ThakurVarshaNalgndaRakesh_2022_finalproject

by Varsha Thakur

Submission date: 26-May-2022 02:27PM (UTC+0200)

Submission ID: 1844583473

File name: 10480_Varsha_Thakur_ThakurVarshaNalgndaRakesh_2022_finalproject_366877_1766210858.pdf (5.25M)

Word count: 20895

Character count: 126656

Kaunas University of Technology
Faculty of Informatics
Department of Applied Informatics

Hotel Management Information System

Final Degree Project (P175B139)
Computer Science (6121BX010)
Specialization: Multimedia Systems

Varsha Thakur
Student

Rakesh Nalgonda
Student

dr.doc. Germanas Budnikas
Lecturer

Kaunas 2022

Kaunas University of Technology
Faculty of Informatics
Department of Applied Informatics

Hotel Management Information System

Final Degree Project (P175B139)
Computer Science (6121BX010)
Specialization: Multimedia Systems

Varsha Thakur
Student

Rakesh Nalgonda
Student

dr.doc. Germanas Budnikas
Lecturer

lekt. dr. Audrius Nečiūnas
Reviewer

Kaunas 2022

Kaunas University of Technology
Faculty of Informatics
Department of Applied Informatics

Hotel Management Information System

Declaration of Academic Integrity

We confirm that the final project of **Varsha Thakur and Rakesh Nalgonda**, on the topic „Hotel Management information system“ is written completely by us; all the provided data and research results are correct and have been obtained honestly. None of the parts of this thesis have been plagiarised from any printed, Internet-based or otherwise recorded sources. All direct and indirect quotations from external resources are indicated in the list of references. No monetary funds (unless required by Law) have been paid to anyone for any contribution to this project.

We fully and completely understand that any discovery of any manifestations/case/facts of dishonesty inevitably results in me incurring a penalty according to the procedure(s) effective at the Kaunas University of Technology.

(name and surname)	(signature)
(name and surname)	(signature)

Kaunas 2022

Kaunas University of Technology
Faculty of Informatics
Department of Applied Informatics

The task of the bachelor's final degree project

1. The topic of the Bachelor's final project:

To create a hotel management system for a small-scale hotel

2. The aim of the Bachelor's final project:

The aim of the system is easy to access to the information about the hotel and to have the best stay at the hotel in order to achieve customer satisfaction and a profitable business.

The system will allow to manage data related to booking, availability of rooms and the services provided at the hotel.

3. The objectives of the Bachelor's final project:

1. To design a websystem which is easy to navigate
2. Automate the room booking process and use AI to provide discounts
3. To have a proper feedback channel

4. The deadlines for submitting the Bachelor's final project:

To the supervisor (*8 work days before the defence at the Department*):	2022 May 22
To the reviewer (*6 work days before the defence at the Department*):	2022 May 26
To the Committee (*3 work days before the defence at the Department*):	2022 June 08

5. Functional requirements for the object or system to be developed:

1. Website Pages
   - Home
   - Gallery
   - Services
   - About Us
   - Testimonials
   - Booking
   - Contact Us : Email to be sent to the provided email address
2. Admin Panel
   - Manage Website Images (CRUD)
   - Testimonials: can delete the feedback if needed
   - Services : Services provided by the hotel (CRUD)
   - Manage Room Categories: Type of the rooms available in the hotel (CRUD)
   - Manage Rooms: Room available in the hotel (CRUD)
   - Manage Booking: See the booking done by staff or add new booking
   - Manage Staff : crerate profile for staff (CRUD)
   - Manage Salary of staff : Enter details of the salary payments done to staff
   - Manage Customers : See and update the customer details

- Statistics of the booking and payment : Graph to show the statistical data
- Discounts : Set up dates for the discount

3. Customer Subsystem
   - Book room : booking engine to book the available rooms from the system
   - Payment : Pay for the booking
   - Register complaint & Feedback : Provide the testimonial which is visible on the website
   - History and Profile : would be able to see the previous actions and the details stored in the database
4. Employee / Staff subsystem
   - Can see the work assigned by the manager
   - Time of working : Last login and logout details will be visible to the admin
   - Feedback for customers: can provide feedback for the customers

6. Non-functional requirements for the object or system to be developed:

1. Security: Password must be hashed. Password should be stored in an encrypted language in the database. Password can be either text or number.
2. Accessibility: Admin should have a separate login page.
3. Administrator rights: The administrator can view as well as alter any information in the system.
4. Employee Rights: Employee is able to see all the information, which is given in the system, even though he/she cannot update it on their own entirely except for few thing
5. Efficient user interface : Number of clicks to perform any operation should not exceed 5 clicks

Supervisor: Germanas Budnikas _____ Date: 2022 Feb 24
(name, last name, signature)

Student: Varsha Thakur _____ Date: 2022 Feb 24
(name, last name, signature)

Student: Rakesh Nalgonda _____ Date: 2022 Feb 24
(name, last name, signature)

Kaunas 2022

Thakur Varsha Nalgonda Rakesh, Hotel Management Information System, Bachelor's Final Degree Project, Supervisor dr.doc. Germanas Budnikas; Faculty of Informatics, Kaunas University of Technology.

Study field and area (study field group): Bachelor of Informatics, Computer Science (6121BX010) Specialization Multimedia Systems (IFU-8).

Keywords: (Information Management, Hotel management system, booking controller, Artificial Intelligence , PHP, Laravel Framework).
Kaunas, 2022. Pages 99.

Summary

This Bachelor's Final Degree Project thesis analysed problems and needs in process of collecting and sorting out some relevant information in the operations of a small-scale hotel. In order to solve these problems and meet hotel management's demand and controlling their website, a targeted Information management system has been developed. In this project, there are some ideas provided for promotions and discounts, managing and optimizing hotel operations and website. Also, Artificial intelligence is used to determine the level of discount that could be offered to the customer based on few factors.

The thesis is composed of four parts: Introduction, Analysis, main body and conclusion. In the section of Introduction, the objective of the project is listed. In Analysis part, analysis of the similar projects already existing in the marking and the technology has been described. In the Main body, all the details about the requirements and implementation details are explained. In Conclusion part, the goals and objectives which listed in the Introduction part of the Hotel Management are described and briefly described about their implementation and result.

Thakur Varsha Nalgonda Rakesh, viešbučių valdymo informacinė sistema, bakalauro baigiamojo laipsnio projektas, vadovas dr.doc. Germanas Budnikas; Kauno technologijos universiteto Informatikos fakultetas.
Studijų kryptis ir sritis (studijų krypčių grupė): Informatikos bakalauras, informatika (6121BX010) Specializacija Multimedijos sistemos (IFU-8).
Raktiniai žodžiai: (Informacijos valdymas, Viešbučių valdymo sistema, užsakymų valdiklis, Dirbtinis intelektas, PHP, Laravel Framework).
Kaunas, 2022. 99 psl.

Santrauka

Baigiamajame bakalauro darbe buvo analizuojamos problemos ir poreikiai renkant ir rūšiuojant tam tikrą informaciją, susijusią su nedidelio masto viešbučio veikla. Siekiant išspręsti šias problemas ir patenkinti viešbučių vadovybės poreikius bei kontroliuoti savo interneto svetainę, buvo sukurta tikslinė Informacijos valdymo sistema. Šiame projekte pateikiamos kelios idėjos dėl akcijų ir nuolaidų, viešbučių ir interneto svetainės valdymo bei optimizavimo. Be to, dirbtinis intelektas yra naudojamas nustatant nuolaidos lygį, kuris gali būti pasiūlytas klientui, remiantis keliais veiksniais.
Darbą sudaro keturios dalys: įvadas, analizė, pagrindinė dalis ir išvados. Įvado skiltyje pateikiamas projekto tikslas. Analizės dalyje aprašyta jau egzistuojančių panašių projektų ženklinimo ir technologijos analizė. Pagrindinėje dalyje paaiškinama visa informacija apie reikalavimus ir įgyvendinimo detales. Išvadų dalyje aprašomi Viešbučių valdymo įvadinėje dalyje išvardyti tikslai ir uždaviniai bei trumpai aprašomas jų įgyvendinimas ir rezultatas.

Contents

List of figures

List of Tables

Abbreviations and Terminology

Sr.no	Term	Meaning
1.	HMS	Hotel Management System
2.	AI	Artificial Intelligence
3.	SaaS	Software as a service
4.	PoS	Point on Sale
5.	OTA	Online Travel Agency
6.	CRM system	Customer relationship management
7.	B&B	Bed and breakfast
8.	IoT	Internet of Things
9.	PMS	Property Management System
10.	PIE	Pricing Intelligence Engine
11.	CRUD	Create Read Update Delete
12.	CLI	Command Line Interface
13.	FR	Functional Requirement
14.	NFR	Non-Functional Requirement

Introduction

The hotel management system was designed for hotels that now handle their operations manually. The administrator can update the hotel information on a daily basis. The hotel's core operations will be automated through this proposed system. The three major users in the hotel management system are the Administrator, Staff, and Customers. Administrators have complete control over the system's operation. The overall purpose of the project is to automate the hotel's daily operations such as assigning tasks to the staff and taking updates on it. The admin has complete access to all system functionalities. Staff has access to limited information.

The Reservation System's objective is to monitor availability and keep track of room and hall reservations [1] . Users may see which rooms are booked and which ones are available using this approach. They can use the reservation module to make a reservation for the hotel. All types of room services are tracked using the Room Management System. Users may keep track of all hotel room data using the room management feature. The user may check the additional services offered such as room laundry service, dining service, and cleaning service using this module. The Inventory Control System keeps track of the hotel's inventory, while information of visitors will be managed by guest management. Everything is monitored by the administrative division. The primary purpose of this system is to computerize all hotel operations.

Check a customer out and vacate the room, as well as compute the bill. This approach makes it simple to check in and leave. When you arrive at the hotel, you may easily check the room availability. When you check out, you may also generate the invoice.

Furthermore to improve and automate the management system, nowadays various technologies such as IoT, Artificial Intelligence, etc are used. Artificial Intelligence can use either supervised or unsupervised learning for the model depending on the requirement and which model suits the need the best.

Purpose and objectives

The main purpose of the project is to automate the hotel's daily operations. You may use this system to keep track of new client admissions and monitor room activities. This approach allows you to check rooms based on client demands and allocate rooms quickly. Checking out a customer from the hotel and updating the system with the room information by simply showing the room as vacant, is simple and clear. When the user completes the checkout procedure, the system will create a final bill after computing all of the invoices. The user may also search the internet for all of the available packages. The user can also make reservations online. They can also use to cancel a reserved accommodation through internet. The list of regular clients and customer reviews may also be examined by hotel management.

The aim of the system is easy access to the information about the hotel and to have the best stay at the hotel in order to achieve customer satisfaction and a profitable business.

The system will allow to manage data related to booking, availability of rooms and the services provided at the hotel.

The foregoing are the project's objectives:

1. To design a web system which is easy to navigate
2. Automate the room booking process and use Artificial Intelligence to get the discounts
3. To have a proper feedback channel to improve the performance of the staff and system

Structure of the document

1. Analysis
   Analysis of similar technologies present in the market and their advantages and disadvantages.
2. Implementation
   Diagrams to show the implementation of the system.
3. Testing and realisation
   Realisation of the project along with test cases description and their results.
4. Conclusion
   Conclusion of the projects and summary of the goals.
5. References
   List of references used while writing the report and for analysis of the systems.

1 Analysis

1.1 General analysis

The current manual system uses work and direct human language communication orally to manage the hotel [2]. This delays information transmission in the building. Booking is completed through phone calls or through visits to the hotel booking office. The guest's personal details like Name, Age, sex, number of days, area unit input throughout the booking. The booking office orders for the preparation of the guest's space before his/ her sign-on date.

Traditional hotel administration does not incorporate all services, such as the consumer's location, the amount of the reservation payment, and what they are doing right now. The hotel is dealing with several challenges that are hurting its daily operations, such as reliability, performance, efficiency, and effectiveness [3].

The documents area unit is transferred manually to the filling department for compilation of the guest's file. On the check-in date of the guest the key is given to his/her assigned space, he or she specifies if needed any room service. The assistant hands over the guest's file to the controller on the subsequent table. The Customer's tab is updated on the routine of his extra expenses if any additional room service is taken. The accounts department generates the bills on commonplace and delivered them to the guests in their rooms by the service maids.

The guest pays at the accounts table, where the receipts area unit is generated. throughout trying out of guests, their expenditure outlines area unit generated day by day before they take a look at the date. The guests receive their outlines at the accounts table as they fight, where they acquire bills balances if any.

With the growth of the tourism business, the hotel industry's competition is growing increasingly fierce as the days are passing by [4]. The effective integration of all types of resources within the hotel can improve labour efficiency, lower transaction costs, and increase the hotel's operational profit. Furthermore, management innovation is a key aspect in determining the hotel's competitiveness. As a result, the concept of a catering management system is proposed. The installation and maintenance of the background database, as well as the development of the front-end application, are the major components of the catering management system [5]. The system necessitates the creation of a database with high data consistency, integrity, and security. Simultaneously, the application's functionality should be complete and also easy to use.

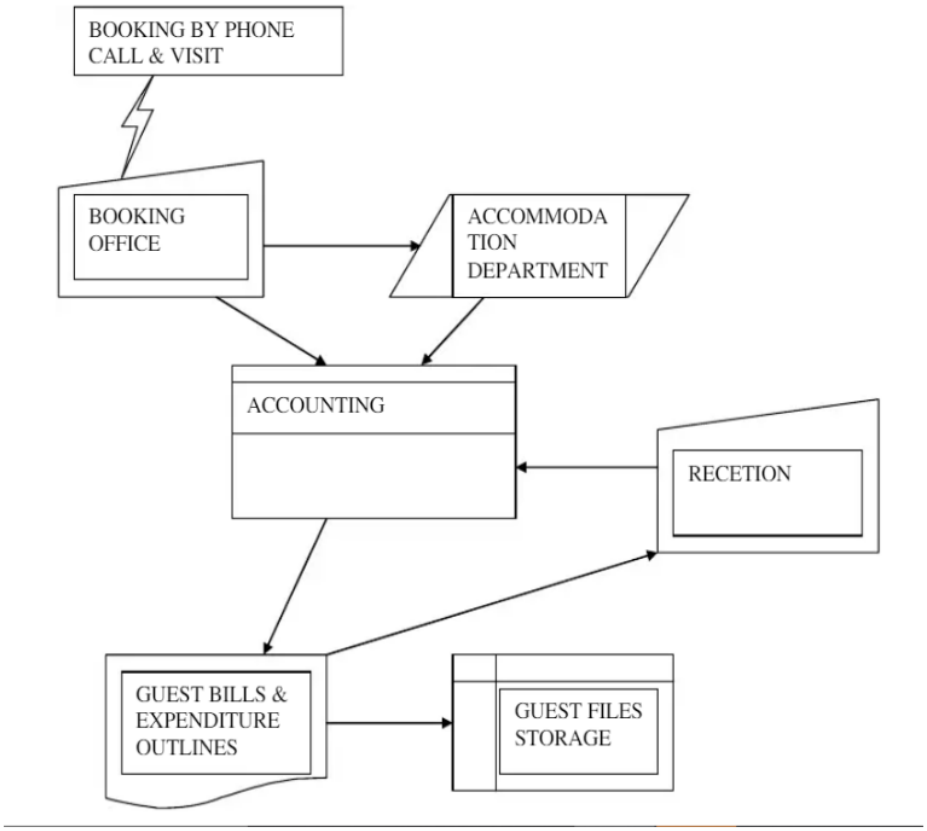


Fig. 1.2. Traditional booking system [6]

Traditionally in hotels all the bookings and accountings were done by person which lead to many problems. Maintaing manual management system is very difficult. If the hotel took booking round the clock and different managers/receptionist were there to attend the guest, it used to lead to errors. Sometimes overbooking caused problem for both the hotel and the clients.
Another big issue was the storage of information about the customer and payments. Paper invoices could get lost and may lead to calculation error at the end resulting in loss for the hotel.

1.1.1 Pros and cons in the manual system

As per our understanding of the manual system following are the list of pros and cons.

- Difficulty in finding the location of guest files:
  Due to associate degree oversize number of guests' files, location of guest files throughout checking in, change of daily expenditures, receipt generation and making a check out is extraordinarily tough for the hotel workers.

- Large storage space occupancy for files:
  The physical files occupy excessive quantity of area of about 2 rooms stuffed with storage cupboards. This occupies the hotel's area which may have otherwise been used for financial gain generation by the hotel.
- Human and machine errors:
  Many errors enabled by the system due to tedious computations needed throughout data processing cost the hotel management heavily
- Poorly generated records:
  Poorly generated records encourage the employees to omit some necessary important data. This lands up in security issues at the hotel like armed robberies [2].
- Visitors' complaints:
  Many incidents of overcharging and charging for services that were not used by the guests were reported due to poor record-keeping inspired by the manual method.
- Ineffective communication:
  Guests are inconvenienced due to a lack of communication between departments. They are frequently presented with services that they did not request [7].
- Data analysis problem:
  In most cases, accountants found it difficult to analyse the guests' data due to the absence of some records, data was lost during the production of expense invoices.
- Reservations made by customers are inefficient.
- They are unable to market the services that they would like to offer.
- The hotel's information is unavailable.
- It's difficult to create reports.
- In the collection and processing of customer data, there are delays in information.

1.1.2 Advantages of manual system

- The manual system does not necessitate any special computer capabilities on the part of the employees [8].
- There is no reliance on computer devices, which, by their very nature, are prone to failure.
- Operation cost is quite low, since system do not necessitate internet access or any kind of energy, as a computerized electronic system would.

1.1.3 Disadvantages of the manual system

- Guest files are easily misplaced or mixed up with other papers in the guest file.
- Files take up a lot of storage space.
- Duplication of data is created unnecessarily.
- Due to insufficient data security levels and standards, files are vulnerable to theft and illegal change.
- It is quite difficult to retrieve guest records.
- Errors are common during the data entering the process.

- Modifying guest records is exceedingly tough because it results in filthy and unpresentable reports.

1.2 Guest Experience

On a large scale, the goal of hospitality management is to create great guest experiences. Hotel businesses, whether they are a luxury four-star boutique hotel, a massive multi-national hotel chain, or a modest B&B off the main path, live and die by their reputations (word of mouth and online). Whether you're running a premium four-star boutique hotel, a major multi-national hotel chain, or a budget B&B off the main path, delivering surprise and delight experiences at scale necessitates the use of technology [7]. Although the appearance and functionality of such technology vary depending on the client, location, and chain size, it is the glue that holds visitors and employees together.
Using guest messaging software, a specified hotel guest-facing app for mobile devices, in-room tablets, or possibly voice-activated gadgets as glue can all be done [9]. In the end, the hotel industry is all about delivering convenience and making tourists feel at ease.

1.3 Revenue Management

Over the last 5-10 years, the revenue management industry has blossomed with innovative software applications [10]. Any revenue management tech stack must have central reservations systems and revenue management systems [11]. Hotel pricing, on the other hand, is exceedingly complicated and necessitates flawless coordination across numerous teams in order to manage inventory and rates over hundreds of channels. OTAs and wholesalers, usually regarded as industry foes, have disintermediated the online booking journey by providing distribution in times of need while often ripping partners on commissions in exchange. At the end of the day, revenue management is all about matching the right guest to the right room at the appropriate price, and there are literally billions of data points to evaluate.
The revenue management tech stack also includes the following essential tools [12]:

- Channel manager: By syncing inventory across hundreds of third-party OTAs, channel management may increase distribution, increase occupancy, and avoid overbooking.
- Upsell software: increase income per guest by selling upgrades, experiences, and other items automatically throughout the guest journey.
- Market intelligence software: the days of revenue managers making sound price decisions based on weekly STR reports are long gone. Markets change by the second, and solutions like Rate Shoppers give you the visibility and analytics you need to stay on top of them.
- Business intelligence software: Do you know how profitable each market segment is, broken down by geos and duration of stay at your hotel? Most likely not. Business intelligence software allows you to cut through the clutter and produce meaningful insights that can help you steer your business strategy.

1.4 Hotel Marketing

Convergence and direct reservations have been driving hotel marketing commercial activities in recent years. Convergence refers to the premise that hotel marketers must be skilled in not only marketing but also sales, distribution, and revenue management in order to maximize hotel bookings. The rising reliance on software solutions to manage all of the complications, such as distribution assignments across online travel brokers, has resulted from the convergence of commercial roles (13). Despite the diverse nature of hotel marketing positions, the primary goal of hotel marketers is to increase direct reservations. These bookings could come through metasearch systems, online review sites, Google and other search engines, or even social media.

This software area is all about how to bring those guests into your direct booking flows and convert them. It's a good idea for marketers to start from the very base of the pyramid and climb their steep. How much of your website traffic is converted by your booking engine? If the stats look good, go to your website and see how much traffic is being directed to your online booking engine. Once you've cleaned up those sections, start searching for ways to drive additional visitors into the funnel, whether it's through your hotel CRM system, email marketing campaigns, or TripAdvisor ad bidding. Eventually, the most crucial thing is to select a fantastic hotel marketing firm that can provide not only technical solutions but also assistance and insights as a true long-term business partner.

1.5 Impacting factors: Market Scenario Analysis

In the future years, the need for SaaS-based hotel management software is likely to move the hotel management software market ahead(14). The hospitality industry's need for SaaS-based hotel management software is driven by its expenditure due to lower installation and maintenance charges. Furthermore, the rapid growth of the hospitality and tourism industries contributes significantly to the growth in the sector of the software made for hotel management. Hotel owners are concentrating their efforts on implementing and forming new strategies and solutions that will increase and also preserve operational efficiencies as well as client engagement.

However, the market's growth is limited to some extent by the difficulties of the software system's integration and the scarcity of technically trained individuals. Increase and boom in upcoming technologies in this era of computer and technology like (internet of things) IoT, machine-learning, AI (artificial intelligence), neural networks on other hand, are predicted to expand the market for cloud-based hotel management software. Hoteliers must ensure that their properties have an internet presence in order to attract customers who proceed to book directly through hotel websites. Travelers are increasingly using the internet to research and schedule their trips. Room availability and rates are directly taken from HMS, reducing the error to book twice and another difficulty for Customers who wants to stay at the hotel.

