Software Requirements Specification

for

Online Guest House Booking System (OGHBS)

Version 1.0 approved

Prepared by:

Akshat Singh Rathore
Anindya Sikdar
Srijanak De

Indian Institute of Technology, Kharagpur

20th March 2021

Table of Contents

Revision History

Name	***Date***	***Reason For Changes***	***Version***
OGHBS-V1.0	***20/3/2021***		***V1.0***

1. Introduction

1.1 Purpose

The purpose of this document is to create a list of detailed requirements for the Online Guest House Booking System (OGHBS-V1.0). This document will capture interactions between different internal web pages, environment scenarios of usage, constraints, and different functionalities of online guest house booking.

Problem Statement

In IIT Kharagpur, a few guest houses are available. This Online Guest House Booking System (OGHBS) will aim to cover the following functionalities:

- ***Guest House Facilities***
  Any visitor who wants to visit the IIT campus can register and verify the identity of the user. (S)he also can check the room availability status for all the guest houses available on the campus. They will also get to know the price per room and the facilities available. Additionally, if the food will be available or not, and the cost for it also can be checked before booking the room.

- ***Booking Credentials***
  Once a room is booked, the remaining room details need to be updated automatically for that particular guest's house. The booking details such as the name of visitors, their address, age, gender, relations, duration of stay, also needs to be filled up. Accordingly, the guest house staff are notified.

- ***Waiting Queue***
  If there is no room available, the visitor can add him/her-self to the waiting queue. If there is any cancelation, the booking would be confirmed automatically. If the waiting queue is nil, the room availability status needs to be updated.

- ***Payment Details***
  On booking a room, the visitor needs to pay 20% of the actual amount for the booking. On cancelation of the room, a fraction of the money would be deducted and the remaining amount would be returned. On the other hand, if the booking does not get confirmed, deposited money would be returned.

- ***Feedback System***
  There must be an online feedback system also for the visitors. Before leaving the guest house, the visitor would be asked to fill up a form with experience details, which would be helpful to upgrade the service quality.

1.2 Document Conventions

The conventions used in the document are as per the IEEE standard provided for formatting the SRS document of a software.

1.3 Intended Audience and Reading Suggestions

This document will capture all the requirements as mentioned in the assignment, different conflicts, and their resolution. It could be potentially used for developing, designing, and testing phases of the project. Thus it will serve as a reference guide in different phases of system development.

The intended readers of this document are the developers of the software testers, guest house staff, and the guesthouse administration. These people are meant to understand the details of the software architecture modeled and used to understand the proper functioning of the software and their individual roles in the software also. Any suggested changes on the requirements listed on this document should be included in the last version of it so it can be a reference to developing and validating teams.

1.4 Product Scope

The scope of the online guest house booking system is to create an online web-interface for the users to check the availability and book the room from any part of the world. This system will be an alternative to the traditional offline way of checking available rooms and booking them. This will also give scope to the guesthouse staff to check bookings, cancel on-spot ones and make arrangements accordingly. The system will have three major types of users: visitors, students/professors, and the guesthouse administration. The bookings can only be made by the students and professors of the institute on behalf of their relatives (or by the administration in special cases).

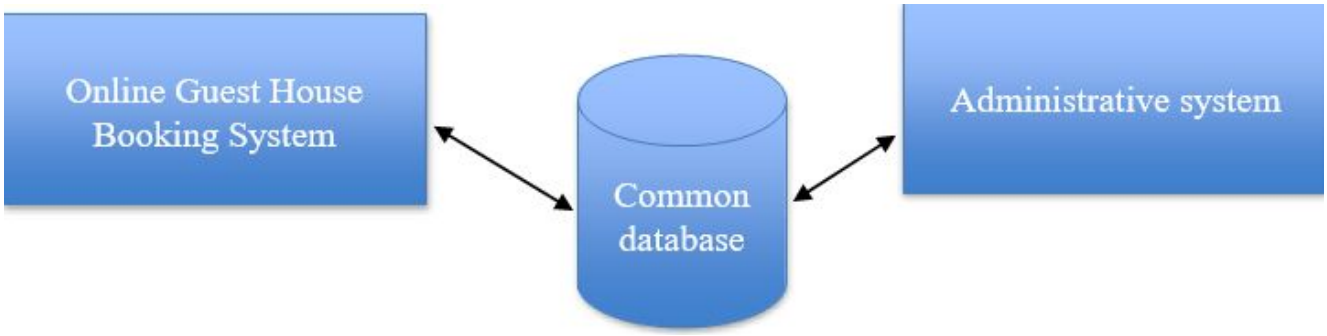
1.5 References

The following reference has been consulted to make a suitable and effective software for online guest house booking system.

• IEEE standard 830-1998 recommended practice for Software Requirements Specifications-Description

2. Overall Description

2.1 Product Perspective



The system considers visitors and guest house administrative staff perspectives. It has OGHBS – where visitors are able to book the rooms and use system functions described in this document, while for the administrative use there is a separate environment, where guest house staff can view all the information about the guest house, booking information and can also cancel a booking or in special cases book a room for visitors. Feedback from visitors can also be seen here. Both systems are based on a common database. They are integrated with each other, meaning the relevant data is exchanged.

