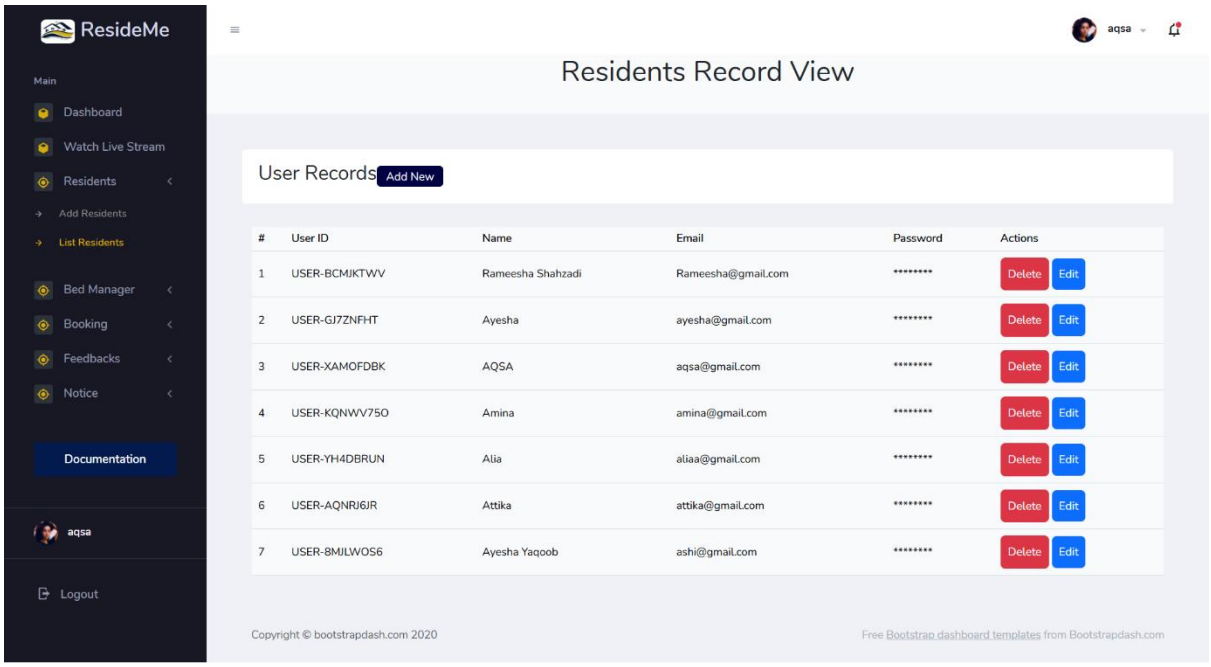
12.1.2 Admin Dashboard



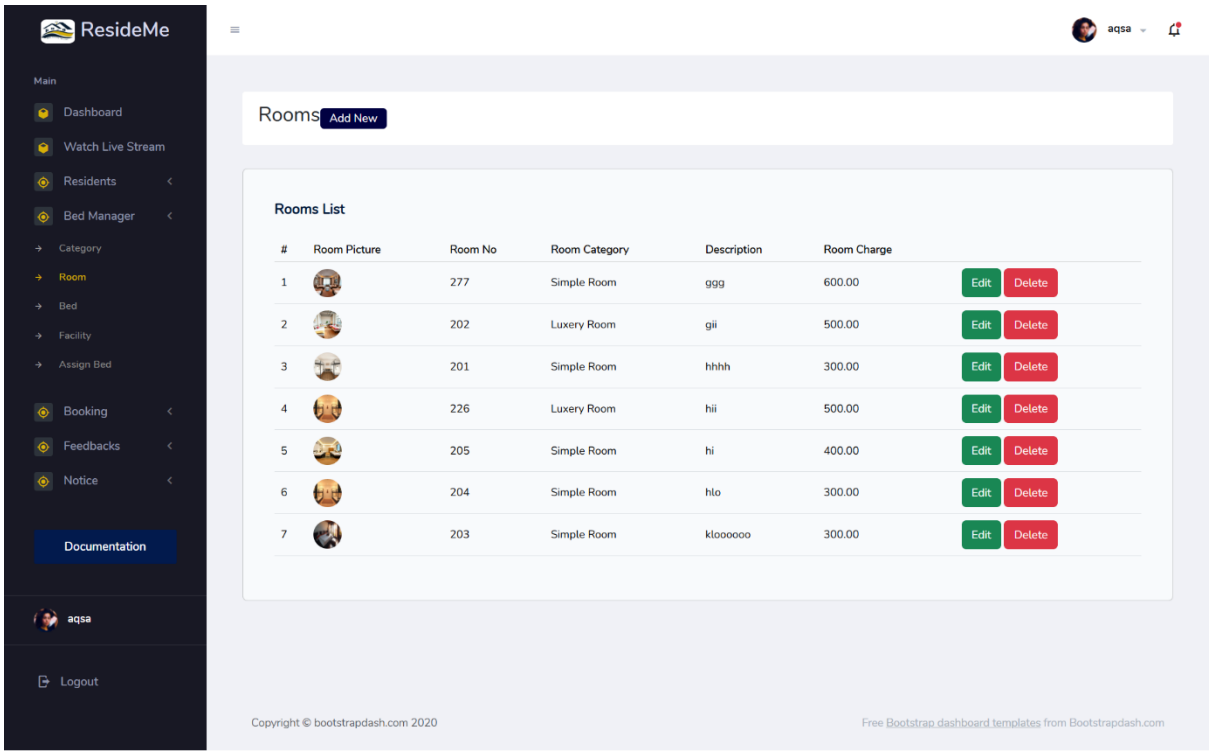
12.1.3 Admin Profile



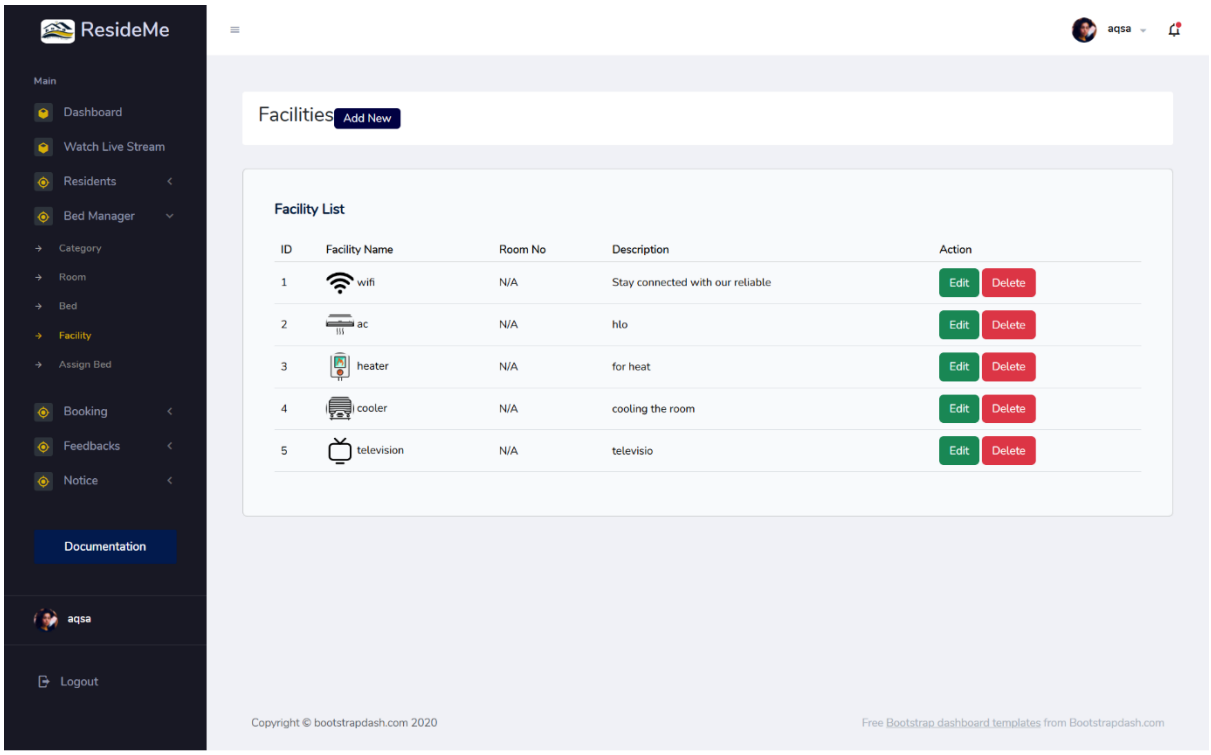
12.1.4 Manage Residents