1.6 Feasibility Analysis

The structure of hotel management information systems is examined primarily through the application of feasibility, technical, and economic factors. Since every hotel has their own needs and

requirements most of the existing systems may or may not fulfil the requirements. It could be said that generalised system cannot be applied for targeted market. Hence before building a software identifying these specific needs and requirement needs a serious analysis (15).
The feasibility analysis employed in the hotel management information system was primarily focused on the operator's current status and environmental factors. In the operation scenario analysis, the hotel-specific segment's attitudes and understanding regarding management information system-specific tasks, such as its technology or section of systems believed not needed or hesitant to use, would surface as a series of obstacles. As a response, give extra attention to certain sectors while performing a feasibility study, especially mission-critical jobs like hotel strategic planning, department manager knowledge, and system orientation. Furthermore, when a system employs feasibility study, it should consider the environment, particularly unexpected conditions in business operations, such as facilitating customer registration and arranging a group of customers.

1.7 Technical Feasibility

Environmental adaptability, reliability, maintenance, scalability, Information processing technology, input/output, and response time, reliability, security among other things, are all aspects that go into evaluating whether technical circumstances can successfully carry out development work, fulfil the demands of software and hardware developers, etc.
It is a database server that uses the MySQL database to handle enormous volumes of data while ensuring data integrity and providing numerous management options that are advanced in nature. Versatility, security, and ease of use of the server make it ideal for the development of database.
As a result, the software platforms for systems are mature and viable, and it could be said to be technically viable.

1.8 Economic Viability

Viability in Economic sector study focuses on analysing and assessing the funding needs and capital efficiency of a building project, includes budget projections, investment payback duration, the input-output ratio, analysis of investment benefit, etc. The cost of development and maintenance of this hotel information system is borne by the hotel, but as a key tool for hotel management, it can drastically improve multiple factors such as department effectiveness, and service quality, business volume, greatly outweighing the initial expenditure (16). As a result, the hotel's management information system selection has a high investment return and good economic feasibility.

1.9 Business Requirement Analysis

- Customers can make reservations via a variety of methods, including the phone directory, the internet, and reservations at the reception.

- The system must be able to categorize room management and offer available price, book discount scheduling function settings such as cheapest cost, simple to meet single guest, and group bookings.
- Keep track of the present status of things, like cleaning the checked-out room and determining whether the visitor stayed. In order to achieve unified administration of the hotel information, example ID numbers for customer queries, based on check-in dates, changes to the contact information of guests. You must provide a customer information management function based on client occupancy information.

1.10 Principle of Database Design

The main goal of hotel management information is to handle information requests resulting from massive volumes of data, which involves massive data storage and administration [17]. As a consequence, developing solid databases and data structures may assist the entire system in managing data more simply, quickly, and accurately, which is one of the major indicators of whether or not the development of a hotel information management system is progressing well. A strong data structure and database should accurately reflect changing conditions, fulfil regulatory requirements at all different levels of the company, and make the following system development work easy, rapid, and inexpensive to maintain. In order to better organize your data, the system in the design typically follows the following principles and is developed to meet the database's real demands. The system is built using the database design concepts listed below:

- By assuring the database's consistency and integrity, data consistency and integrity rules prevent data redundancy. We don't indicate the properties in the database design field since the primary key links with the child table. Associating tables is a coercive approach that adds overhead to the child and parent tables' insert, update, and delete actions.
- Regulatory requirements: Make sure that data is evenly distributed among tables while developing, managing, and maintaining your database. Database normalization is implemented if the table designs are correct. Using the right data structure and database access makes it not only simple to use, but may also greatly simplify other application components like questions, reports, forms, code, etc.

1.11 Similar System Analysis

Several online hotel management software listings wrongly group all hotel software together, while the hospitality sector presents unique difficulties that require distinct point solutions to solve diverse concerns. All-in-one property management software is usually enticing to small hotels.
Your on-property personnel can offer outstanding (and consistent) surprise and delight experiences for guests using hotel operations software. The most prevalent tool in the operations stack is the hotel property management system, sometimes known as front desk software in smaller independent hotels. A PMS, on the other hand, is considerably more. If you're utilizing an out-of-date Hotel PMS system, it's practically hard to properly utilize other IT solutions. (17)

Other important tools in the operations technology stack are:

- Employee participation software: Using staff collaboration software, you can keep your company in sync and streamline internal communications.
- Preventive maintenance software: Preventive maintenance software helps capital investments last longer.
- Housekeeping software provides real-time communication and analytics for one of your property's most crucial employees (especially in the age of COVID!).
- Visitor survey software collects information on the visitor experience that may be utilized to enhance all parts of your hotel's operations.
- Labor-management system: Increase the profitability of your property by improving the way you staff it.
- Concierge software: Using concierge software, manage hundreds of visitor requests during shifts to guarantee your team never misses a beat.

1.11.1 System 1: Cloudbeds HMS

Cloudbeds is a simple-to-use hotel management system that provides hoteliers and hosts with an unified strategy to help them operate their business more efficiently. With Cloudbeds, you only need one login and one platform that is always updated [19]. With over 300 connections, our portfolio comprises a PMS, Booking Engine, Channel Manager and Marketplace, Revenue Management Tool, and Payments. Thousands of customers across more than 150 countries nation world-wide trust the solution, which helps companies acquire more reservations and happier clients.

The firm was established in 2012 and is headquartered in San Diego, California. The firm now employs 553 people.

Cloudbeds is the world's fastest-growing welcome management software package bundle, including features for managing properties of all types and sizes. The Cloudbeds suite comprises a robust PMS, a number one Channel Manager, a conversion-driven Booking Engine, and integrated Payments, and is trusted by over 22,000 hotels, hostels, inns, and other lodgings in over 157 countries. Cloudbeds enables properties simplify their operations, boost income, and adjust procedures with confidence and convenience because to the marketplace's 300+ connectors. Cloudbeds was founded in 2012 and has grown to over 400 members with local groups in forty countries(18).

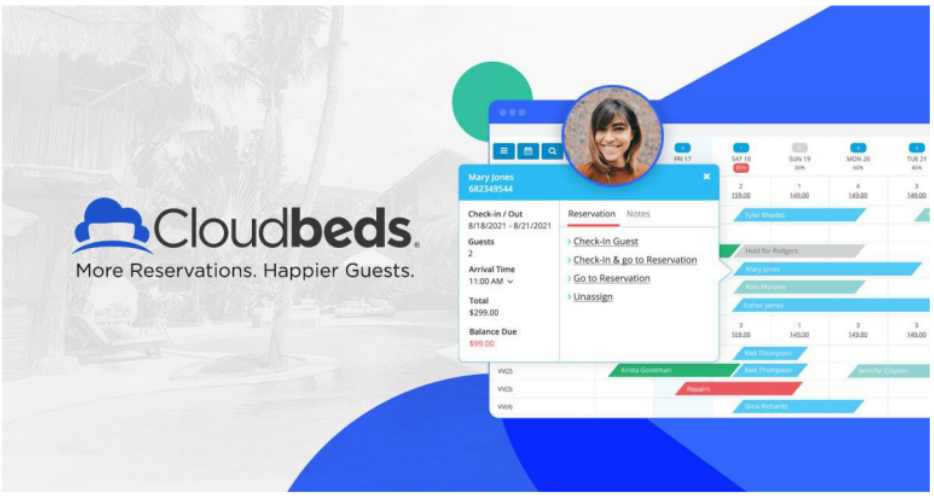


Fig. 2.1 Cloudbeds HMS(19)

Features of Cloudbeds HMS:

- Multi-currency
  The company accepts worldwide currency as payments from clients.
- Multi-lingual
  The website allows the user to select the desired language in the preferences.
- Channel-Manager
  Majority of the bookings come from a single source, a sudden surge in demand will be disastrous. Getting the hotel to as many potential consumers as possible is the simplest way to diversify demand. The Cloudbeds channel manager does precisely that: it distributes the inventory over dozens of online travel agencies and meta-search websites, allowing you to specify channel-specific rates so you may prioritize the most profitable channels, such as those with high consumption and low fee. With such visibility, you may generate money from countries you might not otherwise be able to contact
- Front-desk Software
  Guaranteed property might be a major benefit in a system like CloudBeds, which is an integrated property hotel management system. By communicating properly with booking engine and channel manager, you can minimize overbookings and lost reservations.
  The Cloudbeds is very simple and easy to use: from the centralized calendar read, merely drag and drop reservations to monitor in and out. To regulate reservations, click on the calendar. Every reservation is tied to a guest profile, that puts guest information at your staff's fingertips so that they will track preferences and acknowledge repeat guests. On the selling aspect, the PMS automates your guest communications, therefore you'll send pre-stay emails (with upsells) and post-stay review requests to create your online name.
- Booking Engine

The Hotel website is the calling card, it's your online hoarding showcasing everything that creates your property distinctive. It's an important part of your hotel's identity. With a booking engine that turns lookers into bookers, you'll cut back discharge to OTAs and capture additional commission-free direct bookings. This conjointly implies that you own the guest relationship. Therefore, you'll do things like send scheduled pre-arrival emails with property data and potential upsells (room upgrades, in-destination activities), yet post-stay emails to encourage additional reputation-building online reviews.

Another valuable practicality of a hotel booking engine: works on smartphones and tablets in order that your property can be reserved across all devices; promo codes therefore you'll optimize digital selling for conversions; upsells among the booking flow to boost your total revenue per on the market area and integrated payments, therefore, you'll stop taking payments manually.

- Revenue Management in Pie format

  Smaller properties typically haven't got the budget or have to be compelled to deploy a full-featured revenue management resolution, like ideas or Duetto. However, that doesn't mean they can't have the benefit of one! Cloudbeds puts rating automation and business intelligence within the hands of hotels of all sizes with its rating Intelligence Engine (PIE).

  Price automation is powerful. Instead of manually setting the rates in response to property, market, and challenger dynamics, PIE offers rule-based automation. You set a rule -- like increasing your rates by 100 percent once you hit the eightieth occupancy -- and therefore the system can be in the off state, you are out of the workplace. or else, you'll be able to set the system to trigger a notification once a particular threshold is met, like your challenger dropping its rates by 5 percent. Next time you log in, you'll be able to review these alerts for potential action.

  PIE conjointly offers hoteliers access to a strong knowledge dashboard, which mixes knowledge from your HMS with native market knowledge. you will be ready to track and analyze vital knowledge.

- Hotel App Marketplace

  Even with a fully integrated system, you will realize different solutions that you just wish to integrate with the hotel management system. The most advantage of direct integration is seamless information synchronization. Let's say, as an example, you would like to send a name builder email at check-out. while not Associate in Nursing integration, you'd then need to pull that data from your PMS.

  It makes an enormous distinction once these forms of things can be controlled -- regardless of however busy you're.

Cloudbeds HRM is ranked #1 out of 53 Hotel Management Software.

1.11.2 System 2: RoomRaccoon HMS

The RoomRaccoon Hotel System Includes a property management system with a variety of built-in options, an integrated booking engine that captures 40% more direct bookings on your website, and a

channel manager that divides rates and availability across multiple booking channels - everything you need to run a successful business (20).

The company was founded in 2017 and its headquarters is located in Breda, Netherlands. Currently, the company has 35 employees.

RoomRaccoon is the world's most comprehensive cloud hotel management solution well-liked website by over 1500 accommodation businesses across the world. From one platform, RoomRaccoon provides all the tools you wish to manage each side of your property, together with a powerful property maintenance system (PMS) with a built-in booking engine and channel manager. No additional complicated integrations and awkward stacks. Simply, automation exactness. What makes RoomRaccoon stand come in the jam-packed tech space is our variety of inherent revenue-generating options that job to extend your RevPAR at each client touchpoint. With inherent upselling and yield management tools, you'll have additional management of your property's revenue by doing so much less. They recently launched in North America, wherever we glance forward to partnering with several forward-thinking hotels, hostels, guesthouses, and lots more! If you'd prefer to get connected with a neighborhood member of our team, merely leave your details and we'll get connected to indicate to you the way to become additional economical and increase the guest expertise.

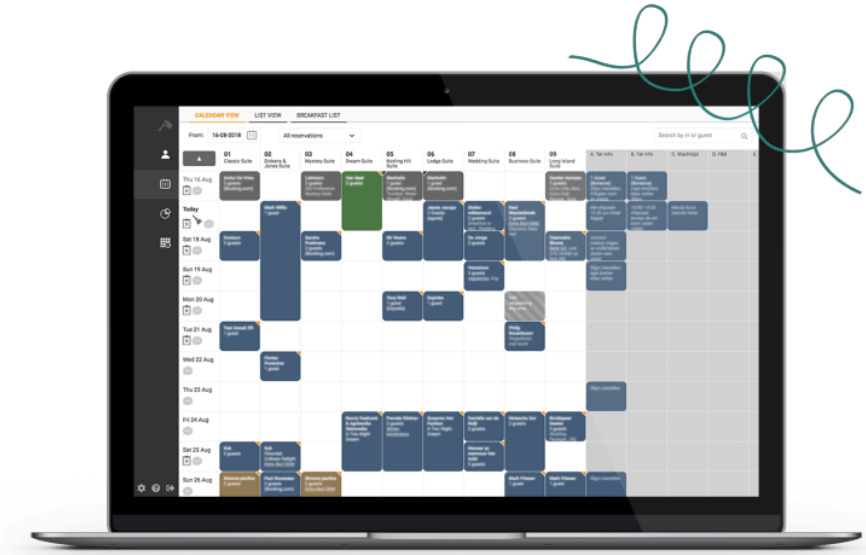


Fig. 2.2 RoomRaccoon HMS Dashboard(21)

Features of RoomRaccoon:

- Multi-currency
- Multi-lingual
- Channel Manager
- Property Management System

- Booking Engine
- Website Development
- Point of Sale
- Group Booking Engine
- Pricing Intelligence
- Gift Vouchers
- Housekeeping
- Revenue Management Module
- Rate Shopping Module

1.11.3 System 3: Little Hotelier

Little hotelkeeper is the one building package that makes it simple to manage your property, attract extra visitors, convert direct reservations, maintain your website, and accept on-line payments. It was created from the ground up to meet the needs of small companies. We provide you with all of the tools you need to manage your business your way, including a mobile app that you can use on the move. Our platform is simple to use, quick to set up, and support is always only a click away.
The company was founded in 2012. Currently, the company has 700 employees (22).

- It is simple to know and use.
- Very clear and undemanding to use, assistance is continuous without delay offered.

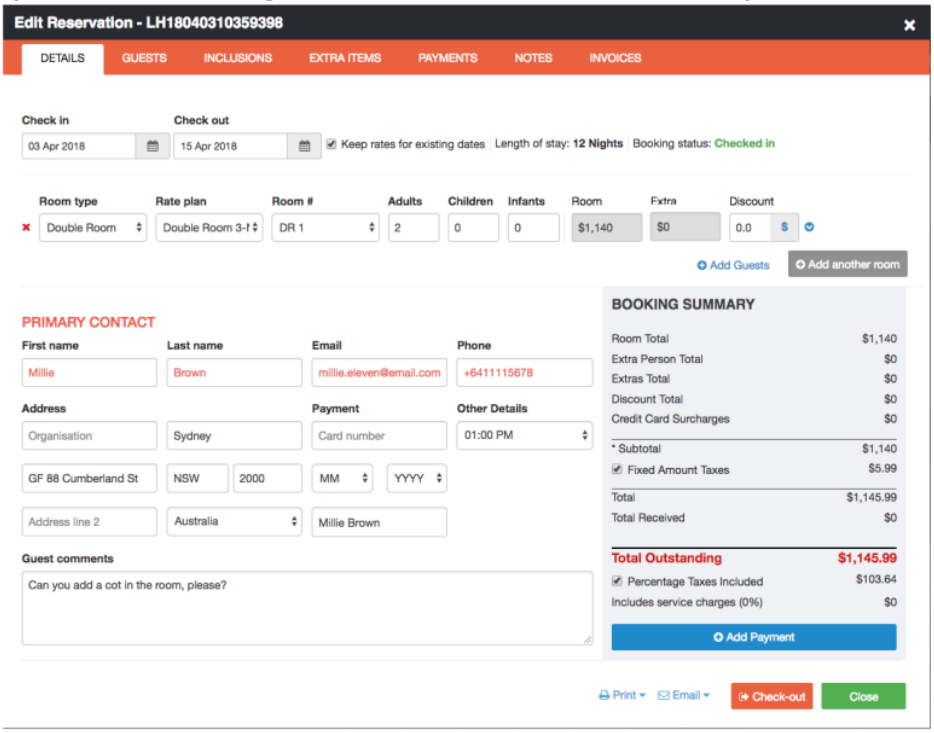


Fig. 2.3 Little Hotelier Interface(23)

Featured of Little Hotelier:

- Pricing Intelligence
- Housekeeping
- Native Email Marketing
- Rate Shopping Module

1.11.4 System 4: RMS Cloud HMS

RMS provides scalable cloud technology that is trusted by over 6,500 properties in sixty countries to manage, run, and expand their hotel lodging companies. RMS has knowledge and insight influencing the continual growth of a robust and integrated platform as an innovative market-leader for over 35 years. Operators can increase revenue, contour operations, and communicate with and keep loyal consumers with the entire range of native alternatives and widening offering. Our goal of ongoing improvement and client knowledge position RMS as a major partner with properties of all sizes – as well as teams looking to optimize multi-national organizations – thanks to our creative approach. Our devoted multinational groups strive to make the firm the world's fastest growing property management software (24).

The company was founded in 1983 and its headquarters is located in San Diego United States. Currently, the company has 145 employees.

RMS makes it straightforward to manage the front and back house operations by using dashboards that suit completely different property varieties. They need dashboards and widgets for reportage and charting that create their PMS software one amongst the foremost intuitive for business choices for hoteliers.

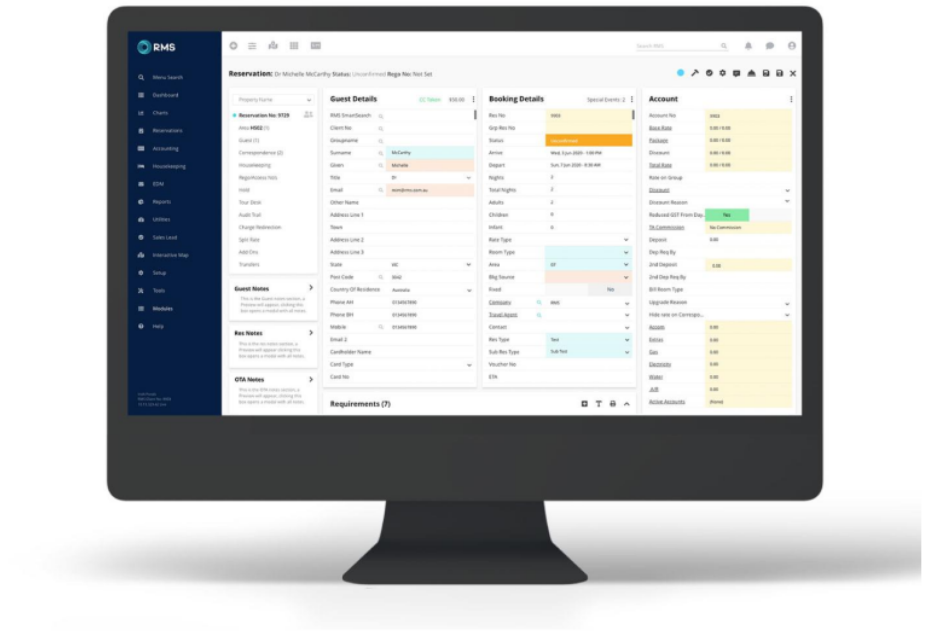


Fig. 2.4 RMS Cloud Dashboard(25)

Features of RMS Cloud:

- Multi-currency
- Multi-lingual
- Channel Manager
- Property Management System
- Booking Engine
- Website Development
- PoS
- Group Booking
- Gift coupons
- Feedback Management from customers
- Email Marketing using Native Content

1.11.5 System 5: innRoad All-in-1 HMS

Thousands of hotels throughout the world rely on innRoad, a successful hotel management software. InnRoad was created exclusively for independent hotels to help them save time, increase occupancy, enhance income, and give visitors with a simple booking experience [24]. Reservation management, a high-converting booking engine, channel management featuring links for 100's of Online travel

agencies, credit-card processing, and revenue management tools are all included in innRoad's all-in-one solution. When compared to their old system, hoteliers employing innRoad have noticed a 20% increase in direct bookings and a 10% increase in room income. Plus, with innRoad's in-house MasterCard process, clients have saved a mean of $57/month on process fees. Customers boast regarding innRoad's 24/7 live client support team and dedicated implementation advisors. With free information migration for each past and future reservation, innRoad can leave no information behind. don't be concerned regarding extra charges, once signed up with innRoad there aren't any hidden fees, no up-charges, and no fine print. Bonus options enclosed in innRoad's edifice management code include daily performance reports, surge alerts, mobile optimization, monthly feature releases, and enlarged security. Able to create the switch to innRoad's all-in-one solution?

The company was founded in 2007 and its headquarters is located in big apple, America state. Currently, the company has 137 employees.