2.2 Product Functions

The product provides the features for online guest house booking which can be used by the students and professors of IIT KGP and their relatives for ease of booking as well as by guesthouse staff for ease of administration.

Any user can freely navigate through the system home page and pages corresponding to each guest house. They can have a view of all the facilities available in each guest house.
Only students and professors can access booking options using their institute email ids as login credentials. They can view their personal details as well as live booking status and booking history. They can book rooms, cancel bookings, add themselves to the waiting list.
Only guest house staff can have access to the Admin Panel. All relevant guest house staff are notified when a booking is made.

Please refer to ***Appendix-A*** *for the Use Case Diagram.*

Use Cases

1. *Visitors use case:*
   - *View Guest House List*

   -Preconditions:
   None

-Postcondition:
A list of all the guest houses is shown.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
The user communicates with the system.

-Trigger:
User visits the website homepage.

-Main Success Scenario:
A list of all the guest houses is shown.

-Extensions/Variations:
None

- *View Guest House Details*

-Preconditions:
None

-Postcondition:
Details of the selected guest house are shown.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
The user communicates with the system.

-Trigger:
The user selects a particular guest house.

-Main Success Scenario:
Details of the selected guest house are shown.

-Extensions/Variations:
None

- *Query about room availability*

-Preconditions:
None

-Postcondition:
Based on the period selected all types of rooms along with availability status will be shown.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
The user communicates with the system.

-Trigger:
The user chooses the option to check the availability status of a particular guest house.

-Main Success Scenario:
Rooms are available for the specified period of duration.

-Extensions/Variations:
None

2. *Students'/Professors' use case:*
   - *User registration*

-Preconditions:
One shouldn't already have an existing account.

-Postcondition:
Users will be taken to the login page.

-Failure Situations:
1. Identity verification through institute email id authentication fails.
2. User already has an account

-Postcondition in case of failure:
1. A message asking users to enter a valid institute email id will be shown.
2. User will be asked to log in and redirected to the login page

-Actors:
Students and Professors of IIT Kharagpur

-Trigger:
The user chooses the option to register to the system.

-Main Success Scenario:
Users are registered into the system.

-Extensions/Variations:
None

- *Edit user details*

-Preconditions:
Users should be logged in.

-Postcondition:
Users' data will be saved according to modifications made.

-Failure Situations:
None

-Postcondition in case of failure:
None.

-Actors:
Students and Professors of IIT Kharagpur

-Trigger:
The user chooses the option to edit user details.

-Main Success Scenario:
Users' changed details are saved.

-Extensions/Variations:
None

- *View Booking History*

-Preconditions:
1. One should be logged in.

-Postcondition:
The user gets the list of all the past bookings along with their details.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Students and Professors of IIT Kharagpur

-Trigger:
The user chooses the option to view booking history.

-Main Success Scenario:
The user gets the list of all the past bookings along with their details.

-Extensions/Variations:
None

- *Cancel booking*

-Preconditions:
1. One should be logged in.
2. There should be at least one active booking.

-Postcondition:
Users will be refunded based on the payment policy.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Students and Professors of IIT Kharagpur.

-Trigger:
The user chooses the option to cancel the booking.

-Main Success Scenario:
Booking is canceled.

-Extensions/Variations:
None

- *Book a room*

-Preconditions:
1. Users should be logged in.

-Postcondition:
Users will be directed to the payment page after filling in relevant details.

-Failure Situations:
1. If the guest house is not functional.
2. If more than the available no of rooms is selected for booking

-Postcondition in case of failure:

1. A message will be shown to users telling them that the guest is not functional.
2. An error message will be shown to limit the no of selected rooms.

-Actors:

Students and Professors of IIT Kharagpur.

-Trigger:

The user chooses the option to book.

-Main Success Scenario:

Users are directed to the payment gateway.

-Extensions/Variations:

None

- *Make Payment*

-Preconditions:

1. Users should be logged in.
2. Users must fill in the relevant details for booking

-Postcondition:

The required amount is deducted as per the payment policy and the desired room in the desired guest house will get booked or the user will be added to the waiting queue based on room availability.

-Failure Situations:

Invalid payment details, weak internet connection.

-Postcondition in case of failure:

A payment failure message will be displayed.

-Actors:

Students and Professors of IIT Kharagpur.

-Trigger:

The user chooses the option to proceed to the payment gateway.

-Main Success Scenario:

1. The required amount is deducted as per the payment policy.
2. The desired room in the desired guest house will get booked or the user will be added to the waiting queue based on room availability.

-Extensions/Variations:

1. The desired room is booked if the desired no of that room is available.

2. The user is added to the waiting queue if the no of rooms desired is more than the available no of rooms.

- *Give feedback*

-Preconditions:

1. One should be logged in.

2. There must be at least one check-outs prior to the date of login.

-Postcondition:

A thank you message will be shown to the users.

-Failure Situations:

None

-Postcondition in case of failure:

None

-Actors:

Students and Professors of IIT Kharagpur.

