



K.L.E.Society's
C.B.KORE POLYTECHNIC
CHIKODI-591201



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

2024-2025

Internship Report On

OJT-1:-Hotel Management System

OJT-2:-Hospital Management System

In partial fulfilment for the award of the diploma of

DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

ABHISHEK SHINTRE

[339CS23701]

DINESH ASHOK JADAGE

[339CS22016]

ISHWAR RAJENDRA GANDH

[339CS22020]

KISHAN NINGANAGAUDA MANGASULE

[339CS22024]

UNDER GUIDANCE OF

Mr. Anoop Desai



K.L.E.Society's
C.B.KORE POLYTECHNIC
CHIKODI-591201



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

2024-2025

Internship Report On

OJT-1:-Hotel Management System

OJT-2:-Hospital Management System

In partial fulfilment for the award of the diploma of

DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

ABHISHEK SHINTRE

[339CS23701]

DINESH ASHOK JADAGE

[339CS22016]

ISHWAR RAJENDRA GANDH

[339CS22020]

KISHAN NINGANAGAUDA MANGASULE

[339CS22024]

UNDER GUIDANCE OF

Mr. Annop Desai

Guide
Mr.Anoop Desai

Cohort Owner
Prof S.S.Khot

Head of Department
Prof S.K.Tavadare

Principal
Prof D.D.Billur



K.L.E. Society's
C.B.KORE POLYTECHNIC
CHIKODI-591201



Department of Computer Science & Engineering

CERTIFICATE

This is to certify that the Internship report entitled. "Hotel Management System & Hospital-Management System "was carried out by Mr. Abhishek Shintre , Mr. Ishwar Gandh , Mr. Kishan Mangasule & Mr. Dinesh Jadage of Sixth semester Diploma in Computer Science and Engineering, K.L.E. Society's C. B. Kore Polytechnic, Chikodi. In practical fulfillment of the Diploma in Computer Science and Engineering award requirement during 2024-2025. This work has been carried out under the guidance of Prof. Anoop. J. Desai, Pranaksh Technologies LLP, Belagavi. The Internship work prescribed for the said Diploma.

It is further understood that by his certificate the undersigned does not endorse or approve any Statement made, opinion Expressed ,or conclusion purpose which it is submitted.

Guide	Cohort Owner	Head of Department	Principal
Mr.Anoop Desai	Prof S.S.Khot	Prof S.K.Tavadare	Prof D.D.Billur

Name and Signature of Examiners:

1. _____

2. _____

INTERNSHIP CERTIFICATION ISSUED BY THE ORGANIZATION



Pranaksh Technologeis LLP

LLPIN: ACC-8635

07 April 2025

TO WHOMSOEVER IT MAY CONCERN

This is to certify **Abhishek Shintre**, bearing USN No: **339CS23701** from KLE Society's Chidanand B. Kore Polytechnic, Chikodi, **Belagavi** has successfully completed 640 Hour internship starting from **16-12-2024** to **31-03-2025** under the mentorship of Pranaksh development team. **Abhishek Shintre** has worked on Full Stack Web Development with Java.

The domain & agile development process exposure was given along with usage of GitHub tool. During the internship, **Abhishek Shintre** demonstrated good coding skills with good design thoughts.

We wish all the best for future endeavor's!



Certificate ID: **MAR2025JFSS166025**

For Pranaksh Technologies LLP
Director



Pranaksh Technologieis LLP

LLPIN: ACC-8635

07 April 2025

TO WHOMSOEVER IT MAY CONCERN

This is to certify **Ishwar Gandh**, bearing USN No: **339CS22020** from KLE Society's Chidanand B. Kore Polytechnic, Chikodi, **Belagavi** has successfully completed 640 Hour internship starting from **16-12-2024** to **31-03-2025** under the mentorship of Pranaksh development team. **Ishwar Gandh** has worked on Full Stack Web Development with Java.

The domain & agile development process exposure was given along with usage of GitHub tool. During the internship, **Ishwar Gandh** demonstrated good coding skills with good design thoughts.

We wish all the best for future endeavor's!



Certificate ID: **MAR2025JFSS166041**

For Pranaksh Technologies LLP
Director

Registered office: 284/1, Plot No. 3, Mutage, Belgaum, Karnataka, India, 591124

M: 8867635488 | www.pranakshtechnologies.com | info@pranaksh.com



Pranaksh Technologieis LLP

LLPIN: ACC-8635

07 April 2025

TO WHOMSOEVER IT MAY CONCERN

This is to certify **Dinesh Jadage**, bearing USN No: **339CS22016** from KLE Society's Chidanand B. Kore Polytechnic, Chikodi, **Belagavi** has successfully completed 640 Hour internship starting from **16-12-2024** to **31-03-2025** under the mentorship of Pranaksh development team. **Dinesh Jadage** has worked on Full Stack Web Development with Java.

The domain & agile development process exposure was given along with usage of GitHub tool. During the internship, **Dinesh Jadage** demonstrated good coding skills with good design thoughts.

We wish all the best for future endeavor's!



Certificate ID: **MAR2025JFSS166027**

For Pranaksh Technologies LLP
Director



Pranaksh Technologieis LLP

LLPIN: ACC-8635

07 April 2025

TO WHOMSOEVER IT MAY CONCERN

This is to certify **Kishan Mangasule**, bearing USN No: **339CS22024** from KLE Society's Chidanand B. Kore Polytechnic, Chikodi, **Belagavi** has successfully completed 640 Hour internship starting from **16-12-2024** to **31-03-2025** under the mentorship of Pranaksh development team. **Kishan Mangasule** has worked on Full Stack Web Development with Java.

The domain & agile development process exposure was given along with usage of GitHub tool. During the internship, **Kishan Mangasule** demonstrated good coding skills with good design thoughts.

We wish all the best for future endeavor's!



Certificate ID: **MAR2025JFSS166024**

For Pranaksh Technologies LLP
Director

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all those who have supported and guided me throughout the successful completion of this project.

First and foremost, I would like to thank Prof. DARSHANKUMAR D. BILLUR, Principal of K.L.E. SOCIETY'S C.B. KORE POLYTECHNIC CHIKODI-591201, for providing the necessary facilities and a motivating environment to carry out this work.

I express my deep sense of gratitude to Prof. SANTOSH K TAVADARE, Head of the Department of COMPUTER SCIENCE AND ENGINEERING, for their constant encouragement and valuable support during the course of this project.

I am especially thankful to my project guide, Prof. ANOOP J. DESAI, for their consistent guidance, suggestions, and valuable feedback that helped shape this project into its final form.

I would also like to thank all the faculty members and staff of the COMPUTERSCIENCE AND ENGINEERING for their help and cooperation throughout the duration of this project. Last but not the least, I am deeply thankful to my family and friends for their unwavering support ,patience ,and encouragement during every stage of this journey.

EXECUTIVE SUMMARY

This Internship explores the integrated management practices in two essential service sectors: healthcare (hospital management) and hospitality (hotel management). Both industries, though distinct in their core functions, share common managerial principles such as customer service excellence, operational efficiency, financial stability, and human resource management. This merged project aims to examine how effective management in hospitals and hotels can improve service delivery, operational workflows, and profitability, while also focusing on customer satisfaction, compliance, and sustainability

Hospital Management: Ensuring quality patient care and satisfaction through streamlined processes, skilled professionals, and patient-centric services. Hospital Management: Effective budgeting, cost control, and revenue cycle management are vital for maintaining financial health. Hospitals must balance medical expenditures with patient care while complying with insurance and billing regulations. Hospital Management: Streamlining operations across departments such as nursing, pharmacy, laboratory, and administrative units to improve workflow and minimize operational costs while enhancing patient outcomes

Hotel Management: Optimizing room management, housekeeping, food and beverage services, and maintenance to ensure smooth operations, reduce costs, and maximize guest satisfaction. In both sectors, leveraging technology and data-driven decision-making is key to improving operations, such as through healthcare management systems in hospitals and property management systems in hotels. Hotel Management: Dynamic pricing, revenue optimization strategies, and cost control measures are essential to ensuring profitability, particularly in managing seasonal fluctuations in demand. Both industries require a keen focus on budgeting, resource allocation, and financial reporting to achieve financial sustainability.

Contents
[OJT-1] Hotel Management System

Sl No	Contents	Page No
1	Abstract	2-2
2	Introduction	3-3
3	Existing System and Literature Survey 1. Existing System 2. Literature Survey	4-6
4	Problem Statement and Objectives 1. Problem Definition 2. Challenges 3. Limitations 4. Applications	7-8
5	Requirements 1. Hardware Requirements 2. Software Requirements 3. Functional Requirements 4. Non-Functional Requirements	9-10
6	System Design and Methodology 1. Methodology 2. Software Architecture 3. Database Design 4. UI Design 5. Module Design 6. ER Diagram	11-14
7	Pseudocode	15-18
8	Experiments and Results	19-21
9	Test Cases	22-27
10	Future Scope	28-28
11	Conclusion	29-29
12	Reference	30-30

Contents
[OJT-2] Hospital Management System

Sl No	Contents	Page No
1	Abstract	32-32
2	Introduction	33-33
3	Existing System and Literature Survey 3. Existing System 4. Literature Survey	34-36
4	Problem Statement and Objectives 5. Problem Definition 6. Challenges 7. Limitations 8. Applications	37-38
5	Requirements 5. Hardware Requirements 6. Software Requirements 7. Functional Requirements 8. Non-Functional Requirements	39-40
6	System Design and Methodology 7. Methodology 8. Software Architecture 9. Database Design 10. UI Design 11. Module Design 12. ER Diagram	41-44
7	Pseudocode	45-47
8	Experiments and Results	48-50
9	Test Cases	51-55
10	Future Scope	56-56
11	Conclusion	57-57
12	Reference	58-58

LIST OF FIGURES

Figure No.	Title of the Figure	Page.no
2.1	ER DIAGRAM	15
2.2	Screenshots of Sign up page	20
2.3	Screenshots of Login page	20
2.4	Screenshots of Home page	21
2.5	Screenshots of Booking Page	21
2.6	Screenshots of Staff Dashboard	22
2.7	Screenshots of Payment	22
3.1	ER DIAGRAM	46
3.2	Screenshots of Admin/Patient/Doctor login	50
3.3	Screenshots of Patient Dashboard	50
3.4	Screenshots of Appointment	51
3.5	Screenshots of ADD Doctor	51
3.6	Screenshots of Patient management	52
3.7	Screenshots of Appointment patient	52

ABBREVIATION / NOATIONS / NOMENCLATURE

Abbreviation / Symbol	Full From /Description
IOT	Internet of Things
API	Application Programming Interface
GUI	Graphical User interface
CPU	Central Processing Unit
RAM	Random Access Memory
HTML	HyperText Markup Language
DFD	Data Flow Diagram
UML	Unified Modeling Language
DBMS	Database Management System
SQL	Structured Query Language
HTTP	HyperText Transfer protocol
IP	Internet Protocol
JSON	JavaScript Object Notation
MBPS	Megabits per Second

CHAPTER-1

COMPANY PROFILE

Pranaksh Technologies, established in 2023, is a dynamic company specializing in technology solutions, digital marketing, and academic training. Initially focused on educational technology services for IT companies, the firm has expanded its expertise to include software development, artificial intelligence, embedded systems, cybersecurity, and the Internet of Things (IoT).

Vision and Mission

Pranaksh Technologies aims to be a trusted leader in delivering high-quality services that bridge the gap between academia and the corporate world. Their mission is to provide exceptional solutions and training that connect academic learning with corporate success, empowering individuals and organizations with cutting-edge skills and knowledge.

Services Offered

- **Corporate Solutions:** Offering smart strategies and tools to boost efficiency, tackle challenges, and drive growth, ensuring success in today's competitive market.
- **Academic Training:** Providing hands-on expertise in fields like AI and Data Science, equipping individuals with the skills needed to stay competitive in a rapidly evolving landscape.
- **Digital Marketing:** Delivering tailored strategies in SEO, content, and social media to captivate audiences and drive impressive results.

Technology Focus

The company emphasizes several key technological areas

- Internet of Things (IoT)
- Data Science
- Artificial Intelligence & Machine Learning
- Web Development

These focus areas underscore Pranaksh Technologies' commitment to innovation and staying at the forefront of technological advancements.

Milestones

Since its inception, Pranaksh Technologies has achieved significant milestones:

- 50+ Happy Clients
- 100+ Students Trained
- 150+ Projects Completed
- 10,000+ Training Hours Delivered

These accomplishments reflect the company's dedication to excellence and client satisfaction.

Why Choose Pranaksh Technologies

The company's value-driven approach ensures cost-effective services without compromising quality. They emphasize innovative partnerships, working closely with clients to deliver creative solutions. With a focus on trust and tangible results, Pranaksh Technologies prepares individuals with corporate-ready skills through programs that mirror real-world expectations.

