**SOFTWARE ENGINEERING ,SOFTWARE ARCHITECTURE AND DEVELOPMENT PLAN FOR**

**HOTEL MANAGEMNET SYSTEM**

**GROUP MEMBERS**

**NARENDRA DESHMUKH (202251036)**

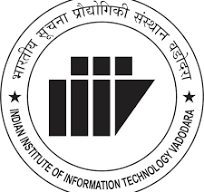
**ROHIT LOKHANDE (202251068)**

**RANISH JAMODE (202251107)**

**MOHIT RITWIK (202251071)**

**SAI TEJA (202251027)**

**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY**



# ABSTRACT

|  |  |  |
| --- | --- | --- |
| **Indian Institute Of Information Technology Vadodara** | **Date**  December 2020 |  |
| **Name of Project**  HOTEL MANAGEMENT SYSTEM | | |
| **Instructor**  Novarun Deb | | **Pages** 6 + 34 |
| In today’s engineering world there are standards to be followed when introducing a new system. A system development lifecycle is a structured process consisting different steps which enables developers to achieve the desired product with minimum risk, cost, and time. A software must go through different phases in its lifecycle development such as analysis, design, coding, testing, deployment, and maintenance.  This project aims at exploring the design phase of software development by using a sample web-based hotel management system. The project discusses observed hotel management problems in developing countries and suggests a solution with detailed description of desired customer and management approach. It also presents the suggested system architecture components and their relationship using different UML diagrams. | | |

**ABSTRACT**

**Key words**

Hotel Management, Requirement Analysis, Software Architecture, System Design, UML. **CONCEPT DEFINITIONS**

|  |  |
| --- | --- |
| DDOS | Distributed Denial of Service |
| HCM | Hybrid connection manager |
| OOD | Object Oriented Design |
| SDLC | System Development Life Cycle |
| SQL | Structured Query Language |
| STRIDE | Spoofing Tampering Repudiation Information disclosure DOS Elevation of privilege |
| UML | Unified Modelling Language |
|  |  |

Technologies Used

PHP

CSS

HTML

XMAAMP

JAVASCRIPT

PYTHON

SELENIUM

CHROME WEBDRIVER ETC…

[ABSTRACT 1](#_Toc164287280)

[1 INTRODUCTION 1](#_Toc164287281)

[2 SYSTEM ARCHTICTURE ANALYSIS 2](#_Toc164287282)

[2.1 Introduction to Software Architecture 2](#_Toc164287283)

[2.2 System Requirement Analysis 2](#_Toc164287284)

[2.2.1 Non-Functional Requirement Analysis 3](#_Toc164287285)

[2.2.2 Functional Requirements Analysis 6](#_Toc164287286)

[3 SYSTEM ARCHITECTURE DESIGN 8](#_Toc164287287)

[3.1 Use Cases Diagram 8](#_Toc164287288)

[3.2 Activity Diagram 19](#_Toc164287289)

[3.3 Sequence Diagram 20](#_Toc164287290)

[3.4 Class Diagram 21](#_Toc164287291)

[3.5 User Interface Design 22](#_Toc164287292)

[REFERENCES 26](#_Toc164287293)

**FIGURES**

FIGURE 1. Hybrid Server-Client Architecture of the Hotel management system .....................................11

FIGURE 2. Use case diagram for hotel management system ................................................................. 13

FIGURE 3. Activity Diagram for customer hotel room reservation ....................................................... 22

FIGURE 4. Activity Diagram for hotel employees and management .................................................... 23

FIGURE 5. Sequence diagram for hotel management system................................................................ 24

FIGURE 6. Class diagram for hotel management system ...................................................................... 25

**TABLES**

TABLE 1. Non-Functional requirements of the hotel management system ............................................. 6

TABLE 2. Functional requirements of the hotel management system ..................................................... 9

TABLE 3. Use case description for hotel management system .............................................................. 13

# INTRODUCTION

The hotel industry continually advances with technology, embracing computerized room reservations, resource management, and innovations like self-check-ins and mobile room keys. However, some hotels in developing countries still rely on paper-based systems, making reservations challenging for customers. This paper proposes a web-based hotel management system to address these issues, offering services like room reservation, cancellation, and online payment processing. The project follows the software development life cycle (SDLC), encompassing analysis, design, coding, and deployment phases.

During the analysis phase, the system's behavioral and functional aspects are identified, including its basic features, primary users, and functional capacity. Functional requirements outline the system's operations, while non-functional requirements define its quality attributes such as performance and security.

The design phase aims to fulfill the identified system requirements by understanding its functionality and creating Unified Modeling Language (UML) diagrams to visualize the system architecture. Threat modeling is also conducted to identify potential threats to the system's confidentiality, integrity, and availability, along with suggesting prevention techniques.

Overall, the project explores the behavioral and structural aspects of a web-based hotel management system, aiming to provide efficiency and organization compared to existing systems in lower standard hotels in developing countries. It outlines the system analysis and design phases and includes sample interface designs for various users. The system suggested in this paper solves most of these problem by providing web-based room reservation system and employee/room management. Since websites are the main platforms to search and find hotel services, the web-based room reservation system provides the hotels with global reach and customers with suitable information and effortless service. The system also enables the hotels to achieve organized and smooth employee and room management by using applications that allows better data storage with reduced data redundancy, secure information access and efficient updates.

# SYSTEM ARCHTICTURE ANALYSIS

## Introduction to Software Architecture

System architecture encompasses both hardware and software components. The hardware component comprises client computers, servers, and the network connecting them. Client computers serve as input and output mediums, while servers process and store data, reflecting the server-client architecture.

On the other hand, software architecture abstracts the behavior of a system's software component, detailing how its elements interact. It offers various architectural views tailored to different stakeholders, including architects, programmers, evaluators, configuration managers, and customers. These views provide insights into the system's functionality, component connections, non-functional features, development management, and software implementation on hardware infrastructure. Software architecture is shaped by the system's primary requirements, which are categorized as functional and non-functional requirements.

## System Requirement Analysis

System requirement analysis is a structured approach of identifying expected features and resources that are needed to achieve the desired product. The requirement analysis in software architecture helps to simplify the modelling and coding phase of system development. Requirements needs to be gathered and analysed before documentation in order to study and understand the practicality of the product to be developed. Using properly analysed system requirements to develop a product helps software engineers not only to meet customer satisfaction but also achieve system security. (Dennis, Wixom & Roth 2012; Software testing help 2020.)

### Non-Functional Requirement Analysis

Non-functional requirements of a system delve into its inherent qualities, helping developers establish constraints on design and delineate the system's scope based on various aspects such as performance, response time, security, and memory capacity. These requirements are broadly categorized into operational, performance, security, and cultural and political requirements.

Operational requirements delineate the necessary operating environment for the system to fulfill its functional objectives and detail its interactions with other systems. This encompasses hardware and software specifications aligned with the operating system, database system, and network system requirements. Additionally, operational requirements identify integration with other systems and how they interact. They also address the system's adaptability to evolving technological landscapes, specifying how it should accommodate changes and upgrades.

Performance requirements outline how effectively the system operates within specified environments and timelines. This includes parameters such as response time, capacity, availability, and scalability. Response time dictates the duration for specific tasks, balancing development costs and user expectations. Capacity assesses the system's ability to handle numerous operations concurrently. Availability ensures the system functions whenever required, while scalability denotes its capability to adjust performance based on application demand.

Security requirements are crucial in safeguarding valuable information in today's high-risk environment. These requirements ensure the confidentiality, integrity, and availability of exchanged data. Measures such as role-based access control, authentication, authorization, encryption, and virus protection contribute to meeting security needs.

Cultural and political requirements vary across countries and encompass standards and regulations governing system development. These requirements address legal obligations imposed by local governments concerning system infrastructure and security. They also include language support specifications for targeted local and international users and consider unstated norms like date and time formats.