-Trigger:

1. Users choose the option to give feedback on their booking history.

2. A feedback form is automatically displayed once the user logs in.

-Main Success Scenario:

Feedback is stored in the system.

-Extensions/Variations:

None

3. *Admins use case:*

- *Superuser login*

-Preconditions:

One should be registered as an admin.

-Postcondition:

The admin gets access to the admin panel.

-Failure Situations:

None

-Postcondition in case of failure:

None

-Actors:
Admin communicates with the system.

-Trigger:
Admin chooses the option to log in.

-Main Success Scenario:
The admin gets access to the admin panel.

-Extensions/Variations:
None

- *View Guest House Details*

-Preconditions:
One should be logged in as admin.

-Postcondition:
Guest House Details of the desired Guesthouse are shown.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Admin communicates with the system.

-Trigger:
Admin selects a Guest House to view its details.

-Main Success Scenario:
Guest House Details of the desired Guest House are shown.

-Extensions/Variations:
None

- *View Bookings*

-Preconditions:
One should be logged in as admin.

-Postcondition:
All the information regarding bookings is shown.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Admin communicates with the system.

-Trigger:
Admin selects the Booking Details tab.

-Main Success Scenario:
All the information regarding bookings is shown.

-Extensions/Variations:
None

- *Change Bookings*

-Preconditions:
One should be logged in as admin.

-Postcondition:
Modified booking will be saved and room availability status will be changed accordingly.

-Failure Situations:
1. Increasing/Changing no/type of rooms won't be possible if all rooms of the desired type are booked.

-Postcondition in case of failure:
1. An error message will be displayed showing all rooms of the desired type are booked.

-Actors:
Admin communicates with the system.

-Trigger:
Admin chooses the option to edit booking details.

-Main Success Scenario:
Modified booking will be saved and room availability status will be changed accordingly.

-Extensions/Variations:
None

- *Edit Guest House Details*

-Preconditions:
One should be logged in as admin.

-Postcondition:
Modified Guest House details will be saved.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Admin communicates with the system.

-Trigger:
Admin chooses the option to edit Guest House details.

-Main Success Scenario:
Modified Guest House details will be saved.

-Extensions/Variations:
None

- *View Feedbacks*

-Preconditions:
One should be logged in as admin.

-Postcondition:
The list of Feedbacks is displayed.

-Failure Situations:
None

-Postcondition in case of failure:
None

-Actors:
Admin communicates with the system.

-Trigger:
Admin selects the Feedback option.

-Main Success Scenario:
The list of Feedbacks is displayed.

-Extensions/Variations:
None

2.3 User Classes and Characteristics

User characteristics
There are three user classes in OGHBS: visitors(or guests), students/professors, and guesthouse administration.

Visitors
Visitors(or guests) are the users who can view different guesthouse details like facilities available, rental costs, food availability, and also check the room availability in a particular guest house according to the desired duration of stay. However, this class cannot make any bookings which they must book through the following class.

Students and Professors
This class of users can do all that a visitor can and in addition they can perform bookings for their relatives who are visiting the campus. Also, they can view their booking history and give feedback about the quality of the stay of their relatives, and rate different guesthouses. They can also cancel bookings.

Guesthouse Administration
This class represents the administrative authority of all the guesthouses. It manages the guesthouses, their bookings and can view the feedback submitted by the previous class. In special cases, this class can also add the booking to any guest house depending on the availability of the rooms.

Class Attributes:

- *Guesthouse Class Attributes:*

  id: int {{unique}}
  name: String
  food_availabilty: bool
  food_cost: float {{Optional}}
  address: String

- *Rooms Class Attributes:*

  id: int unique
  is_ac: bool
  number_of_rooms: int
  type_of_room: RoomTypes
  rental_cost: float
  starting_room_index: int
  booked_rooms: String

- *Booking Class Attributes:*

  id: int {{unique}}
  room_type: String
  room_id: int
  /booking_time: int
  visitors:_count: int
  names_of_visitor: String
  food: bool
  checkin_date: Date
  checkout_date: Date
  booking_status: BookingStatus
  payment_status: bool
  paid_amount: int
  refund_amount: int {{Optional}}

- *Feedback Class Attributes:*

  id: int unique
  comfort_of_stay: rating
  room_cleanliness: rating
  service_quality: rating
  additional_feedback: String
  stars: rating

- *User Class Attributes:*

  id: int unique
  username: String
  name: String
  institute_email_id:String {{unique}}
  password:String
  phone_number:int
  is_active:bool

- *Student Class Attributes:*

  id: int {{ unique }}
  department: String
  roll_number: String {{ Unique }}

- *Professor Class Attributes:*

  id: int {{ unique }}
  department: String
  address: String

- *Admin Class Attributes:*

  username: String
  user_email: String
  unique password: String

- *Views or Functions:*

  1. UserRegister (Interface: REGISTER)
  Stores user details and Registers the user.

  2. SendEmail (Interface: REGISTER)
  Sends verification email for user verification.

  3. EmailVerification (Interface: REGISTER)
  Verifies the user (student or professor of IIT KGP)

  4. UserLogin (Interface: Login)
  Logs in the user if the user is registered.

  5. Logout (Interface: Login)
  Logs out the user.

  6. GerUserDetails (Interface: View User Details)

  Gets the personal information of the user (logged in)