Contact Information

Pranaksh Technologies is located in Belagavi, Karnataka. For inquiries:

- Email: info@pranakshtechnologies.com
- Phone: +91-8867635488

For more detailed information, please visit their official website: <https://pranakshtechnologies.com>

CHAPTER-2

ON JOB TRAINING-1

“HOTEL MANAGEMENT SYSTEM”

Abstract

The Hotel Management System is a comprehensive software application designed to streamline and automate the day-to-day operations of hotels and hospitality establishments. This system facilitates efficient handling of key processes such as room bookings, check-in/check-out procedures, guest management, billing, housekeeping, and staff administration. The aim is to replace manual, error-prone processes with a centralized and user-friendly digital solution.

The system offers a multi-user environment with distinct access levels for administrators, receptionists, and other staff members. Through its intuitive interface, the application allows for real-time room availability tracking, seamless reservation management, automated billing, and secure storage of guest data. Integration of analytics and reporting features enables hotel managers to make data-driven decisions and enhance operational efficiency.

By incorporating technologies such as database management systems and modern programming frameworks, the Hotel Management System ensures accuracy, scalability, and security. Ultimately, the system enhances the overall guest experience and supports hotel staff in delivering prompt, personalized service, thus contributing to improved customer satisfaction and business growth.

INTRODUCTION

Hotel management is the practice of overseeing the day-to-day operations of a hotel, ensuring that all departments, from front desk services to housekeeping and food & beverage, function smoothly and efficiently. It involves a range of responsibilities such as managing guest services, maintaining cleanliness and safety standards, optimizing financial performance, and overseeing marketing and sales strategies.

A successful hotel manager must balance customer satisfaction with operational efficiency, handling everything from staff recruitment and training to budgeting and revenue management. They must also embrace technology, using systems for reservations, guest services, and maintenance, while also responding to guest feedback to improve the overall experience.

In addition to traditional tasks, many hotels are now focusing on sustainability and green practices to attract environmentally-conscious guests. Effective hotel management requires strong leadership, problem-solving, and financial acumen, making it a dynamic and multifaceted career that blends hospitality with business strategy.

The core elements of hotel management include operations, human resources, financial management, marketing, guest services, and technology. In terms of operations, hotel managers must oversee departments like front desk, housekeeping, food and beverage, and maintenance. Each of these departments is crucial to ensuring that the guest experience is seamless and enjoyable. For instance, front desk operations are the first point of contact for guests and handle check-ins, check-outs, and special requests, while house keeping maintains cleanliness and ensures that rooms meet the hotel's standards.

Financial management in hotel management encompasses budgeting, accounting, and revenue management. This includes setting room rates, forecasting demand, and using data-driven strategies to maximize occupancy rates and profitability.

EXISTING SYSTEM AND LITERATURE SURVEY

Existing System

The existing system in hotel management encompasses a combination of traditional practices and modern technology, which together enable the smooth operation of hotels. Historically, many hotels relied on manual processes such as paper logs for guest check-ins, reservations, and housekeeping tracking, but today, most use advanced Property Management Systems (PMS) to integrate various hotel operations like bookings, billing, and guest profiles.

These systems streamline tasks, making day-to-day management more efficient, but many older hotels still use out dated versions that lack integration with newer technologies, leading to potential inefficiencies. Online booking engines and Global Distribution Systems (GDS) have become integral for reservations, allowing hotels to manage bookings across multiple channels like OTAs, travel agents, and direct website bookings.

Additionally, Point of Sale (POS) systems are used for managing transactions in the food and beverage sector, while human resource management systems help with employee scheduling, training, and performance tracking. Financially, hotels use a mix of automated and manual accounting systems, though older methods can lead to discrepancies and errors. Marketing efforts have largely shifted to digital channels, with many hotels focusing on social media, SEO, and online advertising.

Despite the advancements, challenges persist, including the fragmentation of technologies, lack of real-time data, and reliance on outdated systems, which can hinder operational efficiency and affect guest satisfaction. Overall, the existing hotel management system is a mix of manual and automated processes, and many hotels are working to integrate more modern solutions to stay competitive and enhance guest experiences.

Literature Survey

The development of Hotel Management Systems has evolved significantly with advances in information technology, aiming to replace traditional manual methods with efficient, automated solutions. Various studies and existing systems have been reviewed to understand their architecture, functionality, and impact on the hospitality industry.

In early implementations, hotel operations were managed using basic spreadsheets or desktop-based applications with limited functionality and no real-time data synchronization. According to Gupta et al. (2015), manual systems were prone to errors, inefficient during peak times, and lacked proper guest history tracking. Modern systems address these issues through integrated platforms that combine reservation, billing, and guest management in real-time.

A study by Sharma & Jain (2018) emphasized the growing demand for cloud-based HMS, which offer scalability, remote access, and integration with third-party services like payment gateways and online travel agencies (OTAs). These systems enable hotels to manage operations more flexibly and cost-effectively. Similarly, research by Rao et al. (2020) highlighted the importance of mobile-friendly interfaces and real-time updates in enhancing user experience and operational efficiency.

Existing commercial solutions such as Opera PMS, Cloudbeds, and eZee Absolute provide insights into industry standards. They support features like dynamic pricing, analytics dashboards, multi-property management, and CRM integration. These systems also demonstrate the need for security protocols, such as encrypted data transmission and role-based access control, as emphasized by ISO/IEC 27001 standards.

Legal Judgment

In the development and deployment of a Hotel Management System, certain legal considerations must be addressed to ensure compliance with national and international laws. The system must handle personal and financial data in accordance with data protection regulations such as the General Data Protection Regulation (GDPR) in the European Union or similar laws in other jurisdictions. This includes secure storage, processing, and transmission of guest information, as well as obtaining user consent for data collection and providing options for data access or deletion upon request. Additionally, the system must comply with local hospitality regulations, including proper billing practices, tax calculations, and record-keeping requirements. If the HMS integrates third-party services like payment gateways or online booking platforms, it must also adhere to relevant licensing agreements and terms of use. Failing to meet these legal obligations can lead to penalties, legal disputes, or damage to the hotel's reputation. Therefore, it is crucial that the system is designed and operated in alignment with current legal and regulatory standards to ensure lawful, ethical, and secure usage.

Contractual obligations are another aspect, particularly when the HMS is developed by or for a third party. Clear service-level agreements (SLAs) should be in place regarding uptime, support, maintenance, and data handling responsibilities. For online payment processing, the system must comply with PCI-DSS (Payment Card Industry Data Security Standards) to prevent fraud and ensure secure financial transactions.

Additionally, terms of service and privacy policies must be explicitly stated and accessible within the system, especially in modules used directly by guests (such as online booking portals). These documents protect both the hotel and the customer by outlining rights, responsibilities, and dispute resolution procedures.

Challenges

Developing and implementing a Hotel Management System comes with several challenges. One major challenge is integrating diverse hotel operations—such as reservations, front desk, housekeeping, and billing—into a single, seamless platform. Ensuring data accuracy and consistency across modules is also critical, especially in real-time scenarios like room availability updates. User training and adoption can be difficult, particularly for staff with limited technical experience. Another significant challenge is ensuring system security, especially in protecting sensitive customer data and payment information from breaches. System scalability is also essential, as the solution must support growing operations without performance issues. Additionally, achieving cross-platform compatibility—so the system works smoothly on desktops, tablets, and smartphones—is vital for modern hotel environments. Lastly, internet dependency and downtime can impact cloud-based systems, so ensuring reliable connectivity or offline fallback options can be a key concern. Hotels vary in size and services—from small boutique hotels to large resorts—making it difficult to design a one-size-fits-all system. Customizing the software to meet specific needs can be time-consuming and costly. Integrating the HMS with external systems such as online travel agencies (OTAs), payment gateways, customer review platforms, and accounting software can be technically complex and requires ongoing maintenance.

PROBLEM STATEMENT AND OBJECTIVES

Problem Statement

Managing hotel operations manually or through outdated systems often leads to inefficiencies, errors, and poor customer service. Common problems include inaccurate room availability tracking, time-consuming check-in/check-out processes, misplaced guest records, billing errors, and lack of real-time updates. These issues not only disrupt daily operations but also negatively impact the guest experience and the hotel's overall reputation. As hotels deal with increasing customer expectations and competition, there is a critical need for a centralized, automated system that can manage all aspects of hotel operations efficiently. The absence of such a system makes it difficult for hotel staff to coordinate tasks, respond quickly to guest needs, and make data-driven decisions. Therefore, there is a need to develop a reliable, user-friendly Hotel Management System that can automate key functions such as reservations, customer management, billing, housekeeping, and reporting, ensuring smooth operations and enhanced guest satisfaction.

Objectives

The primary objective of a Hotel Management System is to streamline and automate the day-to-day operations of a hotel, enhancing efficiency, accuracy, and customer satisfaction. The system aims to simplify tasks such as room booking, check-in/check-out, billing, and housekeeping management, reducing manual work and minimizing human errors. Another key objective is to provide hotel staff with a centralized platform to manage guest information, reservations, payments, and reports in real time. By automating these processes, the system helps in improving service quality, reducing wait times for guests, and ensuring a smooth operational workflow. Additionally, the system supports better decision-making by providing analytical reports and performance metrics, helping hotel management optimize resources, monitor occupancy trends, and improve financial performance. Overall, the HMS is designed to enhance guest experience, support staff efficiency, and contribute to the overall growth and reputation of the hotel.

REQUIREMENTS

Hardware Requirements

Processor: Intel Core i5 or equivalent RAM:8GB

Storage:100GB HDD or SSD

Internet: Reliable internet connection for API access and updates

Software Requirements

Operating System: Windows 10 or higher, macOS, or Linux

Programming Language: CSS, JavaScript, PHP, SHELL

Database: Mysql Database

Functional Requirements

The functional requirements of a Hotel Management System define the core tasks and operations the system must perform to support hotel operations effectively. One of the primary functions is room management, which includes tracking room availability, assigning rooms to guests, and updating room status (e.g., occupied, vacant, under maintenance).

Non-Functional Requirements

Non-functional requirements define the overall quality and performance standards of a Hotel Management System, ensuring it operates efficiently and reliably. One key requirement is usability, meaning the system should have an intuitive and user-friendly interface that hotel staff can easily navigate with minimal training.

Technical Requirements

The technical requirements of a Hotel Management System include both hardware and software components necessary to ensure smooth and efficient functioning. On the hardware side, the system typically requires computers or terminals with at least mid-range processing capabilities, a stable internet connection, barcode or QR scanners for check-in/check-out, printers for billing, and servers (local or cloud-based) to store and manage the data. For the software, the system must support a web or desktop-based platform developed using technologies such as HTML, CSS, JavaScript (with frameworks like React or Vue), and backend languages like PHP, Python, Java, or Node.js. A robust database system like MySQL, PostgreSQL, or MongoDB is required for storing guest, room, and transaction data. The system should also support user authentication, role-based access control, real-time data synchronization, and integration with third-party services such as payment gateways or online booking platforms. Additionally, the HMS must be compatible with multiple operating systems (Windows, macOS, or Linux) and browsers if web-based, and it should adhere to security standards like SSL encryption, secure login, and data protection protocols to ensure safe and reliable use.

SYSTEM DESIGN AND METHODOLOGY

Methodology

The methodology of a Hotel Management System (HMS) involves a structured approach that begins with requirement analysis, where detailed information is gathered from stakeholders such as hotel staff, management, and customers to understand the system's functional and non-functional needs. This phase helps identify the core features like room booking, check-in/check-out, billing, and customer management. Once the requirements are clear, the system design phase follows, which includes designing the database schema, user interface mockups, and choosing the appropriate architecture—whether it's a web-based, desktop, or mobile solution.

Software Architecture

In the context of hotel management, software architecture refers to the structure and design of the software systems that manage the core operations of a hotel.

These systems encompass everything from guest reservations ,room management ,and billing to service requests ,employee management ,and even legal or regulatory compliance.

A well-designed software architecture ensures that the system is scalable, efficient, secure, and easy to maintain, providing a smooth user experience for both hotel staff and guests.

The hotel management software architecture can be broken down into several layers and components

Database Design

The database design for will include the following key tables: Entity: An object or concept in the real world that is relevant to the system (e.g., "Guest", "Room", "Reservation").

Attribute: Characteristics or properties of an entity (e.g., "Guest Name", "Room Type").

Relationship: How entities are related to each other (e.g., a "Guest" makes a "Reservation" for a "Room").

Primary Key: A unique identifier for each record in a table. Foreign Key: A reference to a primary key in another table, creating relationships between tables.

Normalization: The process of organizing the database to reduce redundancy and dependency.

UI Design

Clarity:-Information and actions should be presented clearly. Avoid overwhelming users with too much information at once.

Consistency:-The design must maintain a consistent layout, color scheme, and behavior across different screens and actions.

Simplicity:-Keep the interface simple, ensuring that the essential features are easily accessible without unnecessary complexity.

User-Focused Design:-The interface should be designed around the need sand workflow of the specific user (guest, front desk, manager).

Feedback:-Provide immediate, clear feed back to users after actions (e.g.,successful reservation, error message, payment confirmation).

