Design Document

for

HallBuddy

Version 1.1

Prepared by

Group #: 12 **Group Name: CodeMonk**

Ritesh Hans	220893	riteshhans22@iitk.ac.in
Mridul Gupta	220672	mridulg22@iitk.ac.in
Rohan Batra	210868	rohanb21@iitk.ac.in
Krutuparna Paranjape	210536	krutuparna21@iitk.ac.in
Mrdul Agarwaal	210632	mrdula21@iitk.ac.in
Apoorv Tandon	220192	apoorvt22@iitk.ac.in
Tanishq Maheshwari	221128	tanishqm22@iitk.ac.in
Taneshq Zendey	221123	taneshq22@iitk.ac.in
Ayush	220259	ayushs@iit.ac.in
Samarpan Verma	220943	samarpanv22@iitk.ac.in

Course: CS253

Mentor TA: *Vaibhav Tanwar*

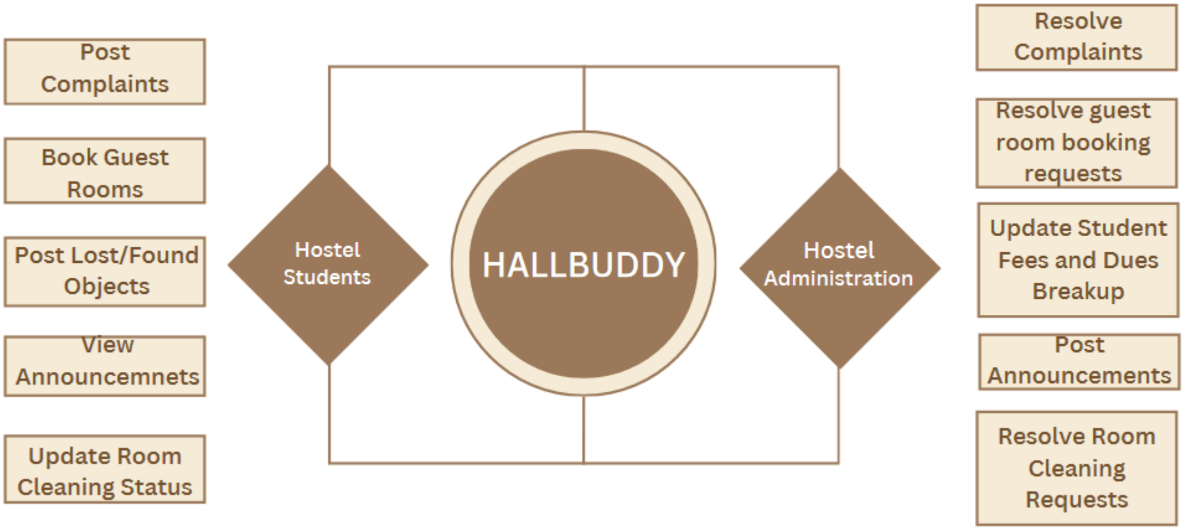
Date: 26.01.2024

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.1	Apoorv Tandon Ayush Mridul Gupta Mrdul Agarwaal Krutuparna Paranjape Rohan Batra Samarpan Verma Ritesh Hans Tanishq Maheshwari Taneshq Zendey	First Draft : This version contains details of the user interface diagrams, use case diagrams, class diagrams, sequence diagrams, and state diagrams for our software.	09/02/24

1. Context Design

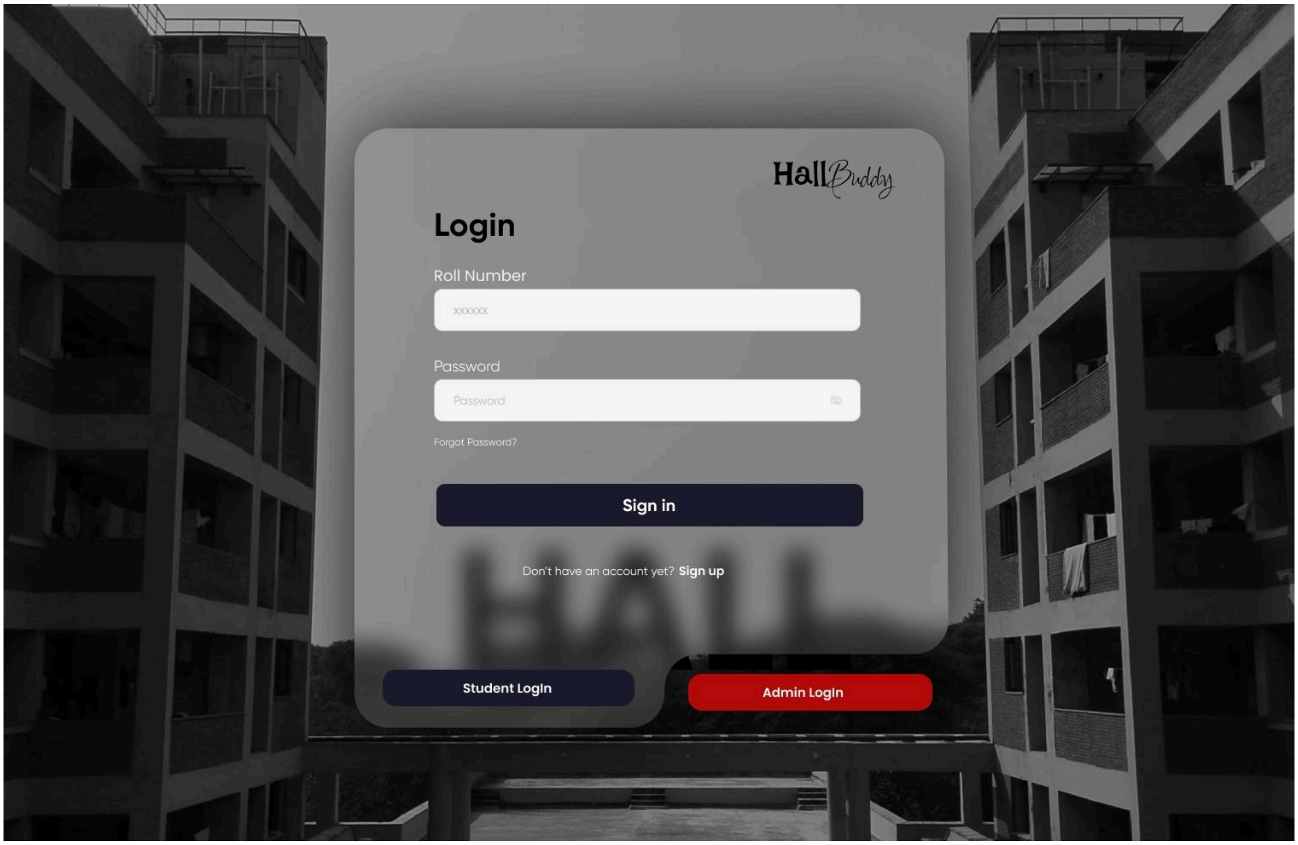
1.1 Context Model



1.2 Human Interface Design

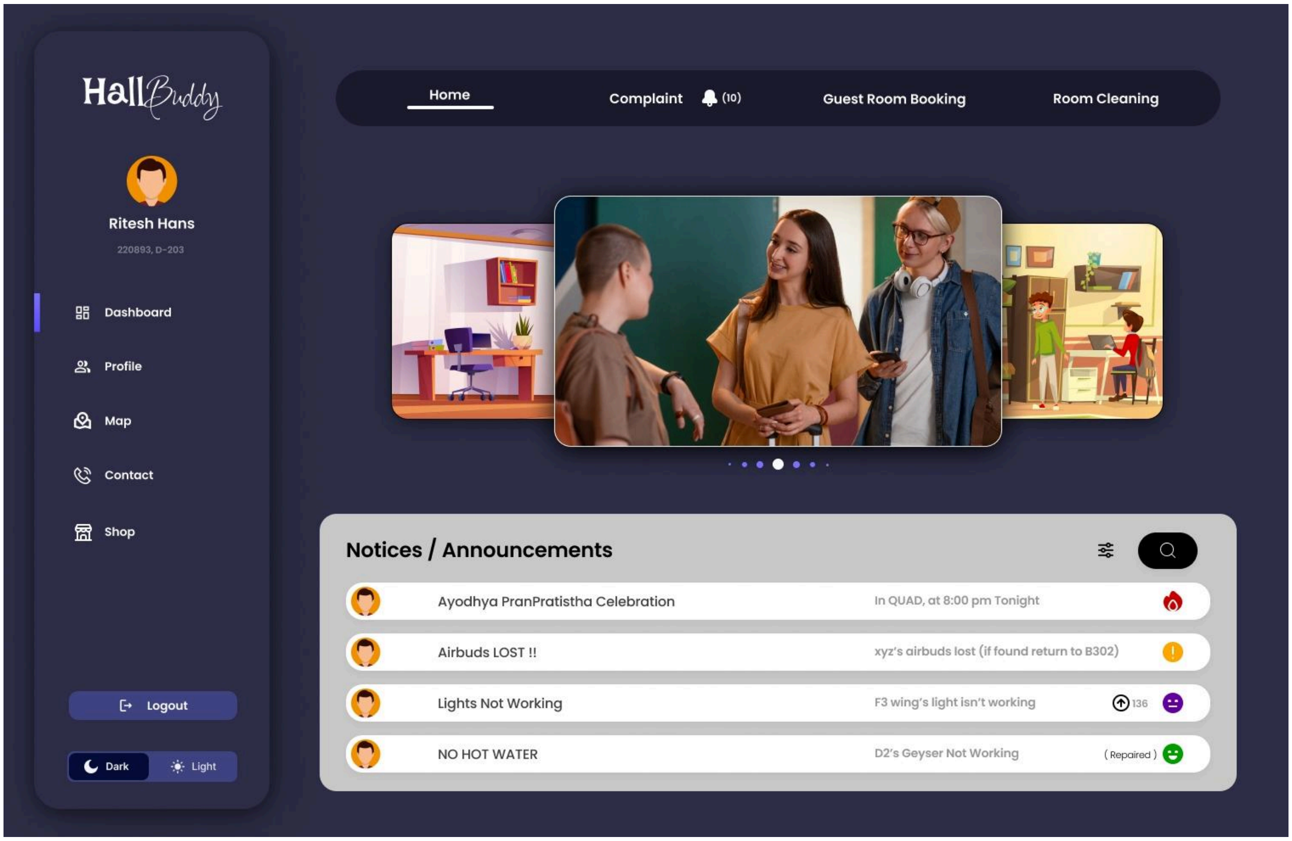
1. Log-In Page

This page allows both students and admin to login and access their account. It also contains fields to signup if the user is not registered as well as a forgot password option if a registered user may have forgotten his password.



2. *Dashboard*

The dashboard contains options to access one’s profile, hall map, lodge a complaint, book the guest room, request/update room cleaning , check the hall shop catalog or contact the warden/HEC.

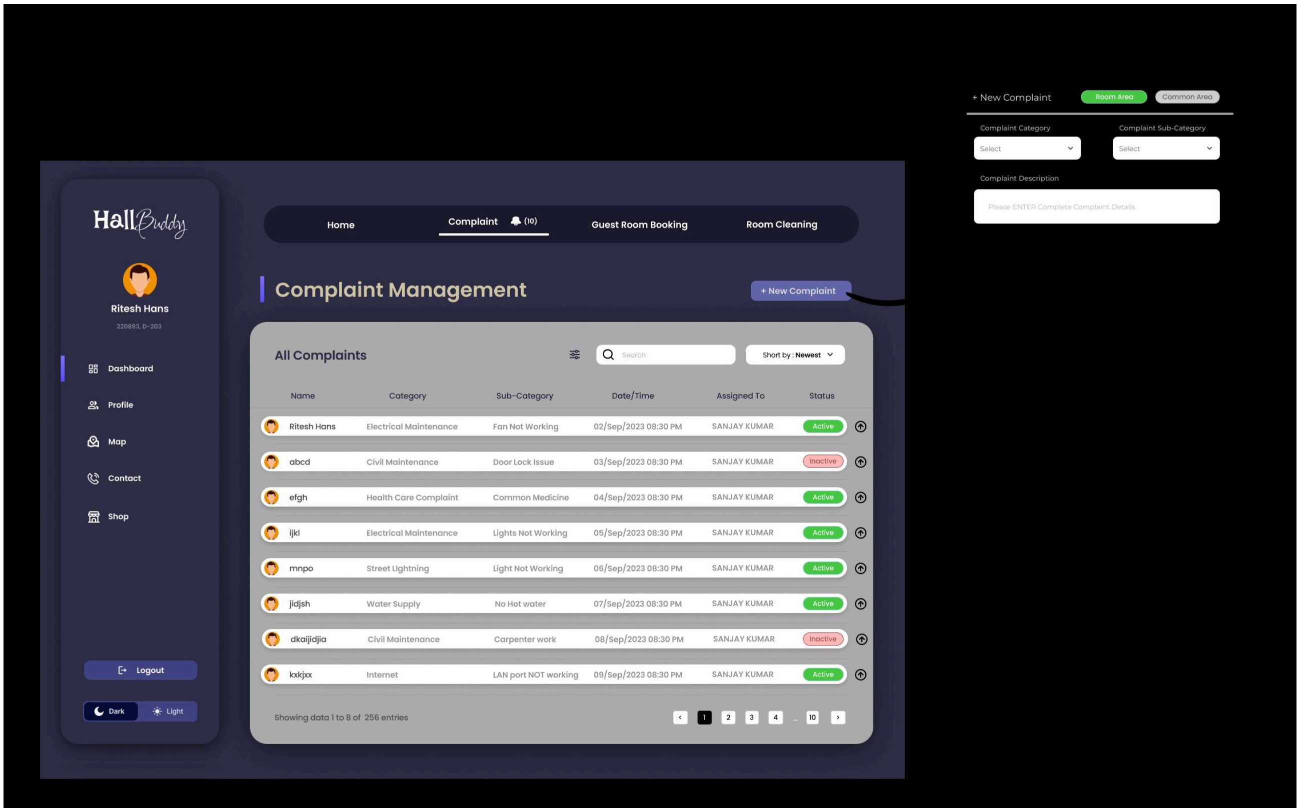


Notices / Announcements

Ayodhya PranPratistha Celebration	In QUAD, at 8:00 pm Tonight	
Airbuds LOST !!	xyz’s airbuds lost (if found return to B302)	!
Lights Not Working	F3 wing’s light isn’t working	↑ 136
NO HOT WATER	D2’s Geyser Not Working	(Repaired)

3. Complaint Management Page

This is the complaints page wherein one can lodge a new complaint by clicking on the new complaint option or view the existing complaints.