The provided table (TABLE 1) outlines non-functional requirements for a hotel management system, categorized into the aforementioned groups

.TABLE 1. Non-functional requirements of the hotel management system

|  |  |
| --- | --- |
| Type of Requirement | Specification |
| Operational Requirements | * The system should be able to work on Internet Explorer, Google Chrome, Safari, Opera, Mozilla Firefox, Microsoft Edge, and other web browsers. * The system should be able to fully function on smart phones and mobile devices * On-premises devices should always have internet connection to allow real-time database updates * The system should be able to operate on different operating systems including every generation of Microsoft Windows, Mac, and Linux. * The system should be able to connect with online payment methods including PayPal and TransferWise. * The system should be able to support maintenance within two weeks of notice. |
| Performance Requirements | * Response time for visible pages for customers should be less than 5 seconds excluding online payment transactions. * Response time for visible staff members should be less than 10 seconds excluding payment transactions. |

|  |  |  |
| --- | --- | --- |
|  | • | The system should be able to support 500,000 customers in total. |
|  | • | The system should be able to support a maximum of 1000 customers simultaneously in peak hours. |
|  | • | The system server should be able to handle 1000 requests per second. |
|  | • | The website should be able to generate 5000 daily customer interactions. |
|  | • | The system should be able to run all times except schedule updates and maintenances. |
|  | • | Maintenance should not exceed a maximum of 8 hours in a day and 16 hours in a month. |
| Security Requirements | • | Customers can only view room availability, book room, cancel room and pay for room reservation through secured payment method. |
|  | • | Front desk can only view hotel room status, update room occupation status, update check in and check outs and process payment through cash register. |
|  | • | Housekeeping manger can edit room cleaning status, update housekeeping shifts and update minibar use. |
|  | • | Housekeeper can edit room cleaning status and update minibar use. |
|  | • | General manager should be able to view and edit all tasks that can be accessed by staff members, can view and update employee information, view and update staff work shifts. |
|  | • | Data transaction from customer to the system through websites will be encrypted. |
|  | • | The system should ask staff members to authenticate themselves using employee Id and password before they are granted any access. |
|  | • | Passwords of every stuff member should be updated every one year |
|  | • | All on-premises devices should be equipped with strong antivirus software. |
|  | • | All security measure should be updated 3 every month |
| Cultural and Political Requirements | • | The system should be able to support Amharic and English languages |
|  | • | Currency should be stated in the system as (United States dollar) USD and ETB (Ethiopian birr). |
|  | • | Date input in the system should follow date month and year format (DD.MM.YYY). |

### Functional Requirements Analysis

The functional requirement analysis in software design is essential in determining the functionality and behaviour of the system to be developed by specifying how the system interact with the user, what input it should be able to take and what tasks it should be able to perform successfully (Dennis, Wixom & Roth 2012). In the hotel management system, there are different users with different roles that the system should interact accordingly. These users are the customers and stuff members. The functional requirements of the system in each user category are listed in the table (TABLE 2) below.

TABLE 2. Functional requirements of the hotel management system

|  |  |  |
| --- | --- | --- |
| User category | Specifications | |
| Customer | * The system should display room availability and price * The system should enable user to input personal information * The system should direct user to secure payment method * The system should send notification to confirm room reservation. | |
|  | • | The system should enable users to cancel room reservation. |
| Stuff member | • | The system should have Authentication page with username password. |
|  | • | System should have ‘forgot password’ option on the authentication page that sends a link to verified email or phone number to set different password. |
|  | • | The system should enable user to view, edit and add employee information. |
|  | • | The system should enable user to view and edit room occupation status |
|  | • | The system should enable user to view and edit room cleaning status including minbar usage. |
|  | • | The system should enable user to process payment using cash register machine. |

# SYSTEM ARCHITECTURE DESIGN

During the architecture design phase of system development, developers define the system's components, interfaces, and data flow. They identify subsystems and their interdependencies through visual modeling, often utilizing the Unified Modeling Language (UML). UML serves as a standard language for specifying, documenting, and developing intricate systems using a set of diagrams.

UML diagrams are categorized into behavioral and structural types, focusing on different aspects of the system. Behavioral diagrams depict dynamic behaviors such as use cases, activities, states, and sequences. Structural diagrams, on the other hand, illustrate static elements and their relationships within the system. These diagrams represent stable structural components, unlike behavioral diagrams which capture changes over time.

Key structural UML diagrams include class, component, and deployment diagrams, which highlight the system's main structural elements and their configurations. While there are over 13 UML diagrams available, this paper focuses on a select few deemed most suitable for describing the hotel management system under development.

## Use Cases Diagram

Use case diagrams are behavioural diagrams that visualizes the interaction between actors and the system. Actors, can be users or other systems, are entities that control the functionality of the system. Use cases diagram provides a simplifies view of what the system does by identifying system functionalities and how they interact with internal and external actors. In the hotel management system there are five actors, the general manager, housekeeping manager, housekeeping, front desk(reception) and customer, which have different roles that the system responds to accordingly. (Otero 2012, Chapter 2.)

The use case diagram for the system is presented on FIGURE 2 below.

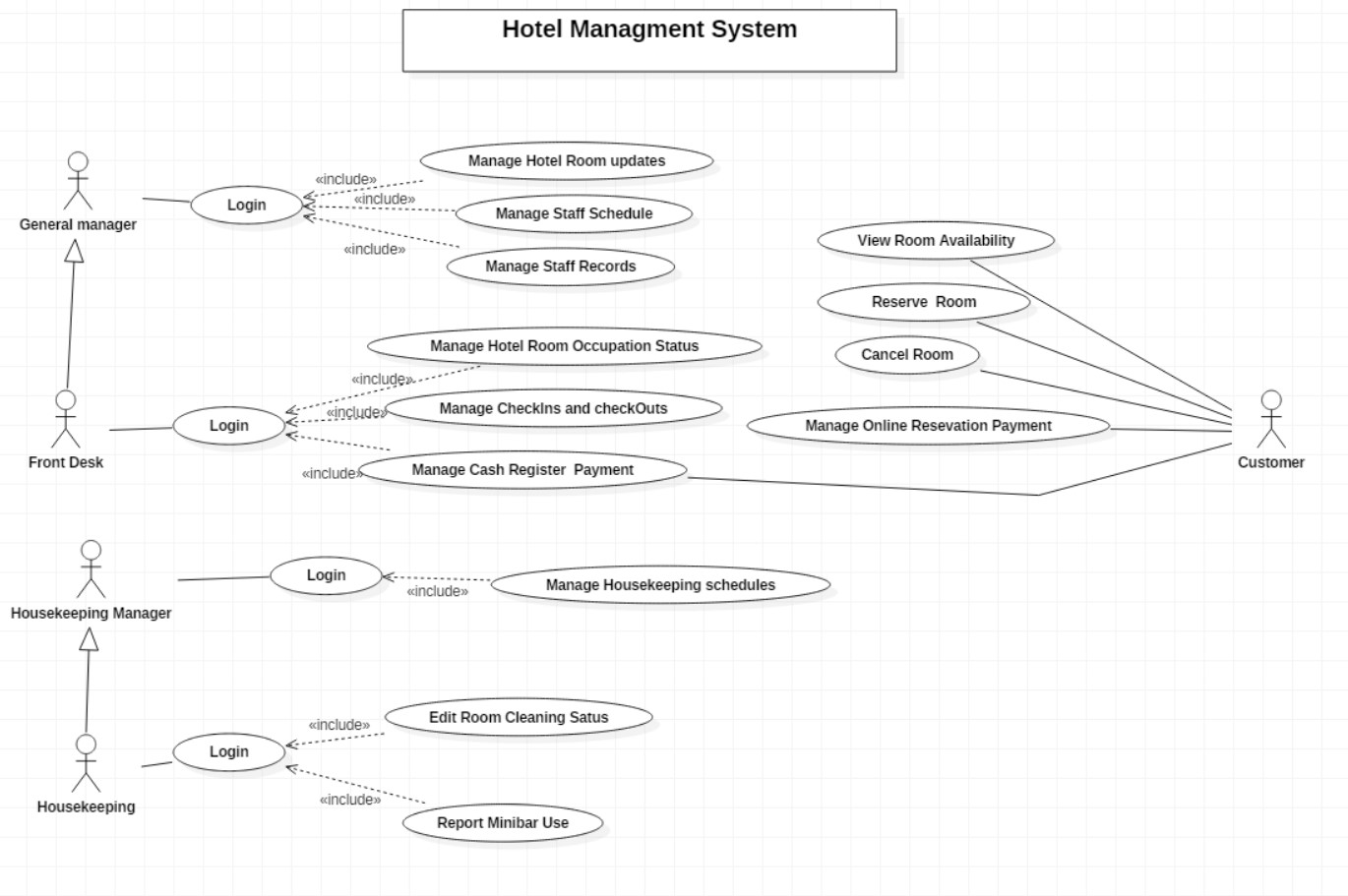


FIGURE 2. Use case diagram for hotel management system.