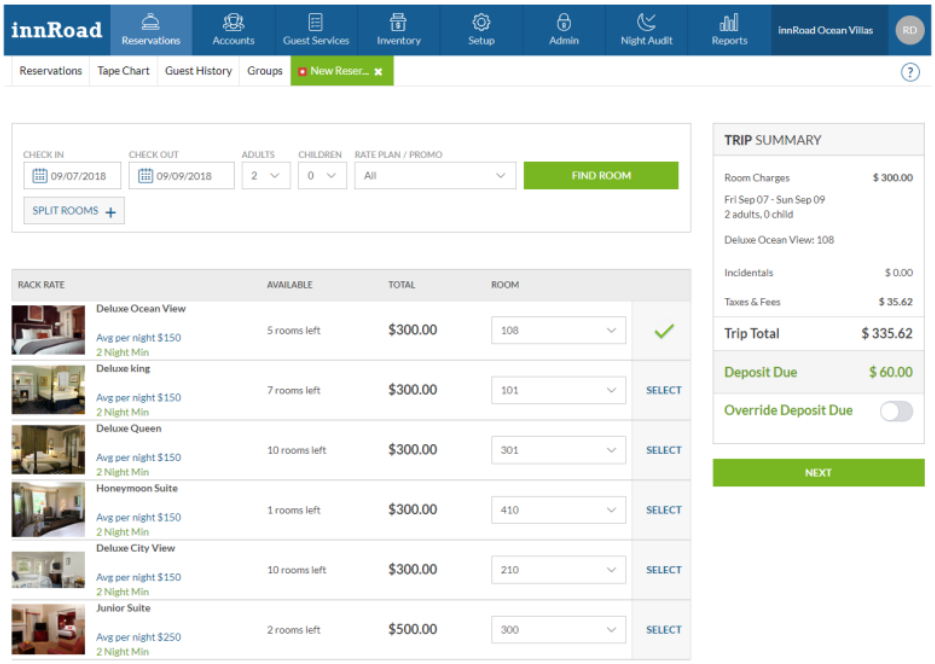


Fig. 2.5 innRoad HMS(26)

Features:

- Multi-currency
- Multi-lingual
- Channel Manager
- Property Management System
- Booking Engine
- Website Development
- Group Booking Engine
- Pricing Intelligence

- Gift Vouchers
- Housekeeping
- Native Email Marketing
- Revenue Management Module

1.12 System Comparison

This tables depicts the comparison between the proposed system and the competitors who already have their strong hold in the market (27)(28)

Table 2.1 Hotel management system comparison

Criteria	**Cloudbeds**	**RoomRaccoon**	**Little Hotelier**	**RMS Cloud**	**innRoad**
Ease of Use	4.5 / 5	4.2 / 5	4.5 / 5	4.2 / 5	4.7 / 5
Ease of Setup	4.5 / 5	4.1 / 5	4.5 / 5	4.4 / 5	4.7 / 5
Ease of Admin	4.5 / 5	4.9 / 5	4.5 / 5	4.5 / 5	4.6 / 5
Time, Attendance,and PTO	4.4 / 5	4.4 / 5	4.7 / 5	4.8 / 5	4.9 / 5
User, Role, andAccess Management	4.6 / 5	4.5 / 5	4.7/5	4.4 / 5	4.6 / 5
Dashboards	4.5 / 5	5 / 5	4.6 / 5	4.3 / 5	4.4 / 5
Entry level price	$10 - $15/room/month	$3-$6/room/month	Less than $500/month	$3 - $6/room/month	Less than $500/month
Implementation	4.4 / 5	4.9 / 5	4.2 / 5	4.1 / 5	4.4 / 5
Functionality	4.5 / 5	4.5 / 5	4 / 5	4 / 5	4.5 / 5

1.13 Review of the Technologies to implement the proposed system

1.13.1 PHP Laravel

We have chosen PHP Laravel as our technology to develop this website is because we have used this framework for almost 2 years for our university projects, so we are more familiar with this technology and to discuss about what it has to offer:

- Laravel outperforms other web frameworks when it comes to developing web applications quickly.
- With clear and reusable code, Laravel aids website programmers in streamlining the development. It's one among the few frameworks with the agility and abilities needed to build websites and online apps.

Technology Advantages (29)**:**

1. Model View Controller (MVC):
    - The model-view-controller (MVC) architectural paradigm is included with Laravel. It is simple to use, making it an excellent choice for creating large-scale or small-scale company websites.
    - MVC can help you work more efficiently by simplifying your coding structure.
    - While working on big projects, MVC support makes it easy to locate the files in their logical folders.

2. Fantastic Layouts with A Template Engine:
    - Laravel offers an amazing template engine called as Blade engine.
    - Blade templating engine offers light weight, powerful and some cool pre-installed template engine that makes the development process smooth.
    - When Blade templating engine is used it smoothly displays the data and extensive layout without slowing down the speed of the application.
    - Blade provides its own kind of structure to help create and view a file that includes conditional statements and loops.

3. Artisan CLI for Quick and Extremely Simple Commands:
    - Laravel offers its own Artisan CLI which helps in making the development simple, straightforward, and fast.
    - It builds apps using the powerful Symfony Console component.
    - It provides the tools to migrate data, manage databases.
    - The Artisan tool optimizes the tedious and repetitive coding processes that developers had to do manually earlier.

4. Eloquent ORM for Easy Interaction with Database:
    - Eloquent ORM by Laravel is just fantastic. It allows you to retain a simple interface with the app database objects utilizing an eloquent or expressing syntax.
    - ORM tool of Laravel is one of the best tools in the market, as it provides the ability to perform database quires with simple PHP syntax which in term saves lot of time as there is no need to write complex SQL code.

5. Proper App Testing Options:
    - Laravel includes a collection of functionalities for using PHPUnit to run unit tests on both online and mobile apps. It created the phpunit.xml file automatically for web development unit testing.
    - It performs tests on features and unit test.
    - Unit tests are used for short coding sections. Laravel can execute numerous unit tests at once to ensure that all the new changes are thoroughly tested.
    - Features test are for bigger codebases that contain lot of objects.

2 Project

The vision behind the project was to make it simple for the small-scale hotel to manage their website and the booking facilities and ease of communication among the customers and staff and admin.
Standardize user activities and increase system quality with thorough project planning and a clear system structure, functional requirements, and non-functional needs.
This project implies the relevance of a dynamically updating websites and the controls for admin. The unique feature which is different from other systems is the “feedback” system. In this project, it has been tried to implement and promote a proper channel for feedbacks. In traditional systems, the customers can provide feedback for the hotel, and it is displayed in either the form of point system or strings on the website. But with this project the employees also have the rights to put feedback on the customer.
Furthermore, Artificial intelligence model has been used to provide discounts on the payments. To get the coefficients, synthetic data has been used. To validate the data 10-fold cross validation method was used.

2.1 Distribution of work

The tasks are distributed according to the subsystems

Table 2.1 Distribution of Work

Task	**Given in percentage of work done (out of 100%)**	
	Varsha Thakur	**Rakesh Nalgonda**
Set up the project	20	80
Design database	60	40
Login and signup	-	100
Email sync	-	100
UI designing	30	70
Website creation	70	30
Room booking functionality implementation	100	-
Additional services list and management	20	80
Implementation of customer login	100	-
staff management	-	100
Implementation of salary page	-	100
discounts algorithm	100	-
Payment gateway implementation	20	80
Creation of employee subsystem	-	100
Creation of customer subsystem	100	-
Analysis	40	60
Testing	60	40
Conclusions	50	50

2.2 Functional Requirements

A Functional Requirement describes the functionality that the code should provide. A function is just the inputs, behaviour, and outputs of a software system. It can be a either manipulation of data or user interaction, or some business process, computation, or any else function that determines what a system do (30).
The basic idea is to distribute system into three different subsystems named and a website

1. Admin subsystem
2. Employee subsystem
3. Customer subsystem

2.2.1 Website Pages

1. Home Page
2. Gallery Page
3. Services
4. About Us Page
5. Testimonials
6. Booking Page
7. Contact Page

2.2.2 Admin Subsystem

1. Login as admin
2. Forget Password/Reset Password
3. Manage Website content
4. Home Page Content
5. Gallery
6. About Us Content
7. Contact Enquiry
8. Testimonials
9. Services
10. Manage Room Categories: The system allows admin to categories the different range of available rooms in the hotel and set the price.
11. Manage Rooms: One of the capabilities that this page provides is room management. Using a front-desk module, the front-office manager can access room status and up-to-date information about all reservations, both current and upcoming. With the help of this feature, room status should be updated quickly. The front-desk module allocates rooms automatically and facilitates a room change.

12. Manage Bookings: The system checks room availability and status, shows free rooms. This function monitors double bookings and allows group reservations. Then it schedules bookings and displays information about current and upcoming bookings on a dashboard.
13. Manage Staff (Housekeeping): The main function of this page is staff management and property maintenance. Manage staff functionality includes management of room status, maid assignment for room cleaning based on a block or floor location, keeping lists of tasks for housekeepers.
14. Manage Salary of staff: This page allows the admin to see the salary of the staff and look at the feedbacks of the customer on the staff experience to provide bonus.
15. Manage Customers: This page allows the admin to fetch the information about the customer from the database and edit the details.
16. Manage Payments of Customers: In this page the admin can get the data of the services that the customers have requested for and price of the booked room. The payment can be done either in the means of debit/credit card or cash.
17. Statistics of the booking and payment: The admin can see the inflow of booking in the form of a statistic pie chart so that the company can take actions to improve the quality and customer satisfaction.
18. Promotions and discounts: Automated offer of promotions for specific facilities in a hotel based on holidays, selected dates, customer profile, etc. Administrative panel has to be used for setting the rules for promotions.

2.2.3 Customer Subsystem

1. Login as customer
2. Book room: This page shows the customer the available rooms. The customer can easily find a free room/resource for booking with intuitive calendar showing room availability side by side. Once booked email confirmations are sent to the customer.
3. Payment: The Payments can be easily done online or can be paid with cash on the front desk.
4. Register complaint: This page allows the customer to register a complaint regarding the behaviour of the staff or any electronics malfunctioning.
5. Feedback: The customer can leave feedback after his experience in the hotel and the services provided during his stay. This will help the company to make changed in the quality of services and staff.

2.2.4 Staff/employee Subsystem

1. Login as staff
2. Can see the work assigned by the admin: Once the staff logins to the system, he/she will be shortlisted with the work assigned for the day.
3. Update the status of work: Staff will be able to update the status of the work whether it is done or still in progress.
4. Time of working: From the time the staff will login into the system the time will be displayed to the admin and same with the logout time.
5. Feedback for customers: The staff can leave feedback about the customer.

2.2.5 List

Table 2.2 List of Functional requirements

Functional Requirement sr.no.	**Description**	**Subsystems involved**
FR1	Login: Login page needs to be implemented to gain access to the particular subsystem	Admin, Customer, Staff
FR2	Forgot/Reset Password: In case the user forgets the password, this function helps to reset the password	Admin, Customer, Staff
FR3	Register: New user registration in the system	Admin, Customer
FR4	Logout: To logout of the particular subsystem and redirect to the home page	Admin, Customer, Staff
FR5	Website content: Website content should be dynamically updated once the admin changes it from the system	Admin
FR6	Contact Us: Any web user is able to send any email to the email account registered in the system.	Website
FR7	Dashboard: Statistics of the systems are shown in form of graphs or counters	Admin
FR8	Room Type: Details about the room type can be added to the system and can be updated or deleted by the actor	Admin
FR9	Room: Admin can add new rooms to the system from the room type	Admin
FR10	Staff: Admin can manage staff which includes making a profile for the staff	Admin
FR11	Staff payment: History about the payments paid to particular staff	Admin
FR12	Services: Manage the content of the services offered by the hotel	Admin
FR13	Testimonial: Customer can write the testimony which is visible on the website but admin can delete the given	Customer, Admin

	feedback	
FR14	Department: Admin can manage departments in the hotel. Staffs are assigned to particular departments.	Admin
FR15	Staff Review: Staff can provide review for particular customer and the review is sent to the admin subsystem.	Staff, Admin
FR16	Task: Admin can add task for the staff and staff can update the status of the work which is visible inside both subsystems	Admin Staff
FR17	Booking and Payment: Room booking according to the availability of the room and payment done	Customer, Admin
FR18	Staff time: Admin can see the login and logout time of the staff	Staff, Admin

2.3 Non Functional Requirements

Non-functional requirements for the information management system are discussed as mentioned:

Table 2.3 List of Non-Functional requirements

Non-Functional Requirement sr.no.	**Description**	**Purpose**
NR1	Password must be hashed. Password should be stored in an encrypted language in the database. Password can be either text or number	Security
NR2	Admin should have a separate login page	Accessibility
NR3	The admin can view and update any information in the system (CRUD)	Admin rights
NR4	Employee is able to see all the information, which is given in the system, even though he/she cannot update it on their own entirely except for few thing	Staff Rights
NR5	Efficient user interface. No more than 4 clicks to reach the desired goal .	User Interface

2.4 User Interface Mock-up

For the admin login page, we have a separate login page. We created this layout as it is more simplistic and attractive. In the left-hand side an image is visible and on the right-hand side the admin can login using his/her credentials.

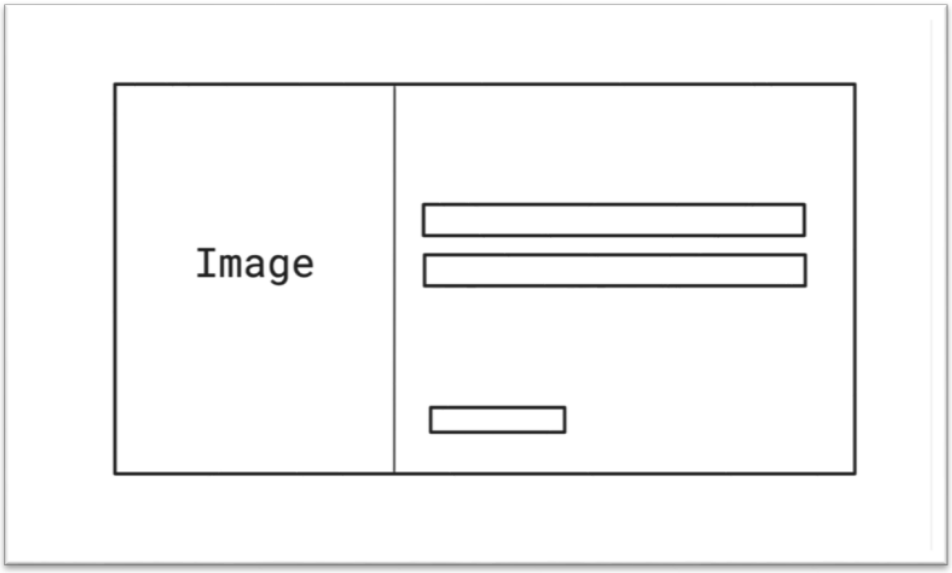


Fig. 2.1 Admin Login mock-up

For the main website, we have chosen an ease looking format where the user can hover over different options provided to the right side of the header. Also a user profile button, which will display the information about the user. As of the content on the main website page, images can be uploaded for attracting the customer's attention. Another sections we have added to the page is services provided by the hotel, Gallery of images and the testimonials of the previous customers.

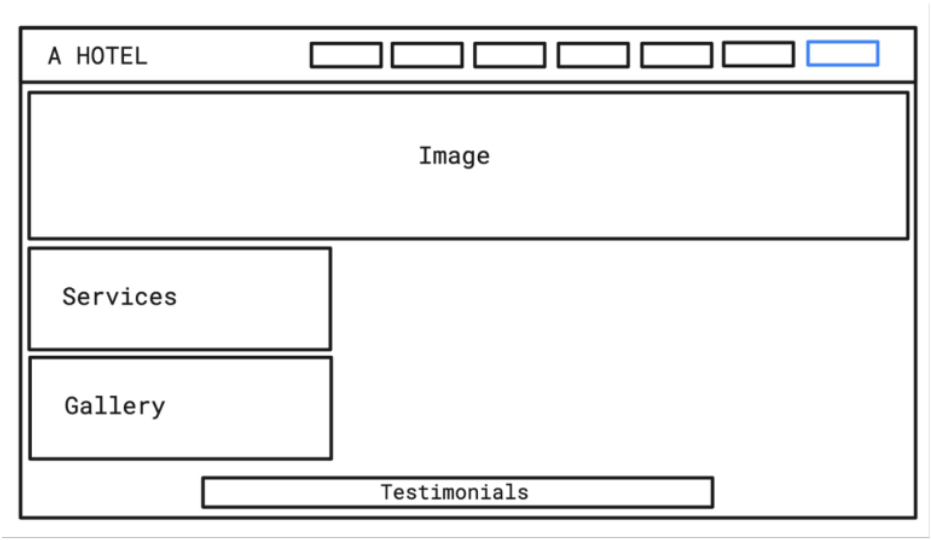


Fig. 2.2 Website Layout mock-up

For the admin dashboard, we have created an amazing looking interface. Where we can see the sidebar with the company icon and the different options with functionalities. On the centre console the admin can look at the statistical data about the bookings and the overview about the bookings.

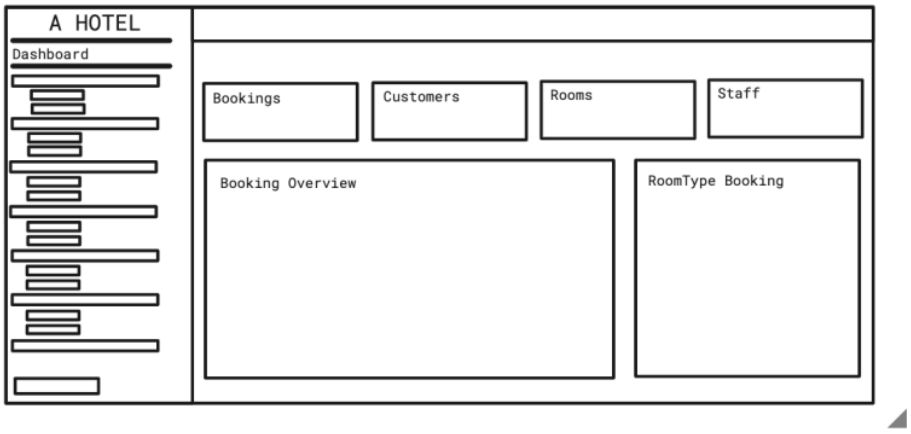


Fig. 2.3 Admin Dashboard mock-up

2.5 UML Diagrams of developing Hotel Management System

2.5.1 Use Case Diagram of proposed system

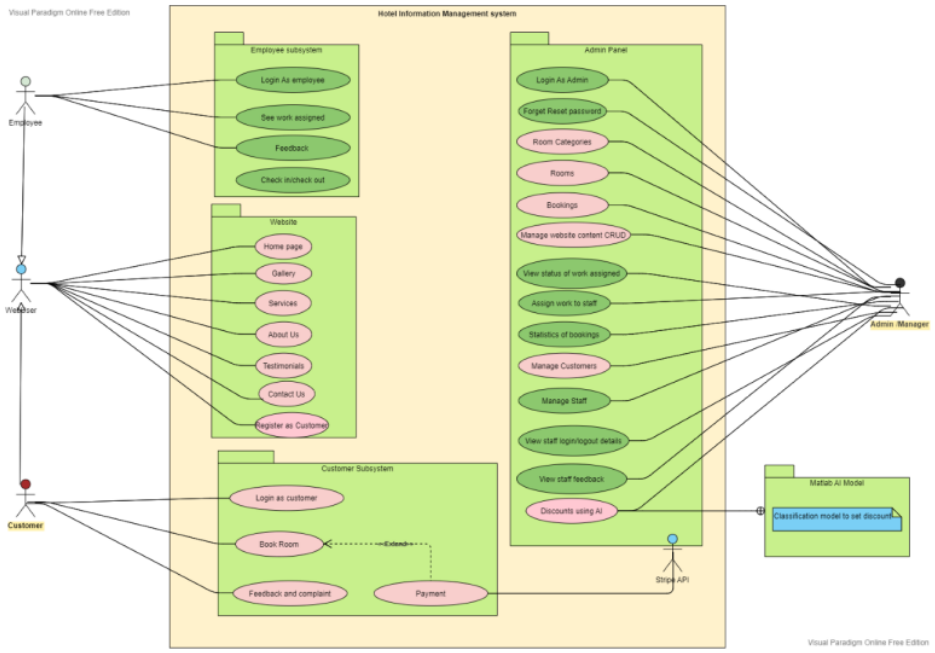


Fig. 2.4 Use Case Diagram

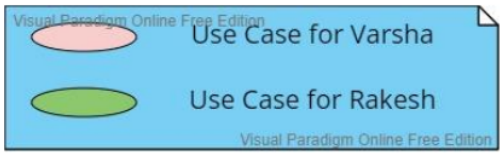


Fig. 2.5 Legend for Use Case

The project consists of four subsystems and the use case. The descriptions of the use cases(31) (32)according to the subsystems are as follows:

1 Employee Subsystem:

Table 2.4 Use case "Login as Employee" specification

Use Case Name	Login as Employee
Goal. Log in to the system	
Description. Employee enters credentials, and is logged into system	
Precondition	Employee is not logged in
Actors	Employee

Postcondition	Employee is logged in

Table 2.5 Use case "See work assigned" specification

Use Case Name	See work assigned
Goal. See the work assigned to the employee	
Description. Employee can look into the daily tasks assigned to him	
Precondition	Employee is at tasks page
Actors	Employee
Postcondition	Employee can see the tasks

Table 2.6 Use case "Feedback" specification

Use Case Name	Feedback
Goal. Employee's feedback on the customer experience.	
Description. Employee can give feedback on customer	
Precondition	Employee should be logged in
Actors	Employee
Postcondition	Employee feedback is done

Table 2.7 Use case "Check in/ check out" specification

Use Case Name	Check in/check out
Goal. To track the employee's working hours.	
Description. When the employee logs in and logs out of the system the details are sent to the admin.	
Precondition	Employee must login
Actors	Employee
Postcondition	Employee's check-in/check-out is noted

2 Customer Subsystem

Table 2.8 Use case "Login as Customer" specification

Use Case Name	Login as Customer
Goal. Log in to the system	
Description. User enters credentials, and is logged into system	
Precondition	Customer is not logged in
Actors	Customer
Postcondition	Customer is logged in

Table 2.9 Use case "Book Room" specification

Use Case Name	Book Room
Goal. To book a room	
Description. Book room according to the room type.	
Precondition	Customer is logged in
Actors	Customer
Postcondition	Room is booked

Table 2.10 Use case "Feedback and complaint" specification

Use Case Name	Feedback and complaint
Goal. Log in to the system	
Description. User enters credentials, and is logged into system	
Precondition	Customer is at feedback page
Actors	Customer

Postcondition	Customer has successfully submitted the feedback.