  7. GetBookingHistory (Interface: View User Details)
  Gets the booking history of the user (logged in)

  8. GuesthousesList (Interface: Guesthouse LIST)
  Returns the list of Guesthouses

  9. GuesthousesDetails (Interface: VIEW GUEST HOUSE DETAILS)
  Returns the details of a selected Guesthouse

10. Search (Interface: CHECK AVAILABILITY)

Searches for availability of all types rooms of a guest house

11. CheckAvailability (Interface: CHECK AVAILABILITY)
Searches for availability of a particular type of room of a guest house

12. ClearQueue (Interface: CHECK AVAILABILITY)
Sets the booking status of a queued booking to confirmed booking if the desired room in the desired guest house is available.

13. BookRoom (Interface: BOOK)
View to book rooms. It validates and stores the relevant data of the customer.

14. Payment (Interface: PAY)
It stores and processes the payment of the customer via a third party service. After Payment is confirmed , the booking is either confirmed or is added to queue.

15. ViewFeedback(Interface : ADMIN VIEW)
Views feedback corresponding to each booking

16. GetGuesthouseDetails(Interface : ADMIN VIEW)
Returns the details of guest houses for the query

17. GetBookingDetails(Interface : ADMIN VIEW)
Returns the details of all the bookings

18. AddBooking(Interface : ADMIN VIEW)
Adds booking in special cases if required

19. EditBooking(Interface : ADMIN VIEW)
Allows admin to edit their booking already made

20. CancelBooking(Interface : ADMIN VIEW)
Allows admin to cancel existing bookings

21. SetGuesthouseDetails(Interface : ADMIN VIEW)
Allows admin to set the guest house details like number of rooms available, type of rooms, food availability etc.

2.4 Operating Environment

The software is designed to run in the following environments :

1. Windows 7 or later
2. Ubuntu 18.04.03 (LTS version)

Any OS with the support of :
1. Python (3 or above)
2. Django (3.1 or above)
3. mysqlclient
4. XAMPP
5. phpMyAdmin

2.5 Design and Implementation Constraints

1. The system has a dependency on the database and the guesthouse's existing traditional system.
2. Internet connection is a constraint for this system

3. Database should be synchronized once every day so that the routine checkout and cancellations are properly handled and tracked.

2.6 User Documentation

User Manual:

- ***User Navigation:***

  Any user can freely navigate through the system home page and pages corresponding to each guest house. In order to book, users must register/log in. Once, logged in, the user can cancel his booking, or navigate to the system home page to book the rooms from the guest house of their choice. Users can also change their personal information and see their booking status and booking history.

- ***User Login/Register***

  Students and professors should register using their institute email ids. Registering with any other email id is not allowed for verification and security purposes. If the user has no account(s) s/he should register.

- ***User Home Page***

  After logging in, the user can see his details, past booking history, live booking status, and navigation to the system home page. A corresponding cancel booking option will be available if the booking status is live. On clicking cancel booking, the user will be asked to confirm. On confirmation, the booking will be canceled and the required amount will be refunded. If the user didn't fill in feedback corresponding to a checkout an option for filling in the corresponding feedback will also be there.

- ***Booking Page***

  This will be an interactive page. Once the user selects a guest house, (s)he will first have to enter the duration of stay. Correspondingly types of rooms in the selected guest houses along with availability status will be shown. On selecting the room type a booking form will be opened

  where the duration of stay and type of room will be auto-filled. Food, number of rooms, number of guests, and name of each guest will have to be filled in. Finally, the user will proceed to the payment gateway. After successful payment, the booking either gets confirmed or is added to the waiting queue based on availability.

- ***Feedback Page(Optional)***

  This will consist of a form asking for various information from the user like the comfort of stay, cleanliness of rooms, quality and behavior of service staff, food description, and additional feedback and suggestions. If the user didn't fill in feedback corresponding to his last checkout the corresponding feedback form will automatically pop-up as a reminder once the user logs in.

- ***Payment Gateway***

  This will give users access to a secured payment channel.

2.7 Assumptions and Dependencies

1. Login identity is verified by institute email-id i.e. emails ending with @iitkgp.ac.in. Hence, this system of verification allows booking through only two types of users, namely, professors and students of the institute.

2. The algorithm of allocating rooms to waiting list users based on room availability is assumed to be first-come-first-served, with the exception that if the number of accommodable guests in available rooms is less than the number of guests for whom the booking is done in the first position of the waiting list, then we go down the list and allocate the rooms to those accommodable.
(For example: Let's assume booking is done for 4 guests in the first position of the waiting list, and booking is done for 2 guests in the second position of the waiting list. Now, if 1 room capable of accommodating 2 guests only is available, then the guests at position 2 of the waiting list are first allocated the room before those at position 1.)

3. Number of people allowed in a room is not more than the given capacity of the room.

3. External Interface Requirements

3.1 User Interfaces

- ***System Home Page***

  This page will show images of different guest houses available in the institute. Each guest house will have an option associated with it to view more details about it. There will also be a login/register option. If the user is logged in and feedback for any previous visit is not yet submitted, a reminder for the same will be displayed.