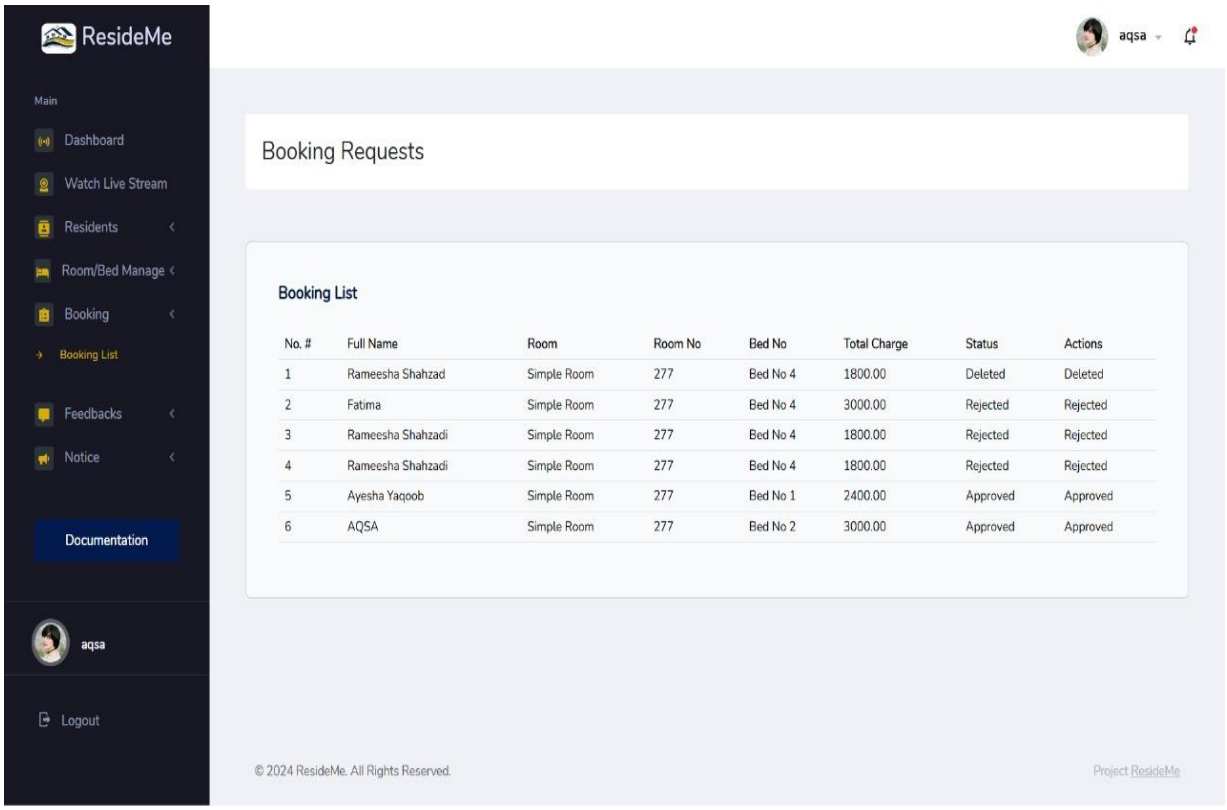
12.1.5 Manage Rooms



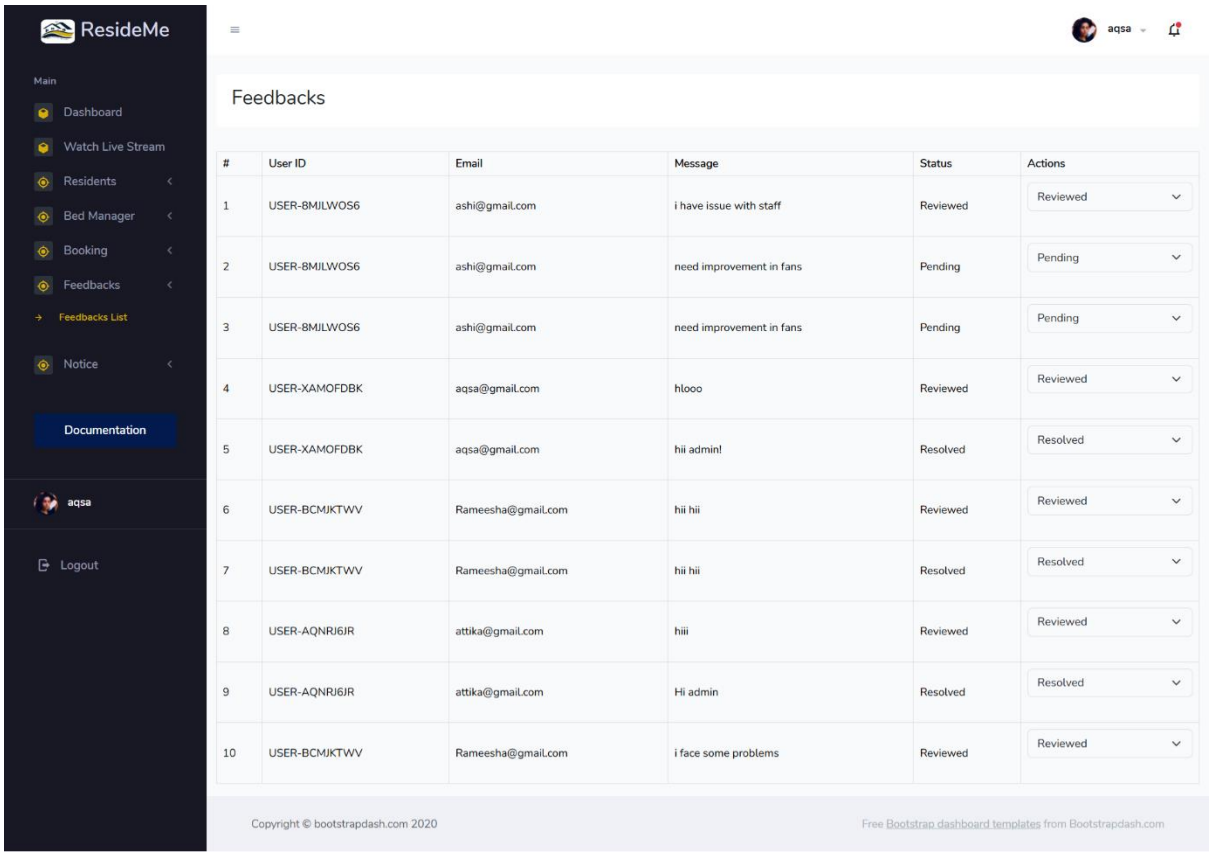
12.1.6 Manage Facilities



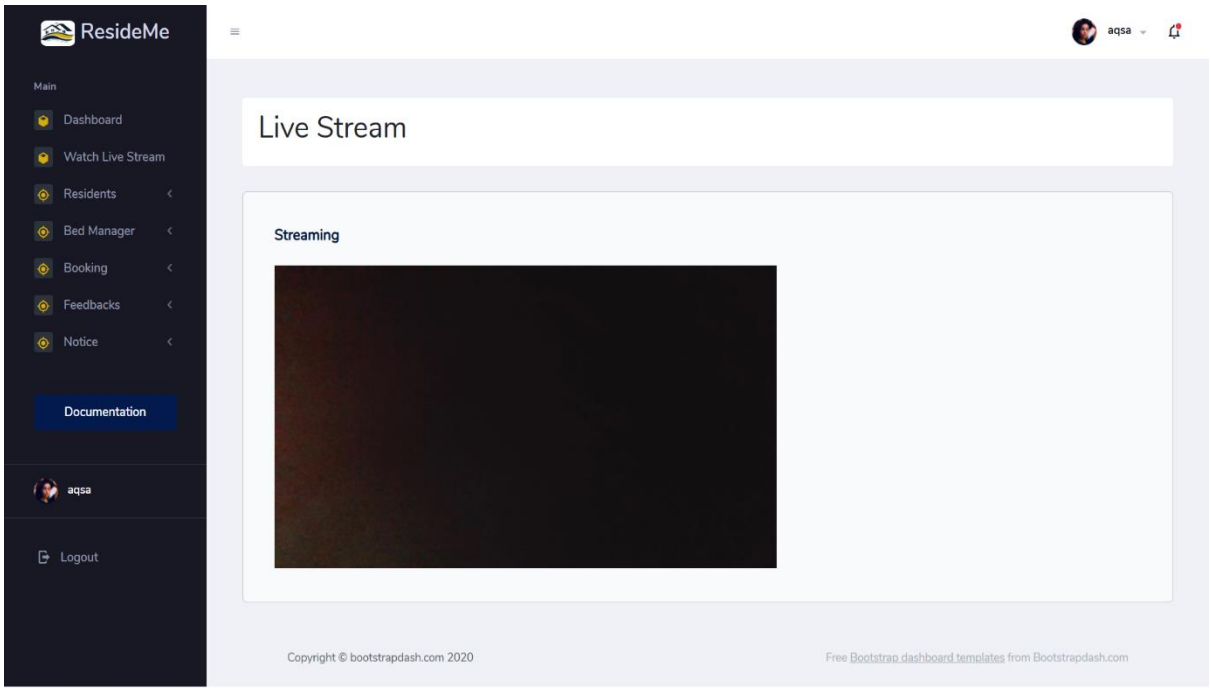
12.1.7 Manage Booking Request



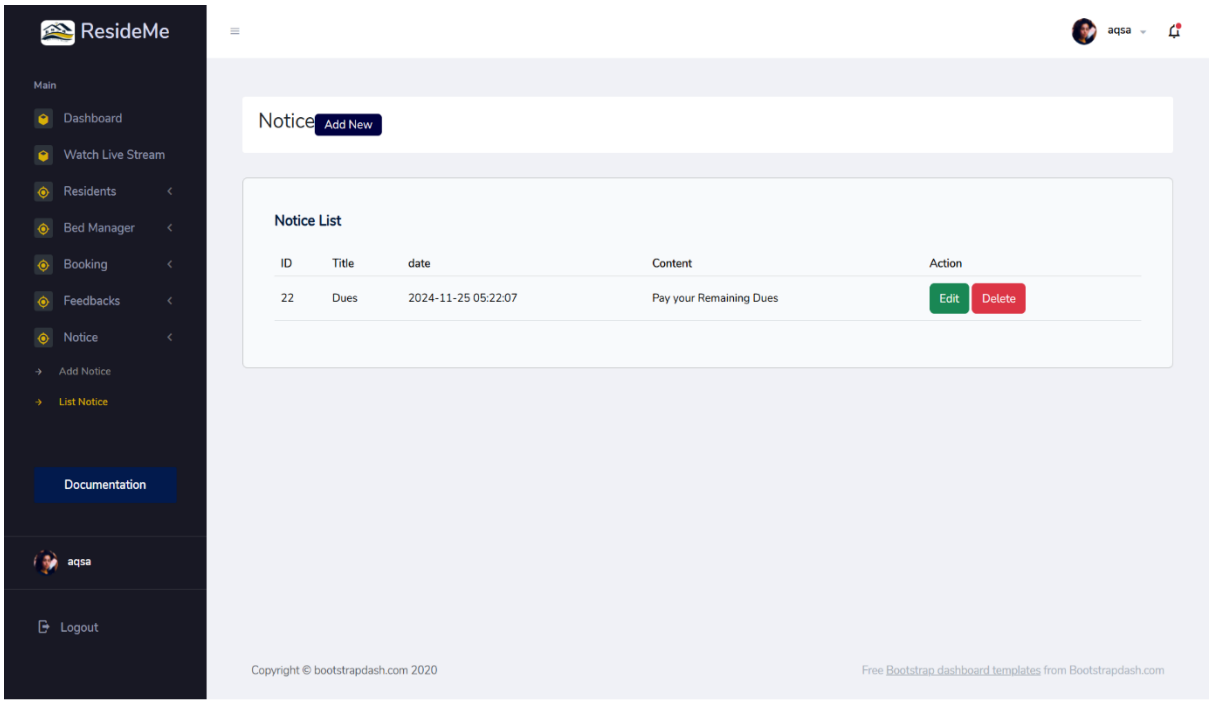