Table 2.11 Use case "Payment" specification

Use Case Name	Payment
Goal. To make payments	
Description. Customer can make payments to the hotel using strip gateway	
Precondition	Customer is at payment portal
Actors	Customer
Postcondition	Payment is done

3 Website

Table 2.12 Use case “Homepage" specification

Use Case Name	Homepage
Goal. To display the information about the available rooms and services	
Description. Information about the rooms, services and the testimonials can be seen on the homepage.	
Precondition	Admin is at homepage
Actors	Admin
Postcondition	Customer can see the homepage with detailed information.

Table 2.13 Use case "Gallery" specification

Use Case Name	Gallery
Goal. To display pictures of the available rooms.	
Description. Pictures help the customer to choose a room as per their needs.	
Precondition	Admin is not logged in
Actors	Admin
Postcondition	Admin is logged in and in gallery

Table 2.14 Use case "Services" specification

Use Case Name	Services
Goal. To add services to the website that are provided by the hotel.	
Description. Services can be added which will be shown on the services page and the customer can select the desired services.	
Precondition	Admin is not logged in
Actors	User
Postcondition	Admin is logged in and add the services.

Table 2.15 Use case "About us" specification

Use Case Name	About us
Goal. To give a brief intro about the hotel	
Description. About us page tells a lot of the hotel and its working.	
Precondition	Admin is on about us page
Actors	Admin
Postcondition	Admin has put the information.

Table 2.16 Use case "Testimonials" specification

Use Case Name	Testimonials
Goal. Customers can add their experience about the stay in the hotel.	
Description. After the stay in the hotel the customer has ability to add testimonials about his experience.	
Precondition	Customer is on testimonials page
Actors	Customer

Postcondition	Customer had added the testimonials

Table 2.17 Use case "Contact us" specification

Use Case Name	Contact us
Goal. To solve the queries of the customer	
Description. Customer queries are solved when the customer contacts the hotel management.	
Precondition	Customer is on contact us page
Actors	Customer
Postcondition	Customer has contacted the hotel management

Table 2.18 Use case "Register as customer" specification

Use Case Name	Register as customer
Goal. The user can register as a customer.	
Description. The user can register as a customer to the hotel by signing up	
Precondition	User is on the website
Actors	User
Postcondition	User has successfully registered

4 Admin Subsystem

Table 2.19 Use case "Login as Admin" specification

Use Case Name	Login as Admin
Goal. Log in to the system	
Description. Admin enters credentials, and is logged into system	
Precondition	Admin is not logged in
Actors	Admin
Postcondition	Admin is logged in

Table 2.20 Use case "Forget Reset password" specification

Use Case Name	Forget Reset password
Goal. Reset password option when the admin forgets the password	
Description. If the admin forgets the password then the admin is able to reset the password using their email.	
Precondition	Admin has forgot the password and on forgot password page
Actors	Admin
Postcondition	Admin has successfully changed the password

Table 2.21 Use case "Room Categories" specification

Use Case Name	Room Categories
Goal. Admin can add different types of room categories.	
Description. Admin can add the room types to the system that are available in the hotel with their prices mentioned.	
Precondition	Admin is on Room Categories page.
Actors	Admin
Postcondition	Admin has successfully added the room categories

Table 2.22 Use case "Rooms" specification

Use Case Name	Rooms
Goal. To show the rooms available in the hotel	
Description. Display the rooms available for stay in the hotel with their prices mentioned.	
Precondition	Admin Is on Rooms page
Actors	Admin

Postcondition	Admin had successfully added the rooms

Table 2.23 Use case "Bookings" specification

Use Case Name	Bookings
Goal. Admin can book the rooms for the customers	
Description. Using the system the admin can book rooms for the customers and accept the payments.	
Precondition	Admin is on the booking page
Actors	Admin
Postcondition	Admin has booked the rooms for the customers

Table 2.24 Use case "Manage website content CRUD" specification

Use Case Name	Manage website content CRUD
Goal. The admin can manage the system content using CRUD method.	
Description. Admin has the ability to manage the system using CRUD method.	
Precondition	Admin must be logged in
Actors	Admin
Postcondition	Admin had changed the content

Table 2.25 Use case "Assign work to staff" specification

Use Case Name	Assign work to staff
Goal. Admin can assign the tasks to the staff	
Description. The admin has the ability to assign the tasks to the staff and also manage the tasks using CRUD method.	
Precondition	Admin is on assign tasks page
Actors	Admin
Postcondition	Admin has assigned the task

Table 2.26 Use case "Statistics of booking" specification

Use Case Name	Statistics of booking
Goal. Admin can look at the statistical data of the number of bookings	
Description. Admin has the access to the statistical data of the number of bookings to the hotel with this the admin can make the changes to the hotel management to improve the count in number.	
Precondition	Admin must log in
Actors	Admin
Postcondition	Admin can see the statistical data of bookings

Table 2.27 Use case "Manage Customers" specification

Use Case Name	Manage Customers
Goal. Admin can manage customers	
Description. Admin can manage the customers using the CRUD method	
Precondition	Admin is on manage customer page
Actors	Admin
Postcondition	Admin can manage the customers

Table 2.28 Use case "Manage Staff" specification

Use Case Name	Manage Staff
Goal. Admin can manage the staff	
Description. Admin can the manage the staff using the CRUD method.	
Precondition	Admin is on the manage staff page
Actors	Admin
Postcondition	Admin can manage the staff

Table 2.29 Use case "View staff login/logout status" specification

Use Case Name	View staff login/logout status
Goal. Admin must see the login and logout time of the staff	
Description. When the staff logs in to the system the “In time” is notes and when logged out the “Out time” is noted and these hours are directly sent to the admin.	
Precondition	Staff must login
Actors	Admin
Postcondition	Staff must logout

Table 2.30 Use case "View staff feedback” specification

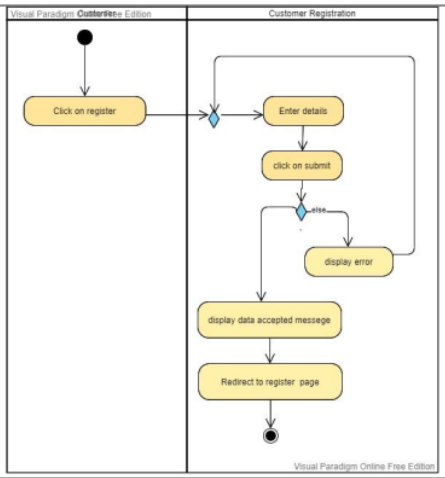
Use Case Name	View staff feedback
Goal. The admin can check the feedback given by the staff	
Description. When the staff gives the feedback then it goes to the admin and the admin can check the feedback.	
Precondition	Staff must login
Actors	Admin
Postcondition	Staff must give the feedback

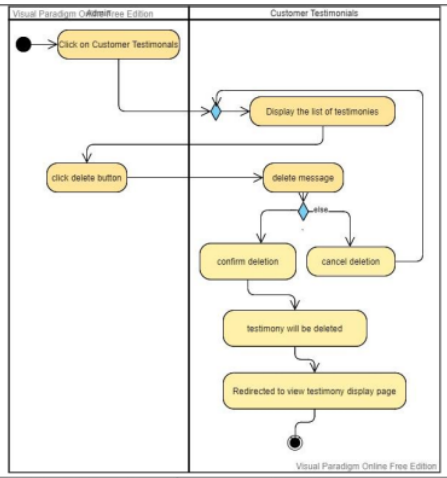
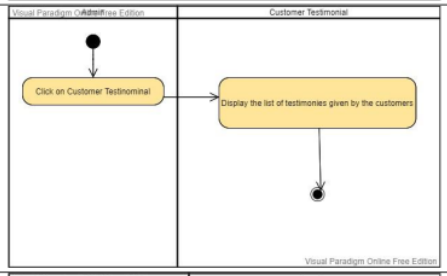
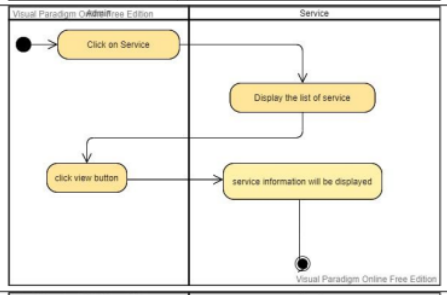
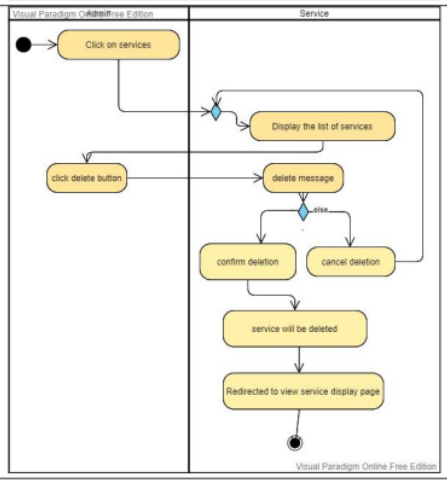
Table 2.31 Use case "View status of work assigned" specification

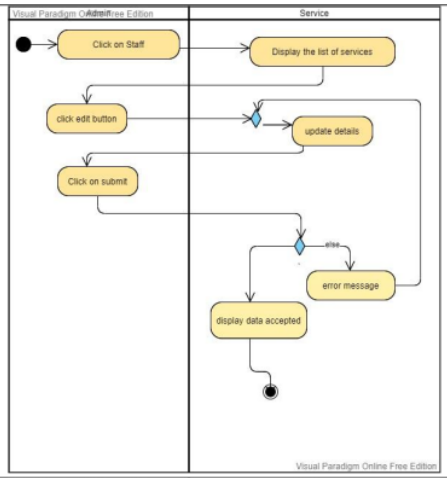
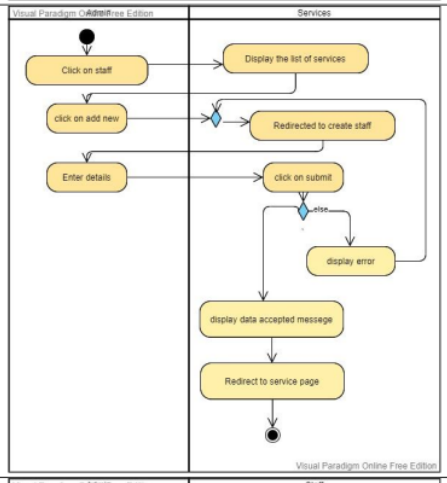
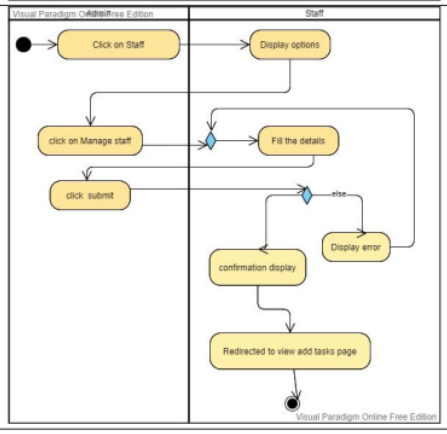
Use Case Name	View status of work assigned
Goal. Admin can see the status of the work which has been assigned to the staff.	
Description. When the staff clicks in the work done checkbox the admin can look at the status of the work assigned to the staff.	
Precondition	Staff must be at the task page
Actors	Admin
Postcondition	Staff must click on the work done checkbox once the work is completed

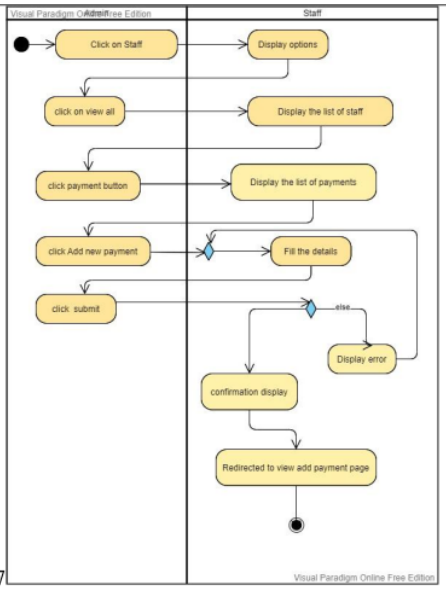
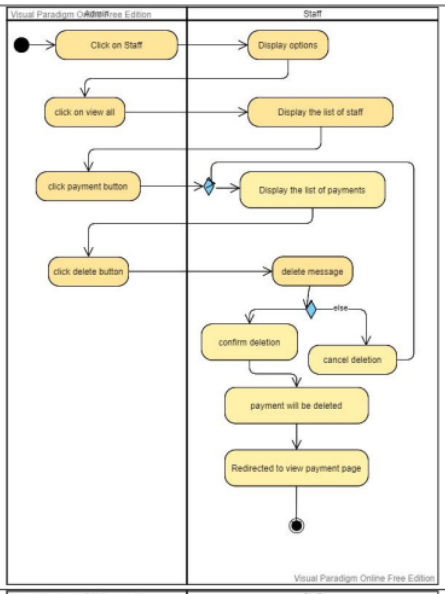
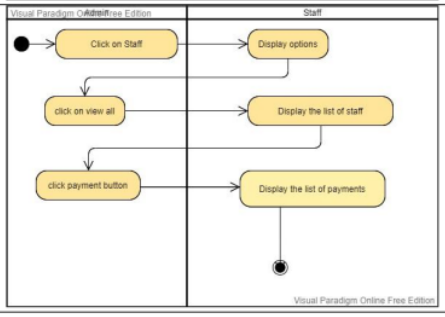
2.6 Activity Diagram for the proposed system

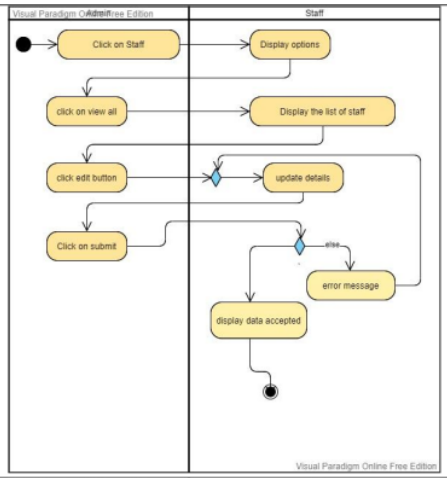
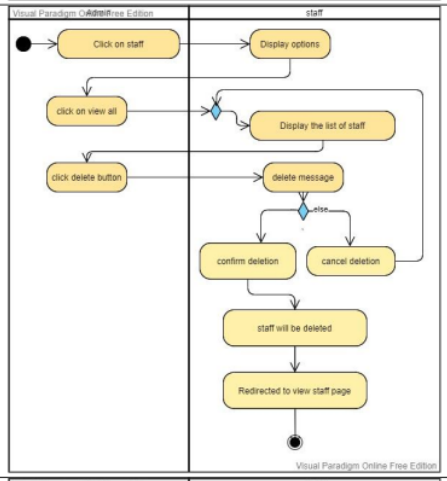
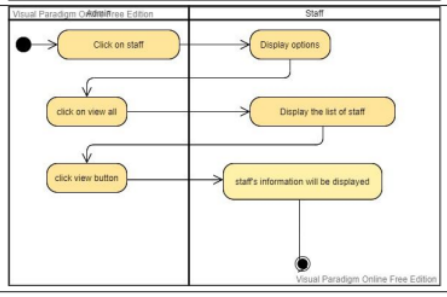
Table 2.32 Activity diagrams for the implemented system

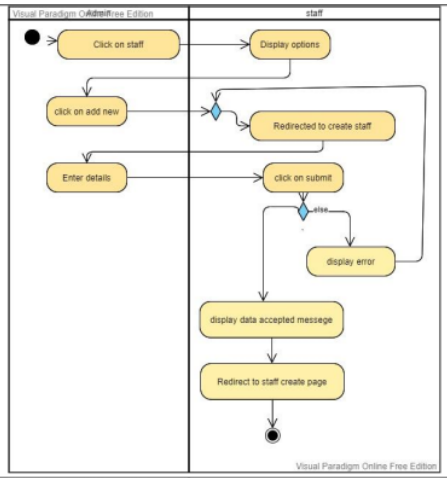
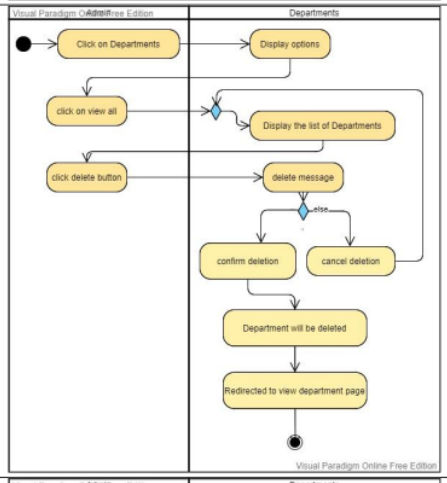
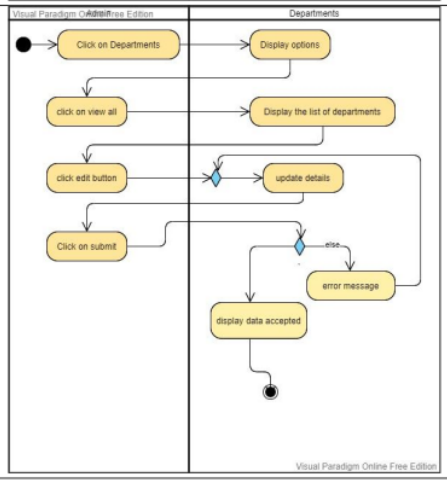
Sr.no	**Diagram**	**Comment**
1.		Register Customer (Website)

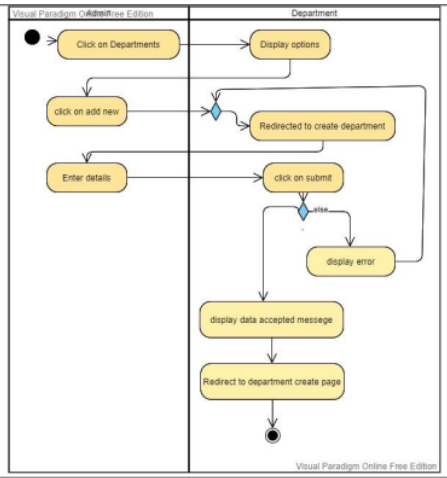
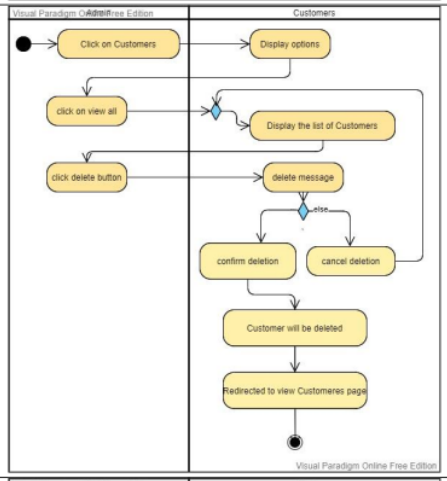
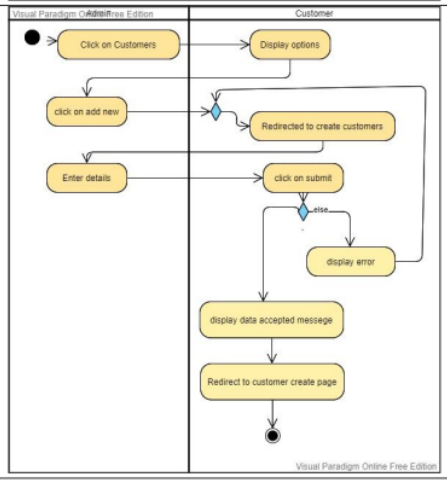
2.		Delete Testimony (Admin)
3.		View Testimony (Admin)
4.		View the list of Services added in the system (Admin)
5.		Delete services from the system (Admin)

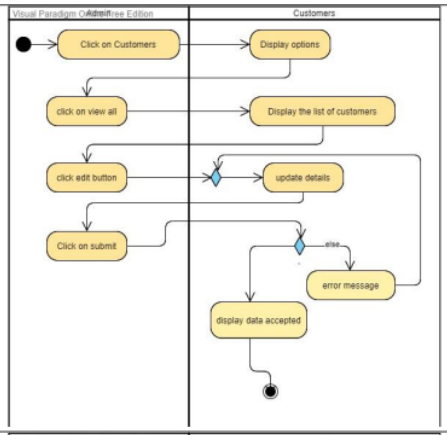
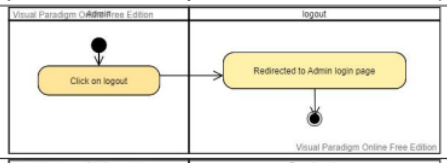
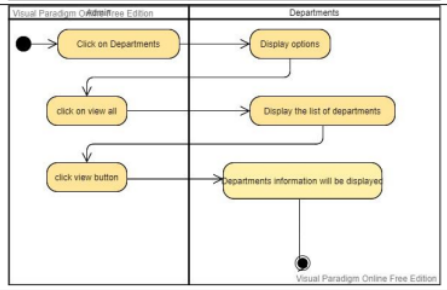
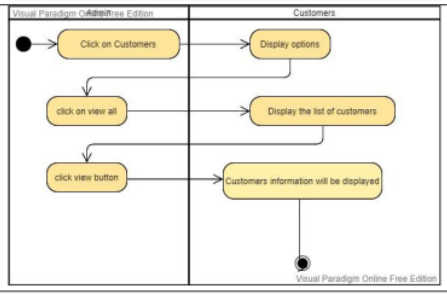
6.		Update Staff details (Admin)
7.		Add a new employee to the system (Admin)
8.		Add task for the staff (Admin)