- ***Guest House Page***

  This page will show facilities available in the guest house with an image gallery pertaining to each guest house. This will also show different types of rooms, prices of rooms, food availability status, and also has a check availability option.

- ***Login Page***

  Login will be done using the institute email address and password.

- ***Register Page***

  This page asks for user details like name, address, gender, email, department, and mobile number. As a method for identity verification, it will just check if the user has the institute email id ending with @iitkgp.ac.in. The user can register as a professor or a student.

- ***User Home Page***

  After logging in, the user can see his details, past booking history, live booking status and navigate to the system home page. A corresponding cancel booking option will be available if the booking status is live. On clicking cancel booking, the user will be asked to confirm. On confirmation, the booking will be canceled and the required amount will be refunded. If the user didn't fill in feedback corresponding to a checkout an option for filling in the corresponding feedback will also be there.

- ***Booking Page***

  This will be an interactive page. Once the user selects a guest house, (s)he will first hathe ve to enter the duration of stay. Correspondingly types of rooms in selected guesthouses along with availability status will be shown. On selecting the room type a booking form will be opened where the duration of stay and type of room will be auto-filled. Food, number of rooms, number of guests, and name and age of each guest will have to be filled in. If a room is not available then the user will be added to the waiting list. Finally, the user will proceed to the payment gateway.

- ***Feedback Page(Optional)***

  This will consist of a form asking for various information from the user like the comfort of stay, cleanliness of rooms, quality and behavior of service staff, food description, and additional feedback and suggestions. The user may also submit ratings for their visited guest house.

- ***Payment Gateway***

  This will give users access to a secure payment channel.

- ***Admin Panel***

  This interface is available only to the guest house administration. Details for all prior bookings and notifications for recent bookings are shown here in a tabular format. New bookings can be made and existing bookings can be canceled at the discretion of the administration. Actions like changing the number of rooms available in any guest house, altering facilities like food and air conditioning, and changing room rentals and food costs are also available.

3.2 Hardware Interfaces

1. A laptop or a desktop is preferable where the local server could be run to use the application.
2. Keyboard and mouse to interact with the GUI are required.

3.3 Software Interfaces

1. Back end: Built using Django and MySQL (for DBMS)
2. Front end: Using Javascript, HTML, CSS

3.4 Communications Interfaces

- *The user must have an active email id with the domain iitkgp.ac.in or kgpian.iitkgp.ac.in for verification purposes.*
- *The admin uses an active email id of the google suite. The following web browser should be available for rendering the HTML pages: Google Chrome (57 or above), Mozilla Firefox(65 or above), or Internet Explorer(79 or above).*
- *HTTP communication standard is used for communication between client and server.*
- *PKDF2 algorithm with a SHA256 hash is used for password encryption.*
- *The databases can be kept in a separate location other than the source computer itself to make it more space-efficient. It can be done by using TCP/IP protocol for network access. It is not mandatory for this specific design but can be taken up as a feature that can be applied to make the software more realistic and user-friendly. This will not only make the software more*

space-efficient but at the same time maintain data security since we will be using a secured channel for data communication.

4. System Features

4.1 Viewing Guest Houses and their details

4.1.1 Description and Priority

The homepage for a visitor sGuesthouse show all the Guesthouse names. On clicking on

any Guesthouse, more details can be viewed about it.

Priority: High

4.1.2 Stimulus/Response Sequences

- *The user can view guest houses by visiting the home page.*
- *The user can view the details of a guest house by selecting the guest house from the guest house list.*

4.2 Registration of Users

4.2.1 Description and Priority

Students and Professors can register with their unique institute email ids in order to perform bookings for rooms and their payment.

Priority: High

4.2.2 Stimulus/Response Sequences

- *The user clicks on the register button and fills in the user details.*
- *A verification mail is sent to the user's email address. After verification, the user is registered in the system and redirected to the home page.*

4.3 Checking Room Availability

4.3.1 Description and Priority

On entering the intended duration of stay for any room type, the number of vacancies available in the guesthouse can be viewed.

Priority: High

4.3.2 Stimulus/Response Sequences

- *The user selects a guest house from the guest house list and the details of the guest house are shown.*
- *The user selects check availability and fills in the check-in date and checkout date.*
- *Based on the check-in date and check-out date all the rooms of the guest house along with their availability status are shown.*

4.4 Booking Room

4.4.1 Description and Priority

A registered user can book his/her choice of room after getting a vacancy in any guest house for the required duration of stay.

Priority: High

4.4.2 Stimulus/Response Sequences

- *The user logs in and checks the availability status of the desired guest house and selects a type of room of his/her choice.*
- *The user clicks on Book Now and fills in the necessary booking details on the booking page.*
- *The user then proceeds to the payment portal by clicking on Pay.*
- *Based on the availability of the room the booking is either confirmed or it's added to the waiting queue.*

4.5 Waiting Queues

4.5.1 Description and Priority

If the user does not get any vacancy for his/her choice of room in the guest house of choice, the user can book a seat in the waiting queue and wait for the check-in date. If the booking does not get confirmed within time, the entire amount paid sGuesthouse get refunded to the user.