Complaint Category: Select

Complaint Sub-Category: Select

Complaint Description: Please ENTER Complete Complaint Details

Complaint Management

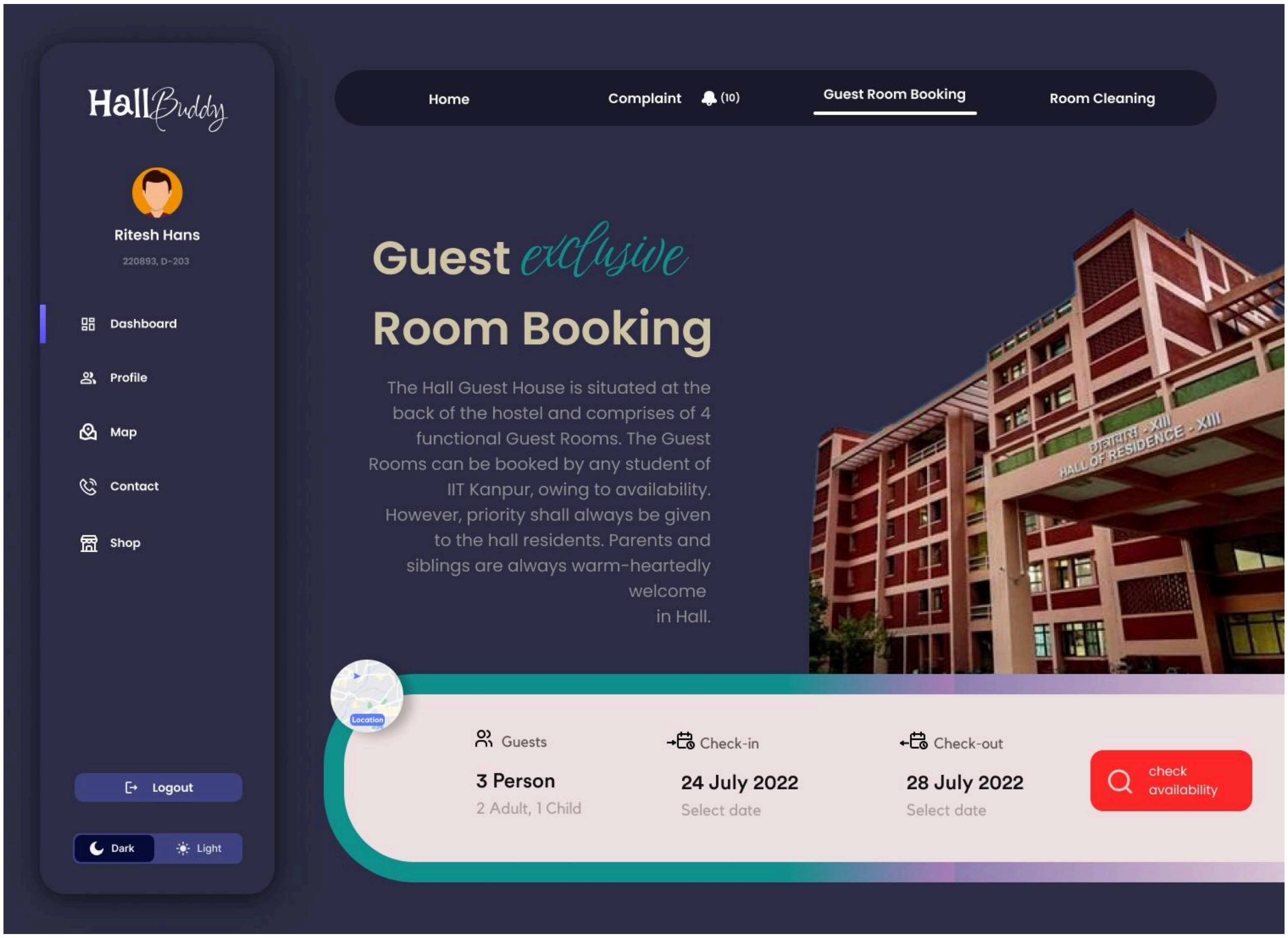
All Complaints

Name	Category	Sub-Category	Date/Time	Assigned To	Status
Ritesh Hans	Electrical Maintenance	Fan Not Working	02/Sep/2023 08:30 PM	SANJAY KUMAR	Active
abcd	Civil Maintenance	Door Lock Issue	03/Sep/2023 08:30 PM	SANJAY KUMAR	Inactive
efgh	Health Care Complaint	Common Medicine	04/Sep/2023 08:30 PM	SANJAY KUMAR	Active
ijkl	Electrical Maintenance	Lights Not Working	05/Sep/2023 08:30 PM	SANJAY KUMAR	Active
mnpo	Street Lightning	Light Not Working	06/Sep/2023 08:30 PM	SANJAY KUMAR	Active
jidjsh	Water Supply	No Hot water	07/Sep/2023 08:30 PM	SANJAY KUMAR	Active
dkaijidjia	Civil Maintenance	Carpenter work	08/Sep/2023 08:30 PM	SANJAY KUMAR	Inactive
kxkjxx	Internet	LAN port NOT working	09/Sep/2023 08:30 PM	SANJAY KUMAR	Active

Showing data 1 to 8 of 256 entries

4. *Guest Room Booking*

The Guest Room Booking Page allows the user to check the price and the availability of the guest room after entering the Check-in Dates, Check-Out Date and the number of guests.



Guest exclusive Room Booking

The Hall Guest House is situated at the back of the hostel and comprises of 4 functional Guest Rooms. The Guest Rooms can be booked by any student of IIT Kanpur, owing to availability. However, priority shall always be given to the hall residents. Parents and siblings are always warm-heartedly welcome in Hall.

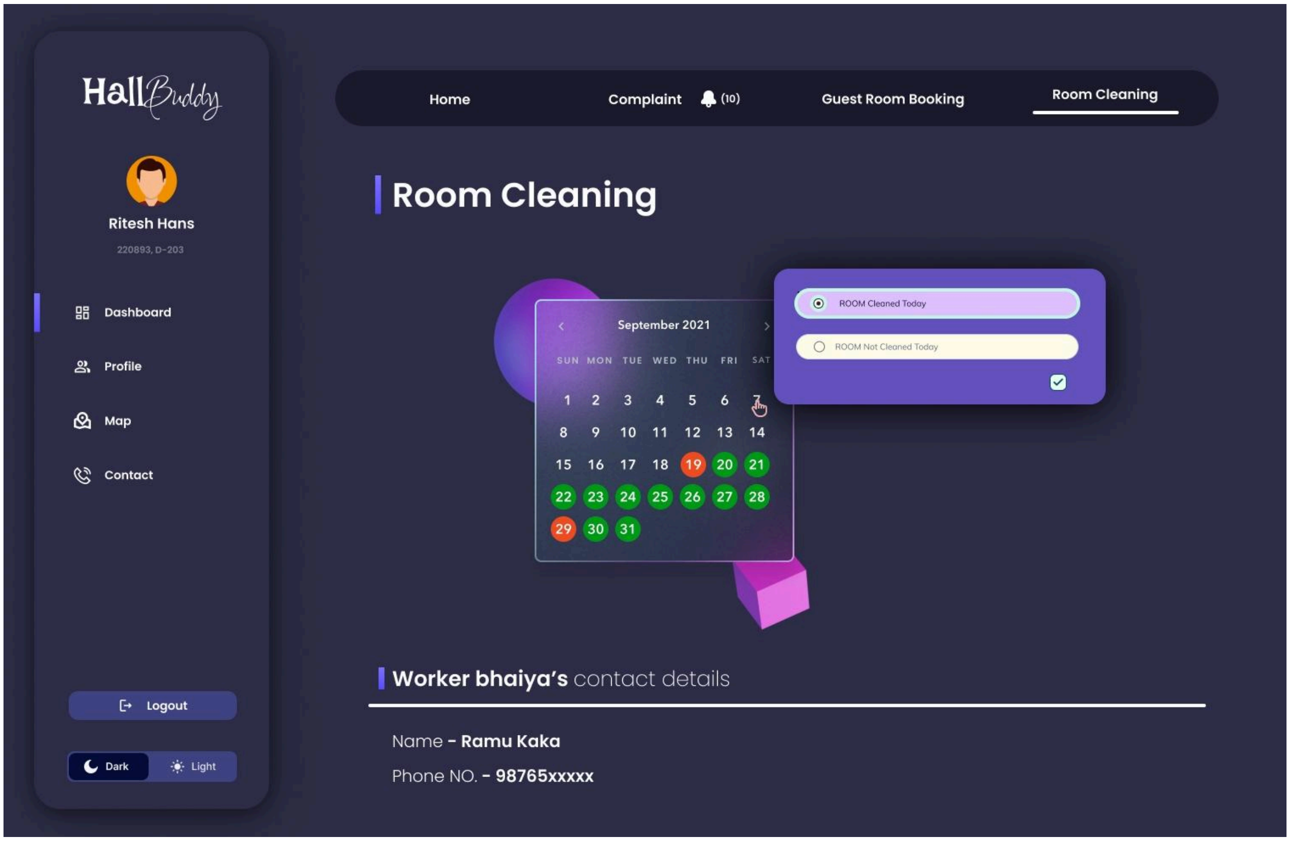
Guests: 3 Person
2 Adult, 1 Child

Check-in: 24 July 2022
Select date

Check-out: 28 July 2022
Select date

5. Room Cleaning

The room cleaning portal would allow the user to update whether the room has been cleaned on a particular day. The student can also contact the respective worker responsible for his wing.