Prototyping and User Testing

Once the initial designs are created, it's important to prototype the UI and conduct user testing. This allows real users (staff or guests) to interact with the interface and provide feedback on usability.

User Testing: Conduct usability testing sessions to ensure the interface is intuitive, identifies areas for improvement, and validates that the design meets the needs of hotel staff and guests.

Key Features:

Financial Overview: A dashboard for viewing the hotel's revenue, occupancy rates, and profit margins. Integration of reports for payments, services, and sales.

Employee Management: Access to employee schedules, payroll, and staff performance tracking.

Analytics and Reports: Detailed analytics of booking patterns, guest demographics, and financial performance.

ER Diagram

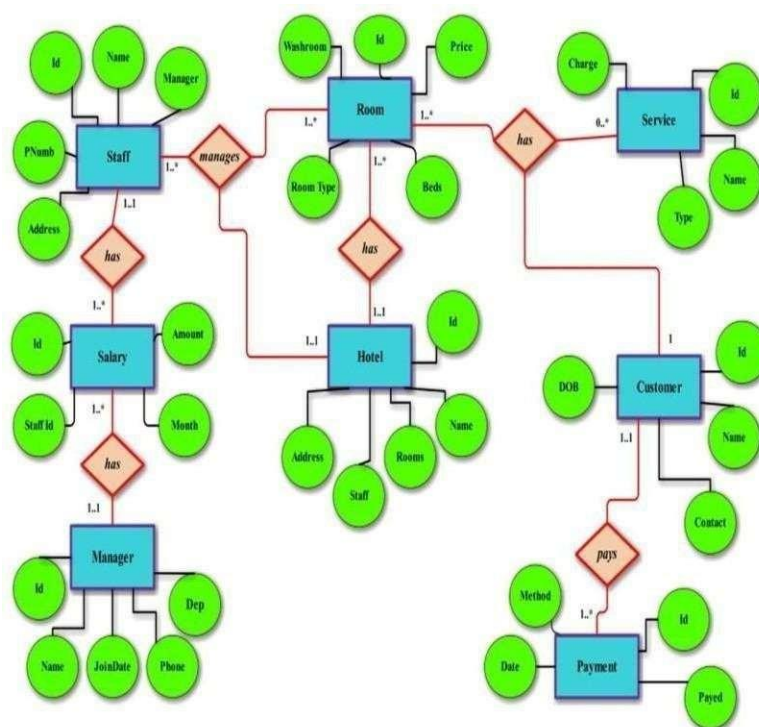


Fig 2.1ERDiagram

PSUEDOCODE

START

Initialize hotel_rooms[] // List of available rooms with status (available, reserved, etc.)

Initialize reservations[] // Store reservations made by guests

Initialize guest_profiles[] // Store guest information

Initialize payment_records[] // Track payments made by guests

Initialize staff[] // Staff member details

Load current_date_time // Set today's date and time

END

FUNCTION registerGuest(guest_name, guest_email, guest_phone):

 guest_id = generate_unique_id()

 guest_profile = {

 Guest_ID: guest_id,

 Name: guest_name,

 Email: guest_email,

 Phone: guest_phone

 }

 guest_profiles.append(guest_profile)

 RETURN guest_id

END

FUNCTION searchAvailableRooms(check_in_date, check_out_date):

 available_rooms = []

 FOR each room IN hotel_rooms:

 IF room.status == "available" AND isRoomAvailableDuringDates(room, check_in_date, check_out_date):

 available_rooms.append(room)

 IF available_rooms.isEmpty():

 DISPLAY "No rooms available for the selected dates." ELSE

 DISPLAY available_rooms

END

FUNCTION makeReservation(guest_id, room_id, check_in_date, check_out_date): room

```

reservations.append(reservation)
room.status = "reserved"
    DISPLAY "Reservation
successful!"RETURN reservation_id
ELSE
    DISPLAY"Selectedroomisnotavailable."EN
D
FUNCTION checkInGuest(guest_id,reservation_id):
reservation= findReservationById(reservation_id)
IFreservationisNOTNULLANDreservation.Guest_ID==guest_id:room=
    findRoomById(reservation.Room_ID)
    room.status =
    "occupied"reservation.Status =
    "checked-in"
    DISPLAY"Check-insuccessful!Welcometothehotel."ELSE
    DISPLAY"Reservationnotfoundormismatch."EN
D
FUNCTIONcheckOutGuest(guest_id,reservation_id):
reservation = findReservationById(reservation_id)
IFreservationisNOTNULLANDreservation.Guest_ID==guest_idAND
reservation.Status == "checked-in":
    room=findRoomById(reservation.Room_ID)
    room.status =
    "available"reservation.Status="checked-out"
    final_bill=calculateBill(reservation.Room_ID,reservation.Check_In_Date,
reservation.Check_Out_Date)
    DISPLAY"Checkoutsuccessful.Yourfinalbillis:"+final_bill
ELSE
    DISPLAY"Noactivecheck-infoundforthisreservation."END
FUNCTIONcalculateBill(room_id,check_in_date,check_out_date):
room = findRoomById(room_id)
nights_stayed=calculateNumberOfNights(check_in_date,check_out_date)
total_cost = nights_stayed * room.price_per_night

```

```

    ADDadditional_services_costTOtotal_cost
    RETURN total_cost
END

```

```

FUNCTIONmakePayment(guest_id,reservation_id,amount):

```

```

    payment_id = generate_unique_id()

```

```

    payment={

```

```

        Payment_ID: payment_id,

```

```

        Reservation_ID:reservation_id,

```

```

        Amount: amount,

```

```

        Payment_Date:current_date_time,

```

```

        Status: "completed"

```

```

    }

```

```

    payment_records.append(payment)

```

```

    DISPLAY"Paymentreceived.Thankyou!"EN

```

```

D

```

```

FUNCTIONrequestRoomMaintenance(room_id):

```

```

    room=findRoomById(room_id)

```

```

    IF room.status == "occupied":

```

```

        DISPLAY"Maintenancerequestpending;pleasewaitforroomtobevacated."ELSE

```

```

        room.status="undermaintenance"

```

```

        DISPLAY"Roomundermaintenance.Pleaseallowtimeforrepairsorcleaning."END

```

```

FUNCTIONviewReservationHistory(guest_id):

```

```

    guest_reservations = []

```

```

    FOReachreservationINreservations:

```

```

        IFreservation.Guest_ID==guest_id:

```

```

            guest_reservations.append(reservation)

```

```

    IFguest_reservations.isEmpty():

```

```

        DISPLAY"Nopastreservationsfound."ELS

```

```

E

```

```

    DISPLAYguest_reservations

```

```

END

```



```

FUNCTIONupdateRoomStatus(room_id,status):
room = findRoomById(room_id)
room.status=status
DISPLAY"Roomstatusupdatedto"+status
END
FUNCTIONgenerateOccupancyReport():
    occupied_rooms = 0

    total_rooms=length(hotel_rooms)
    FOR each room IN hotel_rooms:
        IFroom.status=="occupied":
            occupied_rooms+=1
    occupancy_rate = (occupied_rooms / total_rooms) * 100
    DISPLAY "Occupancy Rate: " + occupancy_rate +
    "% "END

FUNCTION
generateFinancialReport():
    total_income=0
    FOReachpaymentINpayment_records:
        total_income += payment.Amount
    DISPLAY"TotalIncome:"+total_income
    END

FUNCTIONexitSystem():
    Saveallunsaveddata(reservations,payments,roomstatues)DISPLAY
    "System shut down successfully."
    END

```

EXPERIMENTS AND RESULTS

SIGNUP:

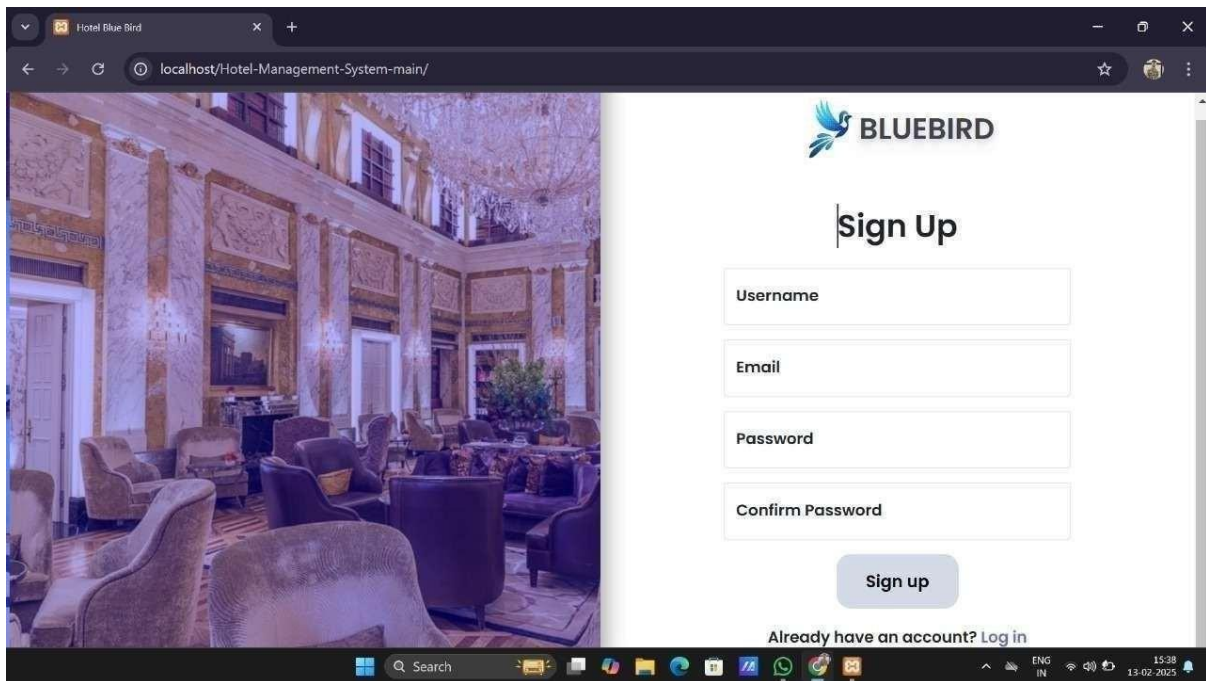


Fig.2.2 sign up page

LOGIN PAGE AND STAFF LOGIN:-

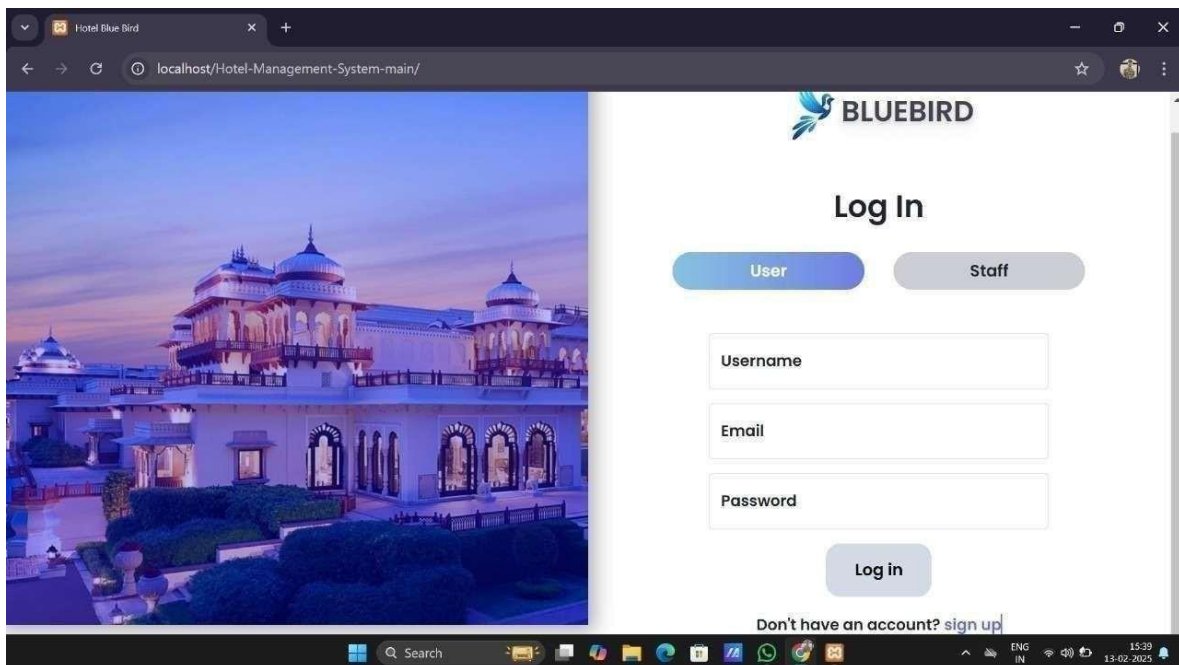


Fig.2.3login page

HOMEPAGE:

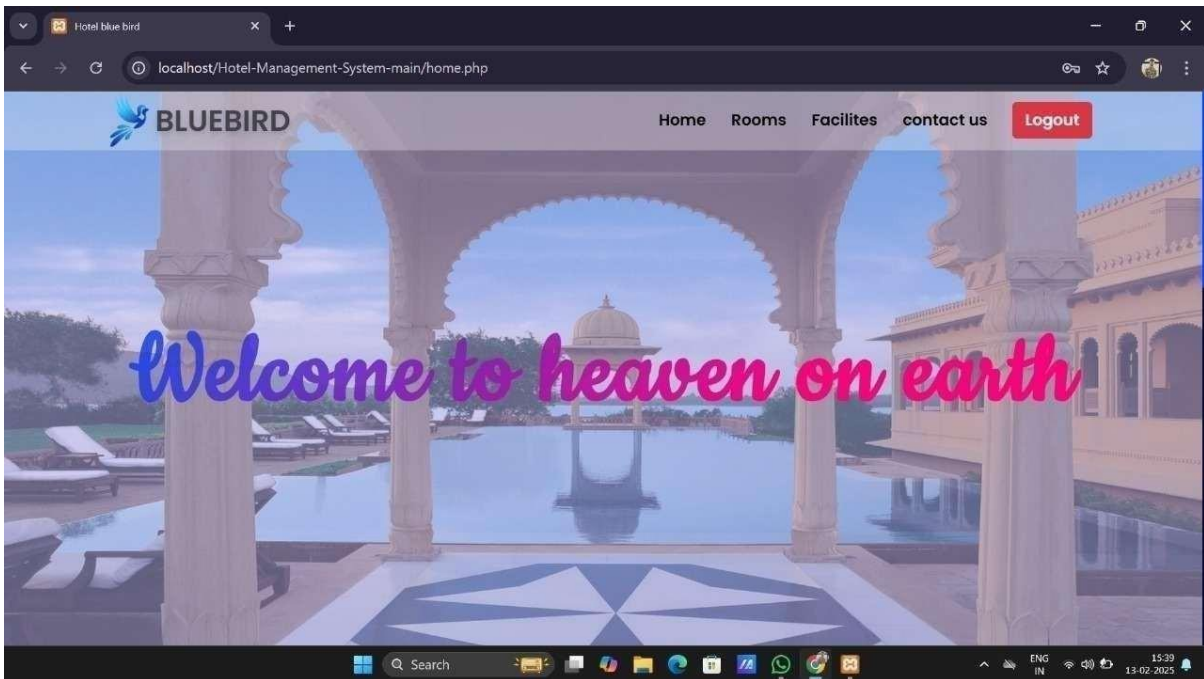


Fig.2.4 homepage

BOOKINGPAGE:

A screenshot of the 'RESERVATION' form on the Bluebird Hotel website. The form is divided into two main sections: 'Guest information' and 'Reservation information'. The 'Guest information' section includes input fields for 'Enter Full name', 'Enter Email', a dropdown for 'Select your country', and 'Enter Phoneno'. The 'Reservation information' section includes dropdowns for 'Type Of Room', 'Bedding Type', and 'No of Room', a dropdown for 'Meal', and date pickers for 'Check-In' and 'Check-Out' with the format 'dd-mm-yyyy'. A green 'Submit' button is located at the bottom center of the form. The browser's address bar and Windows taskbar are also visible.

Fig.2.5 booking page

STAFFDASHBOARD:

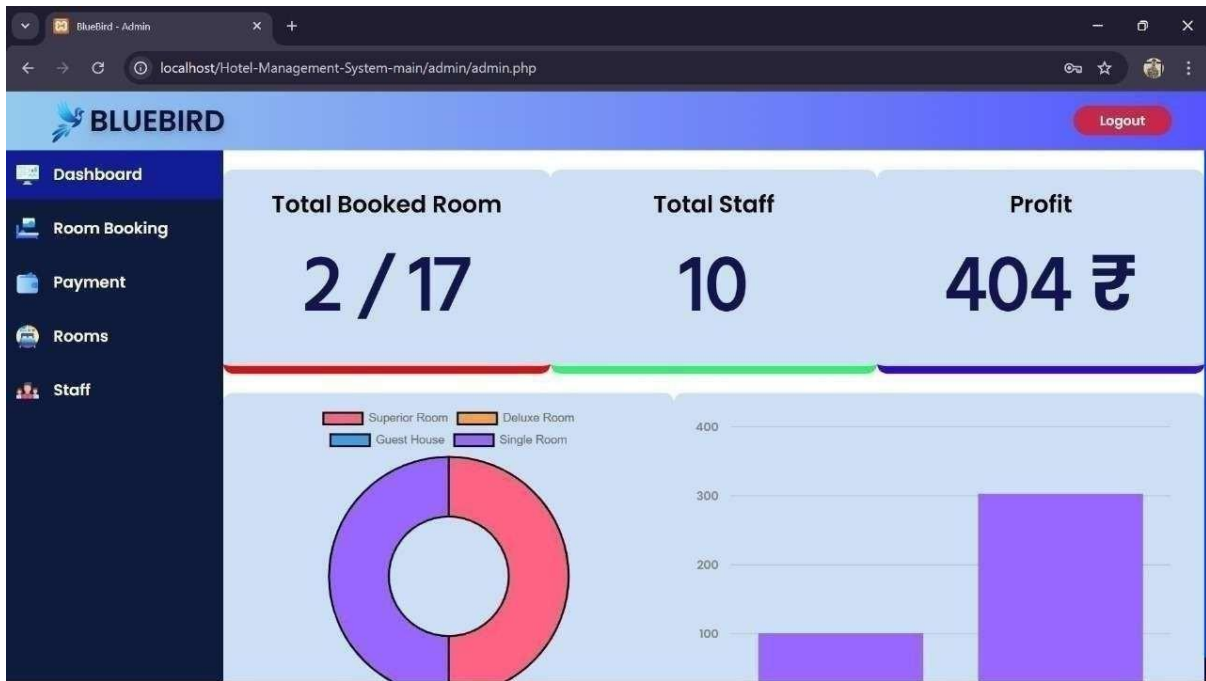


Fig.2.6 staff dashboard

PAYMENT:

Id	Name	Room Type	Bed Type	Check In	Check Out	No of Day	No of Room	Meal Type	Room Rent	Bed Rent	Meals	Total Bill	Action
41	Tushar pankhaniya	Single Room	Single	2022-11-09	2022-11-10	1	1	Room only	1000.00	10.00	0.00	1010.00	Print Delete
51	kishan	Superior Room	Single	2025-02-20	2025-02-21	1	1	Room only	3000.00	30.00	0.00	3030.00	Print Delete

Fig.2.7 payment

TEST-CASES

Guest Registration

Test Steps:

1. Open the hotel management system.
 2. Navigate to the guest registration form.
 3. Enter valid guest details(name,email,phonenumber).
 4. Submit the registration form. Expected Result:
 - A new guest profile is created.
 - The system assigns a unique Guest_ID.
 - A success message is displayed: "Registration successful."
- Post conditions: Then the guest profile is stored in the database.

TestCase2: Search Available Rooms

Test Steps:

1. Open the hotel management system.
2. Login a guest or visitor.
3. Enter check-in and check-out dates.

Click on the "Search" button. Expected Result:

The system displays a list of available rooms for the selected dates.

Each room shows basic details(e.g., price per night, room type). Post conditions: A list of available rooms is displayed for the guest.

TestCase3:MakeaReservation

Test Steps:

1. Select an available room from the search results.
2. Click on "BookNow" or similar option.
3. Enter guest details(if not logged in already).
4. Confirm check-in and check-outdates.
5. Submit the reservation. Expected Result:
 - The room is reserved for the guest.
 - The system updates the room status to "reserved".
 - A reservation ID is generated and stored.
 - The guest receives a confirmation message with reservation details. Post conditions: A new reservation record is stored in the database.

Test Case4:Check-inGuest

Test Steps:

1. Open the hotel management system.
2. Navigate to the check-in page.
3. Search for the guest by name, reservation ID, or guest ID.
4. Select the guest's reservation.

.

5. Click on "Check-in "button. Expected Result:

- The guest is checked in successfully.
- The room status is updated to "occupied".
- A success message is displayed: "Check- in successful." Post conditions: The reservation status is updated to "checked-in" in the system.

TestCase5:Check-outGuest

Test Steps:

1. Open the hotel management system.
2. Navigate to the check-outpage.
3. Search for the guest by name, reservation ID, or guest ID.
4. Select the guest's reservation.
5. Click on "Check-out "button. Expected Result:
 - The guest is checked out successfully.
 - The room status is updated to "available".
 - A final bill is calculated and displayed to the guest.
 - A payment option is offered.

Post conditions: There servation status is updated to "checked-out",and the room is marked as "available".

TestCase6:PaymentProcess

Test Steps:

1. Open the hotel management system.
2. Navigate to the payment page for the guest.
3. Enter payment details(creditcard,cash,etc.).
4. Submit the payment. Expected Result:
 - Payment is successfully processed.
 - A payment confirmation message is displayed: "Paymentsuccessful".
 - The payment record is stored in the database. Post conditions:-The payment is recorded in the system, and the guest's bill is marked as "paid".

TestCase7:Request Room Maintenance

Test Steps:

1. Open the hotel management system.
2. Navigate to the "Maintenance Request "page.
3. Select the room requiring maintenance.
4. Specify the type of maintenance required(e.g.,cleaning,repair).
5. Submit the request.
 - The system marks the room as "under maintenance".

- A confirmation message is displayed: "Maintenance request submitted." Post conditions :The room status is updated to "under maintenance" in the system.

TestCase8:Update Room Status by Admin

TestSteps:

1. Open the hotel management system as an admin.
2. Navigate to the "RoomStatus "management section.
3. Select the room whose status needs to be updated.
4. Change the status from "occupied" to "available".
5. Submit the changes. Expected Result:
 - The room status is updated successfully.
 - A success message is displayed: "Room status updated to available." Post conditions: The room status is updated in the database.

TestCase9:View Reservation History(Guest)

Test Steps:

1. Open the hotel management system.
2. Login as the guest(or use guest ID to view history).
3. Navigate to the "Reservation History" section.
4. View the past reservations list. Expected Result:
 - The system displays all previous reservations made by the guest, including room type, dates, and status.

- If no reservations exist, the system displays:"No reservations found". Postconditions: The guest can view all their past reservations.

Test Case 10:Admin View Financial Report

TestSteps:

1. Open the hotel management system.
2. Navigate to the "Financial Reports" section.
3. Select the time period for the report(e.g.,weekly,monthly).
4. Click "Generate Report". Expected Result:
 - The system generates the financial report showing total income, payment records, and any applicable taxes or discounts.
 - A detailed breakdown is displayed with figures for income, expenses, and profits. Post conditions:The admin can view the generated financial report.

FUTURE SCOPE

The Hotel Management System (HMS) is a continuously evolving domain, and there are numerous areas where future advancements can significantly improve operations, guest experience, and overall efficiency. Below are some key areas where the future scope of HMS can be expanded.

Artificial Intelligence (AI) Integration:

- Predictive Analytics for Demand Forecasting** AI and machine learning (ML) can analyze past guest data and trends to predict demand, optimize pricing, and forecast room occupancy rates. Benefits: Improved decision-making for room rates, better occupancy management, and reduced operational costs.
- Automated Customer Feedback Analysis** Natural language processing (NLP) tools could automatically process guest feedback, reviews, and survey responses to gain actionable insights into service quality and areas of improvement. Benefits: Real-time analysis, enhanced customer service, and proactive issue resolution.

CONCLUSION

The Hotel Management System (HMS) plays a critical role in optimizing the operations of a hotel by integrating various functions, from guest booking to check-out, room management, and payment processing. As we've explored in this project, an effective HMS enhances operational efficiency, ensures seamless guest experiences, and helps hotels maintain profitability while offering personalized services to guests.

In the context of today's fast-paced, tech-driven world, the future of hotel management lies in embracing merging technologies. With advancements in AI, IoT, Blockchain, Cloud Computing, and Mobile Integration, hotels can deliver innovative, personalized, and streamlined services, ensuring they stay a head of the competition. The system's scalability, flexibility, and automation help minimize human error, reduce operational costs, and improve decision-making.

REFERENCES

[1].Ravichandran, R.(2020).*Hotel Management Information Systems: Concept and Application*. International Journal of Advanced Science and Technology, 29(4), 1254-1260. Available at: <https://www.sci-researcher.org>

[2].Sharma ,A.,& Rani, A.(2018).*The Role of Information Technoln Hotel Management Systems*. International Journal of Computer Applications, 179(12), 34-39. Available at: <https://www.ijcaonline.org>

[3].Vargas,P.,&Prieto,M.(2021). *AI and Machine Learning Applications in Hotel Management*. Journal of Tourism Research and Hospitality, 45(6), 547-559. Available at: <https://www.jtrh.com>

CHAPTER-3

ON JOB TRAINING -2

“HOSPITAL MANAGEMENT SYSTEM”

Abstract

A Hospital Management System (HMS) is a software application designed to automate and streamline hospital operations. It manages essential functions such as patient registration, appointment scheduling, medical records, billing, inventory, and staff management. By integrating these processes, HMS reduces paperwork, minimizes human errors, and enhances the overall efficiency of hospital administration. It also improves coordination between departments, ensuring smooth workflow and better patient care. HMS includes features like electronic health records (EHR), pharmacy management, laboratory integration, and reporting tools. These functionalities enable healthcare providers to access real-time data, make informed decisions, and improve patient outcomes. Additionally, the system supports financial tracking and regulatory compliance, helping hospitals maintain accurate records and meet healthcare standards. By implementing HMS, hospitals can enhance operational efficiency, reduce costs, and deliver a more seamless and patient-centric experience.