Priority: Medium

4.5.2 Stimulus/Response Sequences

- *The user has to login first.*
- *The user checks the availability status of the desired guest house and selects a type of room of his/her choice.*

- *The user clicks on Book Now and fills in the necessary booking details on the booking page.*
- *The user then proceeds to the payment portal by clicking on Pay.*
- *If the room is not available the booking is added to the waiting queue. The user can view the status of the booking in the user dashboard.*

4.6 Cancellation of Bookings

4.6.1 Description and Priority

Bookings can be canceled at any point in time whether confirmed or not. For confirmed bookings, half the paid amount sGuesthouse get deducted as a fine, whereas for waiting for queue

bookings, the user gets back the entire amount as a refund.

Priority: Medium

4.6.2 Stimulus/Response Sequences

- *The user logs in and goes to the user dashboard. There the user can view all the past booking history and its current status.*
- *The user can cancel an active booking by clicking on the cancel booking button. The user is asked for a confirmation and if the user confirms the booking is canceled. In case the booking was confirmed 50% of the amount paid will be refunded and in case of queued booking total paid amount will be refunded.*

4.7 Feedbacks

4.7.1 Description and Priority

After checking out the user is requested to fill out a feedback form and give ratings to the guest house according to his/her experience.

Priority: Medium

4.7.2 Stimulus/Response Sequences

- *The user logs in.*
- *There must be at least one check-outs prior to the date of login.*

- *Users choose the option to give feedback on their booking history.*
- *A feedback form is automatically displayed once the user logs in.*
- *A thank you message is displayed after the form is filled.*
- *The feedback gets stored in the database.*

4.8 View Booking History

4.8.1 Description and Priority

After login, the user can see his/her entire booking history showing the room details, facilities opted, paid amount, refunded amount, booking status, and the feedback given.

Priority: Medium

4.8.2 Stimulus/Response Sequences

- *The user logs in with his own credentials.*
- *The user chooses the option to view the booking history.*
- *On selecting it, the user gets the list of all the past bookings along with their details.*

4.9 Booking Details(Admin)

4.9.1 Description and Priority

The admin can view all the details of the current and previous bookings of all the guest houses from the admin panel.

Priority: High

4.9.2 Stimulus/Response Sequences

- *The admin logs in with the superuser credentials.*
- *He/she gets an option to view all the booking details ever made to any guest house.*

4.10 Add Bookings(Admin)

4.10.1 Description and Priority

If required, the admin can add the booking of visitors directly from the admin panel.

Priority: Low

4.10.2 Stimulus/Response Sequences

- *The superuser logs in with the superuser credentials.*

- *Under the Bookings class, the admin gets an option to add new bookings to the database.*
- *On choosing this option, the admin can create a new instance of booking and configure its booking status accordingly.*

4.11 Dynamic Change of Rental Costs(Admin)

4.11.1 Description and Priority

The admin can change the rental costs for any type of room for any guest house at any time without any constraint. Similarly, the cost of food facilities can also be changed.

Priority: Medium

4.11.2 Stimulus/Response Sequences

- *The administration logs in with the superuser credentials.*
- *Under booking details, he/she gets the field rental cost and food cost for the classRooms.*
- *Here the admin can directly change the value of the rentals and food cost.*
- *After changing and saving the new cost, all future bookings follow this new cost system, and accordingly, the guest house details on the home page change.*

4.12 Payment and Refunding

4.12.1 Description and Priority

The payment can be done by the user after confirming the booking details and the amount to be paid is calculated and displayed by the system. A secure payment gateway is used to accept the payment through credit cards, debit cards, net banking, UPI, cGuesthousean, and PayPal. In case the booking is canceled, some amount is refunded based on certain conditions (mentioned in 4.6 Cancellation of Booking).

Priority: High

4.12.2 Stimulus/Response Sequences

- *Registered User can log in and after filling all booking details, he/she gets the option for payment.*
- *After successfully completing the payment, the booking status for the booking changes to 0 if the booking is confirmed and 1 if the booking is added to the waiting queue.*
- *Now, the user can cancel his booking at any point by clicking on the cancel button present in the booking details section.*

- *The appropriate amount is refunded after deduction of the preset fine and the booking gets canceled. Now the booking status for the booking changes to 2 or 3 according to the refunded amount(2 in case of half refunding and 3 for full refunding),*

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Serial No.	*Description*	*Priority*
1	*Every booking submission & modification should be updated in Hotel's common database within 10 seconds after each submission & modification activity*	*High*
2	*Results for cross-checking of availability of rooms in internal database and customer's choice sGuesthouse be in 5 seconds*	*High*
3	*Webpage UI load time should be within 2 secs*	*High*
4	*Redirection page load time sGuesthouse be within 2 secs*	*High*
5	*A confirmation email sGuesthouse be sent to the customer's mentioned email within 2 secs after confirmation page termination*	*Low*