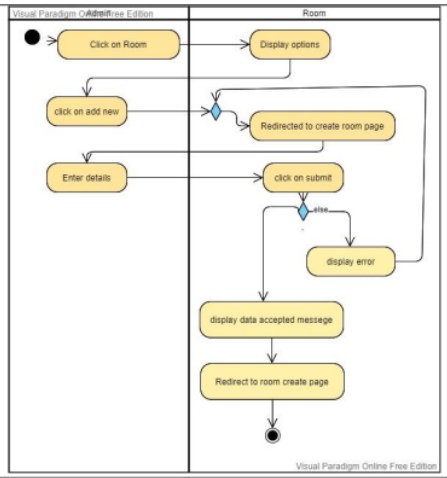
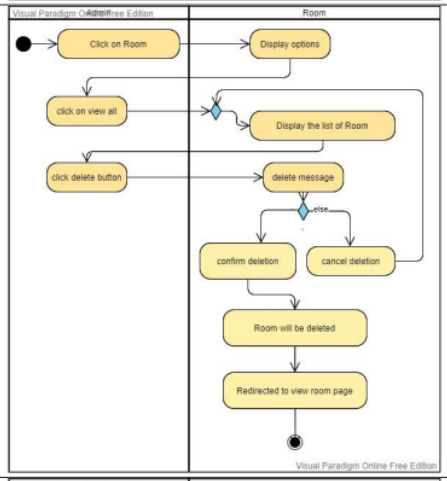
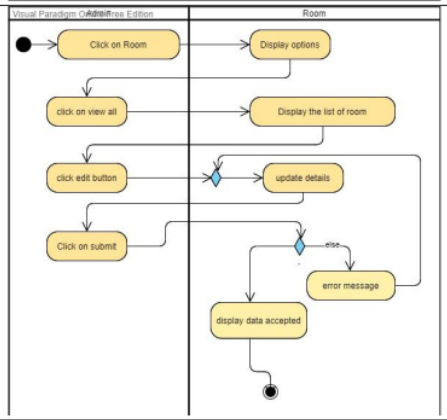
9.		Adding new payment for the staff (Admin)
10.		Delete the payment details from the system (Admin)
11.		View all the payments made to the staff (Admin)

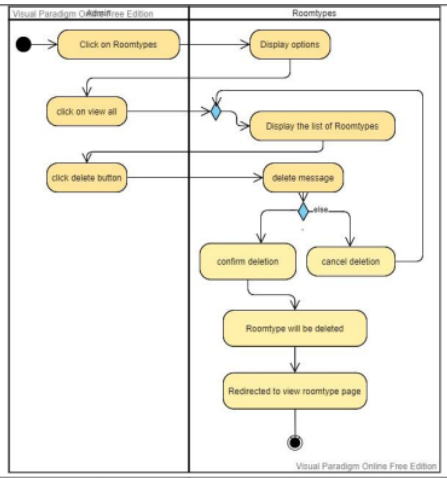
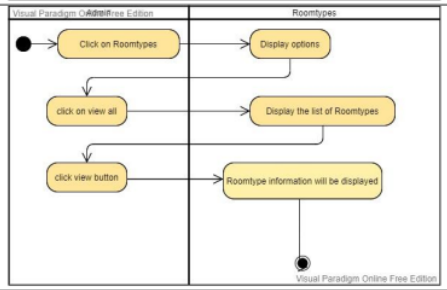
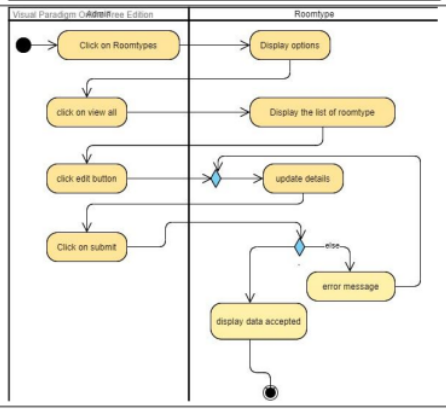
12.		Update staff details (Admin)
13.		Delete staff from the system (Admin)
14.		View list of all the employees (Admin)

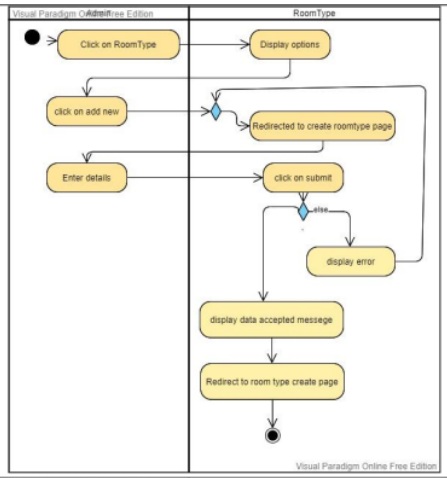
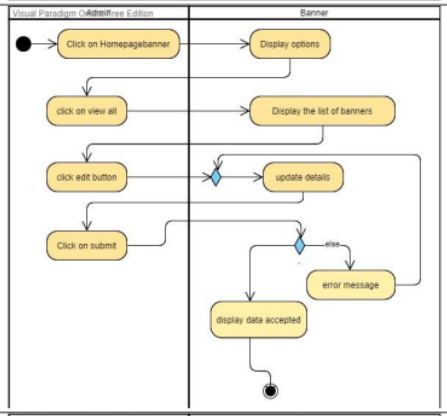
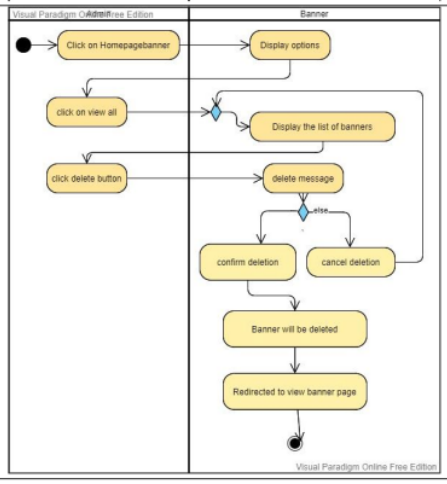
15.		Add a new staff in the system (Admin)
16.		Delete a department (Admin)
17.		Update a department (Admin)

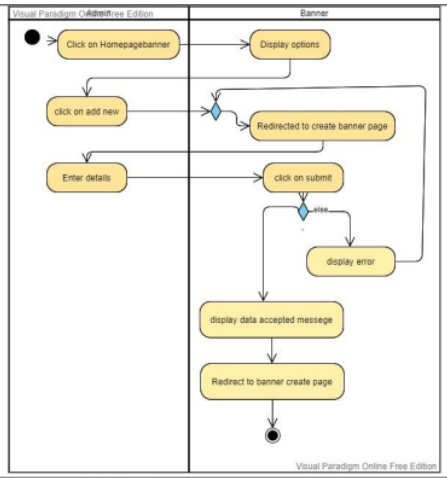
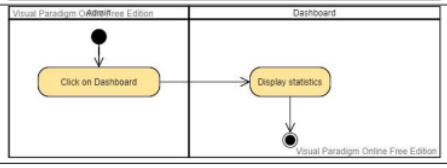
18.		Add new department in the system (Admin)
19.		Delete customer (Admin)
20.		Add new customer to the system (Admin)

21.		Edit customer details (Admin)
22.		Display the statistics (Admin)
23.		View the list of departments (Admin)
24.		Display the list of all the customers registered in the system (Admin)

25.		Create room (Admin)
26.		Delete the room from the system (Admin)
27.		Update the room information (Admin)

28.		Delete room type (Admin)
29.		Display the list of room types (Admin)
30.		Update the room type values (Admin)

31.		Create new room type (Admin)
32.		Update the banner (Admin)
33.		Delete banner (Admin)

34.		Add new images for banner to the database (Admin)
35.		Display Statistics (Admin)

2.7 Navigation Plan for the proposed system

Navigation plan shows how the user can interact with the system and how the user can navigate through the different subsystems(33).

Customer subsystem includes all the pages from website along with booking and history options. In this navigation. The dashboard is on the ribbon with options and each page can be accessed from anywhere in the subsystem. The user must login as a customer after registering into the system in order to access gain into customer subsystem.

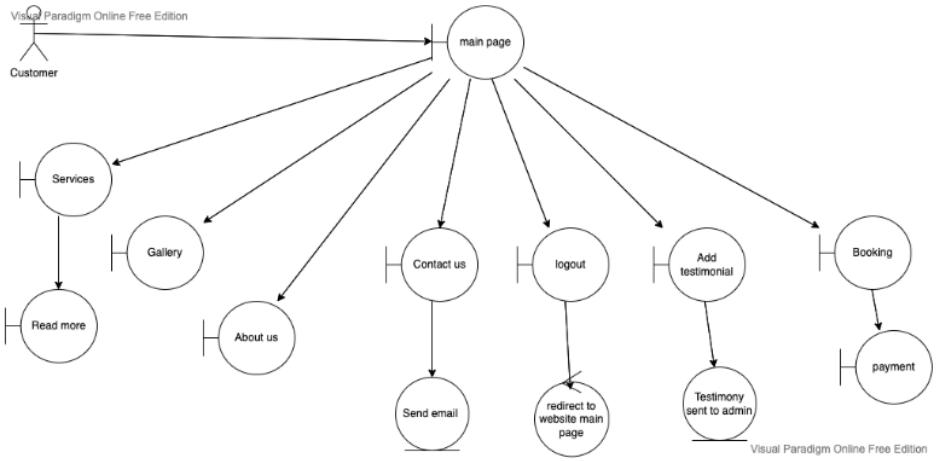


Fig. 2.6 Customer subsystem navigation plan

Employee subsystem extends the features of the website along with additional functions. To navigate through the employee subsystem the user should be registered as an employee and also logged into the subsystem to open the particular features in it.
To exit the subsystem the employee must log out and the user is navigated back to the website.

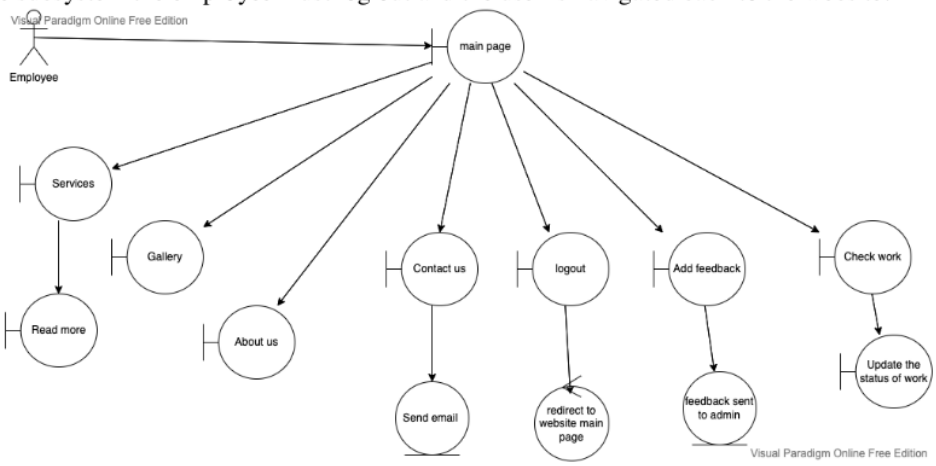


Fig. 2.7 Employee subsystem Navigation Plan

Admin subsystem is the most complex one as it has CRUD for every possible feature. The contents of the website and the system can be managed by the admin. The admin dashboard can be accessed when the user login from the user login and is registered as an admin in the database.

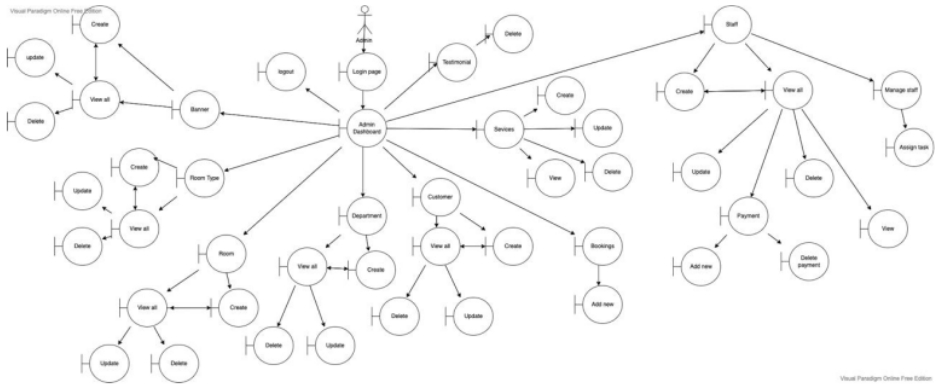


Fig. 2.8 Admin Subsystem Navigation Plan

Home page of the web system is the website and there are multiple login option available for the user to log in as different roles and access the particular subsystem.

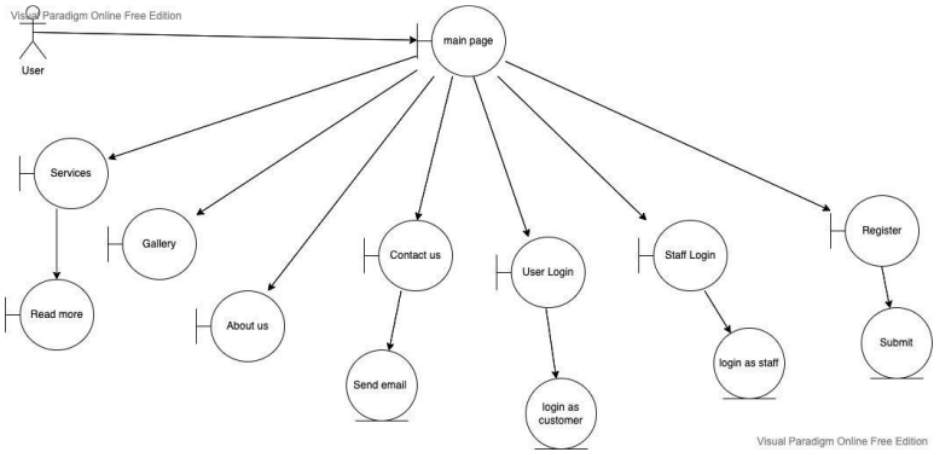


Fig. 2.9 Website Navigation Plan

2.8 Database Model

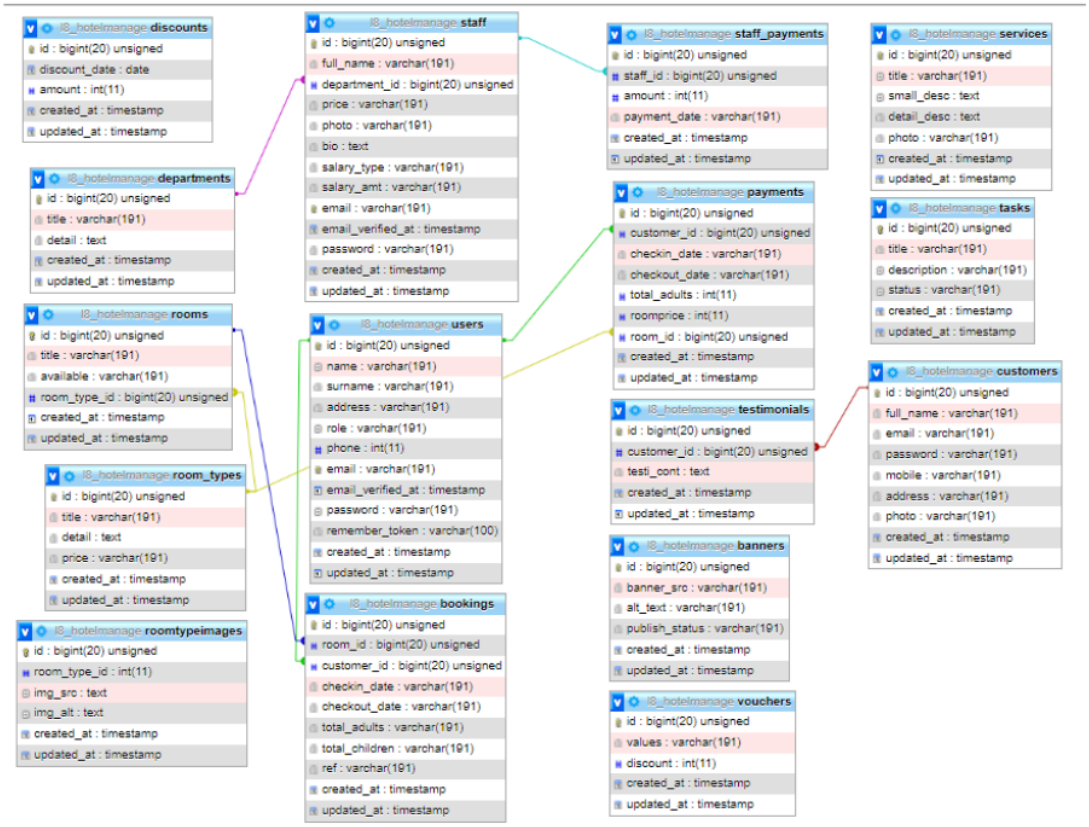


Fig. 2.10 Database Diagram for the proposed system

Table 2.33 Specification of the table Banner

Banner Table to hold images to display on the website			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Banner_src	Image	None	String
Alt_text	Description of image	None	String
Publish_status	Whether or not the image has to be displayed	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	timestamp

Table 2.34 Specification of the table bookings

Booking Table to hold the final bookings done			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Room_id	Foreign key to connect "Room"	None	Integer
Customer_id	Foreign key to connect "Customer"	None	Integer
Checkin_date	Starting date of booking	None	Date
Checkout_date	Ending date of booking	None	Date
Total_adults	Number of adults	None	integer
Total_children	Number of children	Null	integer
Ref	Reference	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.35 Specification of the table Customers

Customers Details of the customers			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Full_name	Full name of the customer	None	String
Email	Email id of the customer	None	String
Password	Password for the account	None	String
Mobile	Mobile number of the customer	None	String
Address	Address of the customer	None	String
Photo	Photo of the customer	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.36 Specification of the table departments

Departments Details about the departments			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Title	Title of the department	None	String
details	Details about the department	None	text
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.37 Specification of the table discount

Discount Discount dates on which admin has put up discount			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Room_id	Foreign key to connect "Room"	None	Integer
Discount_date	Date when the discount should be issued	None	Date
amount	Amount to be paid	None	integer
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.38 Specification of the table payments

Payments Table to hold the payment details			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Customer_id	Foreign key to connect "Customer"	None	Integer
Checkin_date	Starting date of booking	None	Date
Checkout_date	Ending date of booking	None	Date
Total_adults	Number of adults	None	Integer
Total_children	Number of children	None	Integer
Room_id	Foreign key to connect "Room"	None	Integer
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.39 Specification of the table rooms

Rooms Table to hold the values and availability of the rooms			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Title	Title of the room	None	String
Available	Room availability	True	String
Room_type_id	Foreign key to connect "Room_type"	None	Integer
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.40 Specification of the table room_type_images

Room_type_images Table for images of room types			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Room_type_id	Foreign key to connect "Room_type"	None	Integer
Img_src	Image of the rooms	None	Text
Img_alt	Additional images if needed	None	Text
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.41 Specification of the table Room Type

Room_type Information about the type of rooms available in the hotel			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment

Title	Title of the room	None	String
Customer_id	Foreign key to connect "Customer"	None	Integer
Details	Details of the room	None	Text
Price	Price of the room	Null	integer
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.42 Specification of the table services

Services Description of the services provided by the hotel			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Title	Title of the service	None	String
Small_desc	Inshort description about the service	None	Text
Detail_desc	Detailed description about the service	None	Text
Photo	Images of the services	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.43 Specification of the table staff

Staff Details of the staff			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Full_name	Full name of the staff	None	String
Department_id	Foreign key to connect "Department"	None	Integer
Price	Per day wages	Null	integer
Photo	Photo of the employee	None	String
Bio	Information about the employee	None	Text
Salary_type	Monthly/Daily wage	None	String
Salary_amt	Amount to be paid to the employee	None	String
Email	Email of the employee	None	String
Email_verified_at	Time stamp when the email was verified	Null	Timestamp
password	Password for the account	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.44 Specification of the table staff_payments

Staff_payments Description of payments made to staff			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Staff_id	Foreign key to connect "Staff"	None	Integer
Amount	Amount to be paid to the employee	None	integer
Payment_date	Date when the amount has been paid	None	Date
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.45 Specification of the table tasks

Tasks Description of task assigned to staff

Attribute	Comment	Default	Type
Id	Unique id	None	Auto increment
Title	Task	None	String
Description	Task description	None	String
Status	Status of the task	Progress	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.46 Specification of the table testimonials

Testimonials Testimonials provided by the customer			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Customer_id	Foreign key to connect "Customer"	None	Integer
Testi_cont	Content	None	Text
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.47 Specification of the table users

Users Description of Users			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Name	Given name of the user	None	String
Surname	Family name of the user	None	String
Address	Address of the user	None	String
Role	Admin/Customer	Customer	String
Phone	Phone number	None	Integer
Email	Email id of the admin/Customer	None	String
Email_verified_at	Time stamp when the email was verified	Null	Timestamp
Password	Password for the account	None	String
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

Table 2.48 Specification of the table vouchers

Vouchers Voucher's code and amount			
Attribute	**Comment**	**Default**	**Type**
Id	Unique id	None	Auto increment
Values	Voucher code	None	String
discount	Amount of discount	None	Integer
Created_at	Time stamp when the data was created	Null	Timestamp
Updated_at	Time stamp when the data was last updated	Null	Timestamp

3 Realization and Testing

3.1 Description of Controllers

Laravel Framework is MVC i.e. Model, view and controller are used for coding and establishing a web system. In this section, we will give a brief description of all the controllers used in the development of the project.