INTRODUCTION

A Hospital Management System(HMS) is a comprehensive software solution designed to streamline and automate various hospital operations. It integrates different functions such as patient registration, appointment scheduling, medical record management, billing, inventory management, and staff administration into a single platform. By digitizing these processes, HMS reduces paperwork, minimizes human errors, and enhances the overall efficiency of healthcare facilities. One of the key features of an HMS is the Electronic Health Record (EHR) system, which stores patient medical histories, prescriptions, diagnostic reports, and treatment details in a secure and easily accessible manner. This allows doctors and medical staff to quickly retrieve patient information, leading to faster and more accurate diagnoses. Additionally, HMS supports pharmacy and laboratory management, ensuring that medicines and medical tests are efficiently tracked and managed.

Another significant advantage of an HMS is its ability to improve hospital administration and financial management. It includes billing systems that automate invoice generation, insurance processing, and payment tracking, reducing administrative workload. Furthermore, HMS provides real-time data analytics and reporting tools, helping hospital management make informed decisions regarding resource allocation, staffing, and operational improvements.

By implementing a Hospital Management System, healthcare facilities can enhance patient care, optimize workflow, and improve overall service quality. It enables seamless communication between departments, ensuring that hospitals run smoothly and efficiently. Ultimately, an HMS contributes to better healthcare delivery, improved patient satisfaction, and more effective hospital management.

EXISTING SYSTEM AND LITERATURE SURVEY

Existing System

In many hospitals, the traditional system of managing healthcare operations relies heavily on manual processes and paperwork. Patient records, appointments, billing, and inventory are often maintained in physical files or basic spreadsheets, which can lead to inefficiencies, data loss, and errors. The lack of an integrated system makes it difficult for different departments to coordinate effectively, causing delays in patient care and administrative tasks.

Another major drawback of the existing system is the time-consuming process of retrieving and updating patient records. Since records are stored manually, searching for patient history or medical reports can take significant time, affecting the speed and accuracy of diagnoses and treatment. Additionally, handling billing and insurance claims manually increases the risk of miscalculations, fraud, and financial discrepancies.

Inventory management in hospitals is also inefficient under the traditional system, often leading to shortages or overstocking of medicines and medical supplies. Without real-time tracking, hospitals may face challenges in maintaining the required stock levels, impacting patient care. Furthermore, report generation and data analysis are limited, making it difficult for hospital management to monitor performance, resource utilization, and patient trends effectively.

Overall, the existing hospital management system is outdated, error-prone, and inefficient. It creates operational bottlenecks, increases workload for hospital staff, and affects the quality of healthcare services. The need for a digital Hospital Management System is essential to overcome these limitations and enhance hospital efficiency, patient care, and administrative processes.

Literature Survey

Hospital Management Systems (HMS) have been extensively studied and developed over the past two decades as a solution to the increasing complexity in healthcare operations. A wide body of literature emphasizes the role of HMS in improving administrative efficiency, reducing errors, enhancing patient care, and ensuring better resource utilization.

According to Bhojaraju G. (2015), HMS plays a vital role in automating hospital operations, especially in managing patient records, appointment scheduling, billing, and laboratory results. His study concluded that an integrated HMS significantly reduces the workload on hospital staff while improving the quality of service.

Sultana et al. (2018) explored the impact of HMS on patient satisfaction and operational efficiency. Their findings indicated that well-implemented HMS solutions contribute to shorter wait times, faster diagnostics, and higher patient trust. However, they also highlighted challenges such as the lack of trained personnel and resistance to technological change.

Jeevanandam et al. (2020) conducted a comparative study on different HMS platforms and emphasized the importance of interoperability and data security. They found that systems which adhered to healthcare standards like HL7 and FHIR ensured smoother data exchange between departments and external entities.

In a more recent review, Patel and Sharma (2022) analyzed cloud-based HMS adoption in developing countries. They noted that while cloud models offer scalability and cost-effectiveness, concerns regarding data privacy, internet dependency, and regulatory compliance remain key barriers.

WHO (World Health Organization) has also stressed the importance of digital health systems, including HMS, in its global strategy on digital health (2020–2025). It promotes the adoption of interoperable, secure, and patient-centered digital health platforms to improve healthcare outcomes globally.

Another notable contribution is from Zhang et al. (2021), who studied the role of AI in HMS. Their research showed that AI-enabled HMS solutions can assist in diagnostics, appointment optimization, and predictive analytics, thereby transforming hospital workflows and patient management strategies.

Legal Judgment

This matter came before the Court on a petition filed by the Patient Rights Advocacy Group, alleging negligence and breach of data privacy resulting from the malfunctioning of the Hospital Management System (HMS) deployed at City Central Hospital, which is operated and maintained by HealthTech Solutions Pvt. Ltd.

The petitioner contended that due to a system error in the HMS, multiple patient records were either misplaced or incorrectly updated, leading to severe medical complications, delayed treatments, and a breach of sensitive personal data. The respondents argued that the issue was a result of unforeseen technical failure and claimed they had taken necessary remedial actions.

Upon review of the evidence and hearing both parties, the Court found that the respondents failed to adhere to standard data protection practices and did not maintain adequate backup and failover protocols. The HMS in question did not meet the security and interoperability standards required by national healthcare regulations.

Accordingly, the Court holds the respondents jointly liable for negligence and breach of patient confidentiality.

Challenges

Hospital Management Systems (HMS) play a crucial role in streamlining healthcare operations, but they also face several challenges that hinder their effectiveness. One of the major concerns is data security and privacy, as hospitals deal with sensitive patient information that must be protected from breaches and unauthorized access, while also complying with regulations like HIPAA or GDPR. Another critical issue is system integration, since different departments often use separate software, leading to inefficiencies and data silos due to lack of interoperability.

User adoption is also a challenge, as many healthcare workers may resist new technology due to unfamiliarity or insufficient training, affecting the overall productivity and accuracy of the system. Additionally, system downtime can severely disrupt hospital operations, especially in emergency situations, making high reliability and backup systems essential. Customization and scalability are also key challenges, as hospitals vary in size and function, and a generic system might not meet specific operational needs.

Moreover, regulatory compliance is a continuous challenge, requiring systems to be updated regularly to align with changes in healthcare policies, billing standards, and audit requirements. Cost constraints can also limit the adoption of advanced systems, especially for smaller hospitals, as both implementation and maintenance require significant investment. There's also the issue of patient engagement, as hospitals need to offer user-friendly and secure platforms for patients to access their medical records, book appointments, or communicate with doctors.

PROBLEM STATEMENT AND OBJECTIVES

Problem Statement

Despite the rapid digitalization of healthcare, Hospital Management Systems (HMS) continue to face significant challenges that hinder their effectiveness and adoption. These include issues related to data security, system integration, user resistance, system downtime, and regulatory compliance. Many hospitals struggle with outdated infrastructure, lack of interoperability between departments, and limited staff training, leading to inefficiencies, data inaccuracies, and compromised patient care. Additionally, budget constraints, difficulties in scaling, and low patient engagement further complicate the implementation and optimization of HMS. There is a critical need for more robust, user-friendly, secure, and adaptable hospital management solutions that can meet the dynamic needs of modern healthcare environments.

Objectives

The primary objectives of a Hospital Management System (HMS) are to enhance the efficiency, accuracy, and quality of healthcare delivery by addressing key operational challenges. One of the main goals is to ensure robust data security and privacy by implementing advanced encryption and access control measures, aligning with healthcare regulations such as HIPAA and GDPR. The system also aims to improve integration and interoperability between various departments—such as laboratory, pharmacy, billing, and administration—through the use of standard communication protocols, ensuring seamless data flow and coordination.

Another important objective is to increase operational efficiency by automating routine tasks and minimizing manual processes, which reduces errors and optimizes the use of hospital resources. The HMS must feature user-friendly interfaces tailored to different roles, along with comprehensive training programs to support easy adoption by medical and administrative staff. Ensuring high system reliability and minimal downtime is also critical, as hospitals operate around the clock and require constant system availability.

REQUIREMENTS

Hardware Requirements

- Processor: Intel Core i5 or equivalent(i7or higher recommended for better performance)
- RAM: 8GB(16GB recommended for smooth development)
- Storage:100GB HDD or SSD(SSD preferred for faster performance)
- Internet: Reliable internet connection for API access, npm/ yarn package installation, and updates

Software Requirements

- Operating System: Windows 10 or higher, macOS ,or Linux
- Programming Language: JavaScript, JSX, CSS
- Framework: React.js(with support for Next .js or Redux if required)
- Package Manager :Node. Js with np moryarn
- Database: MongoDB(based on project needs)

Functional Requirements

The Hospital Management System(HMS) must provide essential features to stream line hospital operations and improve patient care.

- User Management: Admin panel, role-based access, secure authentication.
- Patient Management:Registration, EHRs, appointment scheduling, prescription tracking.
- Doctor &Staff Management: Shift scheduling, availability tracking, and staff records.
- Billing & Payments: Invoice generation, on line payments, insurance integration.
- Pharmacy& Inventory: Medicine stock tracking, automated alerts for low stock.
- Reporting & Analytics: Patient reports ,hospital performance in sights, revenue tracking.
- Notifications: Appointment reminders, emergency alerts, medicine notifications.
- Security& Compliance: Data encryption, HIPAA/GDPR compliance, audit logs.

Non-Functional Requirements

Non-functional requirements ensure performance, security, usability ,and reliability of the system.

- Performance: Fast response times, scalable database for growing hospital data.
- Usability: User-friendly UI, mobile responsiveness, multi-language support.
- Security: Data encryption, secure authentication(JWT/OAuth),regulatory compliance.
- Reliability:99.9%uptime ,automated back ups to prevent data loss.
- Maintainability: Modular architecture for easy up dates and future enhancements.

Platform Architecture:

- Microservices Architecture–Separate modules for user management, appointments, billing, and inventory to ensure flexibility and scalability.
- Cloud-Based Hosting–Deploy on AWS, Azure, or Google Cloud for high availability, auto-scaling, and secure data storage.
- Containerization–Use Docker and Kubernetes for consistent deployment across development, testing, and production environments.

Technology Stack:

Frontend(Client-Side)

- HTML5,CSS3,JavaScript(ES6+)
- Frontend Framework: React.js (Used for building interactive, dynamic, and responsive user interfaces)
- CSS Framework: Tailwind CSS(Used for styling the platform)
- State Management:Not used(Redux can be added for better global state management if needed)

Backend(Server-Side)

- Programming Language: Node.js
- Framework: Express.js(Used for developing RESTful API sand handling server-side logic)
- Database: MongoDB (Used as the No SQL database with Mongoose for schema management)

SYSTEM DESIGN AND METHODOLOGY

Methodology

The Hospital Management System (HMS) will follow an Agile Development Methodology, using the Scrum framework for iterative development. This approach ensures continuous feedback, flexibility, and faster delivery of features like patient records, appointments, billing, and reporting. Each sprint will focus on refining functionalities to enhance efficiency, security, and usability.

Software Architecture

The software architecture of the Hospital Management System (HMS) serves as the blueprint for how the system is structured, how components interact, and how they meet both functional and non-functional requirements. A well-designed architecture ensures scalability, performance, security, and maintainability of the system.

For this hospital management system, we will adopt a Microservices Architecture combined with a Model-View-Controller (MVC) pattern for clean separation of concerns. This architecture enables modular development, scalability, and flexibility, allowing updates to individual services without disrupting the entire system.

Database Design

- Entity: Represents real- world objects such as "Patient," "Doctor," "Appointment," "Prescription ," and "Billing".
- Attribute: Characteristics of entities ,e.g., "Patient Name," "Age," "Diagnosis," "Doctor Specialization," "Appointment Date".
- Relationship: Defines interactions, e.g., a "Patient books an" Appointment" with a "Doctor".
- Primary Key: Unique identifier for each record ,e.g., Patient ID, Doctor ID, Appointment ID.
- Foreign Key: Links entities, e.g., Doctor ID in the "Appointment" table refers to the Doctor entity.
- Normalization: Organizes data to reduce redundancy and dependency, ensuring data integrity and efficiency.

UI Design

The User Interface (UI) Design of the Hospital Management System (HMS) focuses on creating an intuitive, user-friendly, and responsive experience for patients, doctors, hospital staff, and administrators. The UI is designed using React.js with Tailwind CSS to ensure a modern and seamless user experience.