A use case diagram can be supported by a case description for more understanding. This description can provide more detailed information which are not included in the diagram due to simplicity and readability demands. TABLE 3 includes use case description of the hotel management system.

TABLE 3. Use case description for hotel management system

|  |  |
| --- | --- |
| Use Case 01 | Manage Hotel Room Updates |
| Actor | General manager |
| Description | The system enables the general manager to view, add, and edit hotel room information. |

|  |  |
| --- | --- |
| Pre-condition | The general manager should open hotel management system and login to the system. |
| Post-condition | Undefined |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Hotel Room option. 3. The system displays list of rooms available in the hotel that are sorted according to their size and price. 4. The system displays add room, remove room, edit room, and view room information (including price, room size, bed, accessories, occupation status) options. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |
| Use Case 02 | Manage Staff Schedule |
| Actor | General manager |
| Description | The system enables the general manager to view and update staff work schedule. |
| Pre-condition | The general manager should open hotel management system and login to the system. |
| Post-condition | Undefined |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Staff Schedule option. 3. The system displays date (current date as default), list of department schedules (housekeeping and reception) and create new schedule options. 4. The user selects date and department schedule from the options. 5. The system opens Microsoft excel and displays list of people and shifts hours for the specific day. 6. The system enables the user to view, edit, save, print, and send staff schedule. |
| Alternative Flow | 1. The user login to the system. 2. The user selects Manage Staff Schedule option. |

|  |  |
| --- | --- |
|  | 1. The system displays date (current date as default) list of department schedules (housekeeping and reception). 2. The user chooses Create New Schedule option. 3. The system opens Microsoft excel. |
| Assumptions |  |
| Exceptions | If the user chooses past dates on the Manage Staff Schedule screen, the system will only allow the user to view, print and send the selected schedule option. |
| Use Case 03 | Manage Staff Records |
| Actor | General manager |
| Description | The system enables the user to view and update hotel employee records. |
| Pre-condition | The user should open hotel management system and login to the system. |
| Post-condition | Undefined |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Staff Records option. 3. The system opens excel and displays list of employee records. 4. The system provides view, add, edit, and save options to the user. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |
| Use Case 04 | Manage Hotel Room Occupation Status |
| Actor | Front desk |
| Description | The system enables the receptionist to check room availability, room cleaning status and room booking status. |
| Pre-condition | The user should open the hotel management system on company computer and login. |
| Post-condition | undefined |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Hotel Room Occupation Status option |

|  |  |
| --- | --- |
|  | 1. The system displays list of rooms with cleaning status (ready/ cleaning), occupation status (vacant/booked/occupied), check-in and check-out options. 2. The user clicks on drop down menu on occupation status for a specific room. 3. The system displays list of status options, vacant, booked and occupied. 4. User selects options according to customer check-in and check-out requests. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |
| Use Case 05 | Manage Check-Ins and Check-Outs |
| Actor | Front desk |
| Description | The system enables the user to process check-ins by inputting customer information and check-outs by processing payments. |
| Pre-condition | The user should open the hotel management system on company computer and login. |
| Post-condition |  |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Hotel Room Occupation Status option 3. The system displays list of rooms with cleaning status (ready/ cleaning), occupation status (vacant/booked/occupied), check-in and check-out options. 4. The user clicks on Check-in or Check-out option on the room requested. 5. If (Check-in): - the system displays a page where the user can inter customer information.   If (Check-out): - the system directs user to room usage cost summery and then to cash registry payment processing page. |
| Alternative Flow |  |
| Assumptions |  |

|  |  |
| --- | --- |
| Exceptions |  |
| Use Case 06 | Manage Cash Register Payment |
| Actor | Front desk / Customer |
| Description | The system enables the front desk to process check out payments using Cash registry method. The customer plays a role by inserting card and inputting card pin code. |
| Pre-condition | The user should open the hotel management system on company computer and login. |
| Post-condition | The system prints out receipt. |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Hotel Room Occupation Status option 3. The system displays list of rooms with cleaning status (ready/ cleaning), occupation status (vacant/booked/occupied), check-in and check-out options. 4. The user Check-out option on the room requested. 5. The system opens cost summary page (days stayed \* room price + additional services) and displays Edit, Discard, and Continue options. 6. The user chooses Continue. 7. The system directs to payment page that allows card or cash payment. 8. The customer inserts card and input pin. 9. The system accepts payment and print receipt. |
| Alternative Flow | If customer pays in cash, the receptionist must confirm payment on the system manually. |
| Assumptions |  |
| Exceptions | If the system rejects card payment, the system provides Try Again option.  The customer then removes and insert card again and input pin code. |
| Use Case 07 | Manage Housekeeping Schedule |
| Actor | Housekeeping manager |
| Description | The system enables the user to view and update housekeeping staff weekly schedule. |

|  |  |
| --- | --- |
| Pre-condition | The user should open the hotel management system on company computer and login. |
| Post-condition |  |
| Basic flow | 1. The user login to the system. 2. The user selects Manage Housekeeping Schedule option. 3. The system displays Date, Room Cleaning Status, Housekeeping Schedule, and Create New Schedule options. 4. The user selects date and Housekeeping schedule from the options. 5. The system opens Microsoft excel and displays list of people and shifts hours for the specific day. 6. The user can view, edit, save, print, and send schedule for the specific day. |
| Alternative Flow | If the user selects Create New Schedule option, the system open Microsoft excel. |
| Assumptions |  |
| Exceptions |  |
| Use Case 08 | Edit Room Cleaning Status |
| Actor | Housekeeping |
| Description | The system enables the user to view and update room cleaning status. |
| Pre-condition | The user should open the hotel management system on company tablet and login. |
| Post-condition | Room cleaning status will be set to ready for booking and check-in. |
| Basic flow | 1. The user login to the system. 2. The system displays list of assigned rooms, room cleaning status (ready/cleaning), occupation status (vacant/occupied) and minibar report. 3. The user clicks on drop down menu on the cleaning status section of the cleaned room and selects the option Ready. |
| Alternative Flow |  |
| Assumptions | The Housekeeper can only edit room cleaning status and minibar report. |
| Exceptions |  |

|  |  |
| --- | --- |
| Use Case 09 | Report Minibar Use |
| Actor | Housekeeping |
| Description | The system enables the user to report items used from the minbar that is found in the room. |
| Pre-condition | The user should open the hotel management system on company tablet and login. |
| Post-condition | The system sends list of items used and their price to the cost calculating software. |
| Basic flow | 1. The user login to the system. 2. The system displays list of assigned rooms, room cleaning status (ready/cleaning), occupation status (vacant/occupied) and minibar use report. 3. The user clicks on minibar use option corresponding with the desired room. 4. The system displays list of items available in the minibar, their price, number of items used (with drop down menu that includes number of items) and check box in a table format. 5. The user can click on the check box of items that are used and select the number of items from the dropdown menu. 6. The user clicks on save. |
| Alternative Flow |  |
| Assumptions | The Housekeeper can only edit room cleaning status and minibar report. |
| Exceptions |  |
| Use Case 10 | View Room Availability |
| Actor | Customer |
| Description | The system provides information on the type of rooms, price, and availability on requested dates. The system also enables user to easily search room availability by entering desired date and room type. |
| Pre-condition | The user opens the hotel website on personal computer or mobile device. |
| Post-condition | Room type, price and picture displayed. |
| Basic flow | 1. The opens the hotel website and clicks on Find Room. |