5.2 Safety Requirements

- *A separate copy of the database needs to be maintained locally so that a backup of the dynamic data is available all the time if required. In case the software breaks down due to a glitch all the data should not be lost.*
- *The amount of booking made in a day is kept into account so that if it crosses a threshold value, a new email id from the google suite is used for email id verification in order to prevent server crashing due to SMTP errors.*

5.3 Security Requirements

Serial No	*Description*	*Priority*
1	*Customer's personal details sGuesthouse be encrypted*	*High*
2	*The customer's credit card details sGuesthouse be deleted from the database after the checkout date*	*High*

5.4 Software Quality Attributes

5.4.1 Reliability Requirements

Serial No.	*Description*	*Priority*
1	*The database should be synchronized once every day so that the routine checkout and cancellations are properly handled*	*High*
2	*The system sGuesthouse be accessible 99% of the time*	*High*

5.4.2 Portability requirements

Serial No.	*Description*	*Priority*
1	*The system sGuesthouse be accessible on Google Chrome, Firefox, Safari, Opera, and Internet explorer*	*High*
2	*The system sGuesthouse be accessible on devices with proper hardware interfaces with any screen size, i.e., the web pages should be responsive*	*High*

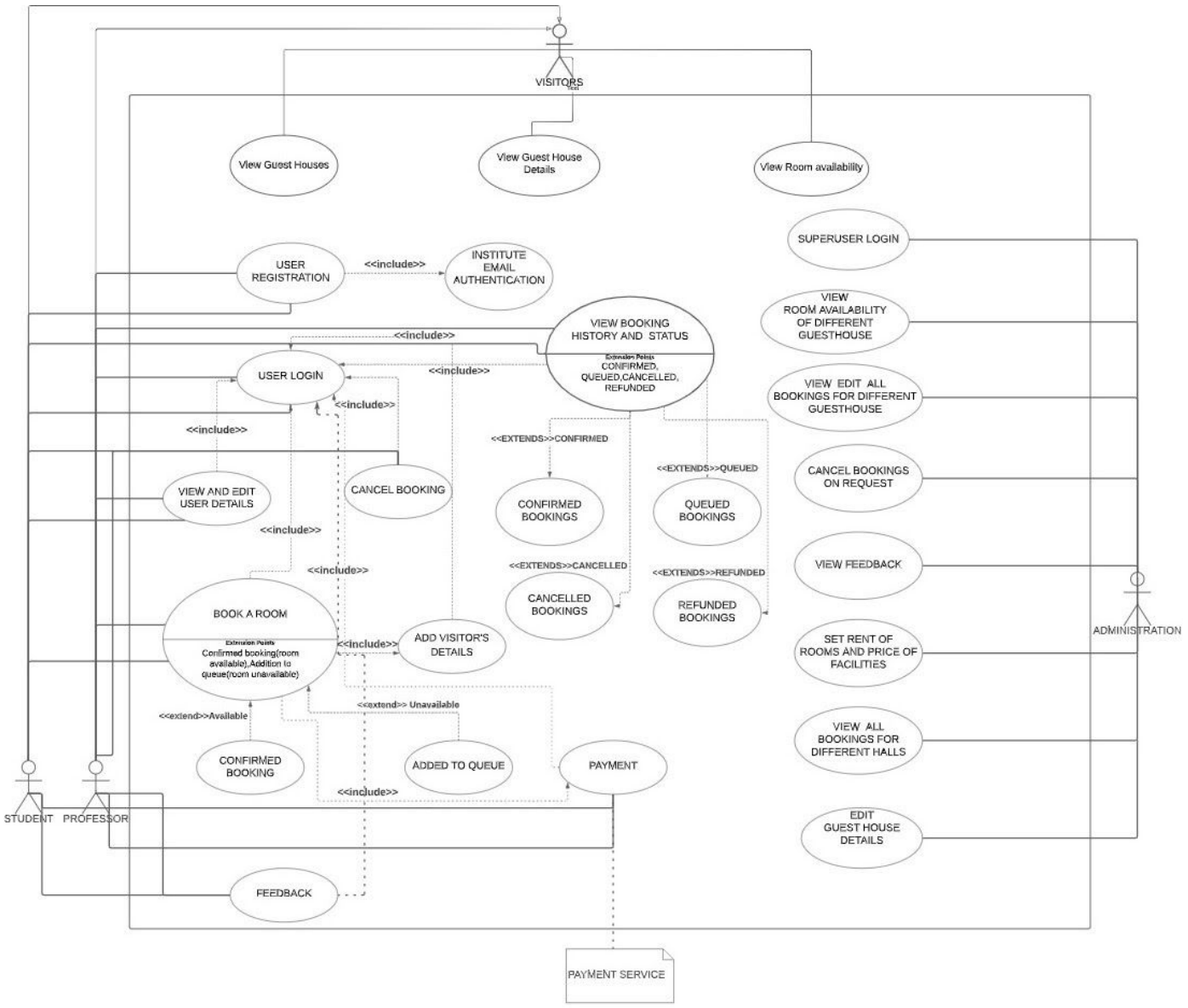
5.4.3 Flexibility requirements

Serial No.	*Description*	*Priority*
1	*All the details about the guesthouses (maximum number of rooms, rental costs, facilities available, food availability, etc.) can all be changed dynamically by the admin(any new change in the guest house details would not interfere with any previous bookings)*	*High*
2	*In cases with no vacant space rooms left in the guesthouse, users can book a room in the waiting queue. Every time any user checks out or a booking is canceled, the queue is checked for any possible booking confirmations in the queue.*	*High*