12.1.8 Manage Feedbacks



12.1.9 Live Streaming

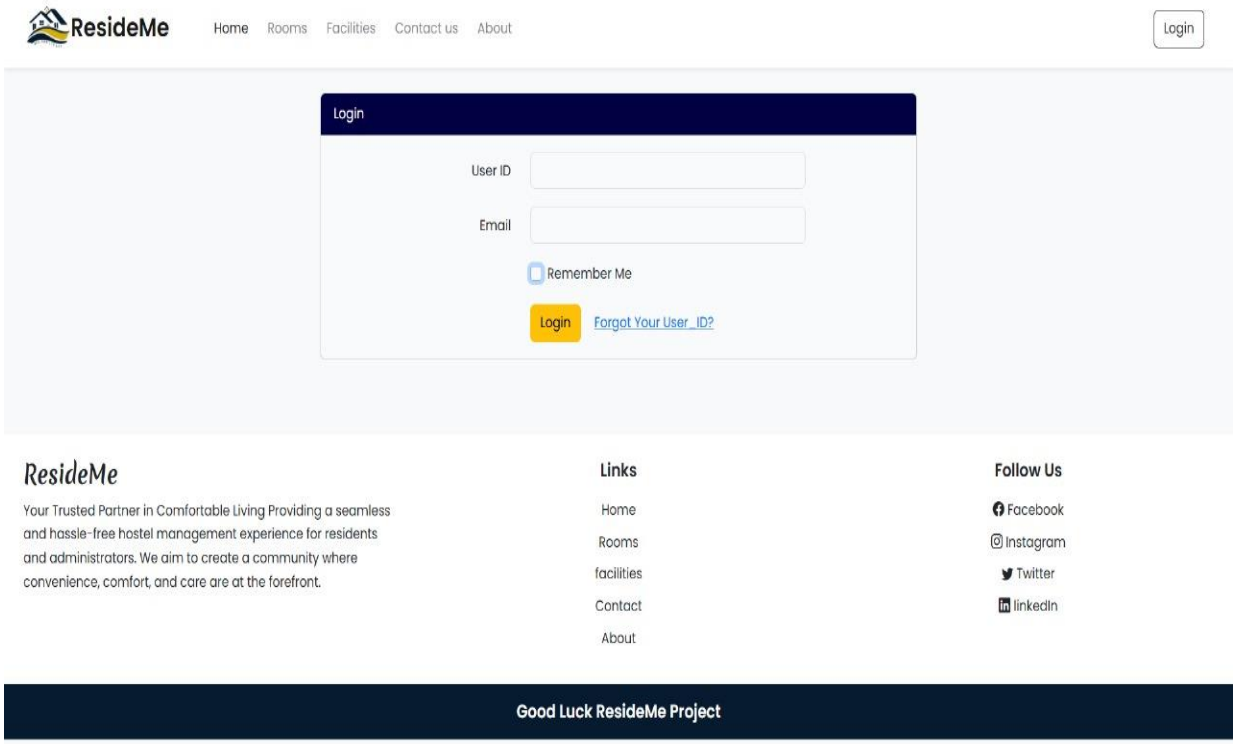


12.1.10 Admin Notice Board

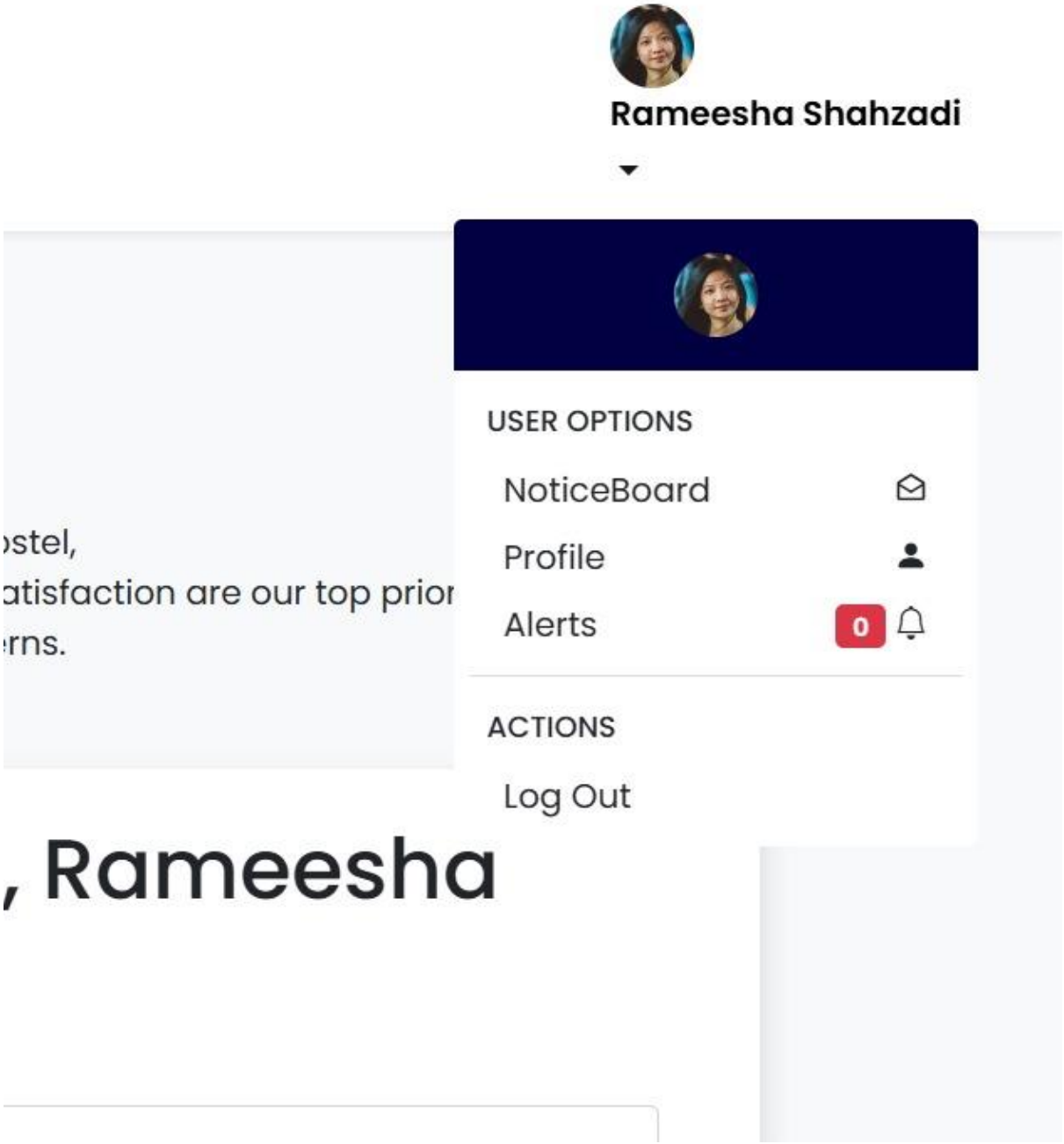


12.2 User

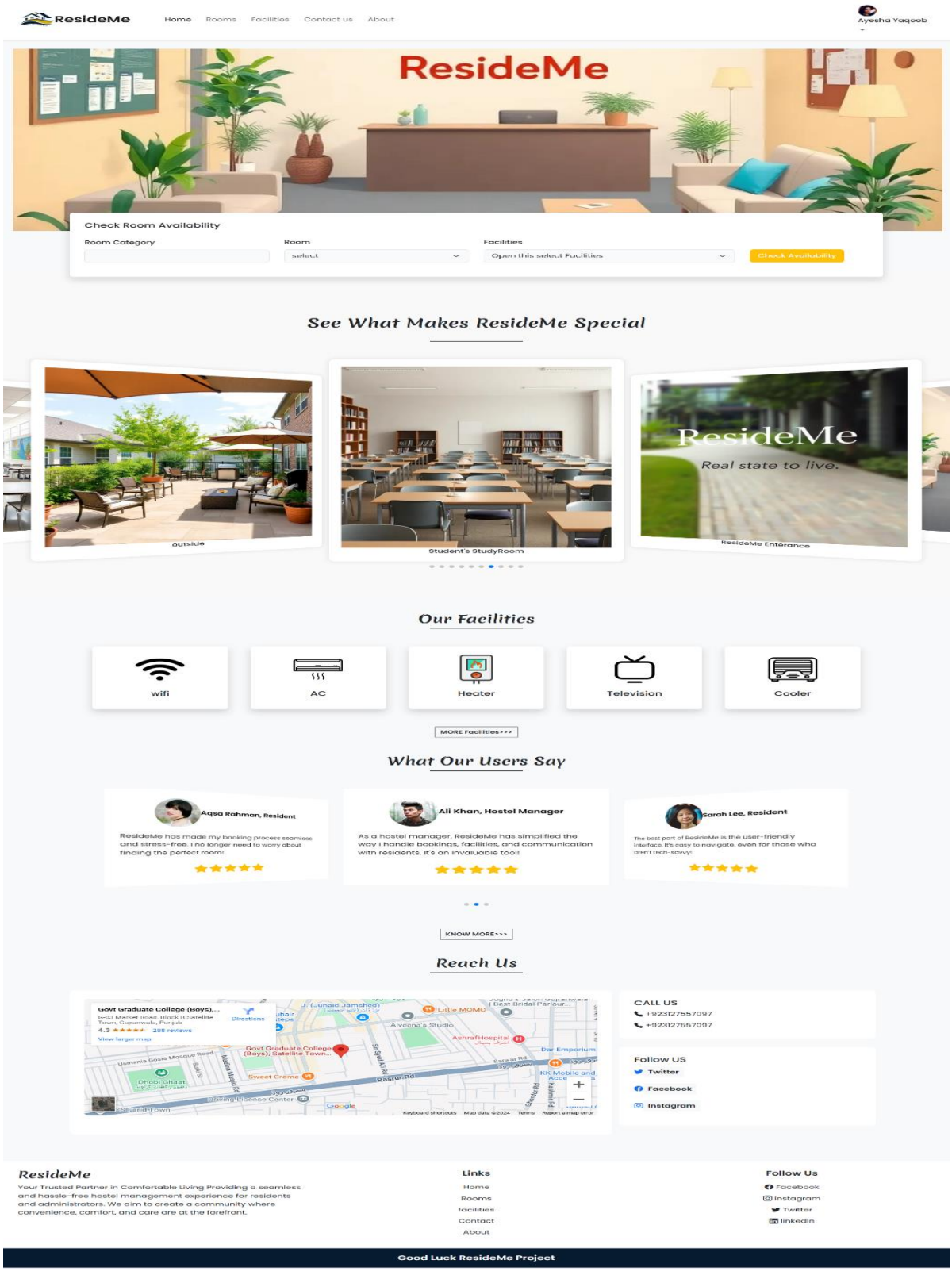