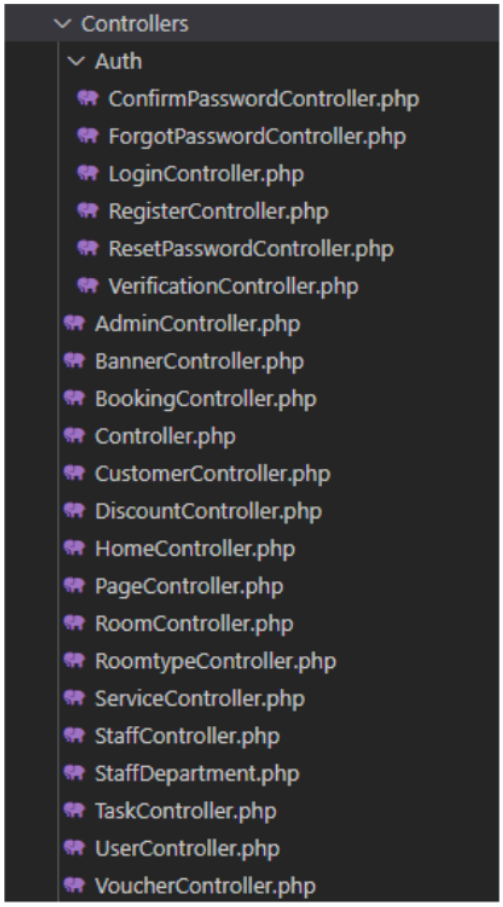


Fig. 2.11 Controllers in Project

1. ConfirmPasswordController : Controller to handle password confirmation
2. ForgotPasswordController : Controller to handle reset password
3. LoginController : Controller to authenticate the users
4. RegisterController: Controller to register new user and validate the data
5. ResetPasswordController : Controller to help reset the password
6. VerificationController : Controller for Email verification
7. AdminController : Controller for admin subsytem

8. BannerController : Controller to manage banner images i.e. the images displayed in the website
9. BookingController : Controller used for booking, implemented AI and Stripe payment Gateway
10. CustomerController : Controller to manage the customer information
11. DiscountController : Controller which helps with the discount date set by the admin which is further used for AI as input
12. HomeController : Controller to display the content on website
13. PageController : Controller for viewing content on website
14. RoomController : Controller for Room management offered by the hotel
15. RoomtypeController : Controller for Room Type Management available in hotel
16. ServiceController : Controller used for managing services provided by the hotel
17. StaffController : Controller for management of staff
18. StaffDepartment : Controller for management of departments
19. TaskController : Controller to manage the task assigned to staff by admin
20. UserController : Controllerto get the history and profile of the user

3.2 Functional Requirement Implementation

3.2.1 FR1: Login

To login into system different pages for different subsystem were implemented. While planning it was thought to make the same login page but after discussion it was established that login could have been done in the same web page but to improve the accessibility to the system and splitting the admin panel from the website different login page idea will be adapted. Hence, different login page are designed and implemented.
Login as a customer and login as a staff could be accessed directly from the website, mentioned in the ribbon.
Login for the admin panel is different which helps make the website design simpler and more minimalistic.

3.2.2 FR2:Forgot/ Reset Password

In login page an option for reset password is provided which further send the email with the link to reset the password.
The password is then changed inside the database as well and the user is redirected to the home page after completion.

3.2.3 FR3: Register

To add new customer into the system, registration is required. The data is then saved into corresponding database which will further be used into different functional requirements such as bookings and payments.
Once the customer ID is generated into the system the details are then used to show the profile once logged into customer subsystem.

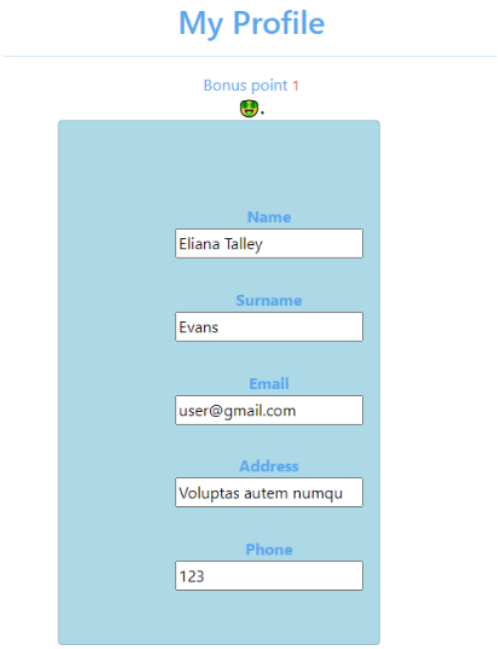


Fig. 2.12 My Profile page

3.2.4 FR4: Logout

To exit the subsystem logout is used. After clicking on logout the user's role is changed back to "WebUser" and if he needs to access the information again, he will need to enter the credentials again and gain access to the corresponding subsystem.

3.2.5 FR5: Website Content

Website content such as banner, services, gallery can be updated dynamically upon the changes done by admin.

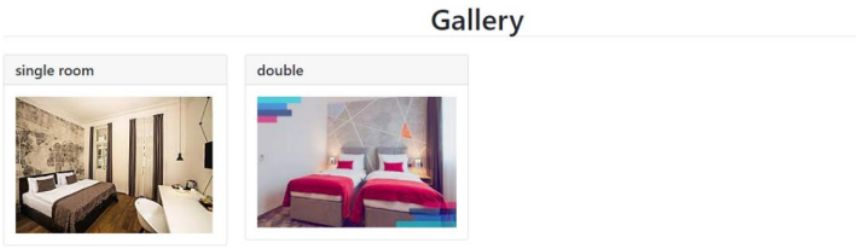


Fig. 2.13 Gallery Page

This content is visible on the website if they are published by the admin and is accessible by all the "WebUser".

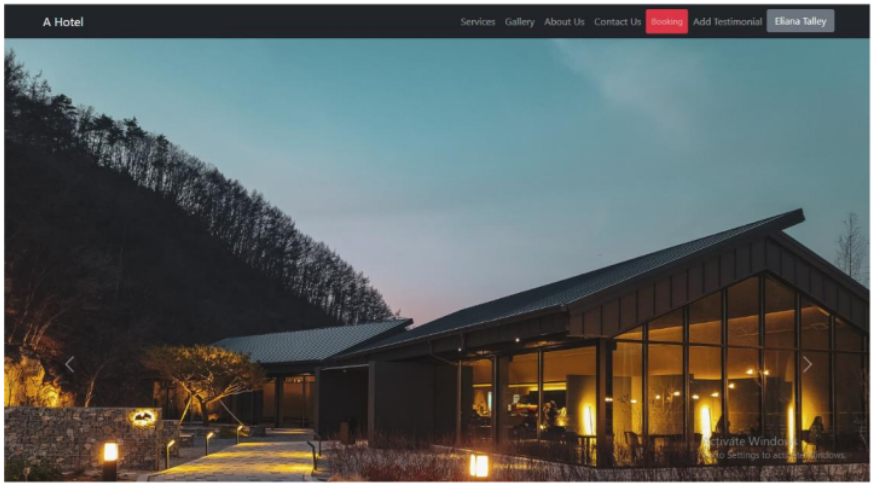


Fig. 2.14 Website Home

3.2.6 FR6: Contact Us

This function allow any WebUser to contact the management directly by send an email which can be anything and the email is then received to the account registered in the project.

Table 2.49 Code Fragment: Contact Us

```
Mail::send('mail', $data, function($message){
```

```
$message->to('ktuhms@gmail.com', 'Rakesh')->subject('Contact Us Query');


        $message->from('ktuhms@gmail.com','zT8D.iE!N6auM7z');
```

Fig. 2.15 Contact Us page

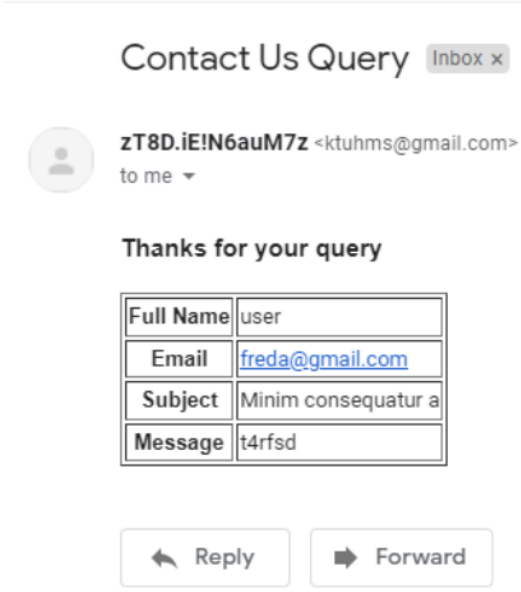


Fig. 2.16 Contact us Result

3.2.7 FR7: Dashboard

Dashboard includes the graphs and statistics from different data tables and using Bootstrap the data is then displayed in the form of pie charts and graphs and counters.

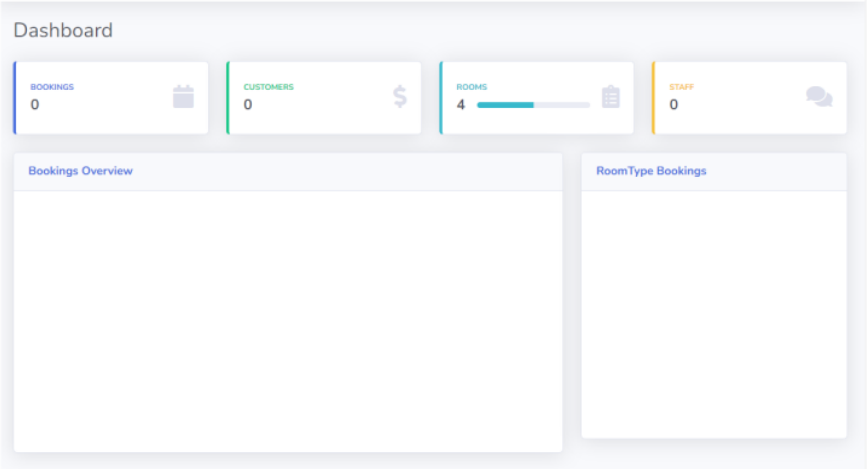


Fig. 2.17 Dashboard

3.2.8 FR8: Room Type

Room Type indicates the room classification of the type of rooms available in the hotel. Admin can add and manage these room types from the admin panel.

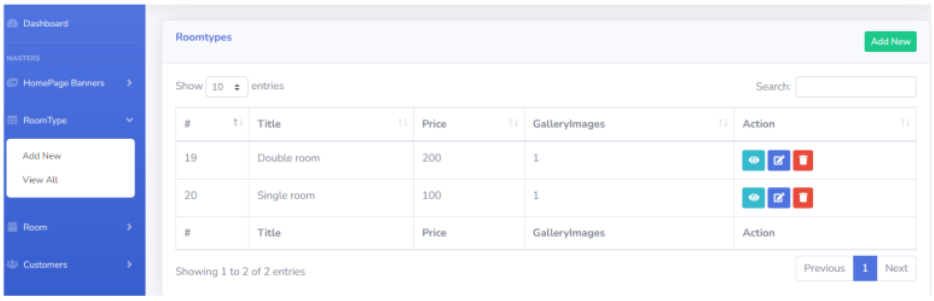


Fig. 2.18 Room Type in Admin Panel

3.2.9 FR9: Room

Room can be added to the system by the admin and to make a new room, room type should be chosen beforehand and description of the room needs to be input. The data is then stored in the system and the admin can change the availability status for the room. Availability can be changed manually by the admin or upon booking.

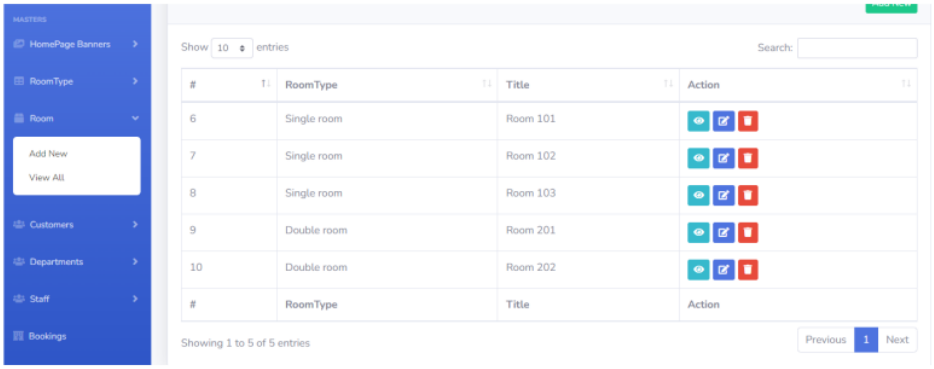


Fig. 2.19 Rooms in Admin Panel

3.2.10 FR10: Staff

Staff account can be made only by the admin. To complete the registration process for the staff. Staff has to be assigned a department and the wage type could be selected. Wage type can be monthly or daily depending on the type of work the staff is doing.

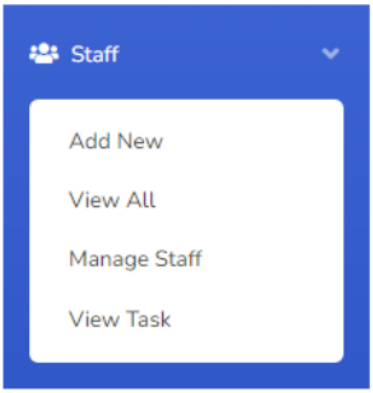


***Fig. 2.20** Dropdown from Staff Menu*

3.2.11 FR11: Staff Payment

Registered staff ,in addition to the CRUD options include one more option called payments. In that, admin can enter the payment details for the particular staff and also manage it.

3.2.12 FR12: Service

Description of the services offered by the hotel can be seen using this blade. The short description can be seen on the website. When clicked on read more option long detailed description can be read. These services can be updated and managed from the admin panel.

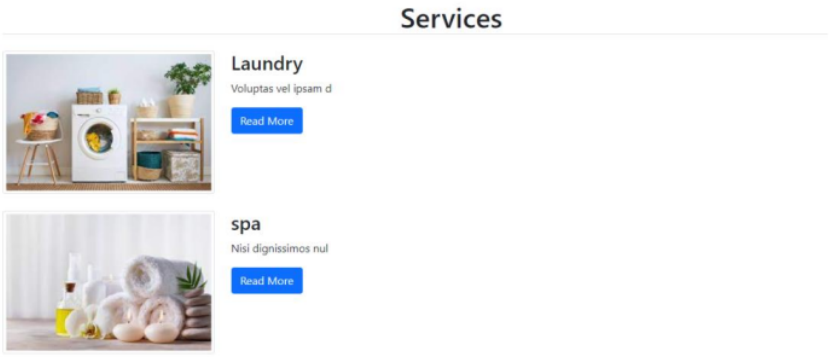


Fig. 2.21 Services page

3.2.13 FR13: Testimonial

Testimonials are provided by the customers and are displayed on the website in the slideshow form. The admin can see the testimonies in the panel and delete them from the system entirely if needed.

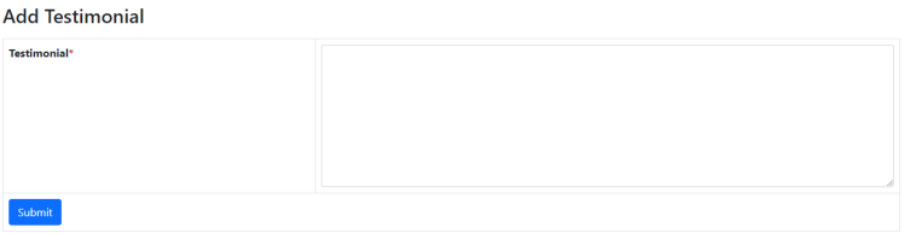


Fig. 2.22 Testimonial addition from Customer panel

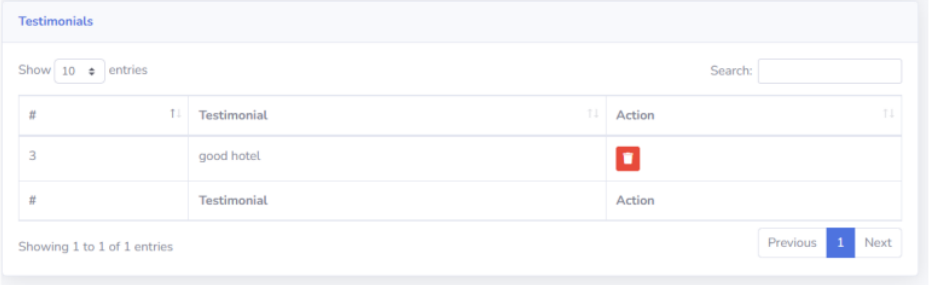


Fig. 2.23 Testimonial in Admin Panel

3.2.14 FR14: Department

Department allows the admin to add departments in the system and assign staff to the departments. This will help the admin while assigning the tasks.

3.2.15 FR15: Staff Review

Reviews can be given by the staff about specific customer if required. For example, if there is something broken or not in the usable condition and some customer was responsible for the damage, the staff can write review about it and the review will be shown in the admin panel. This feature avoid in appropriate blame on the staff and also improves the colleagues inter personal relations because this option eliminated the risk of miscommunication.
Also, admin can ask for compensation from customer for the damages.

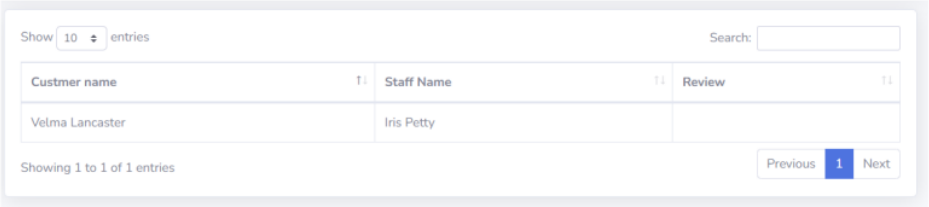


Fig. 2.24 Staff Review in Admin Panel

3.2.16 FR16: Task

Tasks are assigned to the staff from admin panel. The assigned task description can be seen both in admin panel and the staff. The staff is eligible to view the task assigned to him and update the status of the task. Status could be in progress or done. This function makes it easier for the admin as well as staff to avoid miscommunication of work and also this eliminates the risk of task not been achieved.

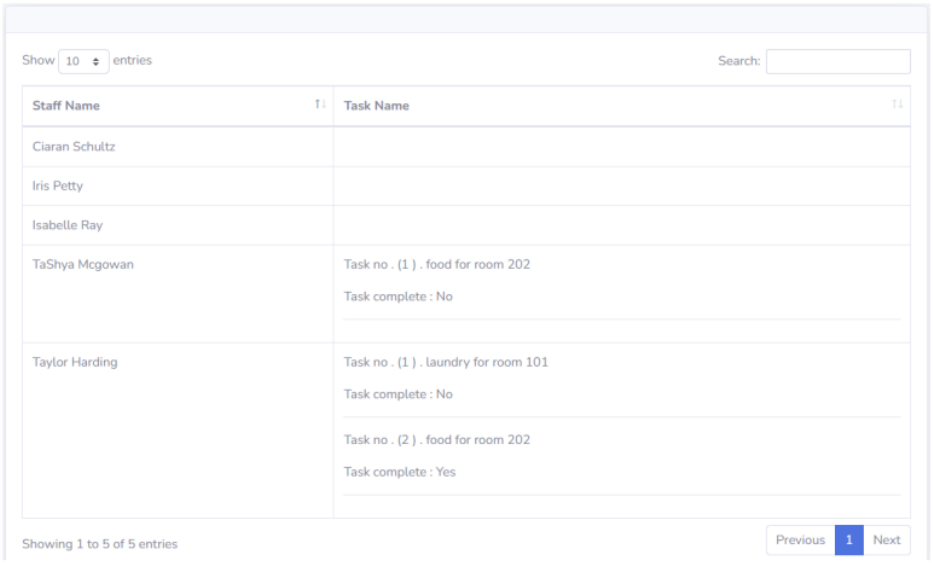


Fig. 2.25 Tasks in Admin Panel

3.2.17 FR17: Booking and Payment

Booking of the available rooms in the system using Stripe Payment gateway and the amount is generated by the output of artificial intelligence which will be explained in further section.

Table 2.50 Code Fragment: Stripe Implementation

```
session($sessionData);            Stripe\Stripe::setApiKey(env('STRIPE_SECRET'));
            $session = \Stripe\Checkout\Session::create(['payment_method_types' => ['card'],
'line_items' => [ ['price_data' => ['currency' => 'inr','product_data' => ['name' =>
$roomstypes, 'description' => 'Before discount :' .$amount ,],
 'unit_amount' => $disocuntprice * 100,], 'quantity' => 1,]],
 'mode' => 'payment',
 'success_url' => 'http://127.0.0.1:8000/booking/success?session_id={CHECKOUT_SESSION_ID}',
'cancel_url' => 'http://127.0.0.1:8000/laravel-apps/hotelManage/booking/fail',]);
```

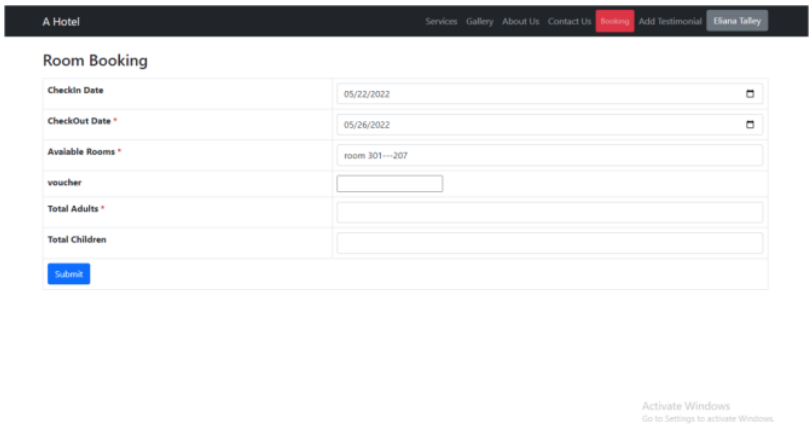


Fig. 2.26 Room Booking

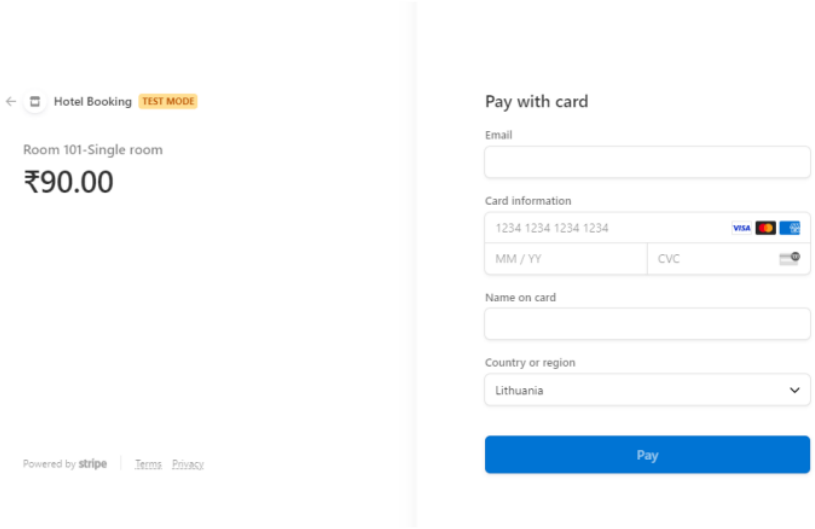


Fig. 2.27 Stripe payment interface

3.2.18 FR18: Staff Time

Login and logout time stamps of the staff can be seen in the admin panel. This data is not accessible by staff since this is the carbon time stamp when the staff actually logged into the system. It gives admin the insight on which staff is available to work and then accordingly the admin can assign work to the active staff personnel.