|  |  |
| --- | --- |
|  | 1. The system displays list of available rooms, price, and search option. The search option has two input fields, Room type and Date (starting and ending period). 2. The user inputs room type and the time length desired to stay in the room on the search field and clicks search. 3. The system displays available requested room types, their price and picture with Book Now option under them. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions | The system displays ‘No Available [room type searched] Rooms at the Moment’, if the system could not find results that match with the search criteria. |
| Use Case 11 | Reserve Room |
| Actor | Customer |
| Description | The system provides information about available rooms and allows user to book desired room. |
| Pre-condition | The user opens the hotel website on personal computer or mobile device. |
| Post-condition | The system sends booking confirmation to user email address. |
| Basic flow | 1. The opens the hotel website and clicks on Find Room. 2. The system displays list of available rooms with Book now option. 3. The user clicks on Book now. 4. The system directs to a page where the user can enter all necessary personal information (full name, Id number, phone number and email address) and process online booking payment through secure payment methods including PayPal and TransferWise. 5. The system sends confirmation email. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |
| Use Case 12 | Cancel Room Reservation |
| Actor | Customer |

|  |  |
| --- | --- |
| Description | The system provides information reserved rooms and allows users to cancel unwanted reservations. |
| Pre-condition | The user opens the hotel website on personal computer or mobile device. |
| Post-condition | The system sends confirmation email for reservation canceling with refund information. |
| Basic flow | 1. The opens the hotel website and clicks on Find Reservation. 2. The system displays a login box with text fields to enter reservation number and email address. 3. The user inputs reservation number (from confirmation email) and email address and press login. 4. The system directs to a page with reserved room information and Cancel Reservation option. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |
| Use Case 13 | Manage Online reservation Payment |
| Actor | Customer |
| Description | The system enables user to process booking payment using secure online payment methods. |
| Pre-condition | The user opens the hotel website on personal computer or mobile device.  The user chooses the best suited room from available rooms list.  The user inputs all necessary information for room reservation |
| Post-condition | User receives payment confirmation together with room confirmation email. |
| Basic flow | 1. The user selects ‘Continue to payment’ option after filling reservation information. 2. The system directs user to a page where the user selects suitable payment method from the available options. 3. The system direct to the chosen payment method. 4. User inputs requested payment credentials and confirm payment. |
| Alternative Flow |  |
| Assumptions |  |
| Exceptions |  |

## Activity Diagram

Activity diagram is a behavioural diagram that shows dynamic aspects of a system. It describes the flow of the system from an initial system triggering act of a user to decision making steps and system outputs.

The diagrams below visualize the workflow of the hotel management system for room reservation (FIGURE 3) and daily system use for hotel stuff members (FIGURE 4). (Gennick 2003.)

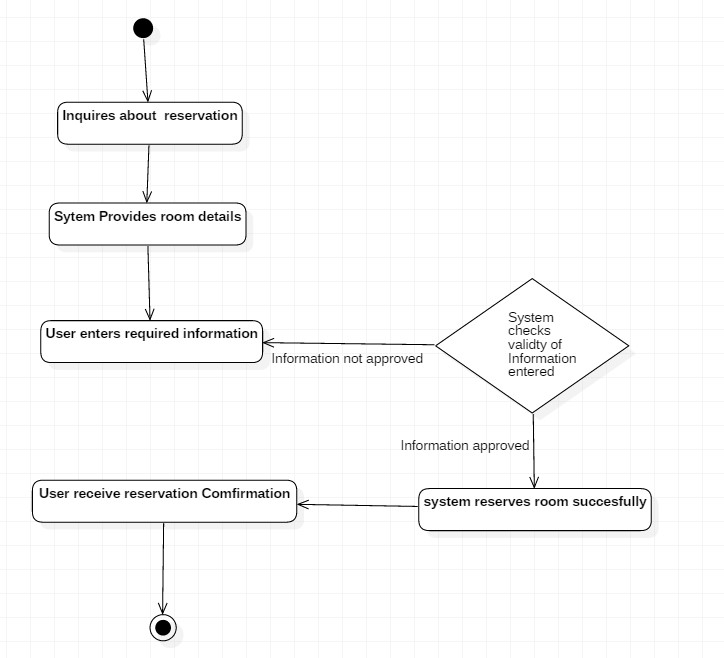


FIGURE 3. Activity Diagram for room reservation (customer).

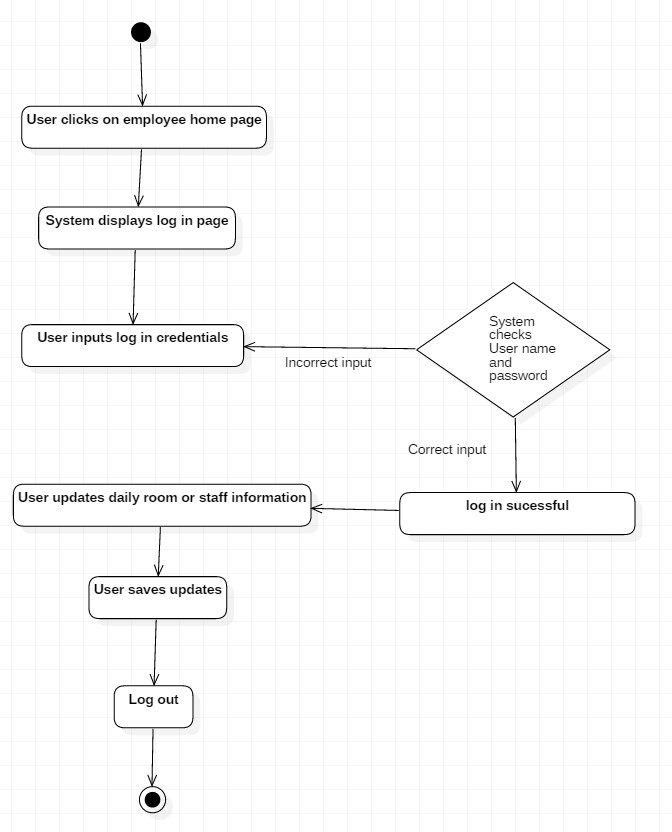


FIGURE 4. Activity Diagram for hotel employees and management.

## Sequence Diagram

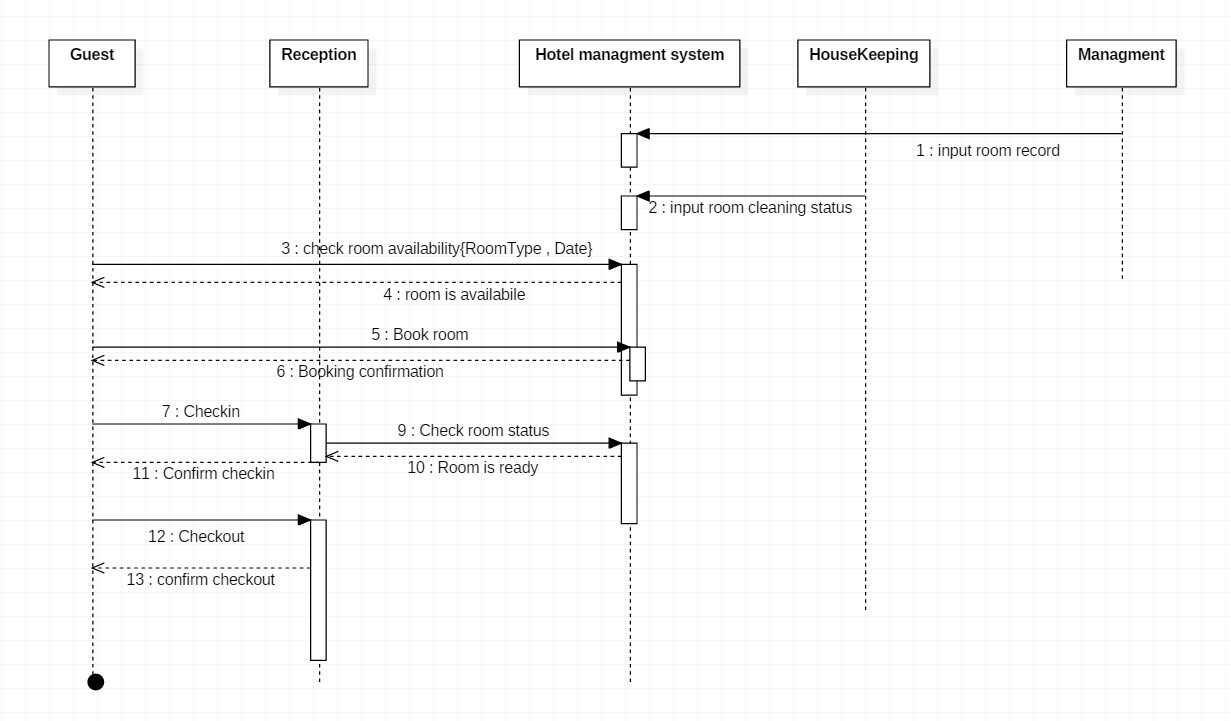
Sequence diagrams illustrate the chronological interaction between system components, depicting time along the vertical axis and system objects aligned horizontally. They showcase the sequence of message execution and decision-making within the system architecture. The provided sequence diagram (FIGURE 5) outlines the step-by-step processes of setting room status, room booking, check-ins, and check-outs over time. Additionally, it identifies the system objects responsible for executing specific actions during these processes.