Ritesh Hans

220893, D-203

Room Cleaning

September 2021

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

● ROOM Cleaned Today

○ ROOM Not Cleaned Today

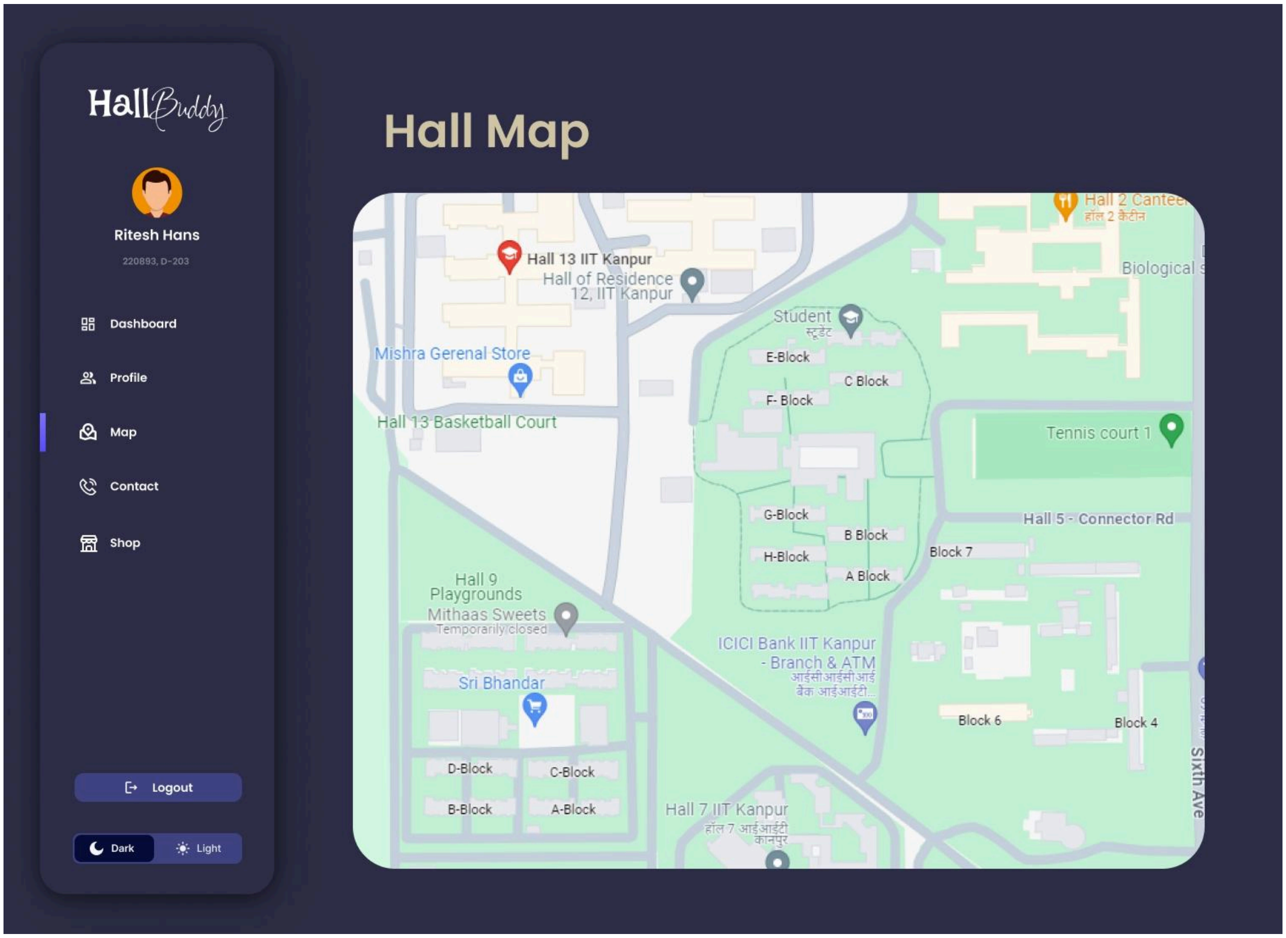
☑

Worker bhaiya's contact details

Name **- Ramu Kaka**

Phone NO. **- 98765xxxxx**

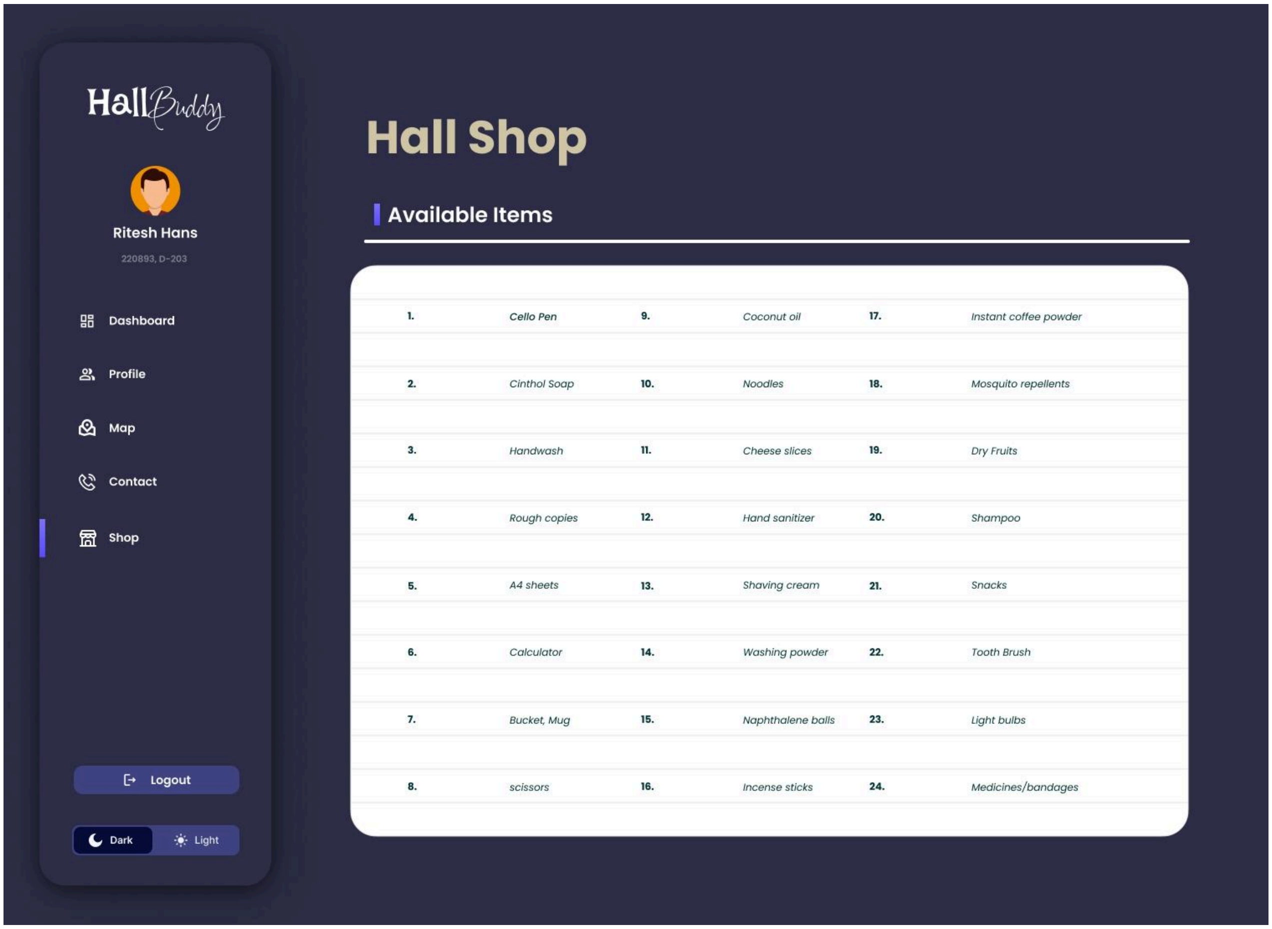
6. ***Hall Map***
   The Hall map contains the map of the hall for easier reachability and access to different hostel facilities.



7. Hall Shop

The hall shop contains the hall shop catalog wherein the student can check if the requisite item is available or not. They can also contact the Hall Owner to request certain items

The hall shop owner would have a different interface wherein he can update the time available with him .



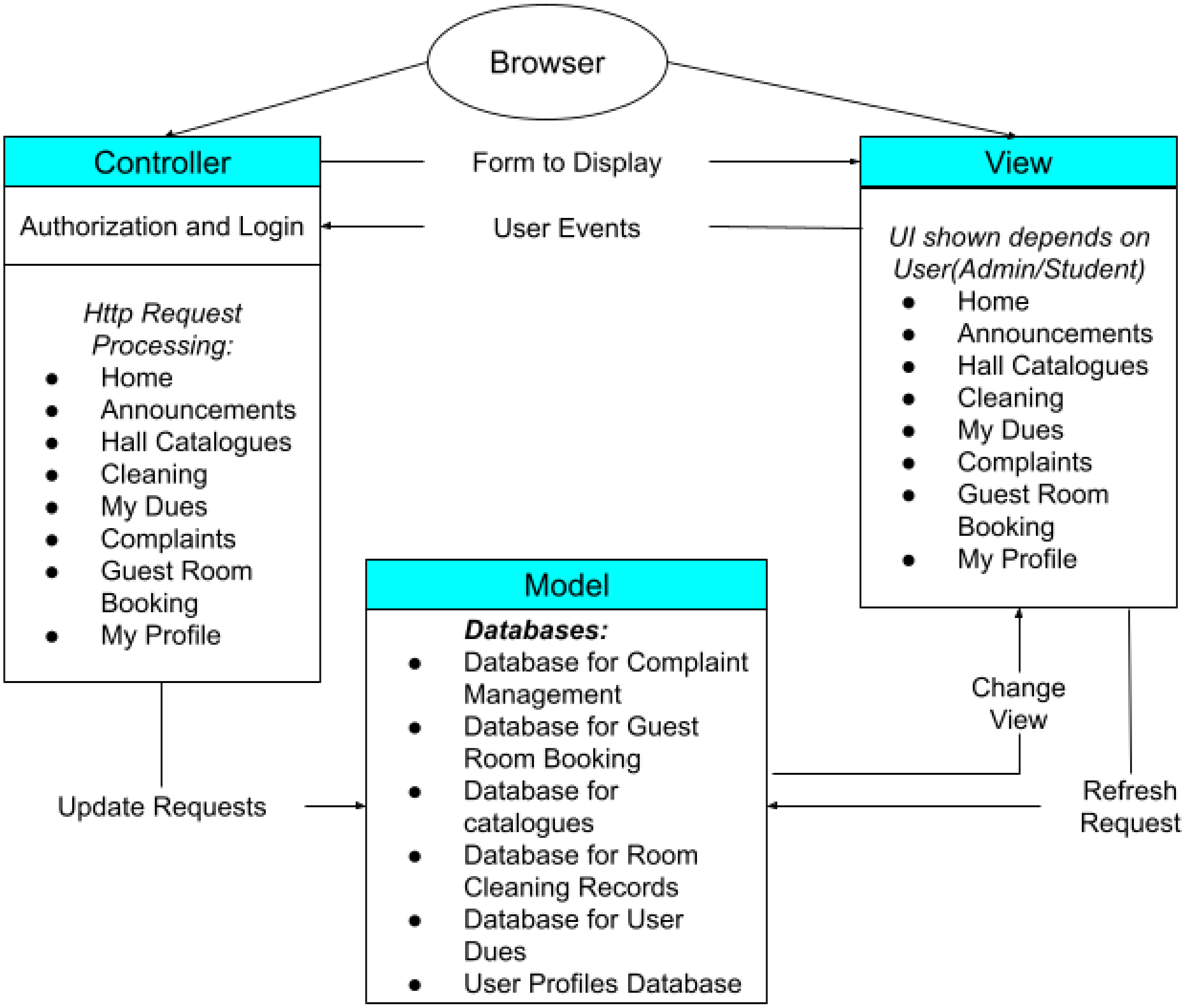
Hall Shop

Available Items

1.	Cello Pen	9.	Coconut oil	17.	Instant coffee powder
2.	Cinthol Soap	10.	Noodles	18.	Mosquito repellents
3.	Handwash	11.	Cheese slices	19.	Dry Fruits
4.	Rough copies	12.	Hand sanitizer	20.	Shampoo
5.	A4 sheets	13.	Shaving cream	21.	Snacks
6.	Calculator	14.	Washing powder	22.	Tooth Brush
7.	Bucket, Mug	15.	Naphthalene balls	23.	Light bulbs
8.	scissors	16.	Incense sticks	24.	Medicines/bandages

2. Architecture Design

Model-View-Control Pattern



The Model-View-Controller architecture is very common amongst Web Applications since it allows maintenance and editing of a uniform database while allowing for different presentations of that data based on request/user.

- This model does not interfere with implementation of security, latency and responsiveness constraints.
- The model ensures reliability of data due to the streamlined mechanisation of data queries, presentation, and editing, as depicted in the model above.

Model:

1. Multiple Databases will be maintained in the software:
    a. Database for Complaint Management
    b. Database for Guest Room Booking
    c. Database for catalogues
    d. Database for Room Cleaning Records
    e. Database for User Dues
    f. User Profiles Database

Controller:

The main parts of the controller will be:

- User Authorization
- HTTP Requests

Authorization: Will include Sign-Up, Sign-In, Forgot Password functionality
HTTP Requests: Will be passed and evaluated based on the current state of the software

- Home
- Announcements
- Catalogues
- Cleaning
- My Dues
- Complaints
- Guest Room Booking
- My Profile

View:

According to their identity, users are shown different UIs amongst:

- Home
- Announcements
- Catalogues
- Cleaning
- My Dues
- Complaints
- Guest Room Booking
- My Profile

The links between Model, Controller and View are as follows:

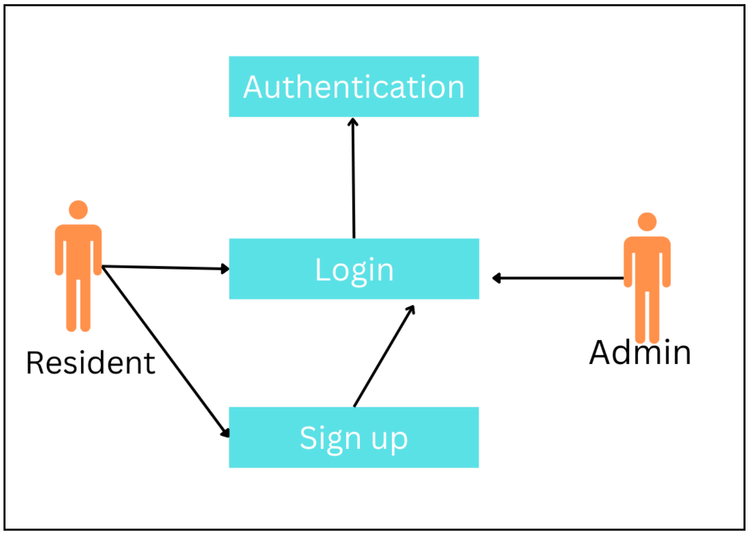
- The Controller authorizes based on User data stores in the model, and implements any updates in the model that are initiated by Admin
- Views communicate with Controllers and Models to cater to requests sent by the user.