3.3 Non-Functional Requirement Implementation

3.3.1 NR1: Security

Password is being hashed before storing it into database. Whenever the system requires the password for checking the credentials it has to reverse the hashing from the database to get the correct password.

3.3.2 NR2: Accessibility

During the designing of the user interface it was kept in mind not to use more than 4-5 clicks to get to complete any operation. Hence the dashboard pattern was used which is accessible at any moment from anywhere in the system. No need to go back to main page to start a new function.
For the customer and staff subsystem, only those options are visible during the session which are required for that particular user and other tabs are hidden.

Table 2.51 Accessibility table

Sr.no.	Panel	Function		Maximum number of clicks required
1.	Admin	Dashboard		1
2.	Admin	Room Type	Create	2
			Read	2
			Update	3
			Delete	3
3.	Admin	Room	Create	2
			Read	2
			Update	3
			Delete	3
4.	Admin	Customer	Create	2
			Read	2
			Update	3
			Delete	3
5.	Admin	Department	Create	2
			Read	2
			Update	3
			Delete	3
6.	Admin	Staff	Create	2
			Read	2
			Update	3

			Delete	3
7.	Admin	Staff Payment	Create	4
			Read	2
			Update	4
			Delete	4
8.	Admin	Staff Task	Create	2
			View	2
9.	Admin	Bookings	View	1
			Create	2
10.	Admin	Customer Testimony	View	1
			Delete	2
11.	Admin	Staff Review	View	1
12.	Admin	Discount	Create	2
			View	1
			Delete	3
			Update	3
13.	Admin	Logout	Logout	1
14.	Customer	Add Testimony	Create	2
15.	Customer	Booking	Create	2
16.	Customer	Profile	View	2
17.	Customer	History	View	2
18.	Customer	Logout	Logout	2
19.	Staff	Task	View	1
			Update	2
20.	Staff	Review	Create	2
21.	Staff	Logout	Logout	1
22.	Website	Banner	View	1
23.	Website	Service	View	2
24.	Website	About Us	View	1
25.	Website	Contact Us	Create	2
26.	Website	Register	Create	2

3.4 Artificial Intelligence Model

Artificial intelligence is utilized in marketing campaigns since it is a very fascinating technology . It has best use where speed is critical. Artificial intelligence systems learns how to engage with clients using data and customer profiles, then give them personalised messages/ promotions etc, without the need for human intervention, ensuring increase in productivity(34, 35).

To automate the process of providing discounts to the customers is the main aim of the Artificial Intelligence implemented in this project. The artificial intelligence model takes in several inputs from the system and according to the trained model and gained coefficients from the training of the model, the model gives a certain output which is then normalised in order to get the desired output(36).

For this model, synthetic dataset has been used in order to train the model. Synthetic data is a dataset which is artificially using some logic just in order to train the model rather than by real-world events. Synthetic data is formed by using algorithms and is used as a stand-in for operational or production data test datasets, mathematical model validation, and, increasingly, machine learning model training.

A	B	C	D	E	F
Sum_of_p...	Season	Visits	Holiday	Vouchers	Discount
Number	Number	Number	Number	Number	Number
Sum_of_pa...	Season	Visits	Holiday	Vouchers	Discount
1500	0	5	0	1	1
1500	0	5	0	0	1
1000	0.33	4	0	1	0.66
1000	0.33	4	0	1	0.66
1500	0.33	4	1	1	0.66
500	1	2	0	1	0
1000	0.66	3	1	1	0.33
1000	0.66	3	0	1	0.33
1000	0.66	3	1	1	0.33
500	0.66	3	1	1	0.33
1500	0.33	4	1	1	0.66
500	1	2	0	1	0
1000	0.66	3	1	0	0.33
500	1	2	1	1	0
1500	0	5	0	1	1
1500	0	5	0	0	1

Fig. 2.28 Synthetic dataset

Dataset of 1299 is being used to implement the AI model. Once the dataset was generated and the further step was to normalise the dataset in order to feed the data as numerical values to the system. The dataset is based on 5 inputs and 1 output. Inputs taken are as follows:

1. Total sum of payments for that customer from previous bookings if there are any.
2. Which month/ season the booking is being made for.
   (Winter normalised as 0, Summer as 1, Autumn 0.33 and Spring as 0.66)
3. Previous count of number of days stayed in the hotel
4. Whether or not the date of booking has any holiday specified by the admin
5. Whether or not the customer has input any voucher code which exists in the system

The output of the system is in form of percentage of discount which should be provided to the customer for that particular booking. The Discount range from 0% up to 20% on each booking.

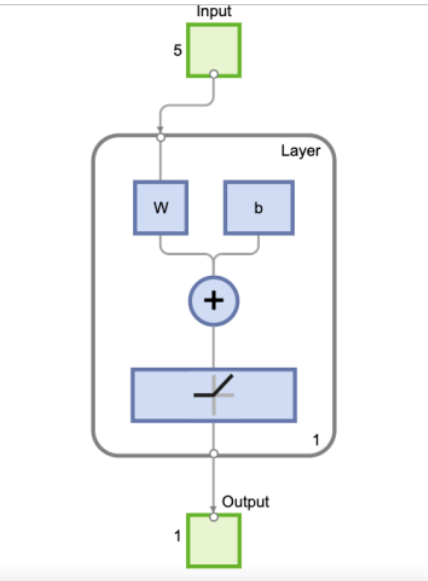


Fig. 2.29 Artificial Intelligence model

The dataset has been divided into three sections to train ,validate and test.

Table 2.52 Data distribution for the AI model

```
net.divideParam.trainRatio = 70/100;
        net.divideParam.valRatio = 15/100;
        net.divideParam.testRatio = 15/100;
```

10

10-fold cross validation would repeat the fitting method ten times, with each fit being conducted on a training set made up of 90/100 of the total set chosen at random, and the pending 10/100 used as to validate hold out set(37).

Table 2.53 10-fold cross validation method

```
indices = crossvalind('Kfold',round(T),10);
        mses = zeros(10, 1);
        for i = 1:10
           test = (indices == i);
           traini = ~test;
           Pu = P(:, traini);
           Tu = T(:, traini);
           S = 1;
           lr = maxlinlr(Pu, 'bias');
           net = newlin(Pu, S, 0, lr);
```

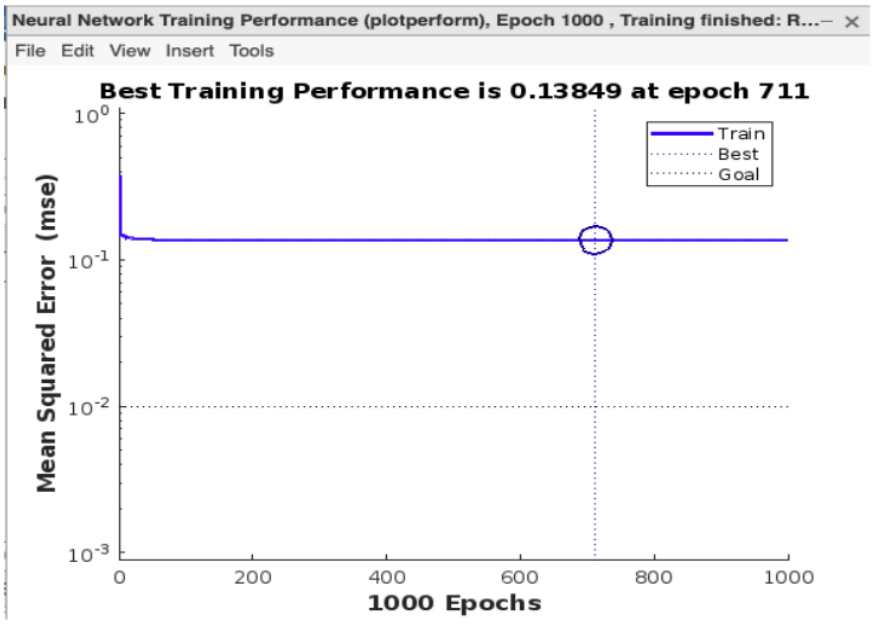


Fig. 2.30 Best training performance

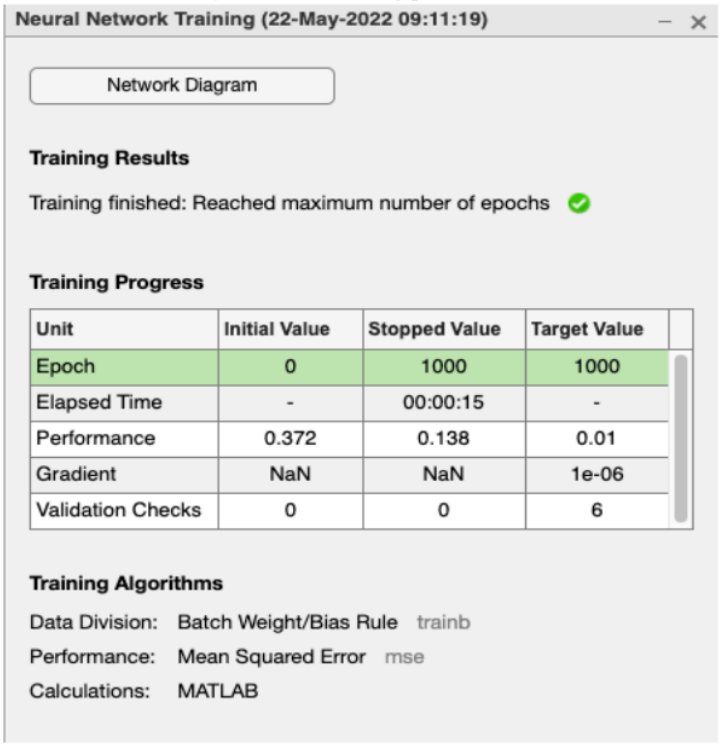


Fig. 2.31 Neural network training

These output weights and bias will be used further in the system to get the discount percentage after putting the output into the sigmoid function.

```
neuron weight coefficient values:
   -0.0740    0.0139   -0.0826    0.0434    0.0316

Bias:
    {[0.5145]}

Mses:
    0.3430
    0.3230
    0.3330
    0.3861
    0.3759
    0.3627
    0.3528
    0.3862
    0.3711
    0.3695
```

Fig. 2.32 Output Weights and bias

Table 2.54 Output after validation

Iteration number	Neural Network weight coefficients					Bias	MSE
	Sum_of_payment	Season	Visits	Holiday	Vouchers		
	Column 1	Column2	Column3	Column4	Column5		
1	-0.0827	0.0145	-0.0678	0.0402	0.0411	0.5082	0.343
2	-0.0815	0.0308	-0.0568	0.0542	0.0243	0.4975	0.323
3	-0.0626	0.0437	-0.0637	0.0527	0.0423	0.4804	0.333
4	-0.0672	0.0155	-0.0816	0.064	0.0178	0.5109	0.3861
5	-0.0876	0.0145	-0.0674	0.0492	0.0334	0.5114	0.3759
6	-0.0998	-0.0029	-0.0564	0.0386	0.0499	0.5186	0.3627
7	-0.0725	0.019	-0.0762	0.0323	0.0469	0.5105	0.3528
8	-0.0701	0.0092	-0.0745	0.0655	0.0423	0.4965	0.3862
9	-0.0642	0.0218	-0.0693	0.0629	0.0311	0.4897	0.3711
10	-0.074	0.0139	-0.0826	0.0434	0.0316	0.5145	0.3695
						Mean MSE	0.3603
						Mean Sqrt MSE	0.6372

3.5 Artificial Intelligence Implementation in the System

Implementation of Artificial intelligence model which was developed needed few steps. First and foremost the data needs to be collected from different databse tables and normalised as it was normalised in the synthetic datasheet.

Table 2.55 Input Normalization Format

Input	Actual Value	Normalised Value
Sum_of_payment: Amount paid by customer previously (previous bookings)	Integer value	No normalization required
Season Which month the booking is being made for	Winter	0
	Summer	1
	Autumn	0.33
	Spring	0.66
Visits How many days the customer has spent earlier in the hotel	Integer value	No normalization required
Holiday During the selected dates is there any day on which admin has specified discount	Yes	1
	No	0
Vouchers Did Customer used any coupon which is registered in the database of the system	Yes	1
	No	0

Data Normalization inside the websystem uses some internal function along with the conditional statements.

Table 2.56 Code Fragment: Input normalization

```
$dateHasDiscount = Discount::whereBetween('discount_date', [$request->checkin_date, $request-
>checkout_date])->get();
            if ($dateHasDiscount != null) {
                $datediscount = 1;
            } else {
                $datediscount = 0;
            }
            $hasVoucher = Voucher::where('values', $request->input('voucher'))->get();
            if ($hasVoucher != null) {
                $voucherFound = 1;
            } else {
                $voucherFound = 0;
            }
            $sumOfPayments = Payment::where('customer_id', Auth::user()->id)->sum('roomprice');
            $currentMonth = Carbon::now('m');
            //dd($currentMonth);
            if (($currentMonth=="01") || ($currentMonth=="02") || ($currentMonth=="03")) {
                $season = 0;
            } elseif (($currentMonth=="04") || ($currentMonth=="05") || ($currentMonth=="06")) {
                $season = 0.66;
            } elseif (($currentMonth=="07") || ($currentMonth=="08") || ($currentMonth=="09")) {
                $season = 1;
            } else {
                $season = 0.33;
            }

            $userPayments = Payment::where('customer_id', Auth::user()->id)->get();
```

```
            $numOfDaysStayed = 0;
            if ($userPayments != null) {
                foreach ($userPayments as $userpayment) {
                    $checkInDate = $userpayment->checkin_date;
                    $checkOutDate = $userpayment->checkout_date;
                    $startCheckIn = new DateTime($checkInDate);
                    $endCheckout = new DateTime($checkOutDate);
                    $interval = $startCheckIn->diff($endCheckout);
                    $numOfDaysStayed += $interval->format('%a');
                }
            } else {
                $numOfDaysStayed = 0;
            }
```

Second step after the data is in the format to befed to the AI system, we have to take the neural network‘s weight cofficients and bias to get a certain output by multiplying corresponding weights and inputs and adding all of them together with the bias

Table 2.57 Code Fragment: Calculation

```
$wPayment = -0.0787;
            $wSeason = 0.0152;
            $wNumOfDays = -0.0578;
            $wDate = 0.0528;
            $wVoucher = 0.0302;
            $bias = 0.5049;

            $ANN = ($datediscount * $wDate) + ($wPayment * $sumOfPayments) + ($wSeason * $season) +
                ($wVoucher * $voucherFound) + ($wNumOfDays * $numOfDaysStayed) + $bias;
```

Next thing to do after obtaining the output is to normalise the output using „Sigmoid“ activation function. This ensures that the output of the AI model is in the range of 0 and 1, hich can be further processed as per the requirement.

Table 2.58 Code Fragment: Activation Function

```
$activationFunc = (1 / (1 + exp(-1*$ANN)));
```

Once the output from the activation function is obtained, the output ranges from 0-1, now to make the output in the required from the output value needs to be intrepreted.

Table 2.59 Output interpretation

Output	**Actual Value Range**	**Normalised Value**

Output from activation function: Sigmoid function classifies the value in between 0 to 1	0	0% Discount
	0-0.33	10% Discount
	0.33-0.66	15% Discount
	0.66-1	20% Discount

Table 2.60 Code Fragment: Output interpretation

```
if (($activationFunc >= 0) && ($activationFunc <= 0.33)) {
                $discountApply = 10;
            } elseif (($activationFunc > 0.33) && ($activationFunc < 0.66)) {
                $discountApply = 15;
            } elseif (($activationFunc > 0.66) && ($activationFunc < 1)) {
                $discountApply = 20;
            } else {
                $discountApply = 0;
            }
```

4 Testing

System testing is necessary in order to see if the program is working in the manner it is supposed to and there are no logical errors or bugs in the system. In order to locate the bugs it is necessary to test the system.
The program's test is the most important aspect of its design. The user believes that the program's code is passed regardless of how it is coded or formatted as long as it can accomplish the fulfils the functions requested by the user. However, if program fails to pass the testing phase, it could lead to bugs in the systems and those bugs could trigger repeatedly, and as a result the application will occasionally crash. In this instance, even though the code is perfectly reasonable, defects will make the user think the software is bad.

4.1 Testing methods

To test the system to achieve and obtain the goals set for the system unit testing method was implemented. Since this project was done in a group which made the testing easier as the work of one programmer can be tested by the other.

1. Unit Testing
   Programmers are responsible for not just creating applications but also for unit testing(38). Unit testing is a method of determining if a source code or function developed by a programmer meets the logical expectations of the system. Each unit is tested using a short bit of code, and each of which must be tested.
2. User Testing
   This part of the testing consists of the feedback from another teammate and taking the input in order to check for any bugs in the system.

4.2 Testing Cases

Test cases specify how a software, web system, code , or application should be verified and tested. A test case is an individual activity or/and instruction that an inspector should follow to verify a specific use case/ feature of a product's or application's operation. If for some reason test fails, company will have to deal with a software fault. Different test cases were implemented and the details of the test cases implemented are mentioned (39).

Table 4.1 Test Case 1: Login as different users

Test Case	1
Function	Login as different users
Description	Check if the different subsystems can be accessed using different login id and password
Test Steps	User input credentials and press button “log in”
Expected Result	To login into subsystems
Test Result	Pass

Table 4.2 Test Case 2: banner CRUD

Test Case	2
Function	Banner CRUD
Description	The banner/ images for the website can be updated or the status of the publish can be updated and accordingly the website is updates
Test Steps	Admin clicks on the banner option and performs the desired CRUD option
Expected Result	Website banner images to be updated accordingly.
Test Result	Pass

Table 4.3 Test Case 3:Room Type CRUD

Test Case	3
Function	Room Type CRUD
Description	Room type can be updated and added as per the hotel admin's requirement and the list of room types can be seen in admin dashboard
Test Steps	Admin clicks on the room type option and performs the desired CRUD option
Expected Result	Message "Data has been saved" to be displayed
Test Result	Pass

Table 4.4 Test Case 4: Room CRUD

Test Case	4
Function	Room CRUD
Description	Room can be added or deleted as per the requirement
Test Steps	Admin can select the room type and add new rooms
Expected Result	Message "Data has been saved" to be displayed
Test Result	Pass

Table 4.5 Test Case 5: Discount

Test Case	5
Function	Discount
Description	Admin can add dates on which the discounts are available
Test Steps	Admin can click on discounts and add the dates on which the discounts are available
Expected Result	To add the dates to database
Test Result	Pass

Table 4.6 Test Case 6: Departments

Test Case	6
Function	Departments
Description	Admin can add departments as per the requirements
Test Steps	Admin can click on the departments tab and do CRUD operation
Expected Result	Data to be saved in the database and to display the data saved message
Test Result	Pass

Table 4.7 Test Case 7: Services

Test Case	7
Function	Services
Description	Admin can add services as per the requirements
Test Steps	Admin can click on the services tab and do CRUD operation
Expected Result	Data to be saved in the database and to display the data saved message
Test Result	Pass

Table 4.8 Test Case 8: Employees

Test Case	8
Function	Employees
Description	Create profile for the employees using the departments already created
Test Steps	Admin can input all the information about the employee and create a login credential for the employee
Expected Result	Profile has been created for the employee
Test Result	Pass

Table 4.9 Test Case 9: Assign task

Test Case	9
Function	Assign task
Description	Admin can assign task to the registered employees
Test Steps	Admin can click on the assign task tab and enter information
Expected Result	Data to be sent to the particular employee dashboard
Test Result	Pass

Table 4.10 Test Case 10: Manage payments

Test Case	10
Function	Manage payments
Description	Admin can enter details of the payments made to the employee
Test Steps	Admin needs to input the details in the particular employee's profile
Expected Result	List of payments to be updated
Test Result	Pass

Table 4.11 Test Case 11: Register as Customer

Test Case	11
Function	Register as customer
Description	New customers can register by inputting their details in the register form
Test Steps	User can click on the register page
Expected Result	Data to be saved in the database and customer registered
Test Result	Pass

Table 4.12 Test Case 12: Services

Test Case	12
Function	Services
Description	Services can be added from the admin panel
Test Steps	Admin can click and do CRUD operations for the services pages
Expected Result	Changes visible in the website
Test Result	Pass

Table 4.13 Test Case 13: Add testimonial

Test Case	13
Function	Add testimonial
Description	Customer can add feedback to the system
Test Steps	Customer can click on add testimonial and write the feedback
Expected Result	Admin can see the testimonies in admin panel
Test Result	Pass

Table 4.14 Test Case 14: Contact Us

Test Case	14
Function	Contact us
Description	User can send email with query to the email
Test Steps	Click on the contact us page and enter the details
Expected Result	Email to be received
Test Result	Pass

Table 4.15 Test Case 15: Room Availability

Test Case	15
Function	Room availability
Description	Customer can check which room is still available in the system
Test Steps	Click on booking, enter check-in and check-out dates
Expected Result	List of available rooms to be displayed
Test Result	Pass

Table 4.16 Test Case 16: Payment

Test Case	16
Function	To pay for the room
Description	After filling in the details the stripe API is invoked to payments
Test Steps	Click on the confirm booking
Expected Result	The data saved in the database and the payment is accepted
Test Result	Pass

Table 4.17 Test Case 17: AI discount

Test Case	17
Function	AI discount
Description	To check how much discount the AI is providing
Test Steps	After confirming the booking the actual amount and discounted prices are displayed
Expected Result	Reduced price as per the function
Test Result	Pass

Table 4.18 Test Case 18: Profile

Test Case	18
Function	Profile
Description	Customer can see the profile and points obtained
Test Steps	Click on the profile and the information is displayed
Expected Result	The points can be seen
Test Result	Pass

Table 4.19 Test Case 19: Logout

Test Case	19
Function	Logout
Description	When clicked on logout button , the user exists out of the subsystem
Test Steps	To click on the log out button
Expected Result	To exit the subsystem
Test Result	Pass

Table 4.20 Test Case 20: History

Test Case	20
Function	History
Description	Customer is able to see the previous bookings

Test Steps	Click on the history button
Expected Result	List of the previous bookings to be displayed
Test Result	Pass

5 Documentation for the user

5.1 Installation

Since the system is on the local server to install the system, few things are needed:

1. XAMPP (40)
2. PHP (41)
3. Composer
4. VS Studio
5. Git Bash

Step 1: Install the above-mentioned software. System is developed using Laravel framework.(42)
Step 2: Open XAMPP "htdocs" folder and unzip the code.
Step 3: Open Git Bash and update the composer
Step 4: Open the unzipped folder inside VS code(43)
Step 5: Delete the storage folder and create the local storage "php artisan storage: link"
Step 6: Run Apache and MySQL from XAMPP control panel
Step 7: Open PhpMyAdmin(44) and create a database named "l8_hotelmange"
Step 8: Migrate the database files "php artisan migrate"
Step 9: Basic setup is done.
Step 10: Run the project "php artisan serve"
Step 11: Click on the server.
Step 12: Register a new user as Admin.