FIGURE 5. Sequence diagram for hotel management

## Class Diagram

Class diagrams are structural diagrams which visualize classes of a system and their relationship. Class is an object that have its own characteristics which are represented by attributes, what data the class contains, and operation which describes what the class does. The diagram contains rectangles with three compartments for class, attribute (with value type next to them in bracket) and operations (also with value type) respectively from top to bottom. These classes are also connected to each other showing the relationship between them. Class diagrams can also be used to generate executable code of the system. The diagram (FIGURE 6.) below explains the classes of the hotel management system and their relationship. (Gennick, 2003.)

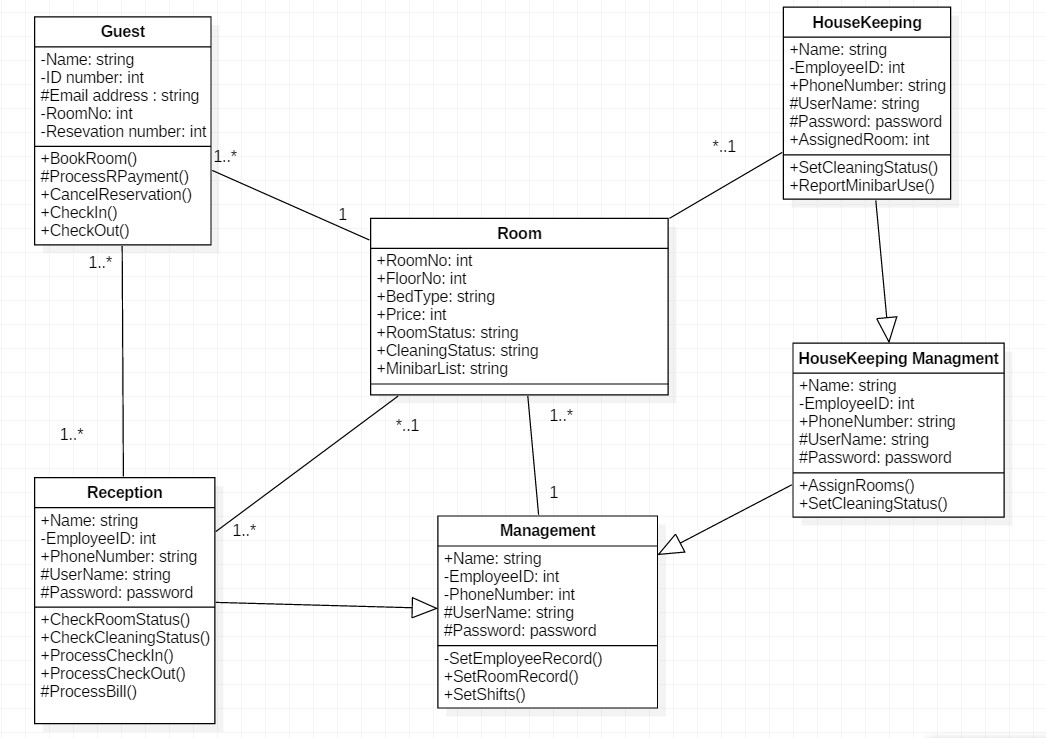


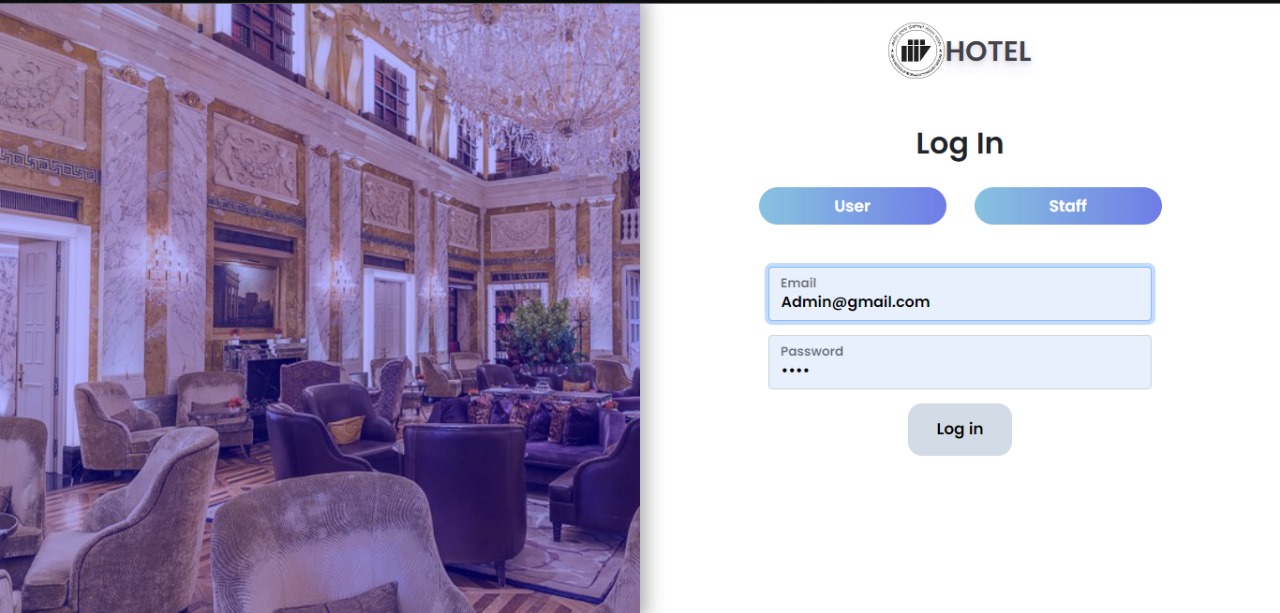
FIGURE 6. Class diagram for hotel management system.