5.5 Business Rules

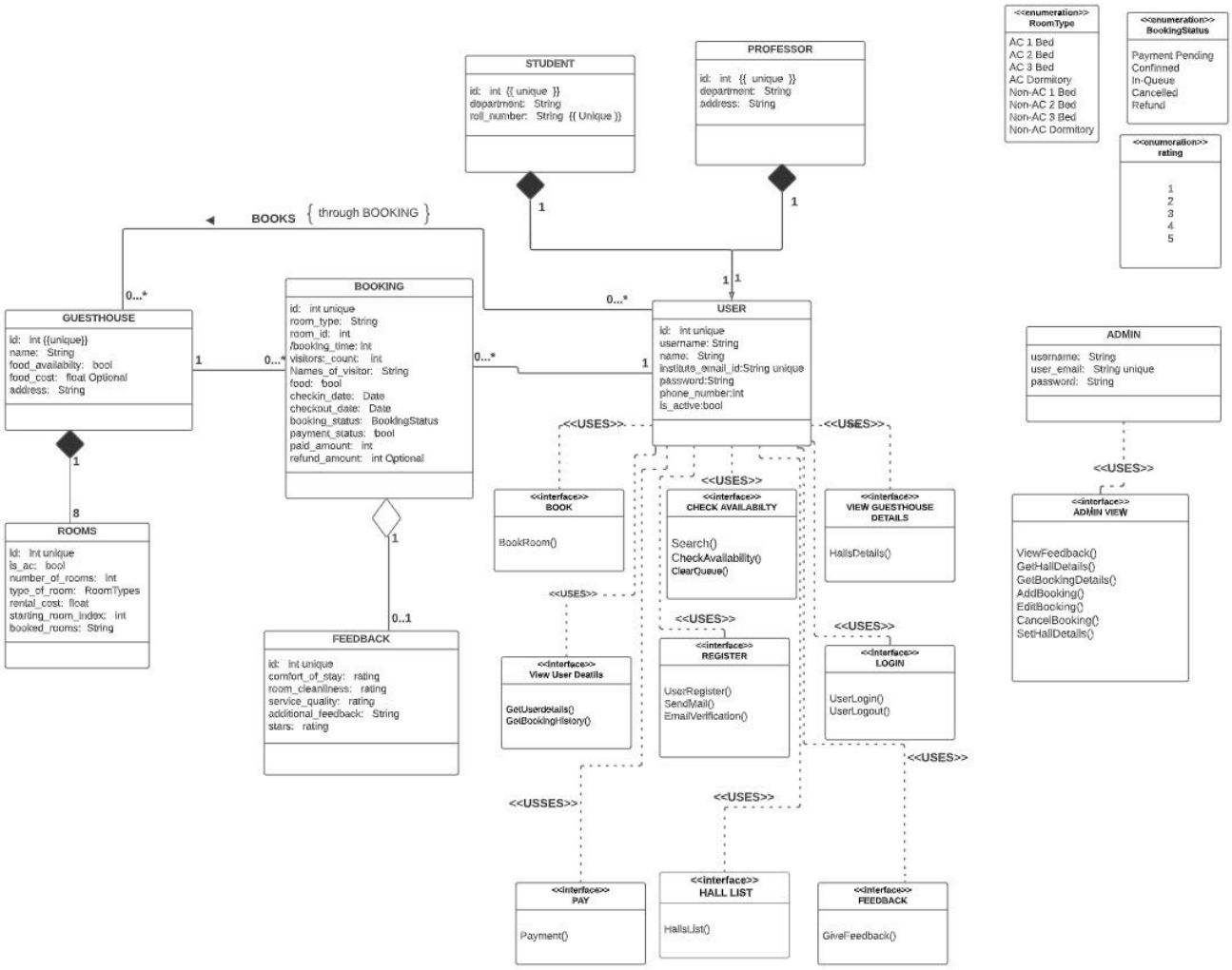
- *A regular user can view the guest house list, guest house details and can also check room availability by mentioning the check-in date and checkout date.*
- *Students and Professors can register and login into the system and only they are authorized to book rooms in guest houses for their relatives. They also have access to the user dashboard where they can view and change their personal information. They can view their past booking history and cancel an active booking if required. They can also provide feedback which will help maintain and improve the service quality of the guest house.*
- *Guest House Administration has all the access that an authorized user has. They can view/change Guesthouse details and services, check the bookings, cancel bookings, register a user, and can also make bookings in special circumstances.*

Appendix A: USE CASE DIAGRAM



USE CASE DIAGRAM FOR
ONLINE GUEST HOUSE BOOKING SYSTEM

Appendix B: ANALYSIS MODEL



CLASS DIAGRAM FOR
ONLINE GUEST HOUSE BOOKING SYSTEM