5.2 Guide

Documentation for the users is available according to the current version of the system and nay update in later stages of development.

1. Stating the application
    - After installation of the system run XAMPP.
    - From XAMPP control panel start apache and MySQL, make sure the port is 3306. If there is any other port you can go to .env file and change the DB port.
    - Open the code in VS code, in the terminal write "php artisan serve" and wait for the system to run.
    - You will find a link after successful run. Click on the link you will directed to the website.

2. Admin Subsystem

    2.1. Register as an admin
        - Click on the register page
        - Fill in the details.
        - Go to "phpmyadmin" from your browser, open the database and select the table users

- Change the role to admin.
- Go to the website and logout.
- Click on user login, enter the credentials and login, user will be redirected to the admin panel's dashboard.

2.2. Homepage banner

2.2.1. Add New

- Click on Add new
- Enter the required fields
- Data has been added to the website

2.2.2. View All

- Click on View All
- List of Homepage banners will be displayed
- Click on the particular image that needs to be updated or deleted.
- Press the relevant button
- Click on submit and the data will be updated or deleted accordingly.

2.3. Room Type

2.3.1. Add New

- Click on Add new
- Enter the required fields
- Data has been added to the website and database

2.3.2. View All

- Click on View All
- List of room type will be displayed or you can search in the search option on the right hand top corner.
- Click on the particular image that needs to be updated or deleted or view in detail.
- Press the relevant button
- Click on submit and the data for room type will be updated or deleted accordingly.

2.4. Room

2.4.1. Add New

- Click on Add new
- Select the room type and the availability status
- Data has been added to database

2.4.2. View All

- Click on View All
- List of room type will be displayed or you can search in the search option on the right hand top corner.
- Click on the particular room that needs to be updated or deleted or view in detail.
- Press the relevant button

- Click on submit and the data for room type will be updated or deleted accordingly.

2.5. Discounts

2.5.1. Add New

- Select the date on which the discount has to be offered.
- Click on submit

2.5.2. View All

- Click on view all
- You'll see all the details with the CRUD option to manage the discount amount.

2.6. Customers

2.6.1. Add New

- Enter the details
- Data will be stored in the database if everything is uploaded correctly.

2.6.2. View All

- Click on View All
- List of customers will be displayed
- Click on the CRUS option or search for the customer from the search option available
- Update the data as per requirement

2.7. Departments

2.7.1. Add New

Enter details for the new department to be added in the hotel

2.7.2. View All

List of departments will be displayed along with the CRUD

2.8. Staff

2.8.1. Add New

- Enter the details
- Data will be stored in the database if everything is uploaded correctly.

2.8.2. View All

- Click on View All
- List of staff will be displayed
- Click on the CRUS option or search for the staff from the search option available
- Update the data as per requirement

2.8.3. Payments

- Click on view all
- Click on a particular staff and click on add payment
- Enter the amount paid and payment date.
- You can delete the payment if it is done.

2.8.4. Manage Staff

Add task for the staff

2.8.5. View Manage Staff

Status of the task assigned

2.9. Bookings

- Click on bookings
- You can add new booking by selecting customer, data and the room and add the booking to the system.
-

2.10. Services

- Click on services
- You can add new services with in short description and a brief description about the service.
- You can view all the services and manage them using CRUD

2.11. Customer Testimonial

When the customer adds the testimonial, the admin can view or delete it from the system.

2.12. Logout

Click on logout, you'll be logged out of the admin subsystem

3. Customer Subsystem

3.1. Register as a customer

- Click on register
- Enter the details
- Then go to user login, put your credentials and you'll be logged in to the customer subsystem.

3.2. Homepage

When the customer login into the system under the name there is a drop bar with three options,

My profile : It shows the details of the customer
My history: History of the bookings done by the customer in the past
Logout: to exit the customer subsystem.

3.3. Booking

- Select the date of the bookings
- Enter the details and select the room

- Payment API will be invoked for payment after discount

3.4. Add Testimonial

Customer can add testimonial which will be sent to the admin panel and also updated in the website.

4. Staff Subsystem

4.1. Login as Staff

Click on the staff login and enter the credentials provided by the admin and user will be logged into the staff subsystem.

4.2. View Task

Employee after logging in can see what tasks has been assigned to him/her by the admin

4.3. Feedback

Employee can go to feedback page and but feedback for particular customers.

4.4. Update the task

Employee can update the status of the task assigned whether it is in progress or done.

Conclusions

Throughout the research, it was established that there are several various kinds of hotel management systems, and these systems have helped multiple companies throughout the world flourish. Systems may aid an organizational framework and operations by saving time, cash, and labor. These technologies have offered businesses with a large database inventory to store and manage all of its personnel and bookings.

System analysis is an important a part of the project. System analysis is required for designing and implementing the plan. The issues are analysed to see the character of the system. System Analysis finds the various phase of a system and components of system design.

The main aim and objectives of the project have been successfully implemented. We have included features and operations in more detailed way.

1. Web system which is easy to navigate has been achieved. The focus was to use the same user interface design and add additional tabs when the user is logged in as different roles. It is easy to navigate since all the options can be found on the ribbon and could be accessed from anywhere in the system.
2. To automate the room booking process the system to checks which rooms are available in the system and when the user tried to book the room the system checks the availability in the given dates. Admin can also put in the information about the all the additional services offered by the hotel. Artificial intelligence to get the discounts using Artificial neural network was achieved and the coefficients and the bias was used in the system with the normalised output. The obtained result is then again classified into the classes and the results are used at the percentage of discounts that can be offered to the customer based on several factors according to the artificial intelligence model.
3. In order to achieve proper feedback channel, all the information had to be linked with the admin subsystem and proper communication between the other subsystems was necessary. Admin subsystem acts as a mediator and can act upon the feedbacks accordingly. The feedback given by the customer is visible on the website and the feedback from the employee is sent to the admin to process.

References

1. Why Hotel Reservation Systems Are Important | SiteMinder. Online. [Accessed 22 May 2022]. Available from: https://www.siteminder.com/r/technology/hotel-cloud-technology/why-reservation-system-important-hotel/

2. Hotel Property Management Systems (PMS): Products and Features | AltexSoft. Online. [Accessed 22 May 2022]. Available from: https://www.altexsoft.com/blog/travel/hotel-property-management-systems-products-and-features/

3. O'CONNOR, Peter and MURPHY, Jamie. Research on information technology in the hospitality industry. *International Journal of Hospitality Management*. 1 December 2004. Vol. 23, no. 5, p. 473–484. DOI 10.1016/J.IJHM.2004.10.002.

4. (PDF) E-Commerce: An Opportunity For Hospitality Industry Growth. Case Study Of Hotels In Accra | GOODLET OWUSU ANSAH - Academia.edu. Online. [Accessed 22 May 2022]. Available from: https://www.academia.edu/13246498/E-Commerce_An_Opportunity_For_Hospitality_Industry_Growth._Case_Study_Of_Hotels_In_Accra

5. JOY, Catherine R. Design and Implementation of an Automated Hotel Management System. *International Journal of Engineering and Advanced Technology (IJEAT)*. Online. 2021. P. 2249–8958. [Accessed 22 May 2022]. DOI 10.35940/ijeat.E2569.0610521.

6. FOUAD, Ali, PHALP, Keith, KANYARU, John Mathenge and JEARY, Sheridan. Embedding requirements within Model-Driven Architecture. *Software Quality Journal*. 2011. Vol. 19, no. 2, p. 411–430. DOI 10.1007/S11219-010-9122-7.

7. Importance of Customer Feedback Software In The Hotel Industry. Online. [Accessed 22 May 2022]. Available from: https://www.pihappiness.com/2019/11/importance-of-customer-feedback-software-in-the-hotel-industry/

8. Advantages & Disadvantages of Manual Accounting Systems. Online. [Accessed 22 May 2022]. Available from: https://smallbusiness.chron.com/advantages-disadvantages-manual-accounting-systems-23862.html

9. Hotel Guest Feedback is Key to Financial Success - HMG Hospitality. Online. [Accessed 22 May 2022]. Available from: https://hmghospitality.com/hotel-guest-feedback/

10. DEETI, Sumanth, MANNE, Sravani and VUNDAVALLI, Veerendra. Hotel Management System Part of the Databases and Information Systems Commons Recommended Citation. Online. [Accessed 22 May 2022]. Available from: http://opus.govst.edu/capstoneshttp://opus.govst.edu/capstones/198

11. HAM, Sunny, KIM, Woo Gon and JEONG, Seungwhan. Effect of information technology on performance in upscale hotels. *International Journal of Hospitality Management*. 1 June 2005. Vol. 24, no. 2, p. 281–294. DOI 10.1016/J.IJHM.2004.06.010.

12. How to choose the best revenue management systems for hotels | Mews | Mews Blog. Online. [Accessed 22 May 2022]. Available from: https://www.mews.com/en/blog/best-revenue-management-systems-hotels

13. Hotel Marketing: The Latest Trends in the Hotel Industry for 2022. Online. [Accessed 22 May 2022]. Available from: https://www.revfine.com/hotel-marketing/

14. SaaS Hotel Management Software: Independent Hotel Benefits – stayntouch. Online. [Accessed 22 May 2022]. Available from: https://www.stayntouch.com/blog/benefits-of-saas-hotl-managment-software-for-independently-owned-hotels/

15. . 0 Objectives: 1.1 Introduction 1.2 Over View of System Analysis and Design 1.3 Business System Concepts 1.4 Characteristics of a System 1.5 Elements of a System 1.6 Types of Systems 1.7 Systems Models 1.8 Categories of Information. .

16. (PDF) Design and analysis of hotel management system based on information technology | Magnon Be7wak - Academia.edu. Online. [Accessed 22 May 2022]. Available from: https://www.academia.edu/42669385/Design_and_analysis_of_hotel_management_system_based_on_information_technology

17. 10 Best Hotel Management Software - 2022 Reviews & Pricing. Online. [Accessed 22 May 2022]. Available from: https://hoteltechreport.com/operations/hotel-management-software

18. Top Hotel Management Software - 2022 Reviews & Pricing. Online. [Accessed 22 May 2022]. Available from: https://www.softwareadvice.com/hotel-management/

19. Award-Winning Hospitality Management Software - Cloudbeds. Online. [Accessed 24 May 2022]. Available from: https://www.cloudbeds.com/

20. RoomRaccoon HMS Reviews: Pricing & Software Features - 2022 - Hotel Tech Report. Online. [Accessed 22 May 2022]. Available from: https://hoteltechreport.com/operations/hotel-management-software/roomraccoon-hms

21. Hotel Program - Time-saving Hotel Software | RoomRaccoon. Online. [Accessed 24 May 2022]. Available from: https://roomraccoon.co.uk/hotel-program/

22. Little Hotelier | We make running your small hotel easy. Online. [Accessed 22 May 2022]. Available from: https://www.littlehotelier.com/?utm_source=google&utm_medium=cpc&utm_campaign=LH_G_EMEA_UK_EN_Exact_Brand_Core&gclid=Cj0KCQjwvqeUBhCBARIsAOdt45aEHMtJSfgOOvcqTUUXvvRhDSpGUop5CgdR55RXYOoMdG54bWbxGDMaAsz1EALw_wcB&gclsrc=aw.ds

23. Little Hotelier Front Desk - Search, view and edit reservations. Online. [Accessed 24 May 2022]. Available from: https://help.siteminder.com/s/article/Edit-reservation?language=en_US

24. About Us | RMS. Online. [Accessed 22 May 2022]. Available from: https://www.rmscloud.com/about-us

25. RMS - The Hospitality Cloud. Online. [Accessed 24 May 2022]. Available from: https://www.rmsnorthamerica.com/

26. Hotel Revenue Management Software | InnRoad. Online. [Accessed 24 May 2022]. Available from: https://www.innroad.com/product/demand-management/

27. RMS Cloud vs Cloudbeds vs Little Hotelier. Online. [Accessed 24 May 2022]. Available from: https://www.softwareadvice.com/hotel-management/rms-profile/vs/cloudbeds/littlehotelier/

28. Compare Hotel Technology Vendors. Online. [Accessed 24 May 2022]. Available from: https://hoteltechreport.com/compare

29. Why Laravel is the Best PHP Framework to Use in 2020. Online. [Accessed 24 May 2022]. Available from: https://www.freecodecamp.org/news/why-laravel-is-the-best-php-framework-to-use-in-2020/

30. What is a Functional Requirement in Software Engineering? Specification, Types, Examples. Online. [Accessed 22 May 2022]. Available from: https://www.guru99.com/functional-requirement-specification-example.html

31. UML Use Case Diagram Tutorial | Lucidchart. Online. [Accessed 22 May 2022]. Available from: https://www.lucidchart.com/pages/uml-use-case-diagram

32. What is Use Case Diagram? Online. [Accessed 22 May 2022]. Available from: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

33. How to Improve Website Navigation (Examples and Why You Should Do It). Online. [Accessed 22 May 2022]. Available from: https://kinsta.com/blog/website-navigation/

34. What is Artificial Intelligence (AI)? | Glossary | HPE United Kingdom. Online. [Accessed 22 May 2022]. Available from: https://www.hpe.com/uk/en/what-is/artificial-intelligence.html?jumpid=ps_3hmavknz8b_aid-520061736&ef_id=Cj0KCQjwvqeUBhCBARIsAOdt45YowQYs-ieE6Vpjz7Yhfs_r9jOEleBJiztV1bQo1xy7N5RsxTgfOO0aAtlGEALw_wcB:G:s&s_kwcid=AL!13472!3!558265259459!e!!g!!what%27s%20ai!14644166551!129071271557&

35. AI @ Scale | AI Consulting and Strategy | BCG. Online. [Accessed 22 May 2022]. Available from: https://www.bcg.com/capabilities/digital-technology-data/artificial-intelligence?utm_source=search&utm_medium=cpc&utm_campaign=digital&utm_description=none&utm_topic=ai&utm_geo=global&utm_content=ai-in-action&gclid=Cj0KCQjwvqeUBhCBARIsAOdt45awRKVL2vjZ7goOZ_yw2gpLDpKyZj-hvR164N5Pdsvu3yCs_1WGWmkaAnxEEALw_wcB

36. PATHAK, Avneesh, SNGHAL, Arun and KUMAR, Pushkar. A New Approach to Hotel Management System. . 2021.

37. ROBERTS, David R., BAHN, Volker, CIUTI, Simone, BOYCE, Mark S., ELITH, Jane, GUILLERA-ARROITA, Gurutzeta, HAUENSTEIN, Severin, LAHOZ-MONFORT, José J., SCHRÖDER, Boris, THUILLER, Wilfried, WARTON, David I., WINTLE, Brendan A., HARTIG, Florian and DORMANN, Carsten F. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*. 1 August 2017. Vol. 40, no. 8, p. 913–929. DOI 10.1111/ECOG.02881.

38. What is Unit Testing? Definition from WhatIs.com. Online. [Accessed 22 May 2022]. Available from: https://www.techtarget.com/searchsoftwarequality/definition/unit-testing

39. What Is a Test Case? Examples, Types and Format - Applause. Online. [Accessed 22 May 2022]. Available from: https://www.applause.com/blog/what-is-a-test-case-examples-types-format

40. XAMPP Installers and Downloads for Apache Friends. Online. [Accessed 22 May 2022]. Available from: https://www.apachefriends.org/index.html

41. PHP: Hypertext Preprocessor. Online. [Accessed 22 May 2022]. Available from: https://www.php.net/

42. Best Laravel Development and Web Services Company in London. Online. [Accessed 22 May 2022]. Available from: https://larasoft.io/?gclid=Cj0KCQjwvqeUBhCBARIsAOdt45YzmpAI-CraWNhQAxh7_ZKgh8ZTpdzBTy1TORKmUy0leTD7QYtIk5waAtV1EALw_wcB

43. Visual Studio Code - Code Editing. Redefined. Online. [Accessed 22 May 2022]. Available from: https://code.visualstudio.com/

44. phpMyAdmin. Online. [Accessed 22 May 2022]. Available from: https://www.phpmyadmin.net/

Appendices

Annex 1: Source Code of the System Developed

ThakurVarshaNalgndaRakesh_2022_finalproject

ORIGINALITY REPORT

5%	2%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.altexsoft.com Internet Source	1%
2	Submitted to University of Wales, Lampeter Student Paper	<1%
3	Submitted to Middlesex University Student Paper	<1%
4	Submitted to Mont Blanc Palace Student Paper	<1%
5	Submitted to Asia Pacific Instutute of Information Technology Student Paper	<1%
6	inba.info Internet Source	<1%
7	Submitted to Macquarie University Student Paper	<1%
8	Submitted to Wellington Institute of Technology Student Paper	<1%
9	sourceforge.net	

	Internet Source	<1%
10	Submitted to Liverpool John Moores University Student Paper	<1%
11	Submitted to Taylor's Education Group Student Paper	<1%
12	Submitted to Saint Mary's University of Minnesota Student Paper	<1%
13	Submitted to Informatics Education Limited Student Paper	<1%
14	Submitted to University of Central Lancashire Student Paper	<1%
15	Submitted to University of Westminster Student Paper	<1%
16	www.roombookingsystem.co.uk Internet Source	<1%
17	Ivo Gamito, Alberto Rodrigues da Silva. "Chapter 37 From Rigorous Requirements and User Interfaces Specifications into Software Business Applications", Springer Science and Business Media LLC, 2020 Publication	<1%
18	Submitted to Southampton Solent University Student Paper	<1%

19	Submitted to Icon College of Technology and Management Student Paper	<1 %
20	Submitted to University of London External System Student Paper	<1 %
21	opus.govst.edu Internet Source	<1 %
22	zestedesavoir.com Internet Source	<1 %
23	Submitted to University of Wolverhampton Student Paper	<1 %
24	www.cs.kuleuven.be Internet Source	<1 %
25	Submitted to Softwarica College Of IT & E-Commerce Student Paper	<1 %
26	www.zentralblatt-math.org Internet Source	<1 %
27	Submitted to University of Hertfordshire Student Paper	<1 %
28	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %

29	Submitted to University of Newcastle upon Tyne Student Paper	<1 %
30	Submitted to Colorado Technical University Student Paper	<1 %
31	smallbusiness.chron.com Internet Source	<1 %
32	Submitted to University of Bedfordshire Student Paper	<1 %
33	hmghospitality.com Internet Source	<1 %
34	Butleris, Rimantas, Audrius Lopata, Martas Ambraziunas, Ilona Veitaitė, and Saulius Masteika. "SysML and UML models usage in Knowledge Based MDA process", Elektronika ir Elektrotechnika, 2015. Publication	<1 %
35	www.guru99.com Internet Source	<1 %
36	Submitted to Hotelschool Den Haag Student Paper	<1 %
37	Submitted to School of Business & Computer Science Limited Student Paper	<1 %
38	Submitted to University of Liverpool Student Paper	

		<1%
39	ideas.repec.org Internet Source	<1%
40	Lawrence Chung, Brian A. Nixon, Eric Yu, John Mylopoulos. "Non-Functional Requirements in Software Engineering", Springer Science and Business Media LLC, 2000 Publication	<1%
41	Yuh-Min Chen, Yann-Daw Jan. "Enabling allied concurrent engineering through distributed engineering information management", Robotics and Computer-Integrated Manufacturing, 2000 Publication	<1%
42	altexsoft.medium.com Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

ThakurVarshaNalgndaRakesh_2022_finalproject

GRADEMARK REPORT

FINAL GRADE

/100

GENERAL COMMENTS

Instructor

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41

PAGE 42

PAGE 43

PAGE 44

PAGE 45

PAGE 46

PAGE 47

PAGE 48

PAGE 49

PAGE 50

PAGE 51

PAGE 52

PAGE 53

PAGE 54

PAGE 55

PAGE 56

PAGE 57

PAGE 58

PAGE 59

PAGE 60

PAGE 61

PAGE 62

PAGE 63

PAGE 64

PAGE 65

PAGE 66

PAGE 67

PAGE 68

PAGE 69

PAGE 70

PAGE 71

PAGE 72

PAGE 73

PAGE 74

PAGE 75

PAGE 76

PAGE 77

PAGE 78

PAGE 79

PAGE 80

PAGE 81

PAGE 82

PAGE 83

PAGE 84

PAGE 85

PAGE 86

PAGE 87

PAGE 88

PAGE 89

PAGE 90

PAGE 91

PAGE 92

PAGE 93

PAGE 94

PAGE 95

PAGE 96

PAGE 97

PAGE 98