## User Interface Design

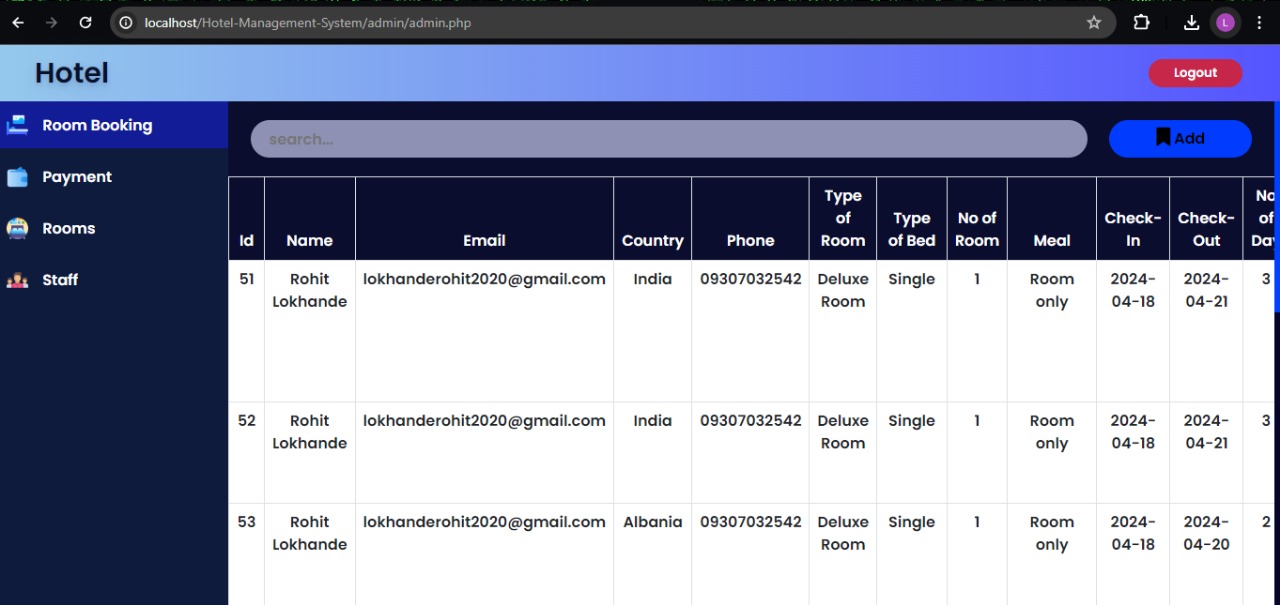
The user interface (UI) of a system serves as its visual framework for user and external system interaction, encompassing navigation fields, input methods, and output displays. It is crucial for UIs to be aesthetically pleasing, intuitive, and informative to facilitate ease of use.

In the hotel management system, various users interact with distinct UIs tailored to their roles. Hotel employees access the system via software installed on company devices, with different UIs customized based on their role-based access permissions. This differentiation is exemplified in the home page UIs designed for management (FIGURE 7), front desk (FIGURE 8), and housekeeping departments (FIGURE 9).