3. Object Oriented Design

3.1 Use Case Diagrams

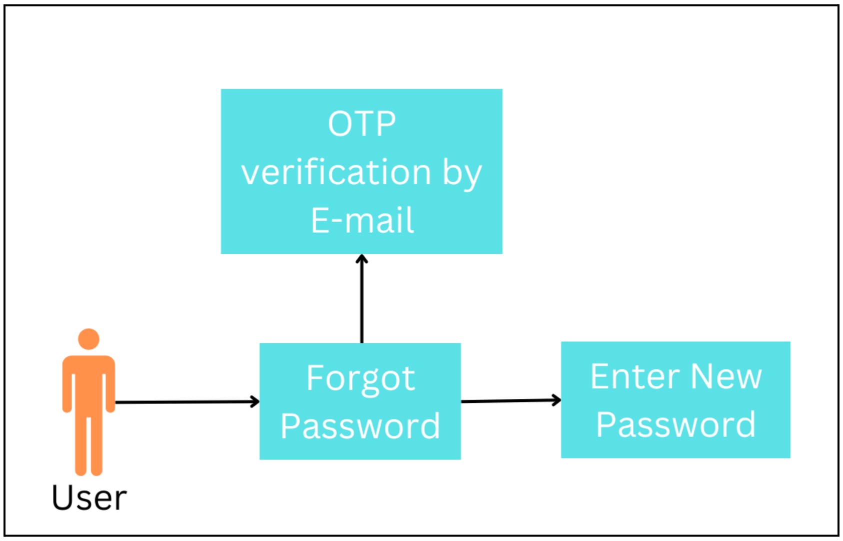
1. Sign Up and Login

Purpose: This use case is used for the Login/ Signup of the Resident and the Admin. Login is used for the registered users and the Signup is used for the first-time users.



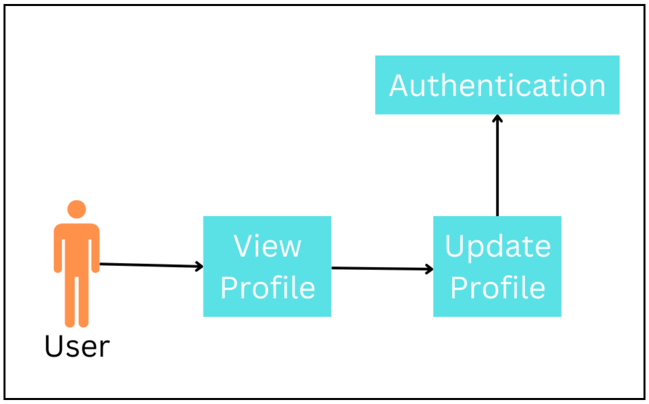
2. Forgot Password

Purpose:This use case is used for new password creation when the user forgets their password.



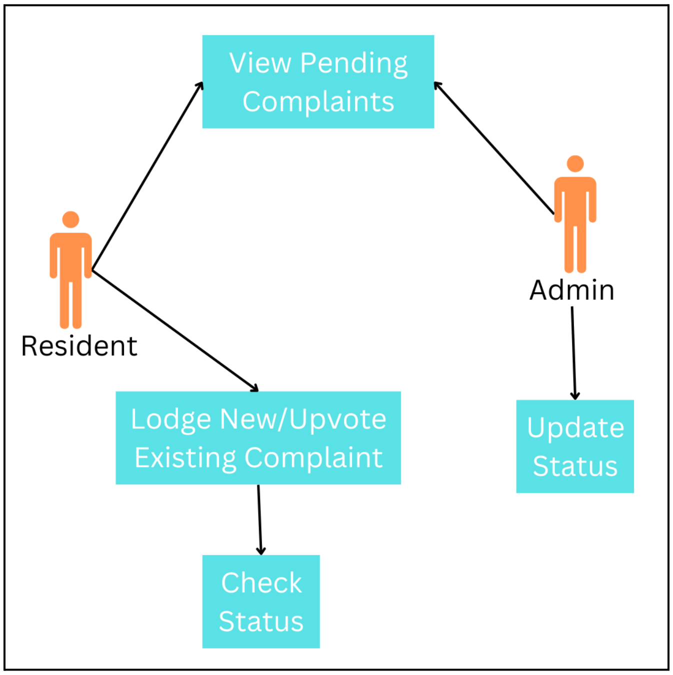
3. View and Update Profile

Purpose: This use case allows the Resident to view and update their Profile Information including Room Number and Contact Details.



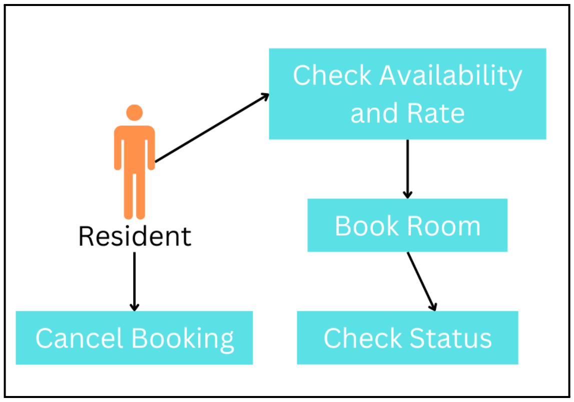
4. Complaint management

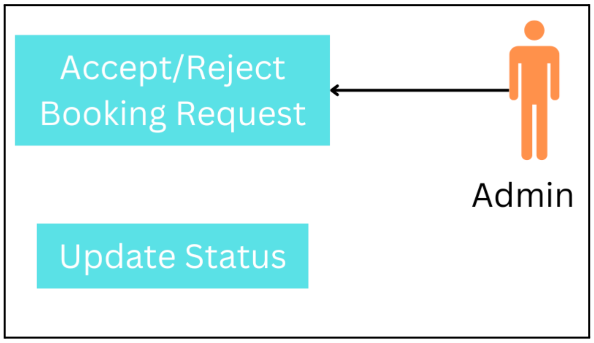
Purpose: This use case allows the users to view the existing complaints and check their response status. The Resident can upvote existing complaints if they concern them. The admin can manage the response status of the complaints as – "Resolved", "In Progress".It also allows the Resident to create new complaints.The resident will be able to enter text description, attach images and select suitable filters based on the category of complaint (e.g. Electrical, Civil maintenance etc.).



5. Guest Room Booking

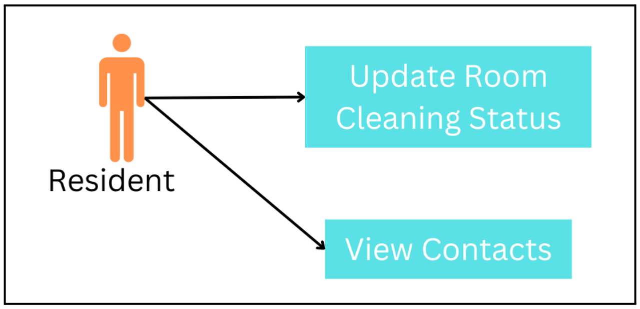
Purpose: This use case is for handling guest room bookings. The user can view prices and availability and book the guest room. The admin must approve the request before the booking is confirmed. This also allows the user to cancel his/her booking

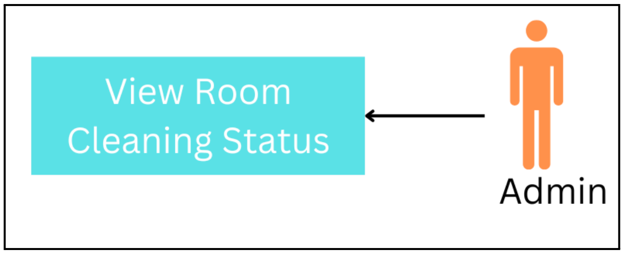




6 .Room Cleaning Status

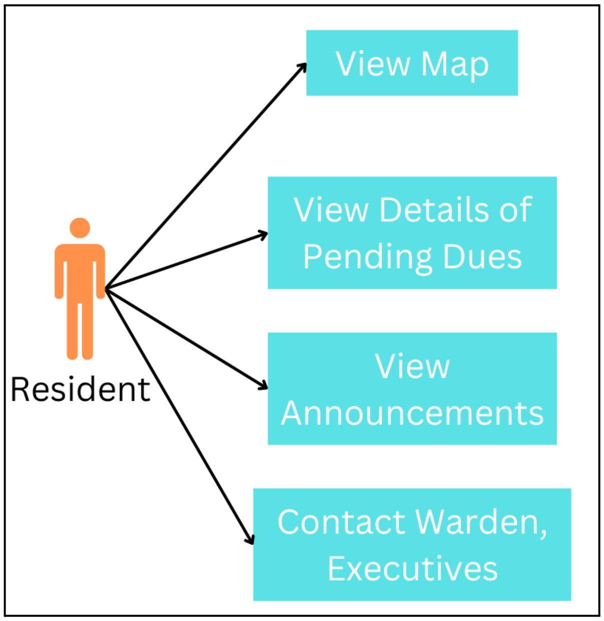
Purpose:The purpose of this user case is to provide users with the ability to update the cleaning status of rooms within the hall management system.

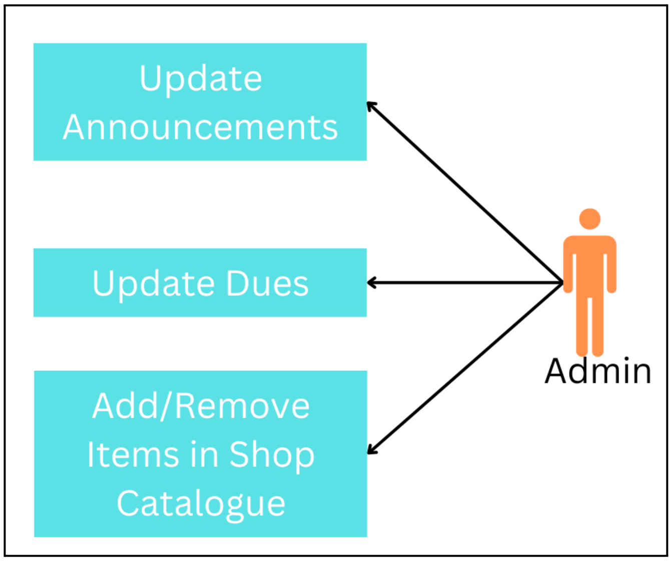




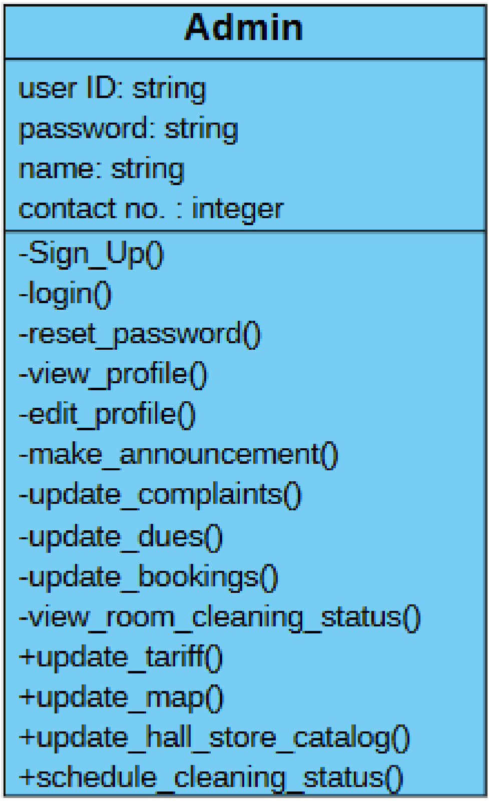
7. Miscellaneous Use Cases

Purpose: This use case covers all miscellaneous actions of the user/admin such as viewing map of the hostel, contacts of the hall executives, view pending dues.