12.2.1 Login



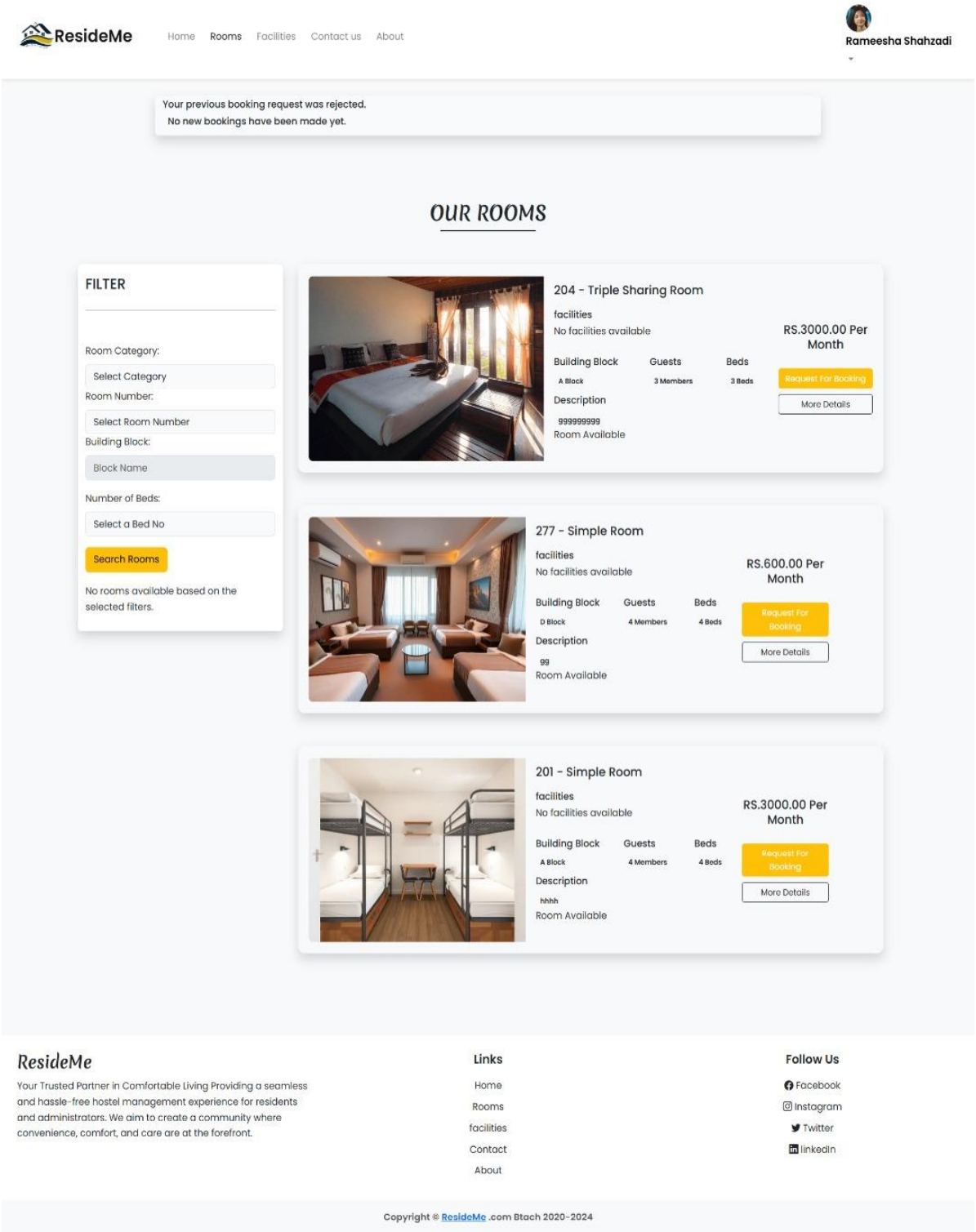
12.2.2 User Profile



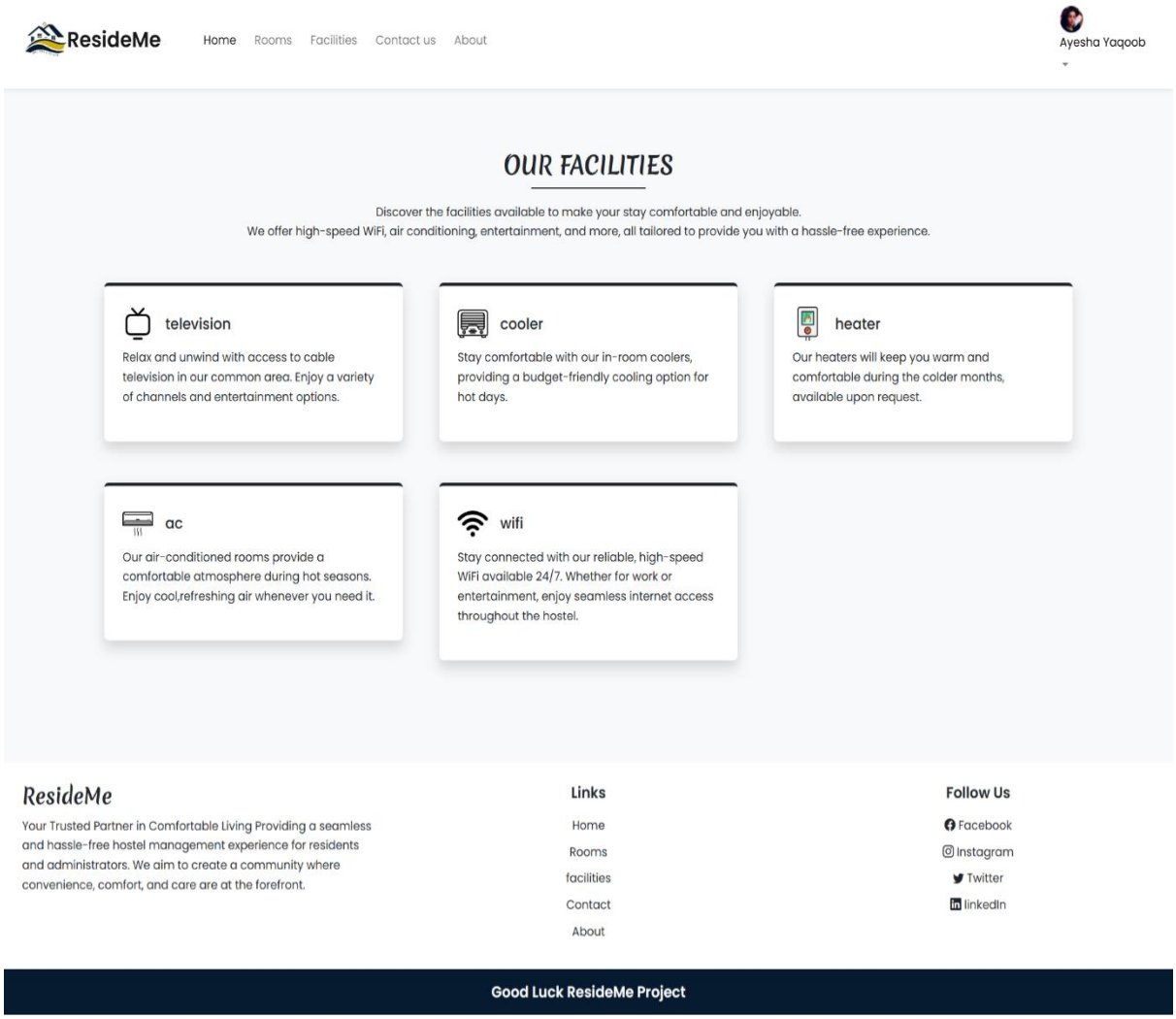
12.2.3 Home Page



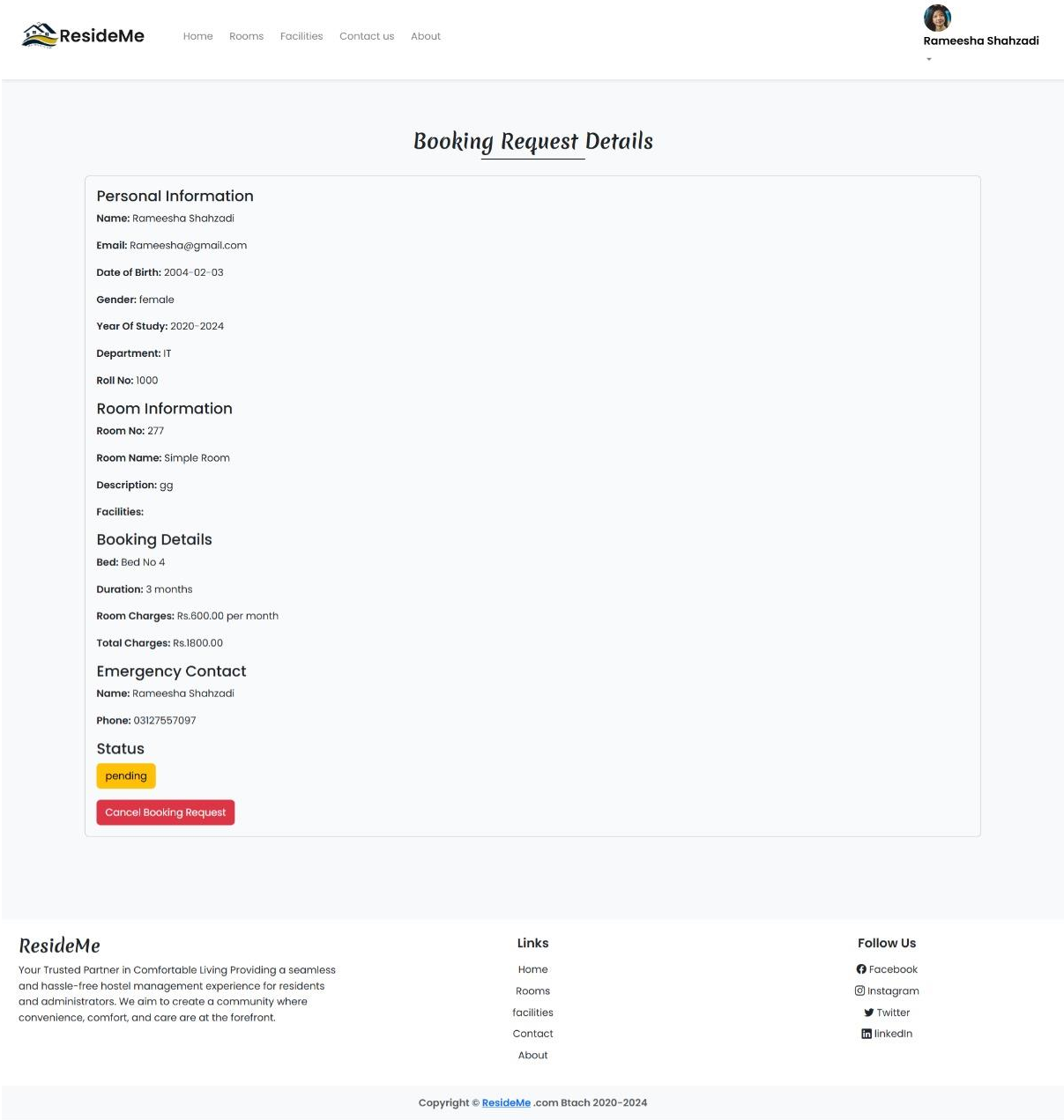
12.2.4 Rooms Page