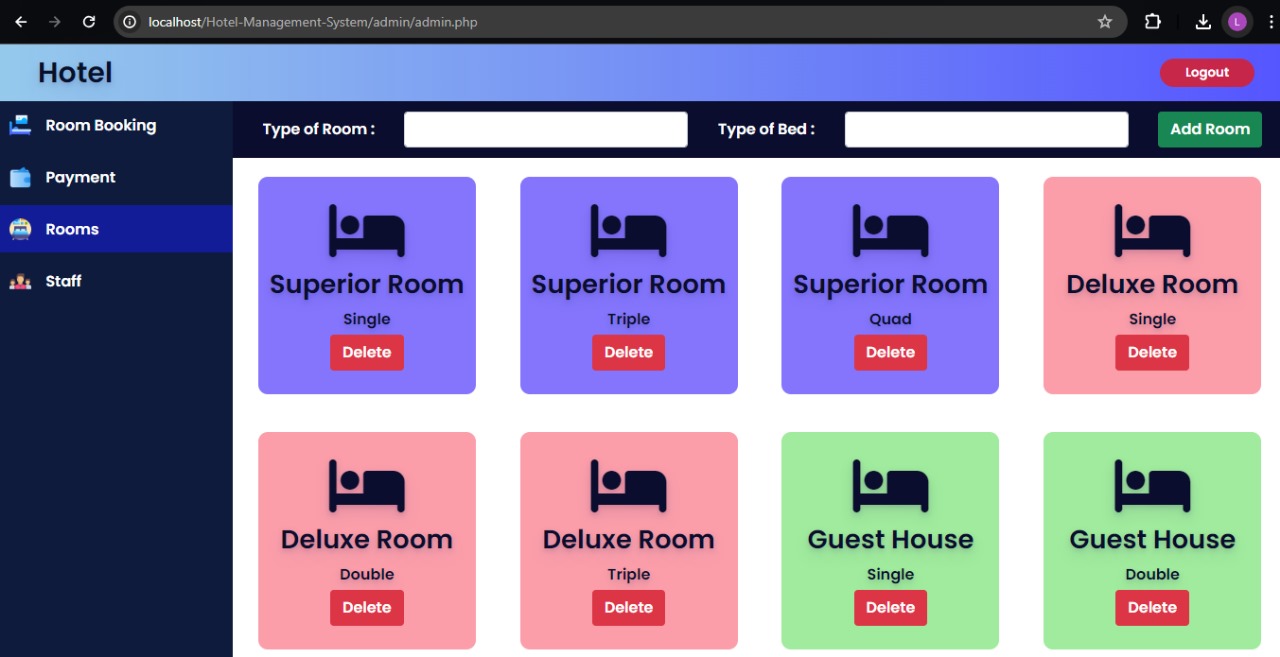
LOGIN PAGE



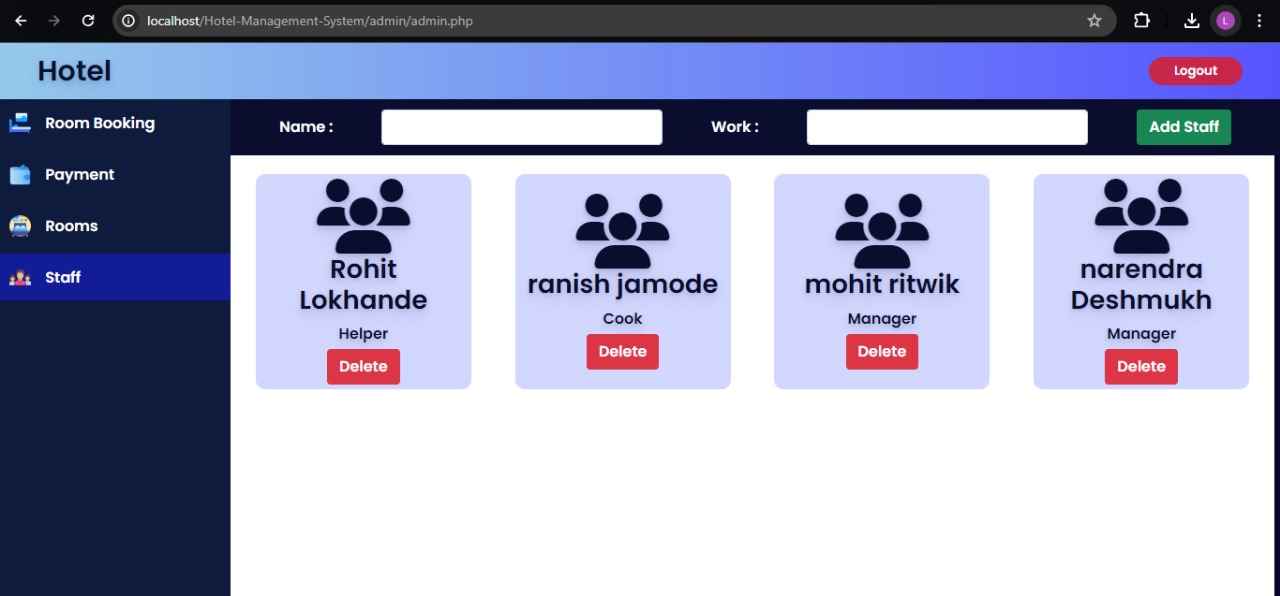
STAFF DASHBOARD



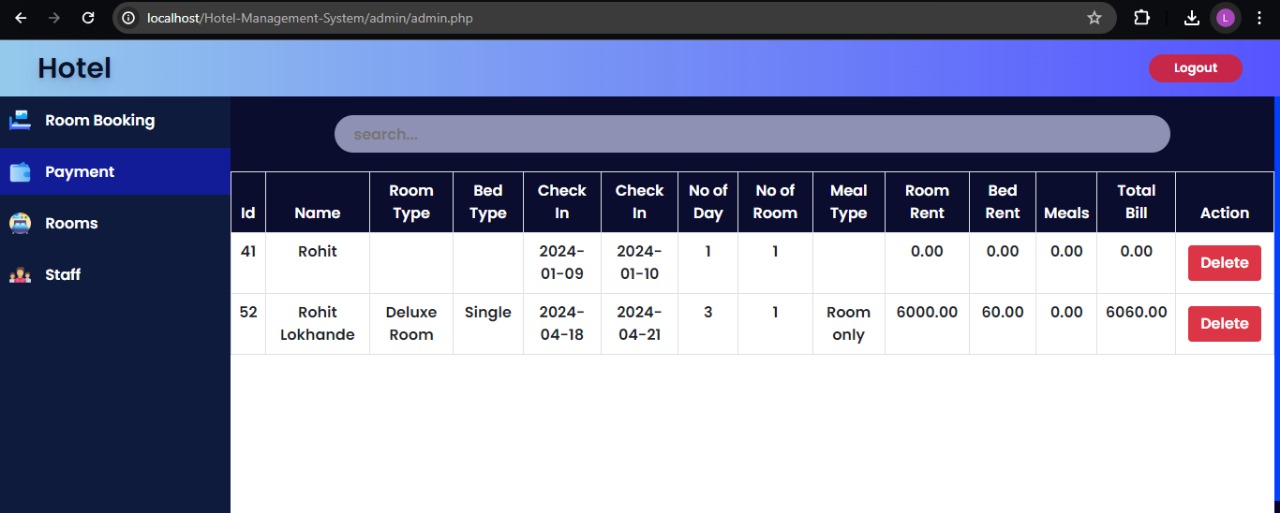
STAFF -ROOM EDIT DASHBOARD



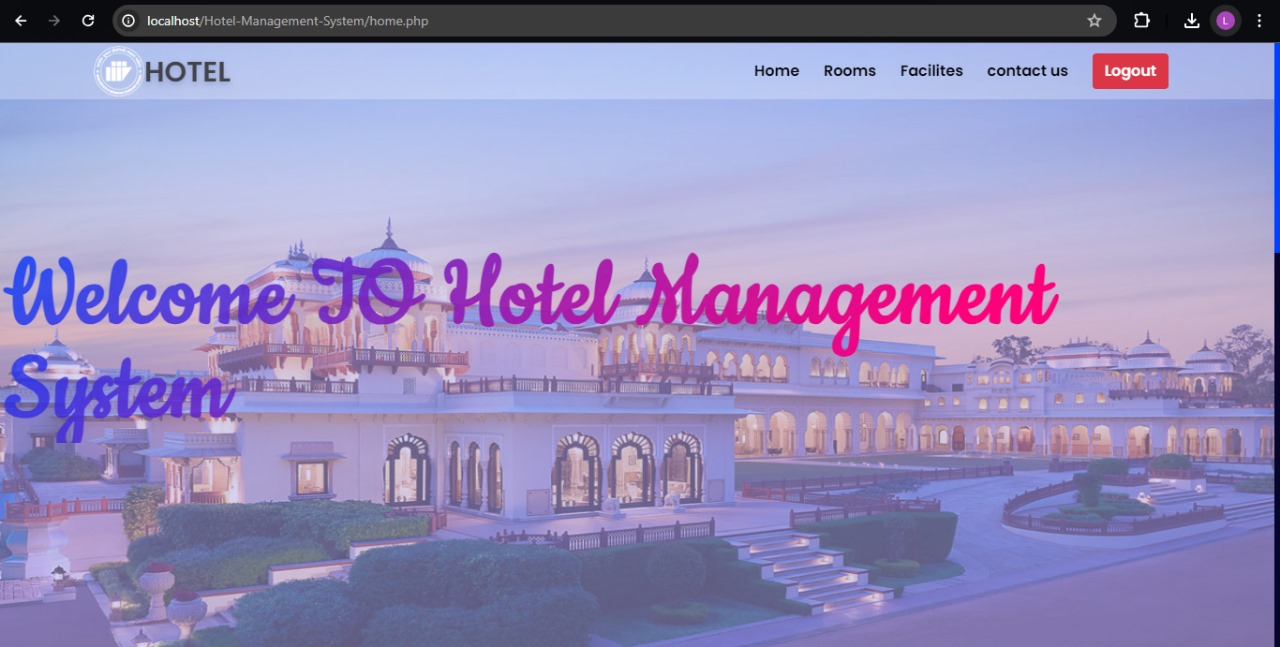
STAFF -JOB MANAGEMENT DASHBOARD



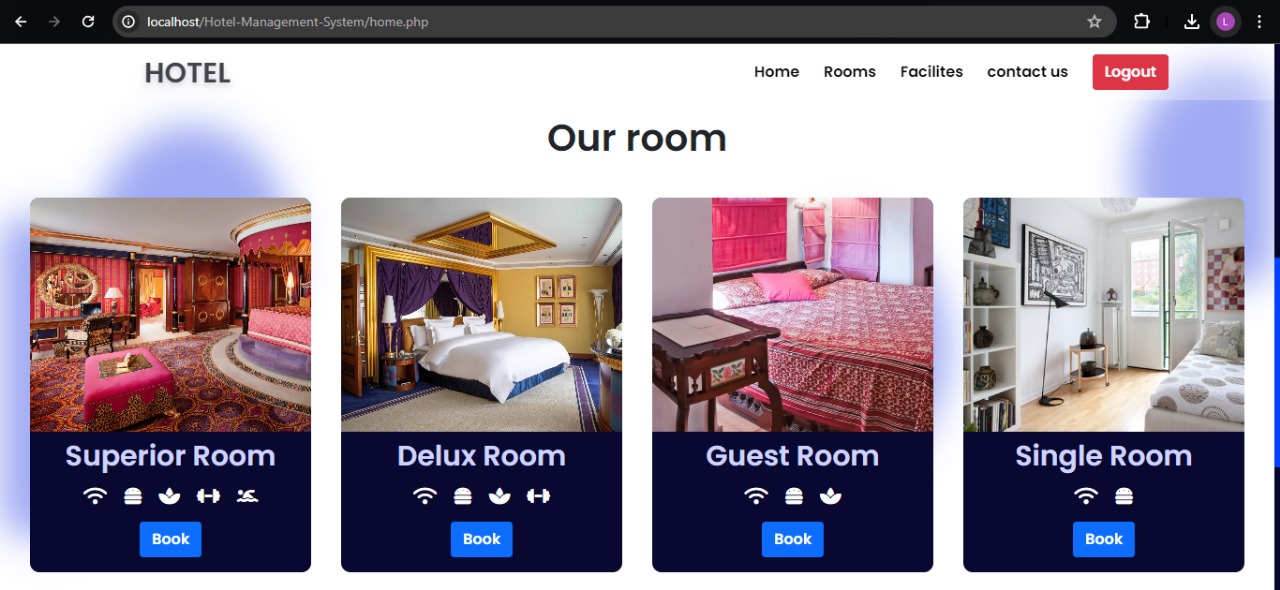
STAFF - PAYMENT CONFIRMATION DASHBOARD

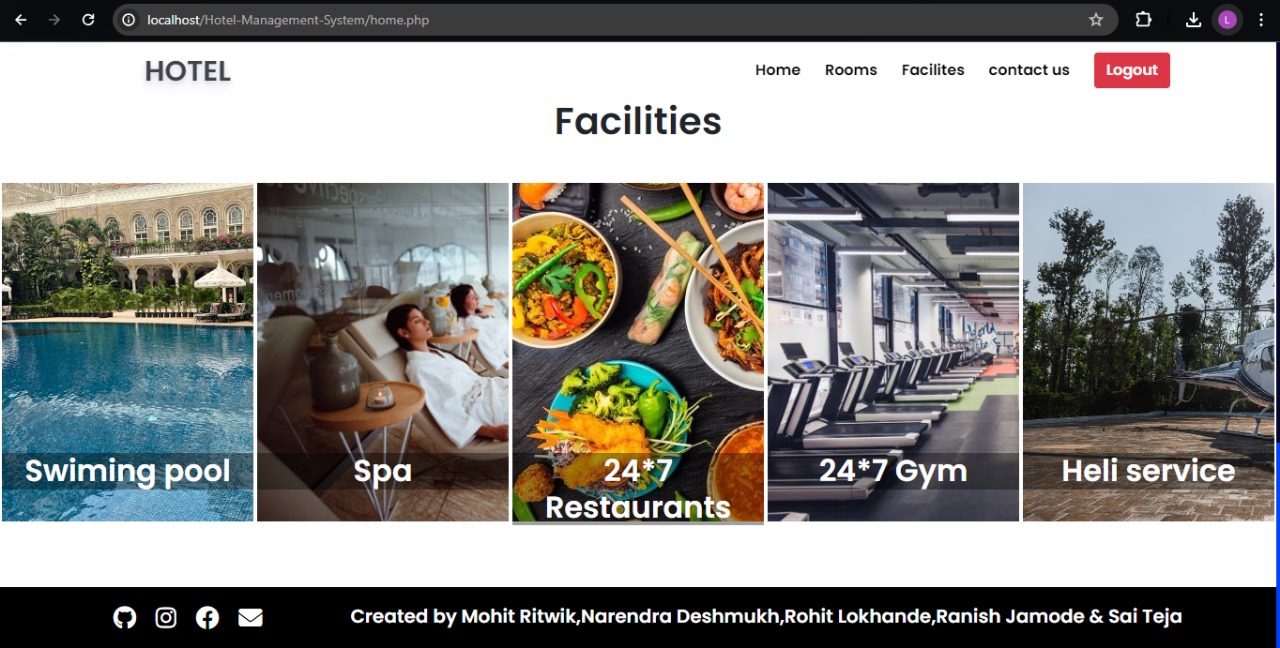


USER – HOMEPAGE



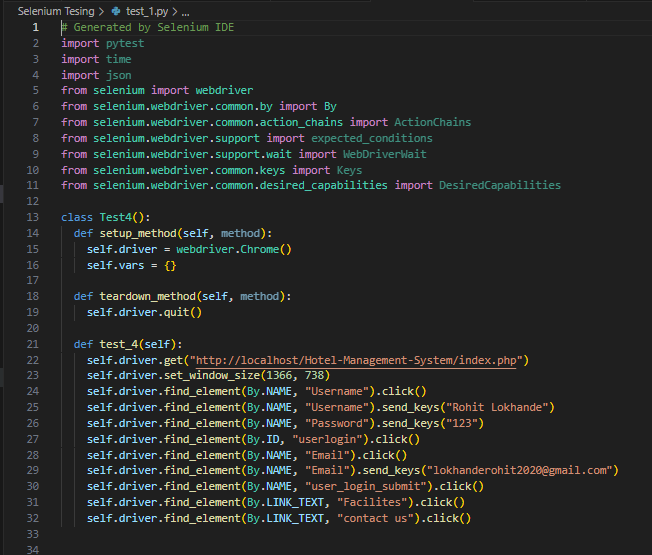
USER – ROOM SELECTION



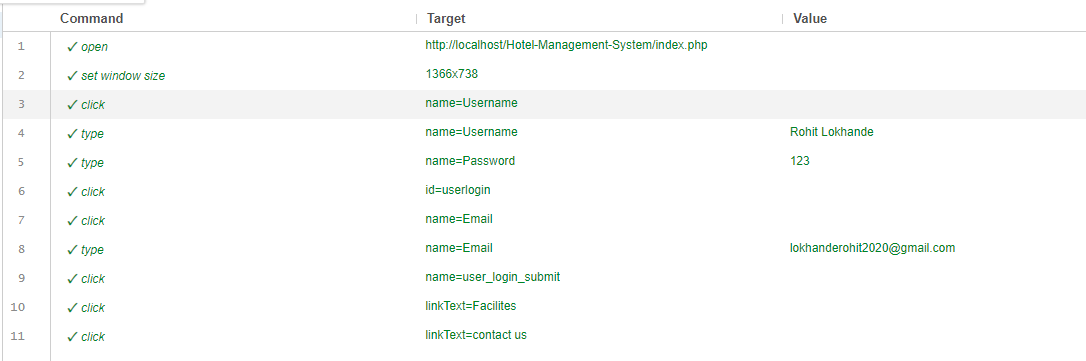


3.6 SELENIUM TESTING

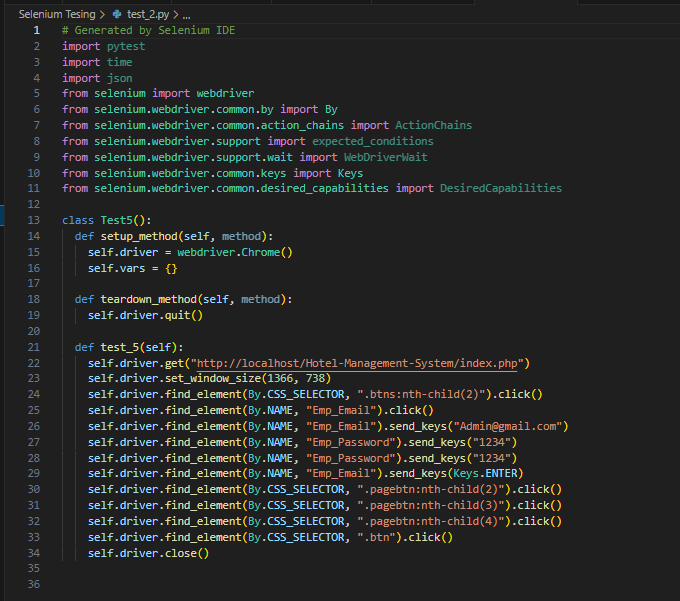
1. Test1:



Output:



1. Test



# Output:

# 

# REFERENCES

Hanmer, RS. 2013, Pattern-Oriented Software Architecture for Dummies, John Wiley & Sons, Incorporated, Somerset.

Tourism Has generated 20% of total world employment since 2013. Available: [https://www.tourismreview.com/tourism-industry-is-the-pillar-of-economy-news11210.](https://www.tourism-review.com/tourism-industry-is-the-pillar-of-economy-news11210) Accessed 12 December 2020.

Dennis A. Wixom B.H., Roth R.M 2012, system analysis and design, John Wiley & Sons. Available:

[https://learning.oreilly.com/library/view/system-analysis-and/9781118057629/17\_chap08.html.A](https://learning.oreilly.com/library/view/system-analysis-and/9781118057629/17_chap08.html.0)ccessed 16 August 2020.

Lano, K. 2005, Advanced Systems Design with Java, UML and MDA, Elsevier Science & Technology, Burlington.

What is requirements analysis and gathering in SDLC. Available: [https://www.softwaretestinghelp.com/requirement-analysis-in-sdlc/.](https://www.softwaretestinghelp.com/requirement-analysis-in-sdlc/) Accessed 16 August 2020.