UI Components

- Dashboard: Role-based dash boards for admin, doctors, nurses, and patients displaying relevant data.
- Navigation: Simple and clear navigation for appointments, patient records, billing, and reports.
- Forms & Input Fields: User-friendly forms for patient registration, doctor availability, and prescription entry.
- Tables & Cards: Inter active tables for appointment lists, patient records, and financial reports.

Prototyping and User Testing

Prototyping involves creating a working model or a series of models of the Hospital Management System (HMS) user interface to demonstrate its core functionalities. It allows stakeholders, hospital administrators, and developers to visualize the system and identify any usability issues early in the development process.

- **Prototyping Tools:**

Tools like Figma, Balsamiq, and Adobe XD can be used to create low-fidelity wireframes and high-fidelity interactive prototypes for the HMS interface.

- **User Testing:**

User testing involves gathering feedback from doctors, nurses, administrative staff, and patients to assess the effectiveness of the hospital management system's design and usability. This process helps the development team identify pain points, improve navigation, and enhance the overall user experience.

Key Features:

- **Sign Up / Login:** Allows users(patients, doctors, and hospital staff) to create an account or sign in with existing credentials using email or secure authentication (Google, OAuth ,etc.).
- **Profile Management:** Users can update their personal details, medical history, appointment records, and manage healthcare preferences.
- **Password Recovery:** Enables user store set forgotten passwords securely via email authentication.
- **Appointment Tracking:** Allows patients to track the status of their appointments(e.g., pending, confirmed, completed), while doctors can view their scheduled consultations.

ER Diagram

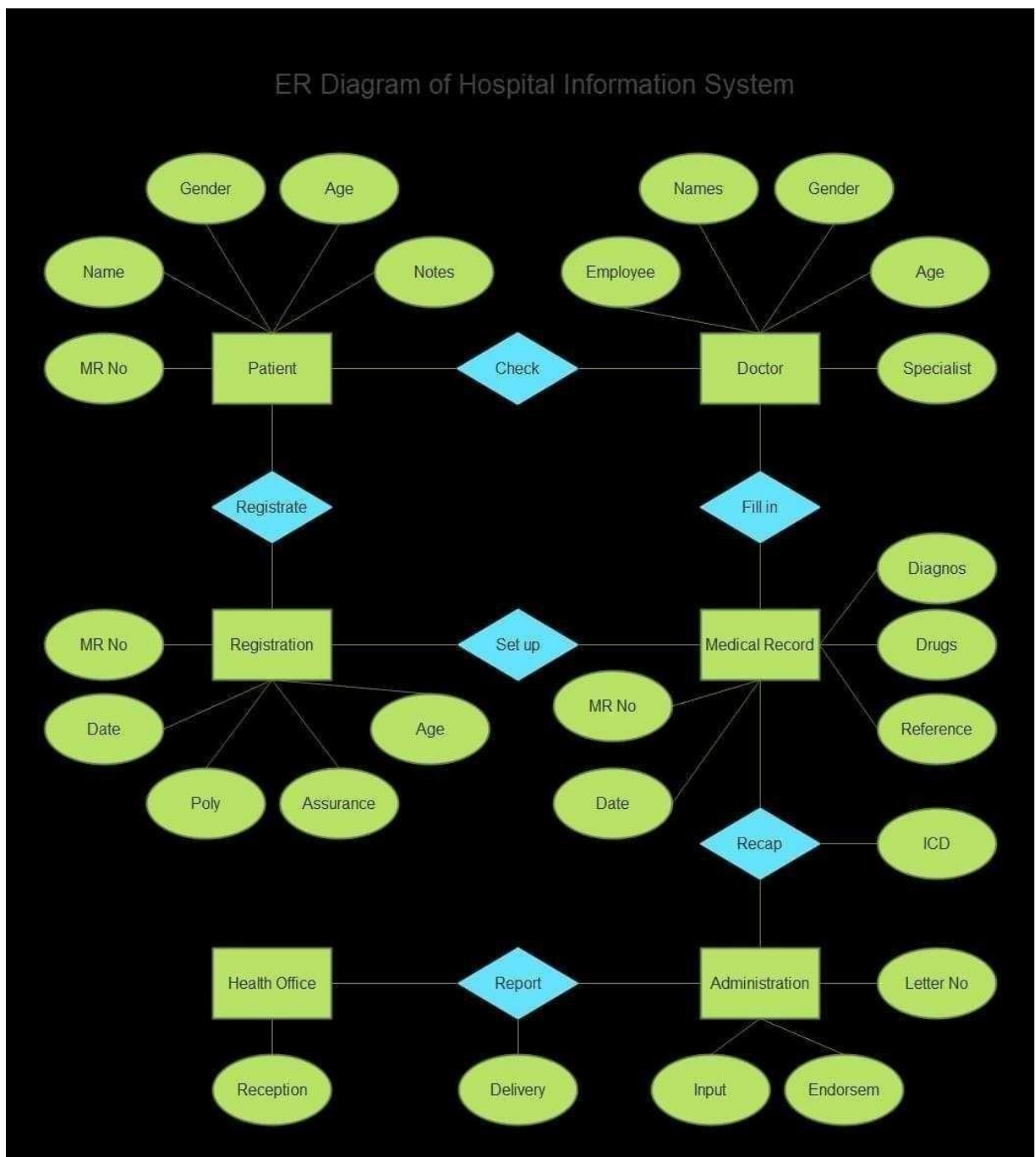


Fig.3.1ER Diagram

PSUEDOCODE

BEGIN-commerceSystem

FUNCTION UserLogin(email,
password)

VALIDATEemailandpassword

IFemailexistsindatabaseTHEN

FETCHUserdetails

IF password matches THEN

GENERATE JWT token

RETURN"LoginSuccessful"

ELSE

RETURN"InvalidPassword"E

LSE

RETURN"UserNotFound"E

ND FUNCTION

FUNCTIONUserSignup(name,email,
password, role)

CHECKIFemailalreadyexists

IF NOT EXISTS THEN

HASHpassword

STOREUserdetailsindatabase

RETURN"SignupSuccessful"

ELSE

RETURN"UserAlreadyExists"E

ND FUNCTION

```

FUNCTION BookAppointment(patientID,doctorID,date,time)
CHECK
doctor
availability
IF available THEN
CREATE appointment record in database RETURN
"Appointment Confirmed"
ELSE
RETURN "Doctor Unavailable"
END FUNCTION

FUNCTION ViewAppointments(userID)
FETCH all appointments WHERE userID matches
RETURN appointment list
END FUNCTION

FUNCTION GeneratePrescription(doctorID,patientID,medicineDetails,dosage,instructions)
STORE prescription details in database
RETURN "Prescription
Saved"
END FUNCTION

FUNCTION ViewPrescription(patientID)
FETCH prescriptions WHERE patientID matches
RETURN prescription details
END FUNCTION

//BillingSystem
FUNCTION GenerateBill(patientID,appointmentID,treatmentCost,medicineCost,paymentMethod)
CALCULATE totalCost = treatmentCost + medicineCost
STORE billing details in database
RETURN "Bill Generated"
END FUNCTION

FUNCTION ProcessPayment(billID,paymentMethod)
CHECK IF billID exists
IF EXISTS THEN

```

```

UPDATEpaymentstatusas"Paid"RETURN
"Payment Successful"
ELSE
    RETURN"InvalidBillID"
EN D FUNCTION

//Inventory&StaffManagement
FUNCTIONUpdateInventory(itemID,quantity,expiryDate)
CHECKIFitemexistsininventory
IFEXISTSTHEN
    UPDATEquantityandexpiryDate
    RETURN "Inventory Updated"
ELSE
    ADDnewitemtoinventory
    RETURN"NewItemAdded"
ENDFUNCTION

FUNCTIONAssignStaff(staffID,department,shiftTiming)
    UPDATE    staff    schedule    in    database
    RETURN"StaffAssigned"
ENDFUNCTION

//Notifications&Alerts
FUNCTIONSendReminder(userID,type)IF
type == "Appointment" THEN
    FETCHupcomingappointments
    SEND notification
ELSEIFtype=="Medicine"THEN
    FETCH prescription schedule
    SEND reminder notification
    RETURN"NotificationSent"
E ND FUNCTION