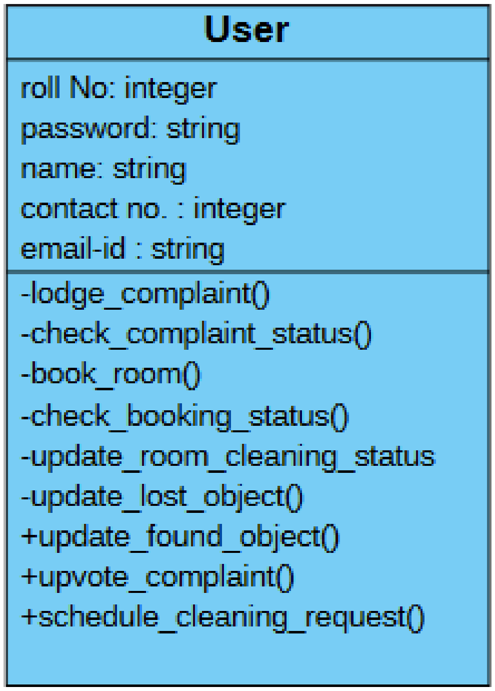
3.2 Class Diagrams



Admin

user ID: string
password: string
name: string
contact no. : integer

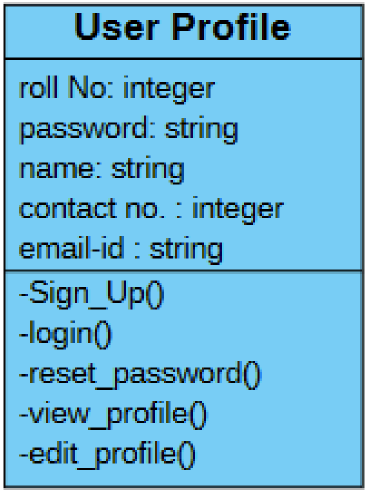
-Sign_Up()
-login()
-reset_password()
-view_profile()
-edit_profile()
-make_announcement()
-update_complaints()
-update_dues()
-update_bookings()
-view_room_cleaning_status()
+update_tariff()
+update_map()
+update_hall_store_catalog()
+schedule_cleaning_status()



User

roll No: integer
password: string
name: string
contact no. : integer
email-id : string

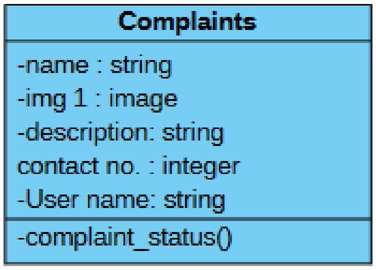
-lodge_complaint()
-check_complaint_status()
-book_room()
-check_booking_status()
-update_room_cleaning_status
-update_lost_object()
+update_found_object()
+upvote_complaint()
+schedule_cleaning_request()

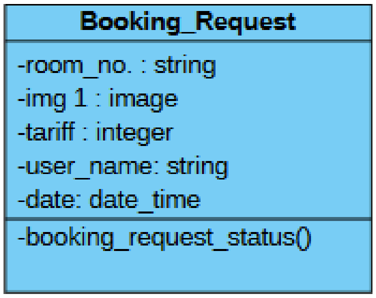


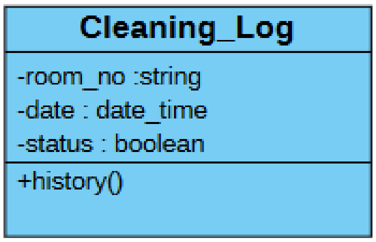
User Profile

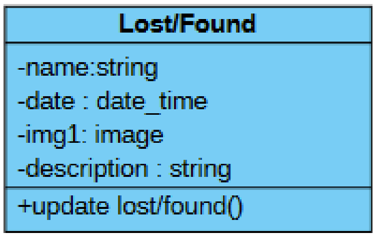
roll No: integer
password: string
name: string
contact no. : integer
email-id : string

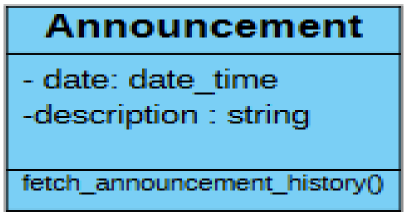
-Sign_Up()
-login()
-reset_password()
-view_profile()
-edit_profile()

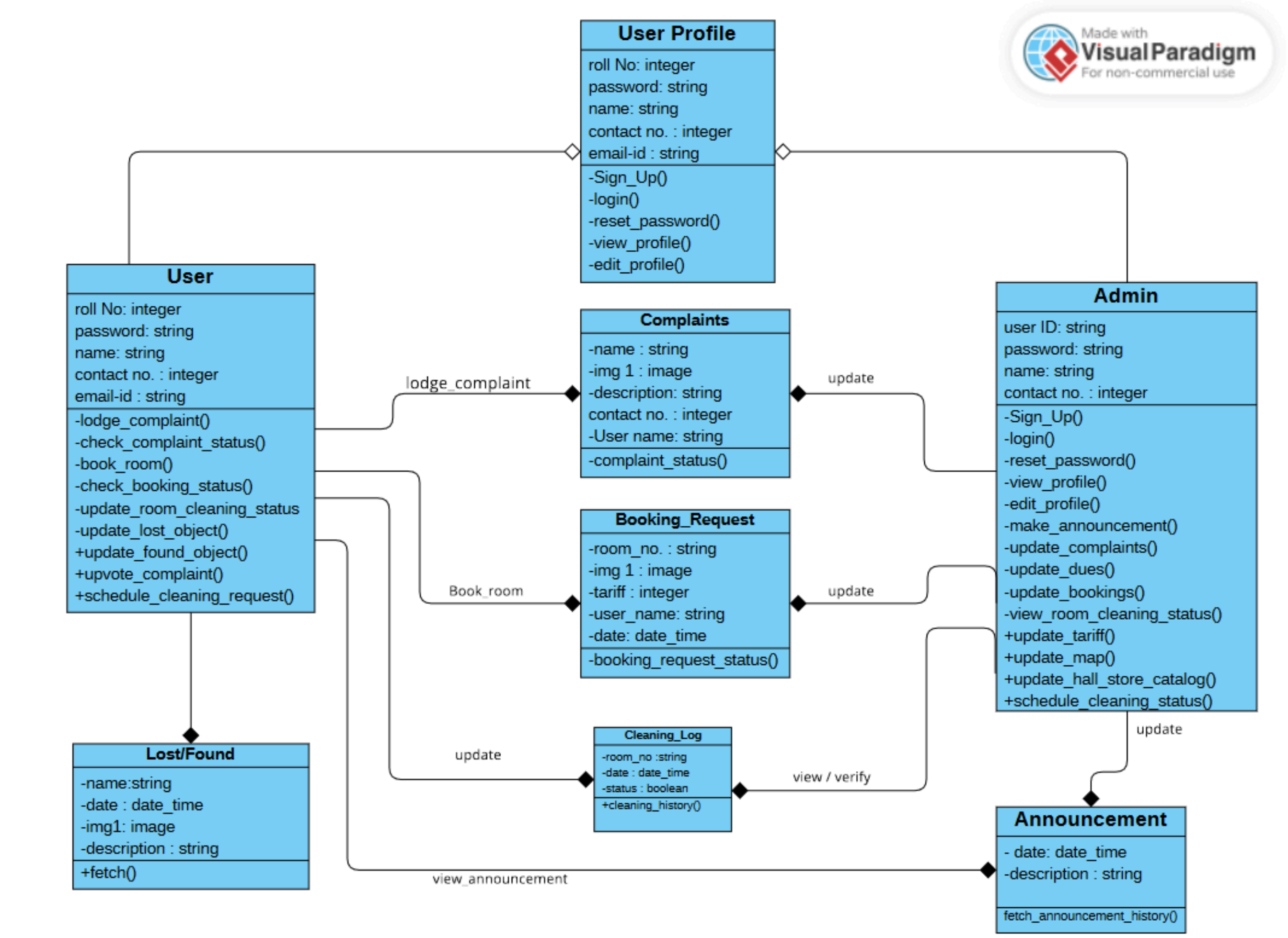

Complaints
-name : string
-img 1 : image
-description: string
contact no. : integer
-User name: string
-complaint_status()


Booking_Request
-room_no. : string
-img 1 : image
-tariff : integer
-user_name: string
-date: date_time
-booking_request_status()


Cleaning_Log
-room_no :string
-date : date_time
-status : boolean
+history()

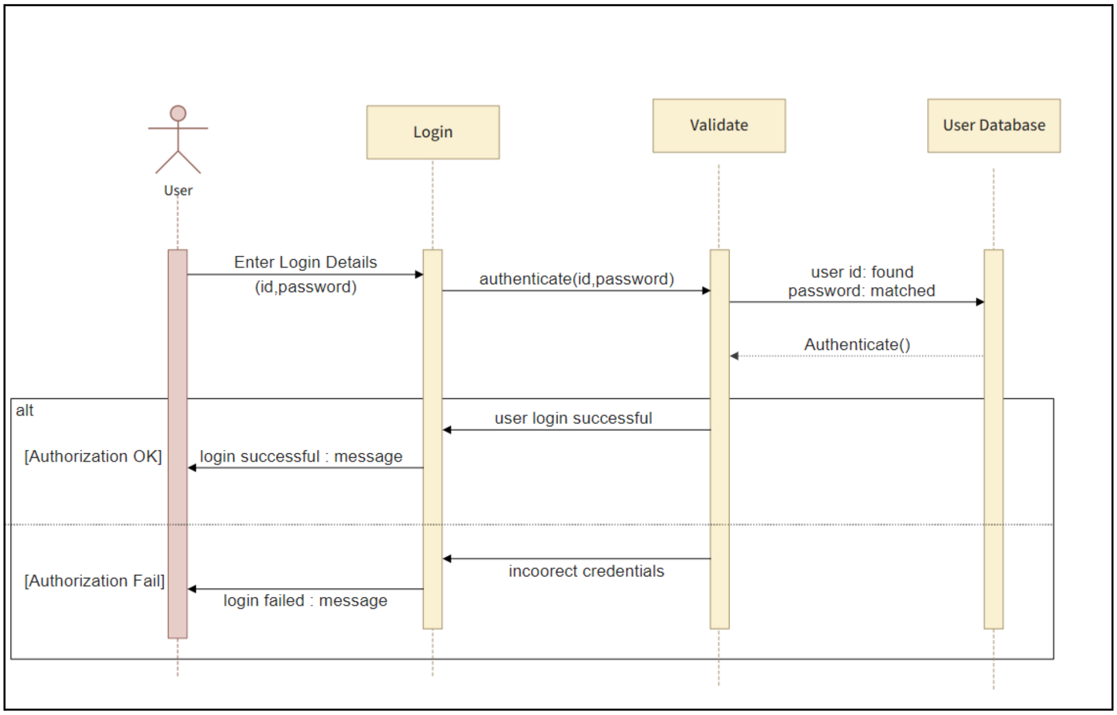

Lost/Found
-name:string
-date : date_time
-img1: image
-description : string
+update lost/found()


Announcement
- date: date_time
-description : string
fetch_announcement_history()

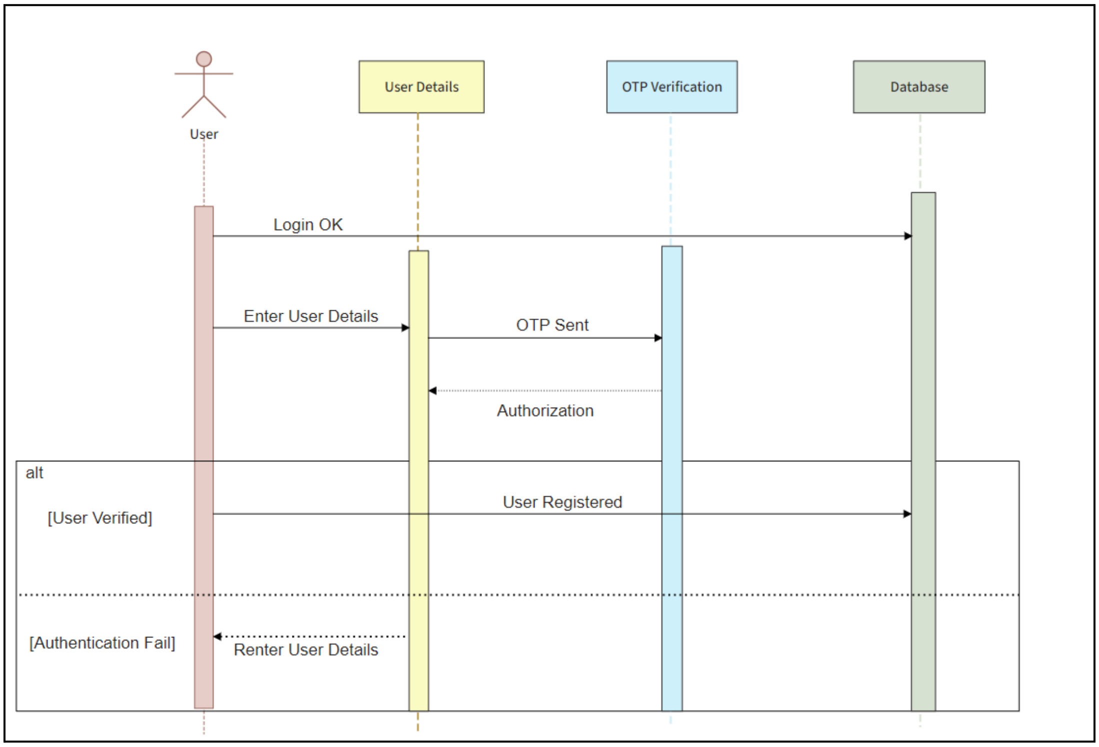

User Profile
roll No: integer
password: string
name: string
contact no. : integer
email-id : string
-Sign_Up()
-login()
-reset_password()
-view_profile()
-edit_profile()
Made with Visual Paradigm For non-commercial use
User
roll No: integer
password: string
name: string
contact no. : integer
email-id : string
-lodge_complaint()
-check_complaint_status()
-book_room()
-check_booking_status()
-update_room_cleaning_status
-update_lost_object()
+update_found_object()
+upvote_complaint()
+schedule_cleaning_request()
lodge_complaint
Complaints
-name : string
-img 1 : image
-description: string
contact no. : integer
-User name: string
-complaint_status()
update
Admin
user ID: string
password: string
name: string
contact no. : integer
-Sign_Up()
-login()
-reset_password()
-view_profile()
-edit_profile()
-make_announcement()
-update_complaints()
-update_dues()
-update_bookings()
-view_room_cleaning_status()
+update_tariff()
+update_map()
+update_hall_store_catalog()
+schedule_cleaning_status()
Book_room
Booking_Request
-room_no. : string
-img 1 : image
-tariff : integer
-user_name: string
-date: date_time
-booking_request_status()
update
Lost/Found
-name:string
-date : date_time
-img1: image
-description : string
+fetch()
update
Cleaning_Log
-room_no :string
-date : date_time
-status : boolean
+cleaning_history()
view / verify
update
Announcement
- date: date_time
-description : string
fetch_announcement_history()
view_announcement