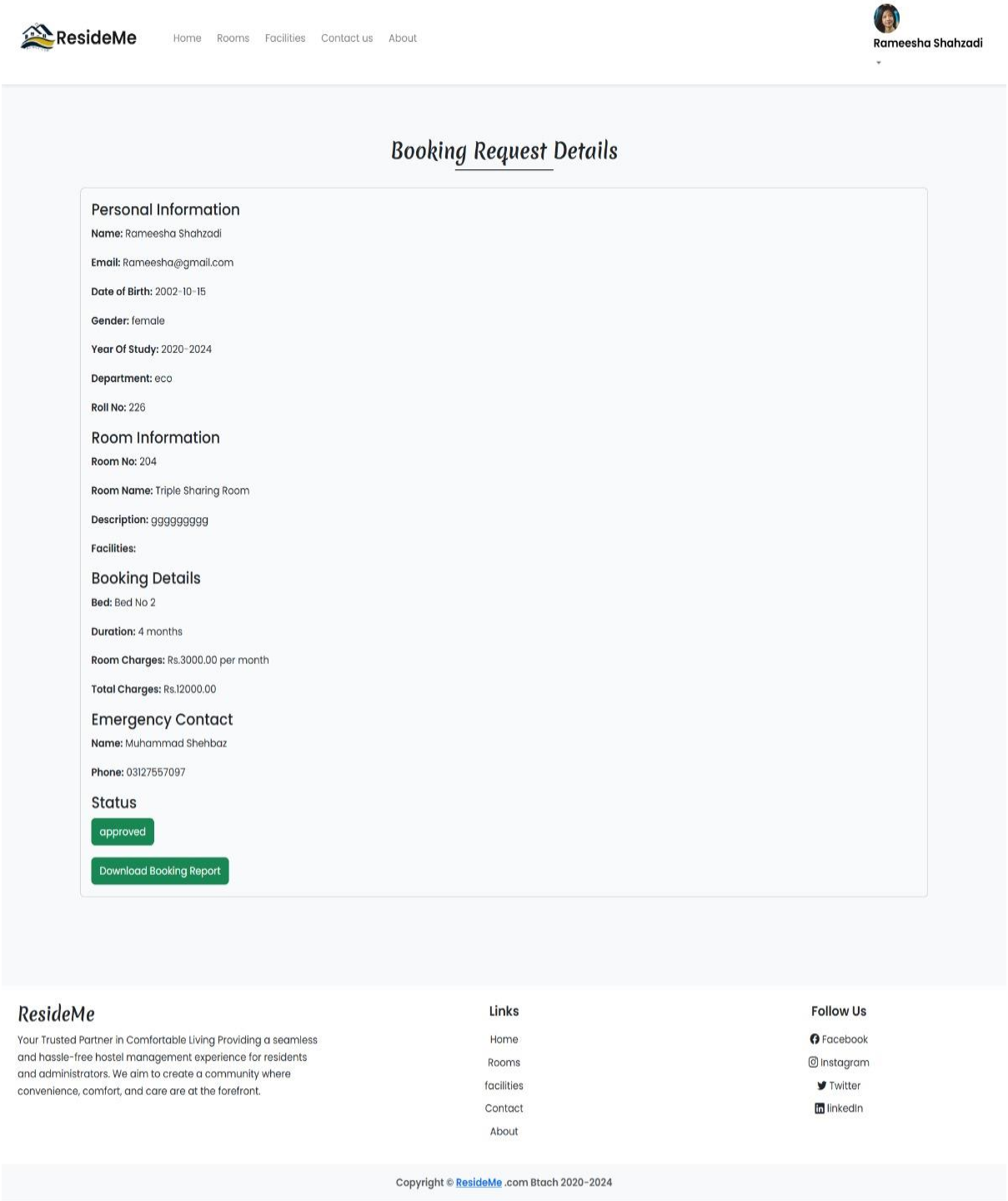
12.2.5 Facilities Page



12.2.6 Booking Request Details



12.2.7 Approved Booking Request Detail



12.2.8 User Report

Booking Report

Personal Information

Name: Rameesha Shahzadi

Email: Rameesha@gmail.com

Date of Birth: 2002-10-15

Gender: female

Year of Study: 2020-2024

Department: eco

Roll No: 226

Room Information

Room No: 204

Room Name: Triple Sharing Room

Description: ggggggggg

Facilities:

Booking Details

Bed: Bed No 2

Duration: 4 months

Room Charges: Rs. per month

Total Charges: Rs.12000.00

Emergency Contact

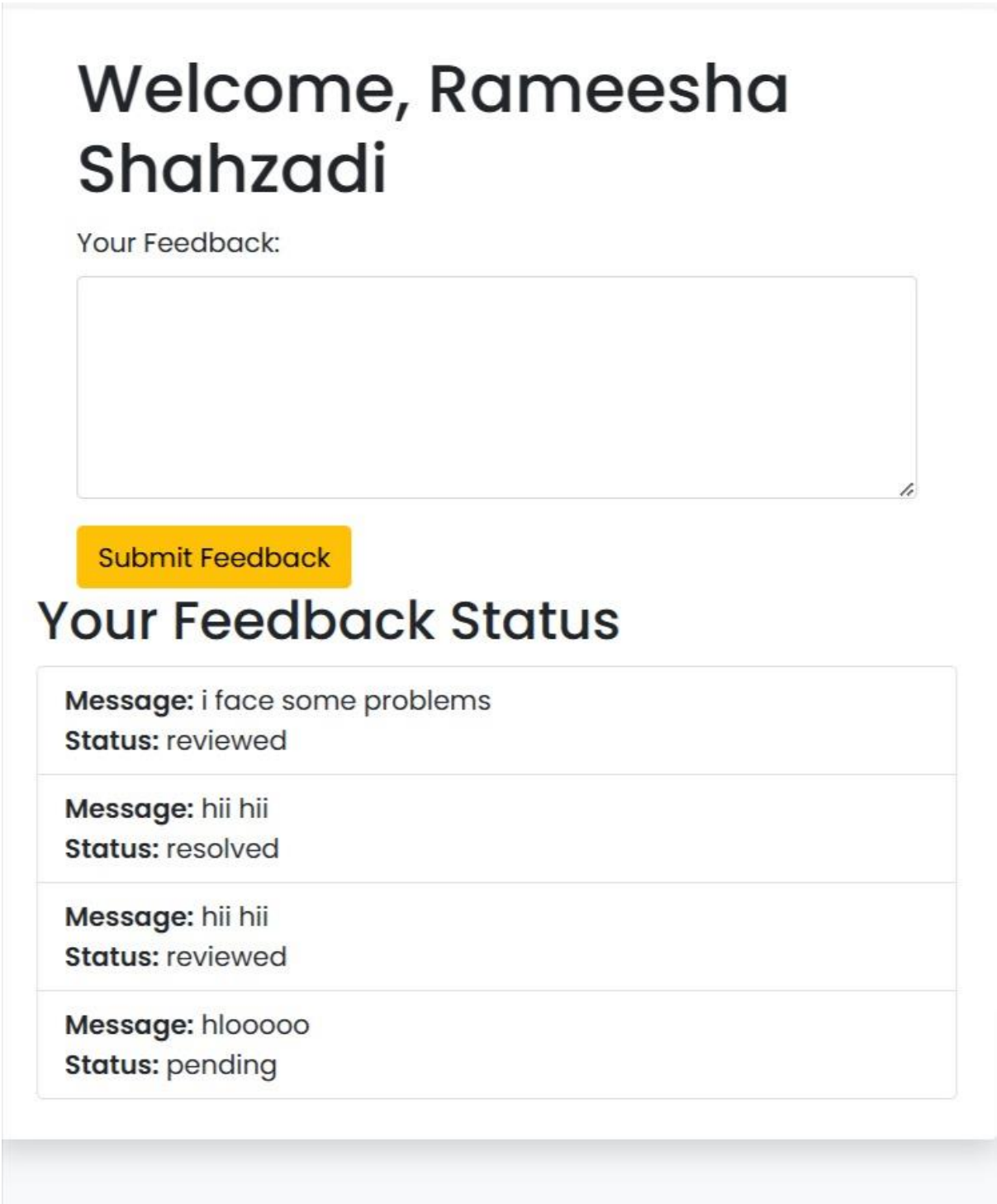
Name: Muhammad Shehbaz