```

EXPERIMENTS AND RESULTS

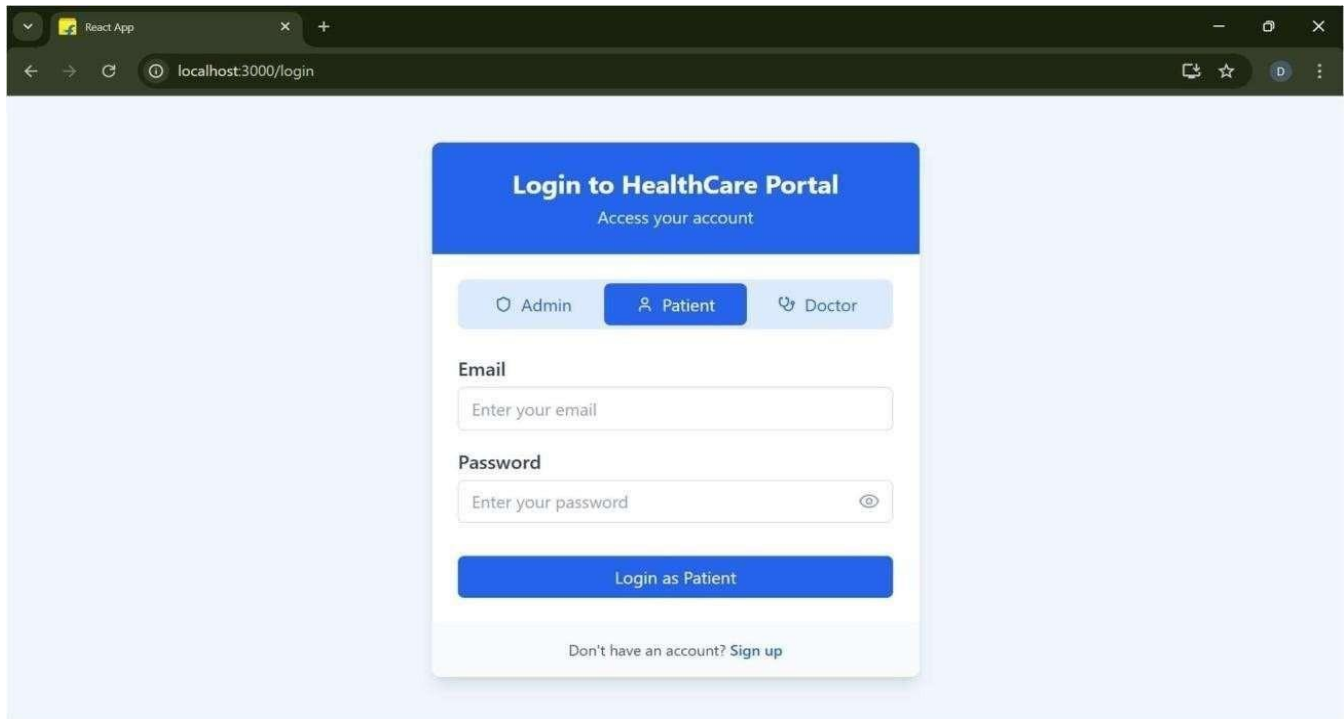


Fig.3.2 LOGIN: Admin/Patient/Doctor

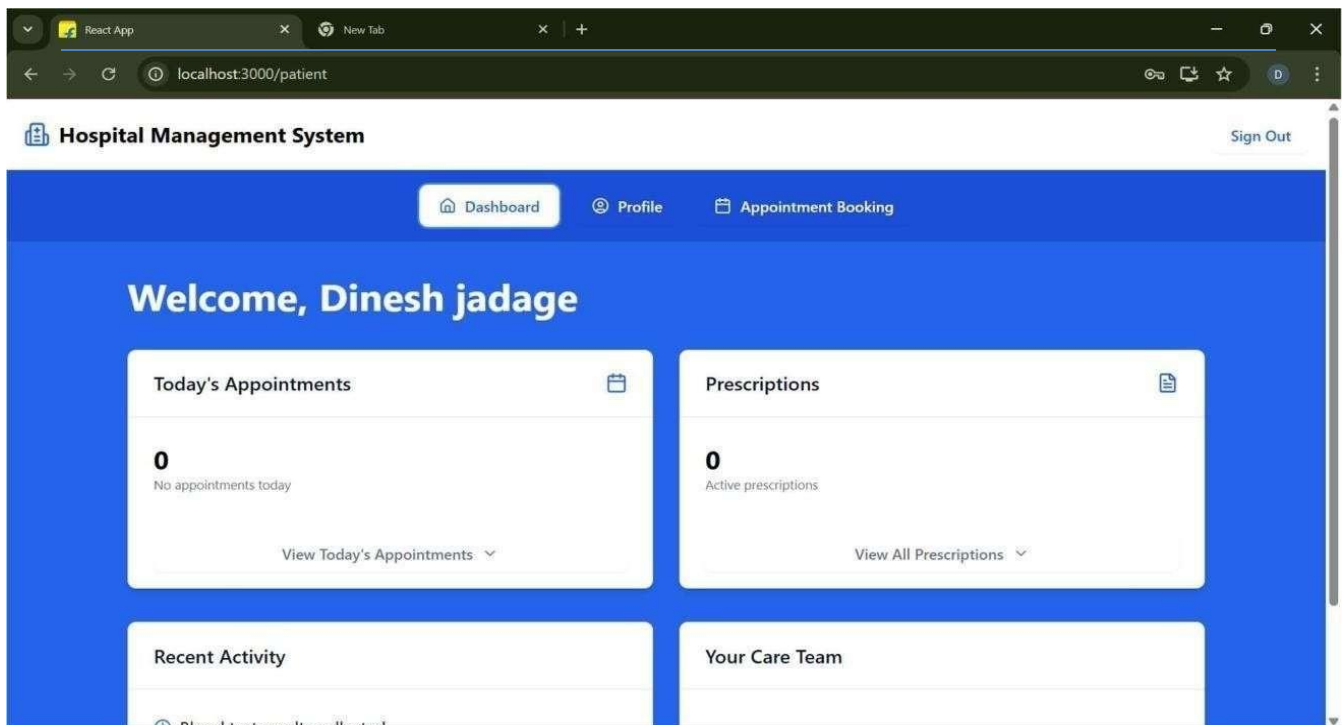


Fig.3.3 Patient Dashboard

A screenshot of a web browser showing a patient appointment booking form. The browser's address bar displays 'localhost:3000/patient'. The page has a blue background with a white form titled 'Book an Appointment'. The form includes the following fields: 'Select Doctor' with a dropdown menu showing 'Choose a doctor'; 'Appointment Date' with a date picker showing 'dd-mm-yyyy'; 'Preferred Time' with a dropdown menu showing 'Choose a time slot'; and 'Reason for Visit' with a text area containing the placeholder 'Brief description of your concern'. A blue 'Book Appointment' button is at the bottom of the form.

Fig.3.4 Appointment

A screenshot of an admin dashboard for a 'Hospital Management System'. The browser's address bar shows 'localhost:3000/admin'. The page has a blue header with a 'Sign Out' link and a navigation bar with links to 'Dashboard', 'Profile', 'Add Doctor', and 'Add Admin'. The main content area has a blue background with a white form titled 'Add New Doctor'. The form includes the following fields: 'First Name' and 'Last Name' (text inputs); 'Email' (text input); 'Specialty' (dropdown menu showing 'Choose a specialty'); and 'License Number' (text input).

Fig.3.5 ADD Doctor

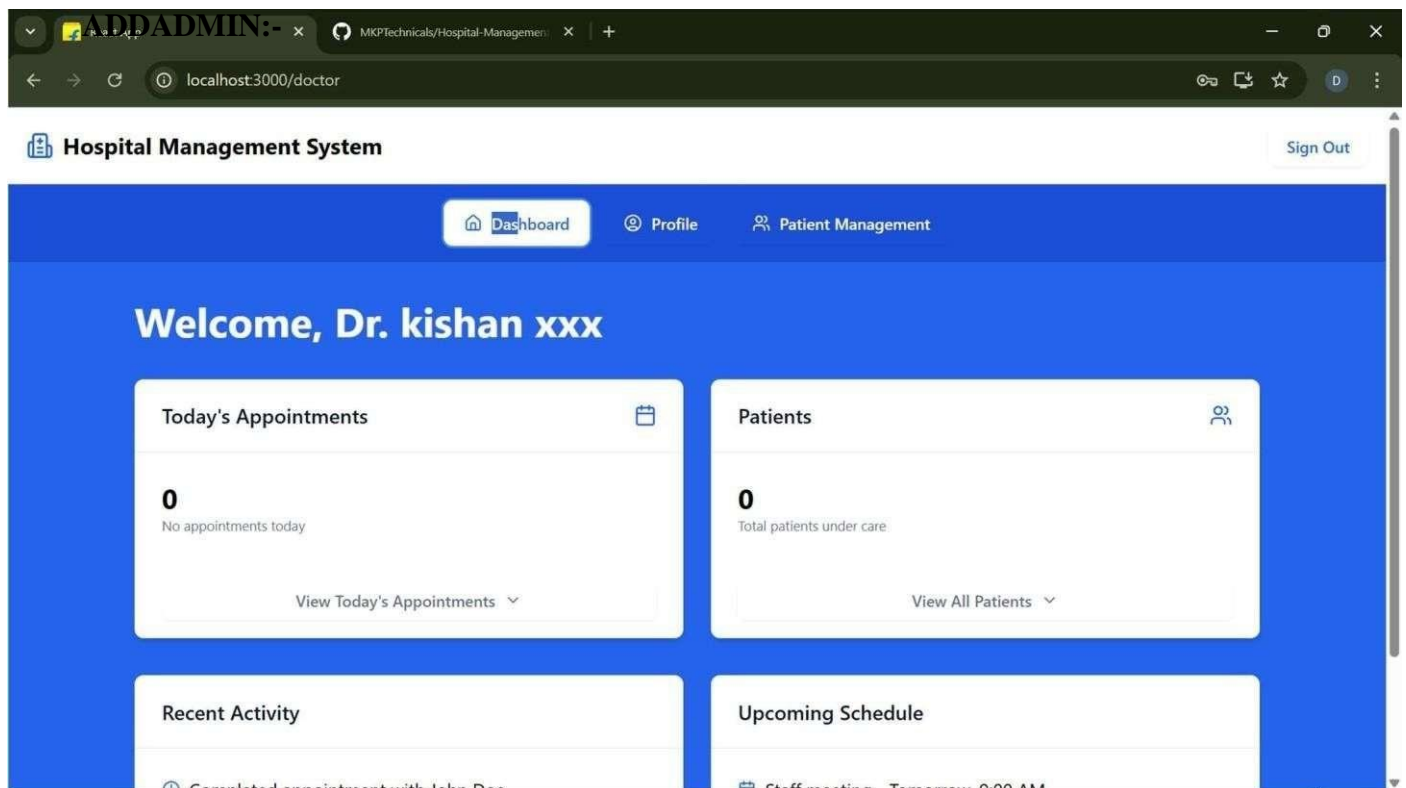


Fig.3.6 Patient management

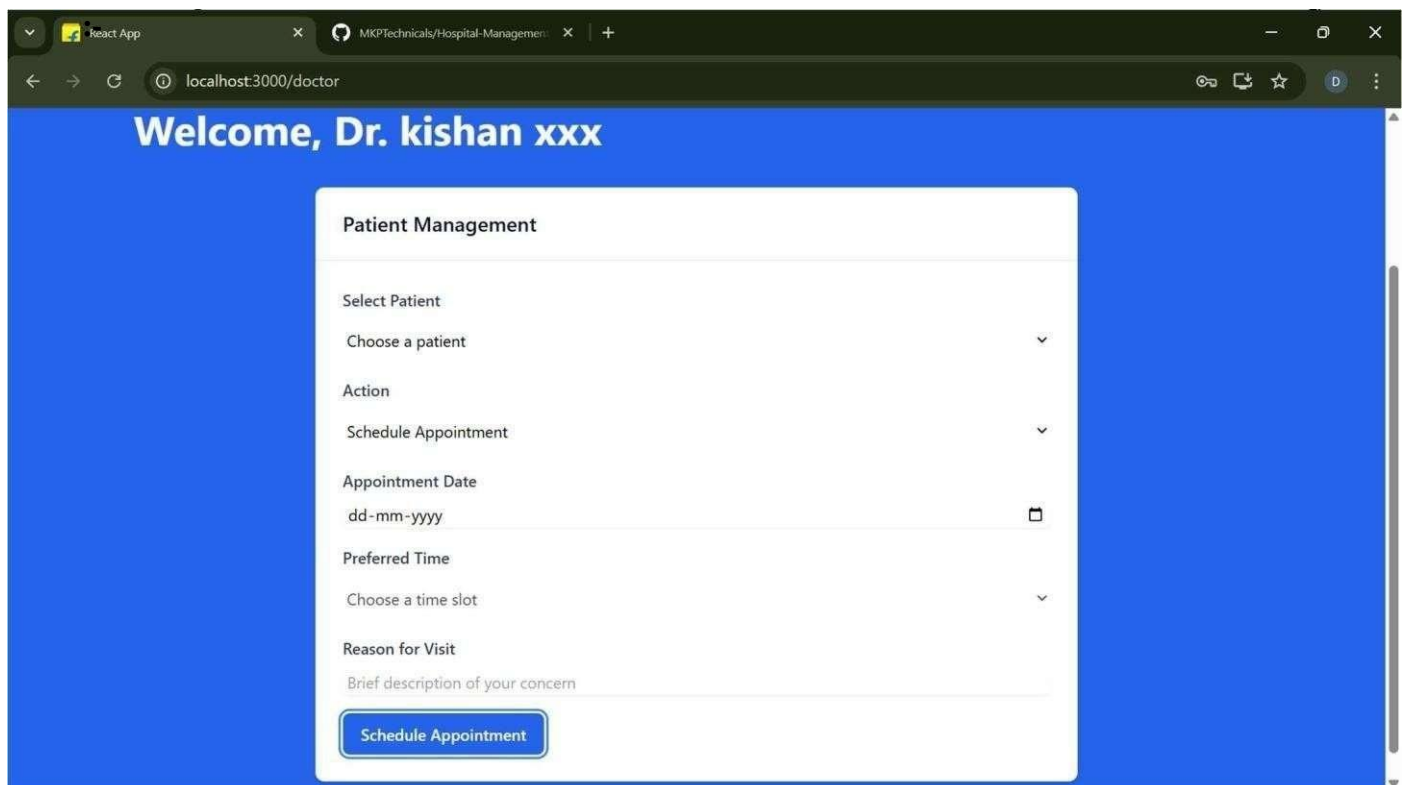


Fig.3.7 Appointment

TEST CASES

1. User Authentication

Test Case 1: Valid User Login

Input: Correct email and password

Expected Output: User successfully logs in and receives a JWT token

Test Case 2: Invalid User Login

Input: Incorrect email or password

Expected Output: "Invalid credentials" error message

Test Case 3: User Signup with Existing Email

Input: Email already registered

Expected Output: "User already exists" message

2. Appointment Booking

Test Case 4: Book Appointment with Available Doctor

Input: Patient selects available doctor, date, and time

Expected Output: Appointment is confirmed and saved in the database

Test Case 5: Book Appointment with Unavailable Doctor

Input: Patient selects a doctor who is already booked at the selected time

Expected Output: "Doctor unavailable" message

Test Case 6: View Patient Appointments

Input: Patient requests appointment list

Expected Output: Display list of upcoming and past appointments

3. Prescription Management

Test Case 7: Doctor Generates Prescription

Input: Doctor enters medicine details, dosage, and instructions

Expected Output: Prescription is saved in the database and linked to the patient

Test Case 8: Patient Views Prescription

Input: Patient requests prescription details

Expected Output: Prescription history is displayed

4. Billing & Payment Processing

Test Case 9: Generate Bill for Appointment

Input: Patient completes appointment, and billing details are entered

Expected Output: Bill is generated with correct treatment and medicine charges

Test Case 10: Process Successful Payment

Input: Patient pays bill using a valid payment method

Expected Output: Payment is processed, and status is updated to "Paid"

Test Case 11: Process Failed Payment

Input: Invalid card details or payment failure

Expected Output: "Payment failed" message

5. Notifications & Alerts

Test Case 12: Send Appointment Reminder to Patient

Input: Patient has an appointment scheduled for the next day

Expected Output: Automated SMS/Email reminder is sent

6. User Role & Access Control

Test Case 13: Admin Access Verification

Input: Admin logs in and tries to access staff management

Expected Output: Admin successfully accesses staff management panel

Test Case 14: Doctor Trying to Access Admin Panel

Input: Doctor logs in and tries to access hospital-wide reports

Expected Output: "Access Denied" message

7. Patient Management

Test Case 15: Register New Patient

Input: Enter valid patient details (name, age, gender, contact)

Expected Output: Patient record is created and stored in the database

Test Case 16: Fetch Patient Medical History

Input: Doctor searches for a patient's past medical records

Expected Output: Display complete medical history of the patient

Test Case 17: Update Patient Details

Input: Patient updates address and contact details

Expected Output: New information is saved successfully

8. Inventory & Pharmacy Management

Test Case 18: Add New Medicine to Inventory

Input: Admin adds a new medicine with quantity and expiry date

Expected Output: Medicine is successfully added to the inventory

Test Case 19: Check Medicine Stock Availability

Input: Pharmacist searches for a medicine

Expected Output: Display available stock and expiry date

Test Case 20: Low Stock Alert

Input: Medicine stock falls below the threshold level

Expected Output: System sends an alert to restock

9. Emergency & Critical Cases Handling

Test Case 21: Emergency Case Registration

Input: ER staff registers a critical patient without full details

Expected Output: System allows partial registration with "Emergency" status

Test Case 22: High-Priority Notification to Doctor

Input: Critical patient assigned to a doctor

Expected Output: Doctor receives instant notification

10. System Performance & Security

Test Case 23: System Load Handling

Input: 100+ users try to login simultaneously

Expected Output: System handles multiple logins without crash or slowdowns

Test Case 24: SQL Injection Attack Attempt

Input: Malicious SQL query in login form

Expected Output: System prevents attack and shows an error

Test Case 25: Data Backup & Recovery

Input: Simulate server failure and restore from backup

Expected Output: System restores all records without data loss

11. Doctor Management

Test Case 26: Add New Doctor to System

Input: Admin enters doctor details (name, specialization, contact, availability)

Expected Output: Doctor profile is successfully created and stored

Test Case 27: Doctor Updates Availability Schedule

Input: Doctor changes available time slots

Expected Output: New availability is reflected in the appointment system

Test Case 28: View Assigned Patients for the Day

Input: Doctor logs in and checks their scheduled appointments

Expected Output: System displays a list of today's patients

12. Staff & Shift Management

Test Case 29: Assign Shift to Nurse/Staff

Input: Admin assigns a nurse to a specific shift

Expected Output: Shift assignment is saved successfully

Test Case 30: Staff Checks Work Schedule

Input: Nurse logs in and checks their shift details

Expected Output: System displays assigned shift timing

Test Case 31: Unauthorized Staff Trying to Access Patient Data

Input: Receptionist attempts to view a patient's medical history

Expected Output: "Access Denied" message

13. Reports & Analytics

Test Case 32: Generate Monthly Revenue Report

Input: Admin selects a date range and requests financial reports

Expected Output: System generates a report with revenue from appointments and billing

Test Case 33: Generate Doctor Performance Report

Input: Admin selects a doctor and requests performance analytics

Expected Output: Report shows number of patients attended, feedback ratings, etc.