3.3 Sequence Diagrams

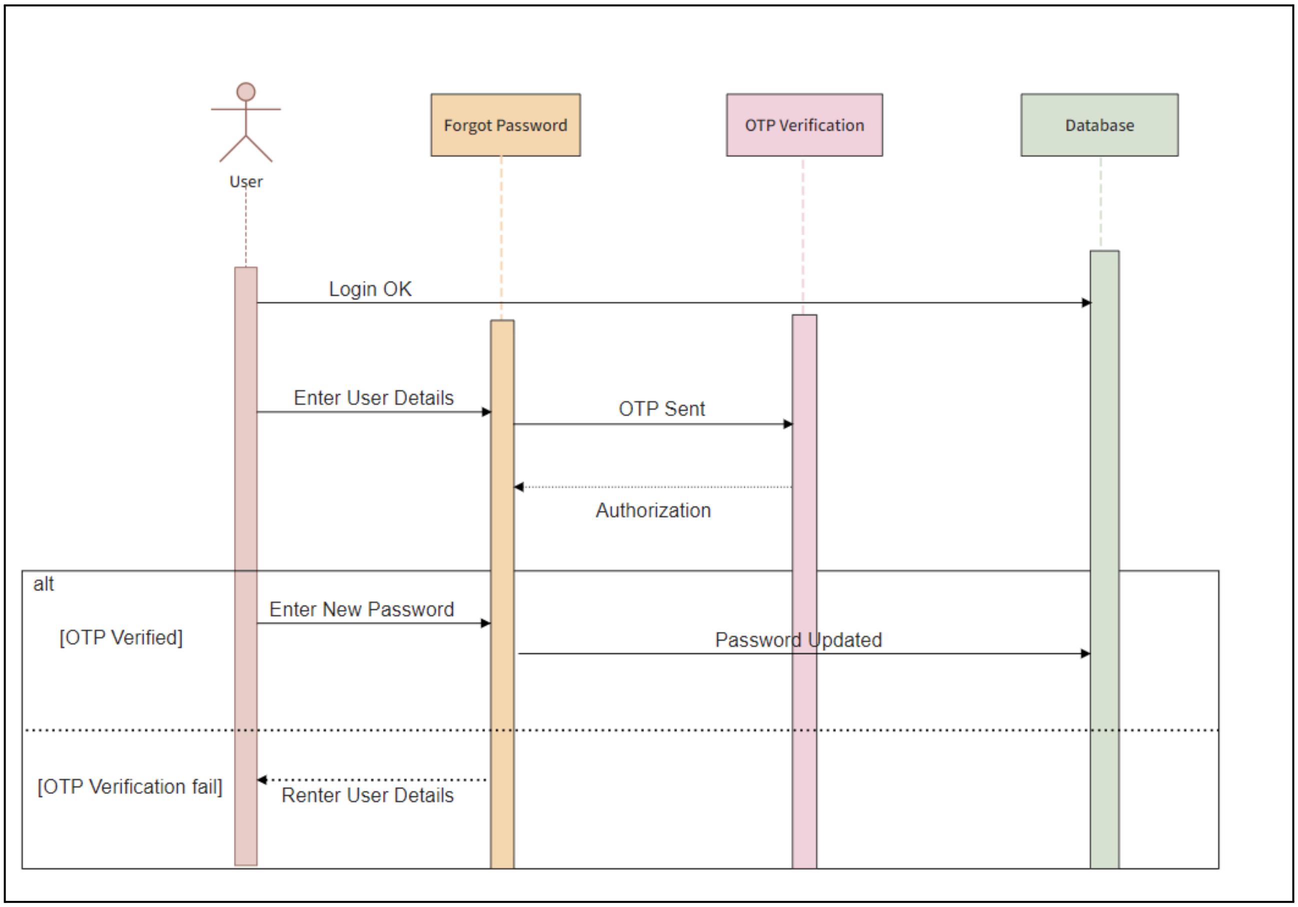
Login:



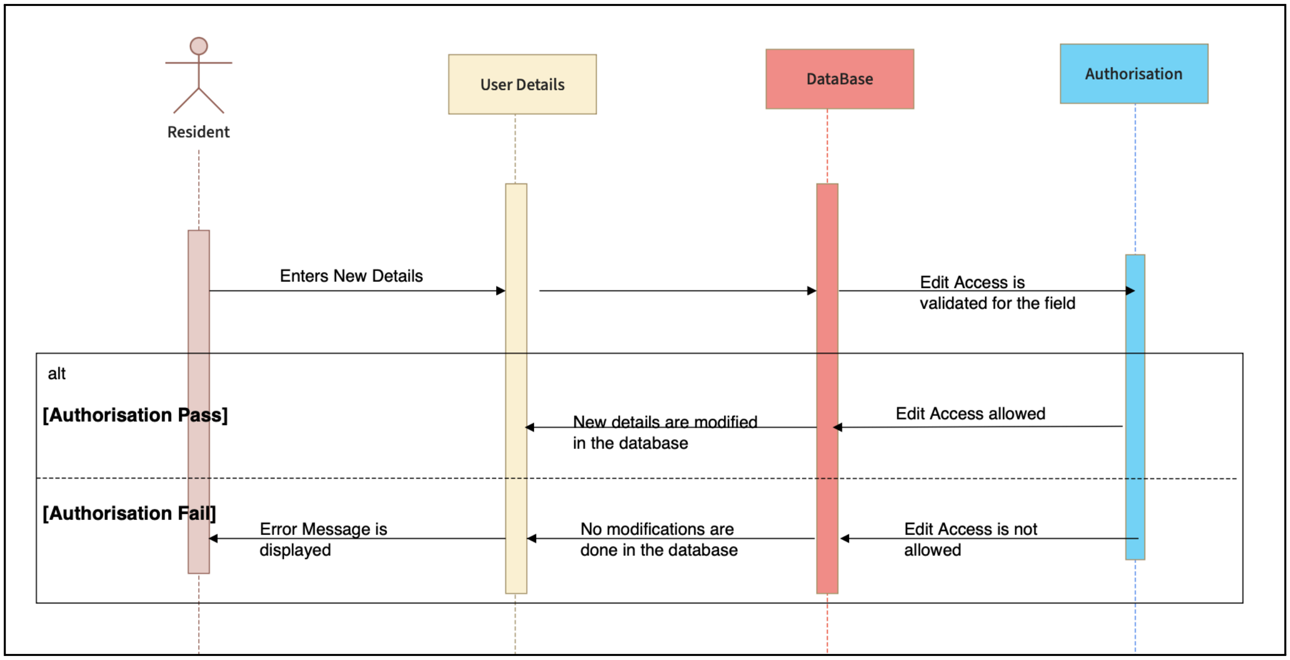
Sign Up:



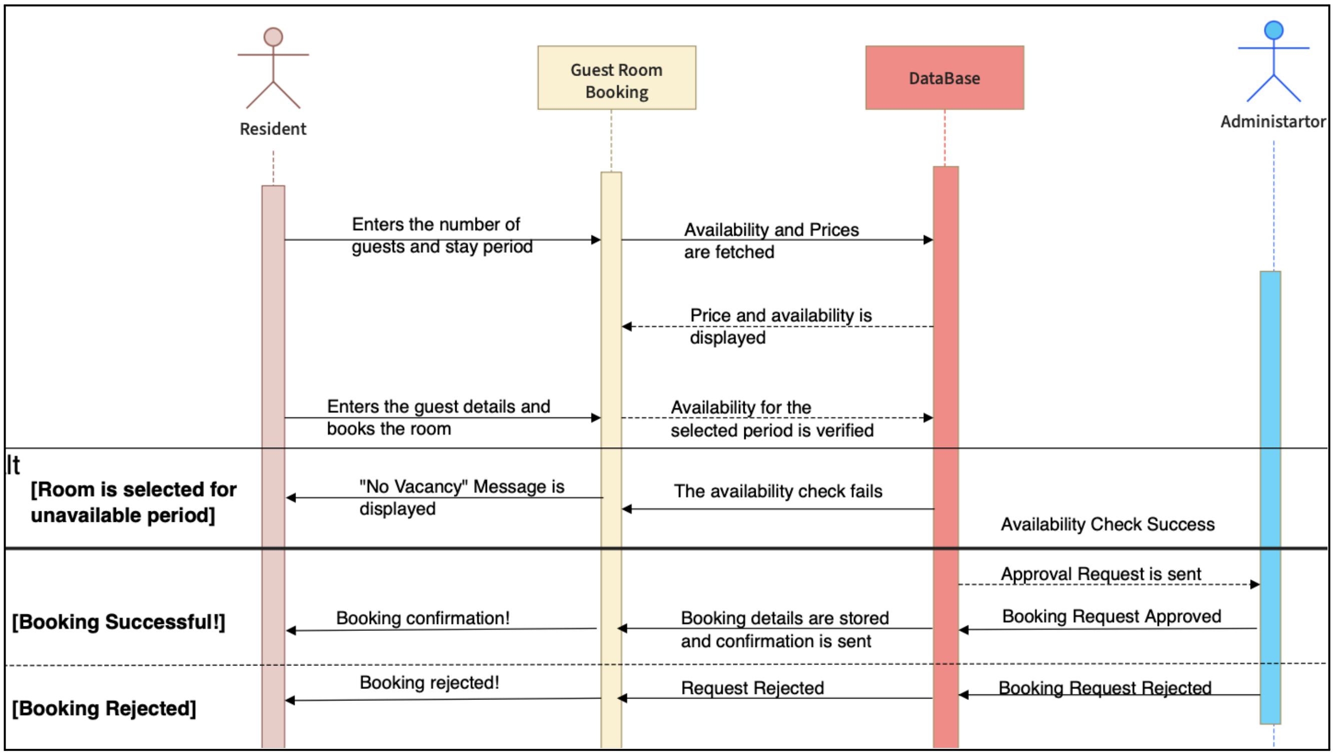
Forgot Password:



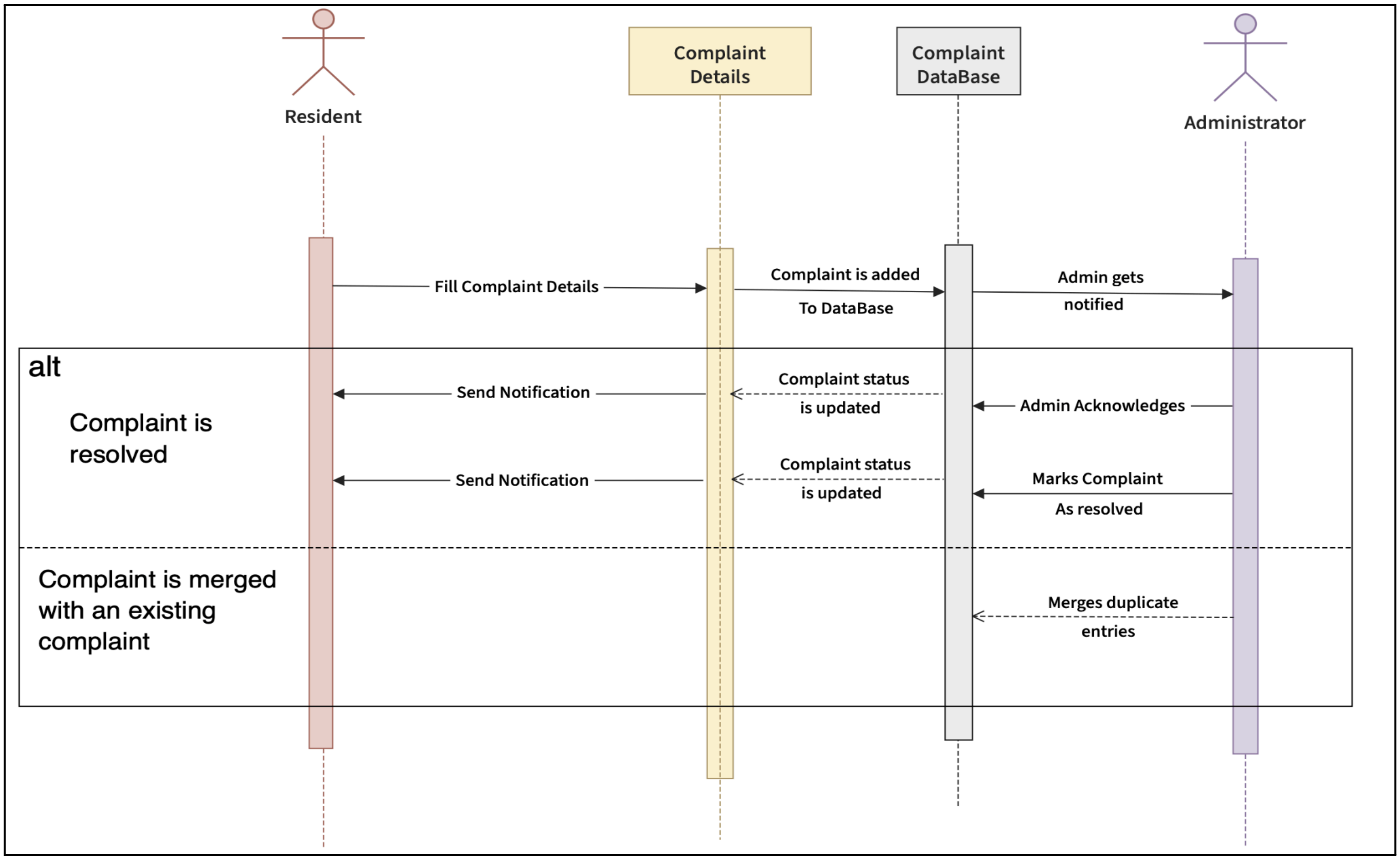
Profile Updation:


Resident
User Details
DataBase
Authorisation
Enters New Details
Edit Access is validated for the field
alt
[Authorisation Pass]
New details are modified in the database
Edit Access allowed
[Authorisation Fail]
Error Message is displayed
No modifications are done in the database
Edit Access is not allowed

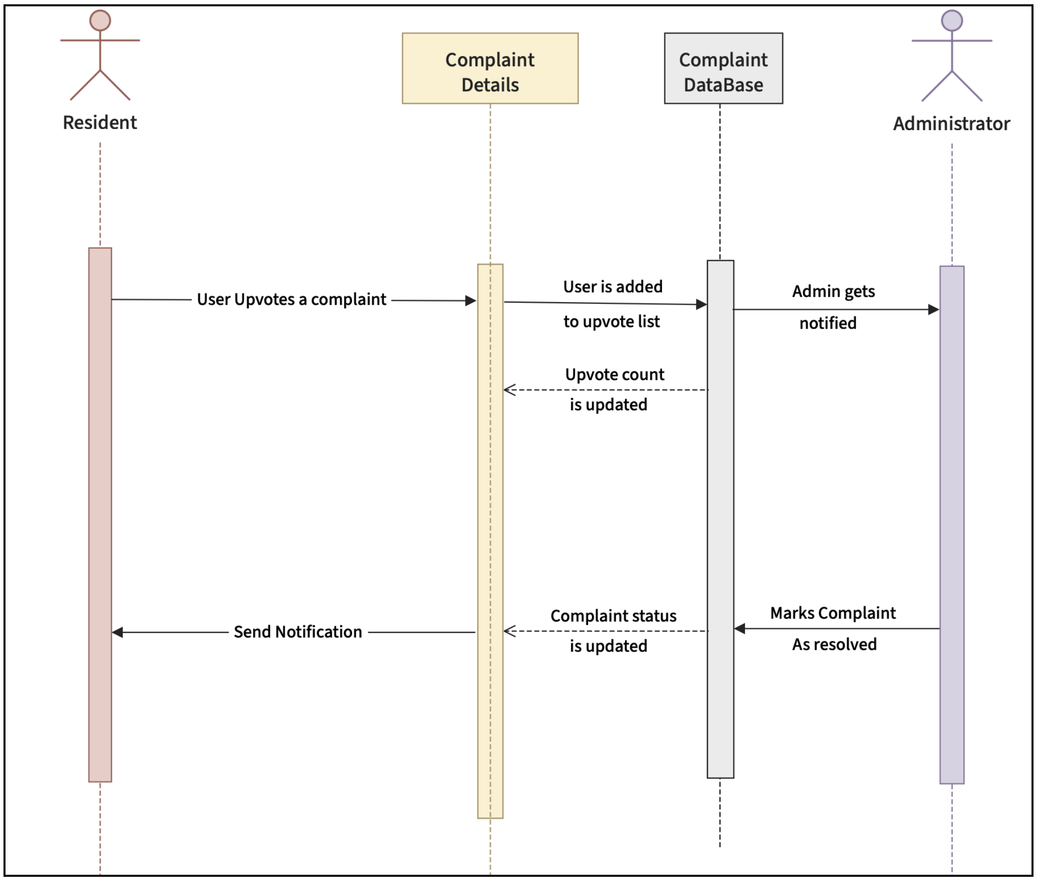
Guest Room Booking:


Resident
Guest Room Booking
DataBase
Administartor
Enters the number of guests and stay period
Availability and Prices are fetched
Price and availability is displayed
Enters the guest details and books the room
Availability for the selected period is verified
lt
[Room is selected for unavailable period]
"No Vacancy" Message is displayed
The availability check fails
Availability Check Success
Approval Request is sent
[Booking Successful!]
Booking confirmation!
Booking details are stored and confirmation is sent
Booking Request Approved
Booking rejected!
Request Rejected
Booking Request Rejected
[Booking Rejected]

Create New Complaint:

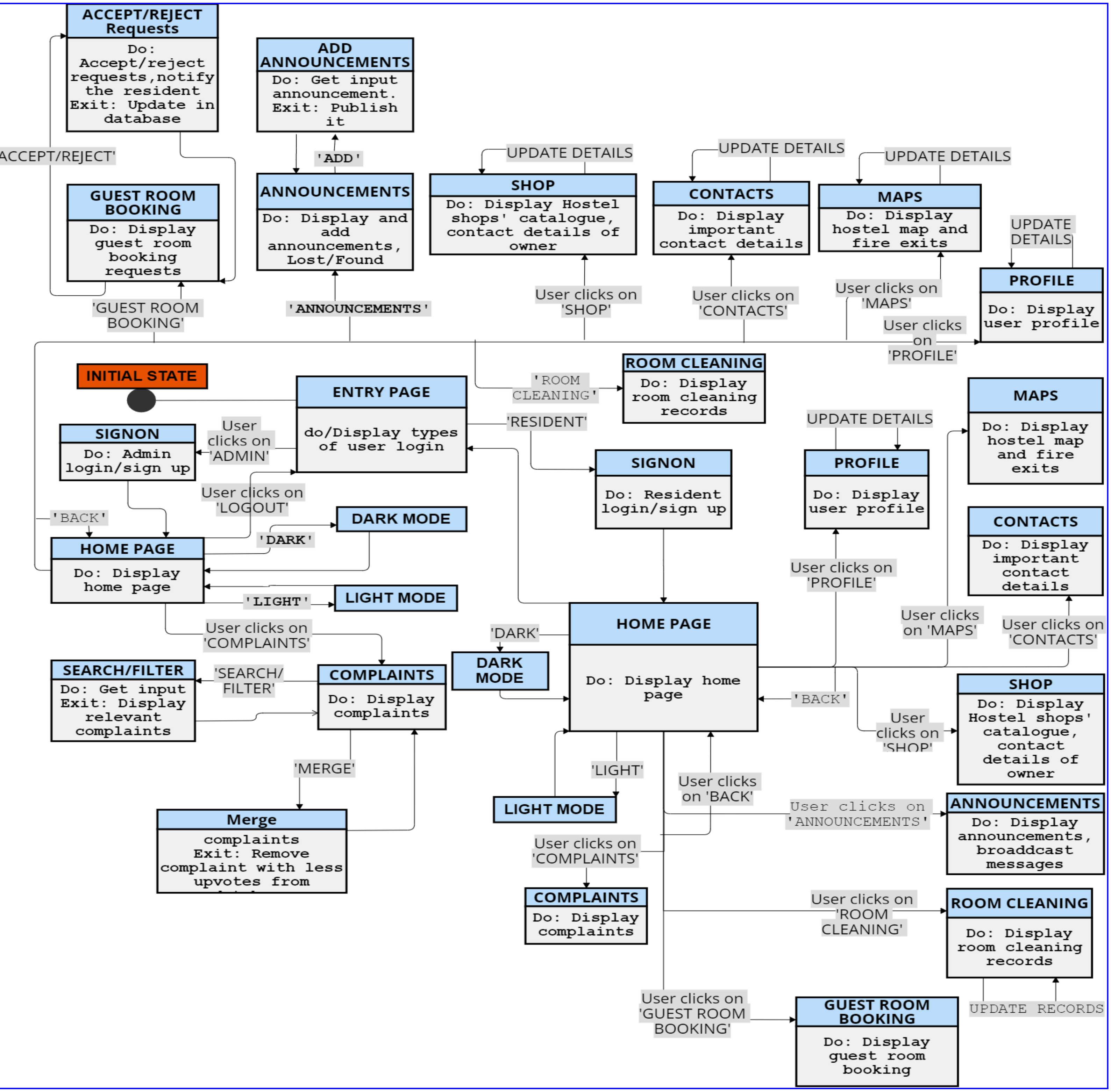


Upvote Complaint:

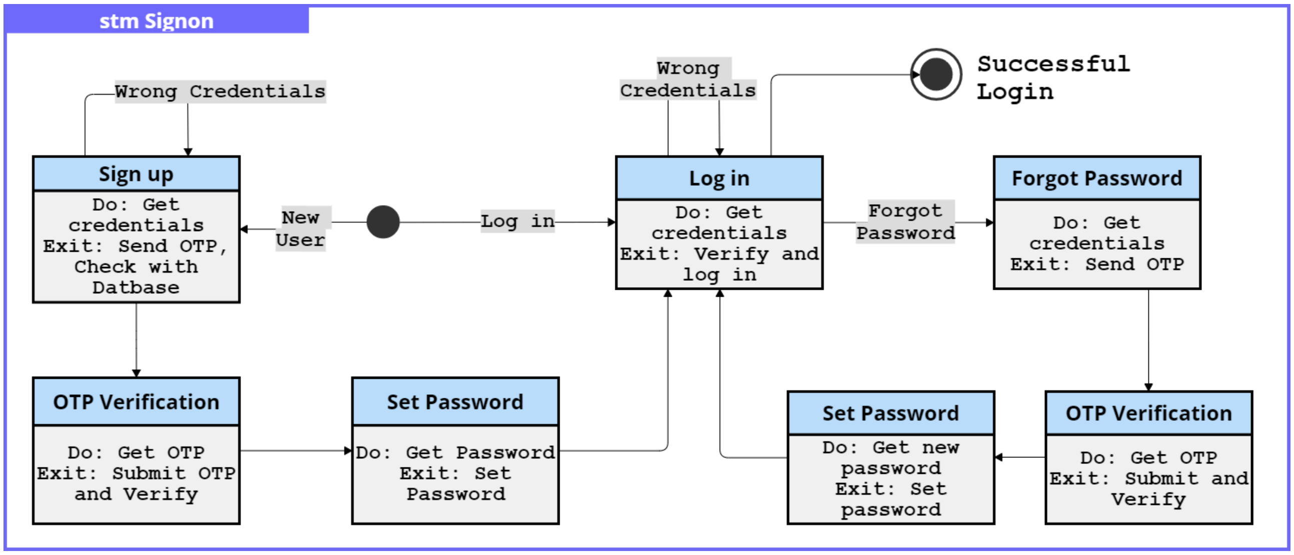


3.4 State Diagrams

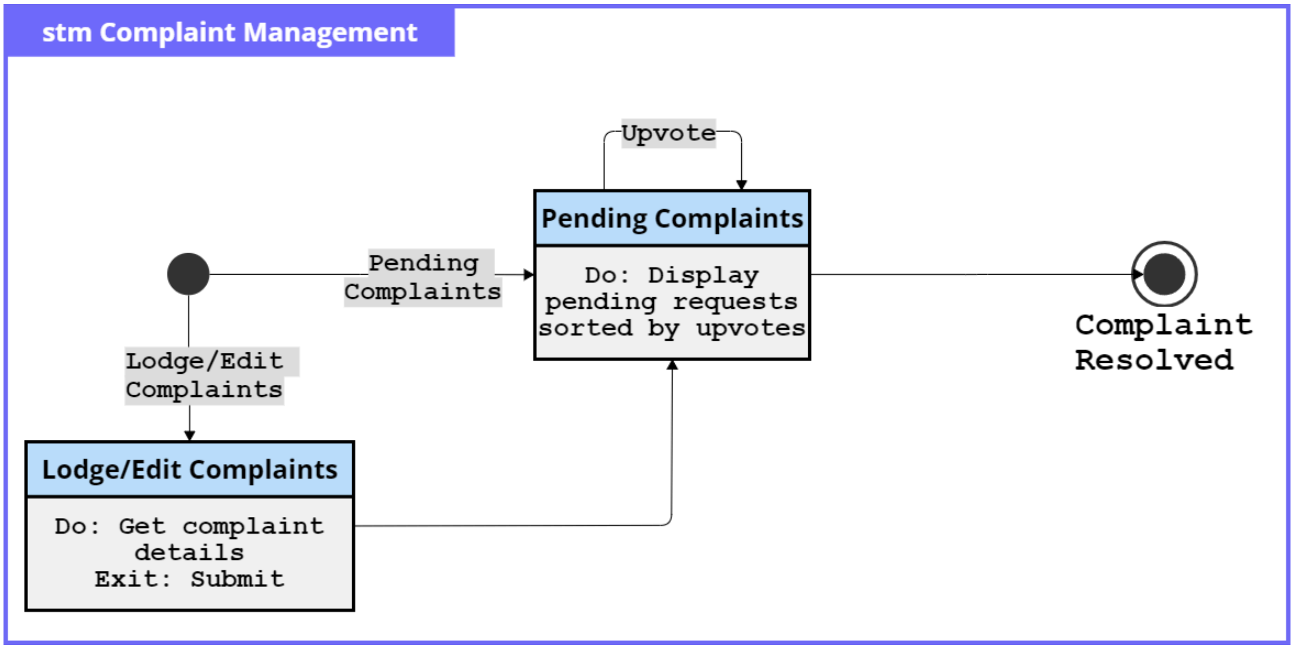
Overall state diagram:



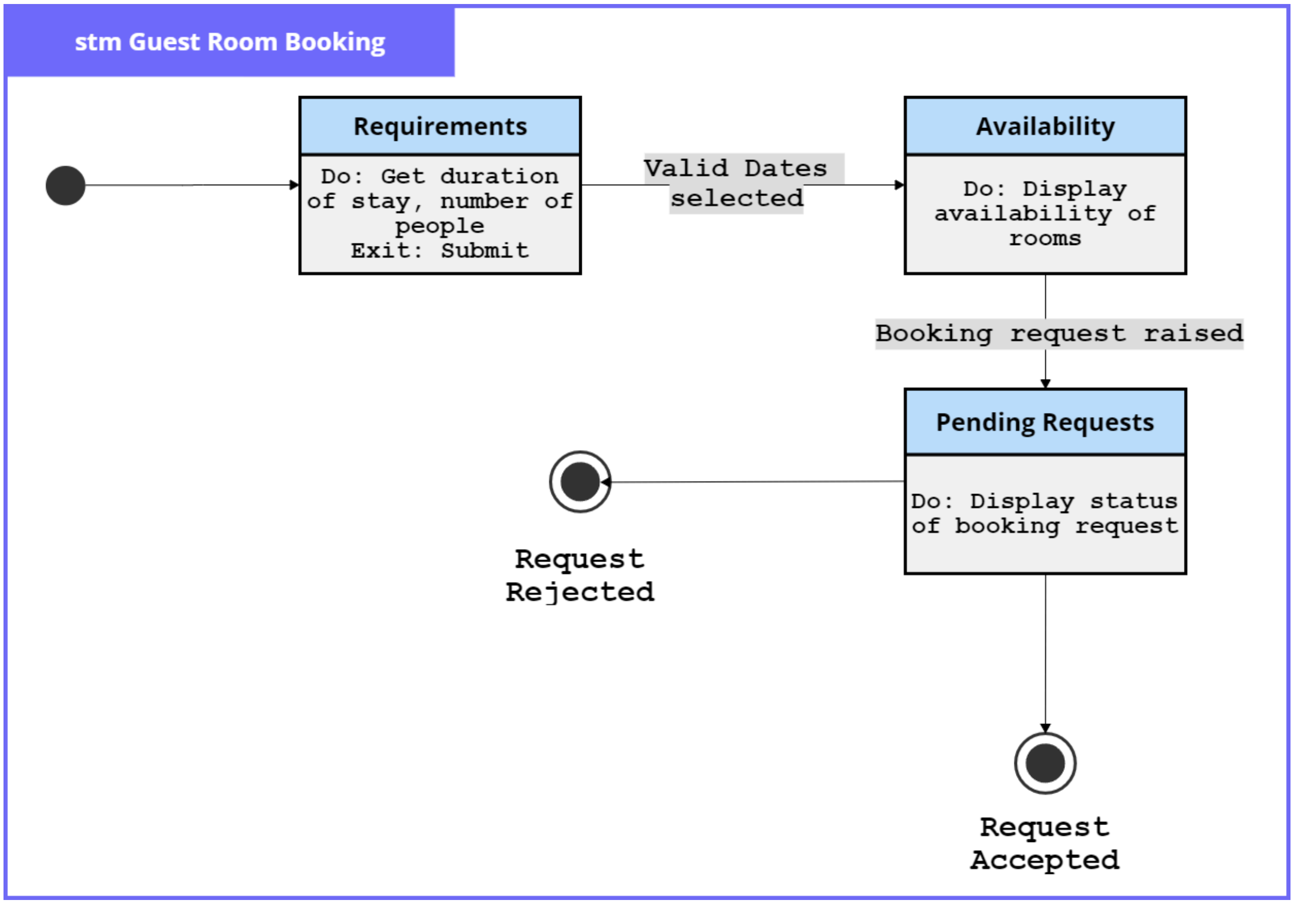
1. Signon



2. Complaint Management



3. Guest Room Booking



4. Project Plan

Timeline of development :

1. 12th January 2024 - 26th January 2024: Requirement generation,Analysis and Documentation of Software Requirements Specification (SRS).
2. 27th January 2024 - 9th February 2024: Documentation of Software Design Document
3. 9th February 2024 - 15th March 2024: Implementation, unit testing, integration testing
4. 15th March 2024 - 29th March 2024: System testing, manual preparation for beta testing
5. 29th March 2024 - 12th April 2024: : Code Improvement and Beta Testing:- With the results of the previous testing sessions, improvements on the software would be done to increase speed, security and its overall quality
6. 12th April 2024 - 19th April 2024: Addressing Beta Testing Feedback and Delivering the Final Project Report.

Broad classification of the tasks along with the team members working on it are :-

1. **Front End Design:** A front-end designer not only designs the look and layout of a product but also translates these designs into code. Any interactive element on a website that users can engage with falls under the category of front-end development.

The members responsible for the front-end design are:

- Ayush
- Ritesh Hans
- Mrdul Agarwaal
- Rohan Batra

The following software might be useful for Front End Development:-

a. **React JS**: A free and open-source front-end JavaScript library for building user interfaces based on UI components.

b. **JavaScript**: JavaScript, often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS.

c.**HTML**: A standard markup language for documents designed to be displayed in a web browser.

d.**CSS**: Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.

1. **Back End Design:** Anything that is not directly visible to the user and contains and processes the data behind the scene comes under the back-end. It focuses mainly on APIs,backend logic,databases etc.

The members responsible for the back-end development are:

- Apoorv Tandon
- Tanishq Maheshwari

- Taneshq Zendey
- Samarpan Verma
- Krutuparna Paranjape
- Mridul Gupta

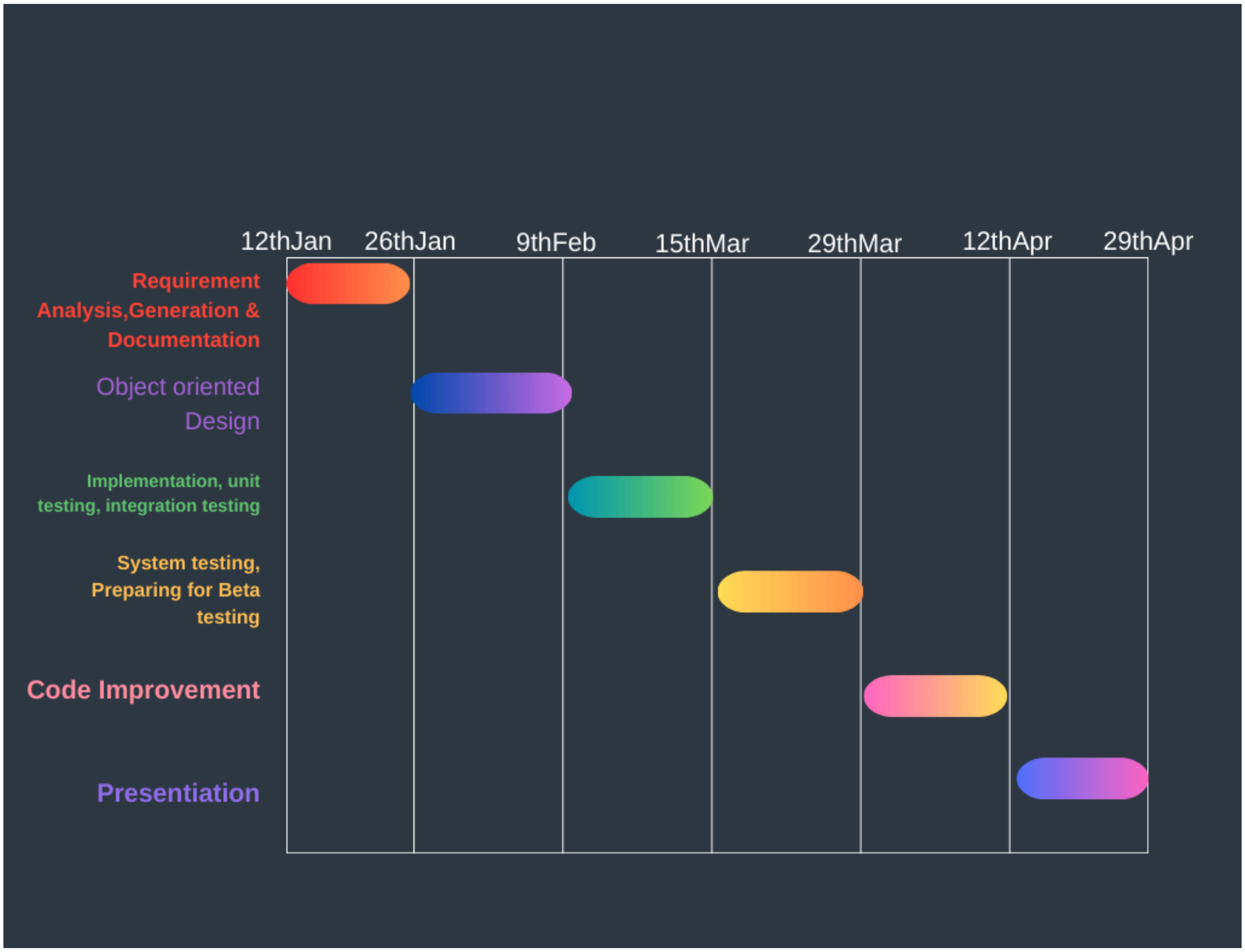
The following software might be useful for Back End Development:-

a.**JavaScript**: Can be used for both Front and Back End Development.

b. **MySQL**: MySQL is an open-source relational database management system.

c.**Node.JS:** Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser.

d. **MongoDB**: MongoDB is a free and open-source document-oriented database that is very much popular among web developers.



Appendix A - Group Log

<*Please include here all the minutes from your group meetings, your group activities, and any other relevant information that will assist in determining the effort put forth to produce this document*>

All the group members were in constant touch with each other and the TA through the WhatsApp group. Suggestions were given by the TA regarding some corrections and elaboration in the architecture diagram, class diagrams, and state diagram. All those suggestions have been seriously incorporated into the document. Work was divided and team-meetings were held regularly.

Date	Members Present	Topic Of Discussion
29/01/2024	All group members	Collaboratively discussed the appearance of user interfaces and the required features for each webpage.
02/02/2024	All group members	Distributed work on context, use case, class, sequence, and state diagrams.
04/02/2024	TA & All group members	Addressed and resolved TA's questions and suggestions regarding our software architecture, determining suitable classes for inclusion in the class diagram, and clarifying uncertainties related to the state diagram.
08/02/2024	All group members	Examined diagrams and interfaces to ensure consistency of features, functions, and attributes across the design document.