Phone: 03127557097

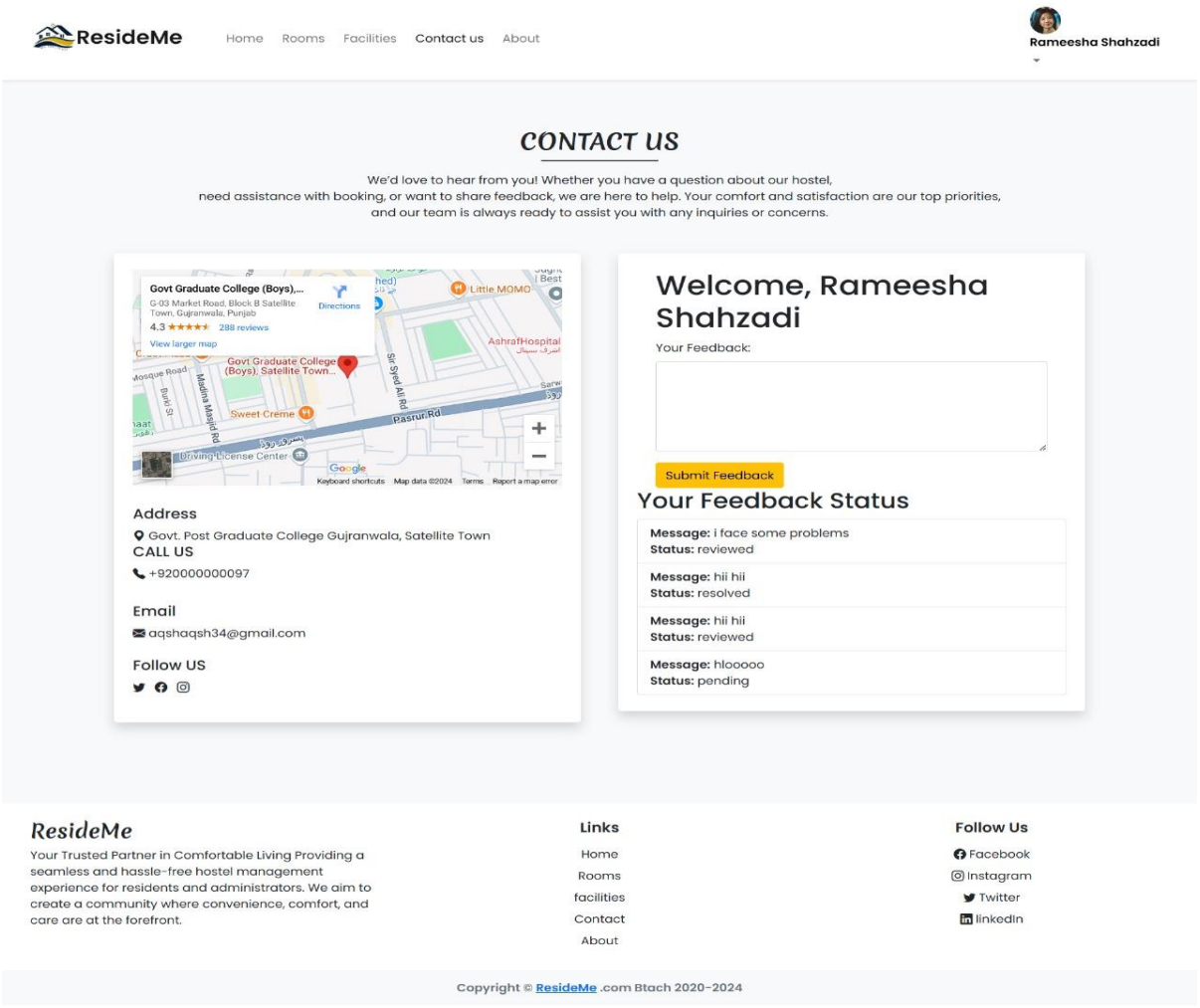
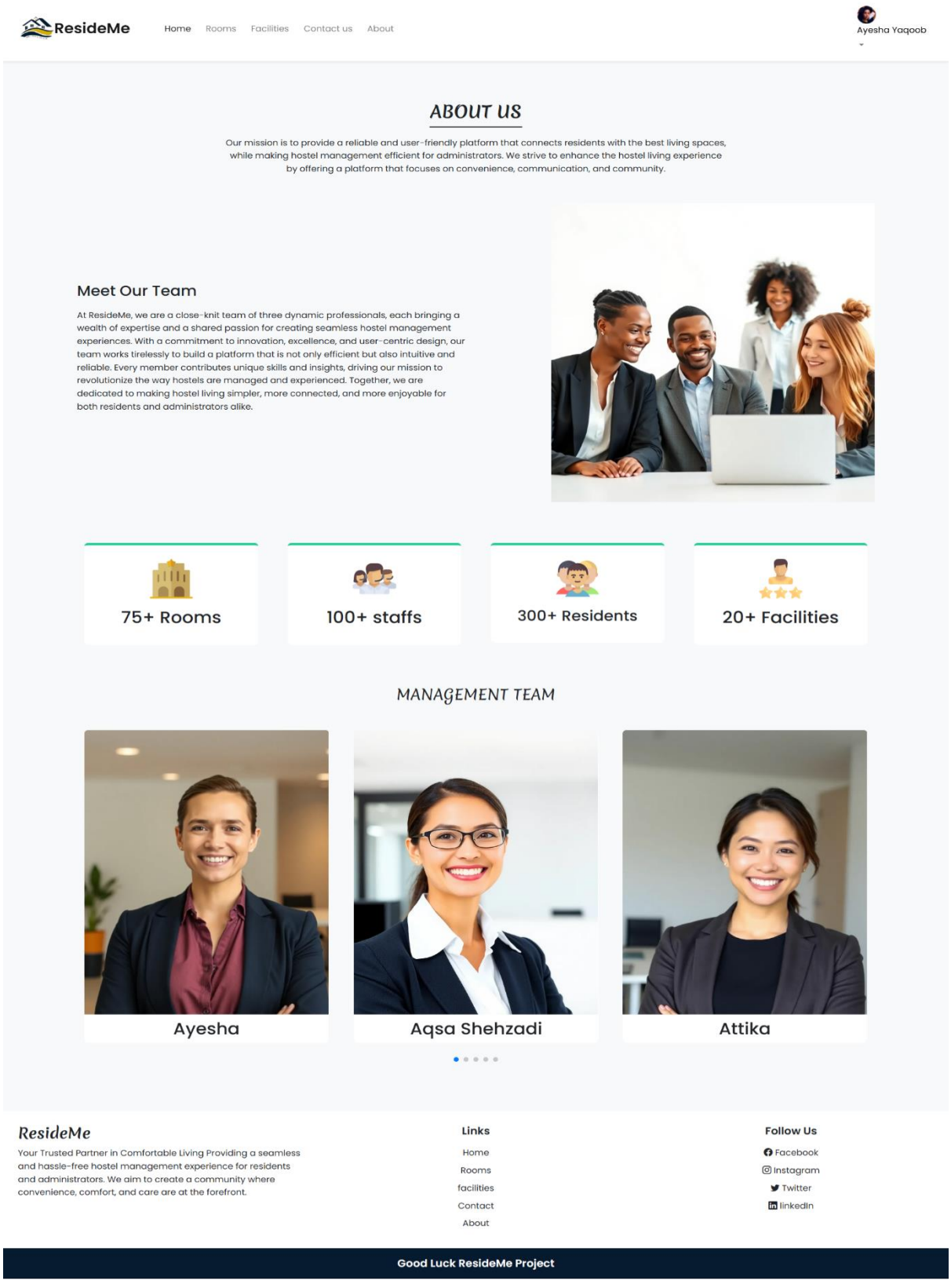
Status

Status: approved

12.2.9 User Feedbacks



12.2.10 Contact Us Page



12.2.11 About Page