Performance by design. Available: [http://www.1202performance.com/atricles/how-to-write-performance-requirements-with-example/.](http://www.1202performance.com/atricles/how-to-write-performance-requirements-with-example/) Accessed 16 August 2020.

Azure App Service Hybrid Connections. Available: [https://docs.microsoft.com/en-us/azure/app-service/app-service-hybrid-connections#:~:text=The%20relay%20agent%2C%20Hybrid%20Connection,to%20access%20the%20desired%20endpoint.](https://docs.microsoft.com/en-us/azure/app-service/app-service-hybrid-connections" \l ":~:text=The%20relay%20agent%2C%20Hybrid%20Connection,to%20access%20the%20desired%20endpoint) Accessed 6 September 2020.

Azure Fundamentals. Available: [https://docs.microsoft.com/en-us/learn/paths/azure-fundamentals/.](https://docs.microsoft.com/en-us/learn/paths/azure-fundamentals/) Accessed 6 September 2020

What is UML. Available: [https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/.](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/) Accessed 23 November 2020.

Otero, C. 2012, Software Engineering Design: Theory and Practice, Auerbach Publishers, Incorporated, London.

Gennick, J. 2003, Learning UML, O’Reilly media Inc. Available: <https://learning.oreilly.com/library/view/learning-uml/0596003447/index.html>. Accessed 23 November 2020.

The STRIDE Threat Model. Available:  [https://docs.microsoft.com/en-us/previous-versions/commerceserver/ee823878(v=cs.20).](https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)) Accessed 16 November 2020.

Fruhlinger, J. 2020, Threat modelling explained: A process for anticipating cyber-attacks. Available:

[https://www.csoonline.com/article/3537370/threat-modeling-explained-a-process-for-anticipatingcyber-attacks.html.](https://www.csoonline.com/article/3537370/threat-modeling-explained-a-process-for-anticipating-cyber-attacks.html) Accessed 16 November 2020.