Test Case 34: Generate Patient Visit Statistics

Input: Admin requests data on total patient visits per department

Expected Output: System displays statistical breakdown

14. Multi-User & Session Management

Test Case 35: Prevent Duplicate Login from Multiple Devices

Input: User tries to login from two different devices

Expected Output: System restricts second login or logs out the first session

Test Case 36: Auto Logout After Inactivity

Input: User remains inactive for 30 minutes

Expected Output: System logs out the user for security reasons

Test Case 37: Simultaneous Appointment Booking Conflict

Input: Two patients try to book the same time slot with a doctor

Expected Output: System prevents double booking and notifies the second patient

15. Notifications & Communication

Test Case 38: Send Email Confirmation for New Appointments

Input: Patient books an appointment

Expected Output: Email confirmation is sent with appointment details

Test Case 39: SMS Alert for Prescription Refill Reminder

Input: Patient has a recurring prescription that is about to expire

Expected Output: System sends an SMS reminder to renew medication

Test Case 40: Emergency Broadcast to All Staff

Input: Admin triggers an emergency alert in the system

Expected Output: All doctors and medical staff receive instant notifications

FUTURE SCOPE

The Hospital Management System (HMS) is continuously evolving with technological advancements. Future enhancements can improve efficiency, security, and patient care through innovative solutions. Predictive Analytics for early disease detection and risk assessment. AI-powered Chatbots to assist patients with FAQs and appointment scheduling. Automated Diagnosis Assistance using machine learning models.

Wearable Device Integration to track patient vitals in real time. Remote Patient Monitoring for chronic disease management. Smart Hospital Beds & IoT Sensors for enhanced patient care. Decentralized Patient Records to ensure privacy and security. Tamper-proof Medical History for secure access across hospitals. Smart Contracts for Insurance Processing to automate claims. Cloud Storage for Medical Data to enable real-time access from anywhere. Telemedicine & Virtual Consultations for remote healthcare services. E-Prescriptions & Digital Health Reports to eliminate paperwork.

CONCLUSION

In conclusion, The Hospital Management System (HMS) plays a crucial role in modernizing healthcare operations by automating patient management, appointments, billing, and medical records. By leveraging technologies such as React.js for the frontend, Node.js for the backend, and MongoDB for data storage, the system ensures efficiency, accuracy, and ease of access for patients, doctors, and hospital administrators. This project enhances patient care, reduces administrative workload, and streamlines hospital workflows, making healthcare management more organized, scalable, and secure. With future advancements in AI, IoT, cloud computing, and blockchain, HMS can evolve into a fully integrated, intelligent healthcare ecosystem, improving medical decision-making and overall patient experience.

REFERENCES

- [1]. Tan, J., & Payton, F. C. (2021). *Healthcare Information Systems: Future Trends and Challenges* (3rd ed.). Springer. This book provides an in-depth analysis of health care information systems, including hospital management, electronic health records(EHRs),and emerging technologies in healthcare IT.
- [2]. Shortliffe , E. H.,& Cimino ,J.J.(2021).*Biomedical Informatics:Computer Applications in Health Care and Biomedicine* (5th ed.). Springer. A comprehensive resource covering the role of informatics in hospital management, patient care, and decision support systems.
- [3]. Gupta,P.,& Bhatt,R.(2020)."Machine Learning in Health care: Applications and Future Prospects." *Journal of Medical Systems*, 44(8), 152-168.This paper discusses how AI and machine learning improve hospital management systems, from patient diagnostics to resource allocation and workflow optimization.
- [4]. Davenport, T., & Kalakota, R. (2019). "The Potential for Artificial Intelligence in Healthcare ." *Future Healthcare Journal*, 6(2), 94-98.Analyzes the role of AI in hospital management, including predictive analytics, patient monitoring, and automation of administrative tasks.
- [5]. Kim,J.,& Lee,S.(2021)."The Evolution of Digital Health Systems:Trends and Innovations in Hospital Management. " *International Journal of Healthcare Information Systems*, 39(5), 220- 245.This article provides in sights into advancements in digital hospital management systems, focusing on cloud computing, telemedicine, and data security in healthcare IT.

CHAPTER-4

USE CASES - 1

HOTEL MANAGEMENT SYSTEM

Front Desk Operations

1. Guest Check-in/Check-out

- Automate registration and payment.
- Generate digital or printed invoices.

2. Room Assignment

- Assign rooms based on guest preferences and availability.

3. Reservation Management

- Handle bookings (online, walk-in, OTA integration).
- Modify, cancel, or extend stays.

4. Channel Management

- Synchronize room availability and rates across OTAs (e.g., Booking.com, Expedia).

5. Group Bookings

- Manage large bookings (weddings, conferences, etc.).

6. Booking Engine Integration

- Allow direct booking from the hotel website.

Billing & Payments

7. Invoice Generation

- Create detailed invoices with room charges, services, taxes, etc.

8. Multi-currency & Tax Handling

- Automatically apply local taxes and support international payments.

9. Split Payments & Corporate Billing

- Handle different payment methods or split charges between guests.

Housekeeping Management

10. Room Status Tracking

- Mark rooms as clean, dirty, under maintenance, etc.

11. Task Assignments

- Assign rooms and floors to housekeeping staff.

12. Inventory Management

- Track usage of linens, toiletries, minibar items.

Food & Beverage Integration

13. Restaurant & Room Service Orders

- Post F&B charges to guest folios directly.

14. Table Reservations

- Enable guests to book tables in the hotel restaurant.

Reporting & Analytics

15. Occupancy Reports

- Show room usage by date, type, etc.

16. Revenue Reports

- Daily, monthly, and annual earnings by source.

17. Guest Behavior Insights

- Track return guests, preferences, and reviews.

Use Case 2: Hospital Management System

Patient Management

1. Patient Registration

- Capture personal details, contact info, and medical history.

2. Appointment Scheduling

- Book, reschedule, or cancel appointments with doctors.

3. Patient Check-in/Check-out

- Automate admission, discharge, and transfer processes.

4. Emergency Case Handling

- Fast-track registration and treatment for emergency patients.

Doctor & Clinical Staff

5. Doctor Scheduling

- Manage duty rosters and doctor availability.

6. Access to Patient Records

- View patient history, prescriptions, lab reports, and diagnoses.

7. Electronic Medical Records (EMR)

- Update clinical notes, diagnoses, and treatments in real time.

Pharmacy Management

8. Drug Inventory Management

- Track stock levels, expiry dates, and restocking needs.

9. Prescription Fulfillment

- Dispense medication based on doctor prescriptions.

10. Billing Integration

- Add medicine charges directly to patient bills.

Laboratory Management

11. Test Ordering

- Doctors can order lab tests directly from the system.

12. Sample Collection & Tracking

- Record sample status and ensure traceability.

13. Result Entry & Sharing

- Upload test results and notify doctors/patients.

Billing & Payments

14. Inpatient and Outpatient Billing

- Generate bills for consultation, treatment, room charges, etc.

15. Insurance Claim Processing

- Handle pre-authorization and claims with insurance companies.

16. Payment Gateway Integration

- Enable online payments and print receipts.

Ward & Bed Management

17. Room Allocation

- Assign beds/wards based on availability and patient needs.

18. Bed Availability Tracking

- Real-time status of each bed (occupied, cleaning, available).

19. Discharge Summary

- Automatically generate discharge reports and follow-up instructions.

Reporting & Administration

20. MIS Reports

- Generate reports on revenue, patient counts, occupancy, etc.

21. Regulatory Compliance

- Maintain records for audits, accreditation, and government bodies.

22. Data Analytics

- Analyze trends in treatment, outcomes, and resource use.



RESUME

NAME :ABHISHEK SHINTRE

DIPLOMA BRANCH : COMPUTER SCIENCE & ENGINEERING

COLLEGE : K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI, 591201

ADDRESS : Chikodi

AGGREGATE : 74.00%

Career Objectives:-

I am seeking for an opportunity to start my career with your organization which will make me to grow competitively, which will enhance my growth by meeting social concern.

Educational Qualification:-

a) Diploma in COMPUTER SCIENCE & ENGINEERING:-

	INSTITUTE/BOARD	YEAR OF PASSING	CLASS OBTAINED	PERCENTAGE%/ SGPA
6 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]			
5 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]			
4 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	8.08
3 RD	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	8.23
2 ND	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]			
1 ST	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]			

b) Prior Education:-

	Institute	Year of passing	Marks obtained	Percentage %
SSLC	CBSE	2021	366	66.8%
PUC	STATE	2023	373	62.7%

Software efficiency:-

Language	HTML, CSS, PYTHON, JAVA, DBMS
----------	-------------------------------

Name : ABHISHEK SHINTRE

Personal profile:-

Father name : MAHADEV SHINTRE

Mother name :SHUBHANGI SHINTRE

Nationality : Indian

DOB : 29/06/2004

Hobbies : CRICKET

Language known : KANNADA, ENGLISH, HINDI, MARATHI

Marital status : SINGAL

Alternate contact no : 7406009525

Address : A/P:- SOUNDALGA DIST: Belgaum, Karnataka

Email :-abhishekshintre1@gmail.com

Declaration:-

I hereby informed that above mentioned details are true the best of my knowledge.

Place:- SOUNDALGA

Date:

abhishek shintre
(student name.)

NAME :DINESH ASHOK JADAGE

DIPLOMA BRANCH : COMPUTER SCIENCE & ENGINEERING

COLLEGE : K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI, 591201

ADDRESS : Chikodi

AGGREGATE : 86.40%

Career Objectives:-

I am seeking for an opportunity to start my career with your organization which will make me to grow competitively, which will enhance my growth by meeting social concern.

Educational Qualification:-

a) Diploma in COMPUTER SCIENCE AND ENGINEERING:-

SEM	INSTITUTE/BOARD	YEAR OF PASSING	CLASS OBTAINED	PERCENTAGE%/SGPA
6 TH	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	WATING	WATING	WATING
5 TH	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2024-25	WATING	WATING
4 TH	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	9.69
3 RD	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	9.62
2 ND	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2022-23	DISTINCTION	9.18
1 ST	339-[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2022-23	DISTINCTION	8.89

b) Prior Education:-

	Institute	Year of passing	Marks obtained	Percentage %
SSLC	STATE	2022	562	89.92%

Software efficiency:-

Language	HTML, CSS, PYTHON, JAVA, DBMS
----------	-------------------------------

Personal profile:-

Name : DINESH ASHOK JADAGE
Father name : ASHOK SHIVAPPA JADAGE
Mother name : SATYAWA SHANKAR VITEKAR
Nationality : Indian
DOB : 26/10/2006
Hobbies : PLAYING CHESS
Language known : KANNADA, ENGLISH, HINDI,
Marital status : SINGLE
Alternate contact no 8088219156
Address : NEAR SAI MANDIR ATHANI, DIST: Belgaum, Karnataka
Email :jadagedinesh12@gmail.com

Declaration:-

I hereby informed that above mentioned details are true the best of my knowledge.

Place:- NEAR SAI MANDIR CHURCH ROAD ATHANI.

Dinesh Jadage

Date:

NAME :KISHAN NINAGANGAUDA MANGASULE

DIPLOMA BRANCH : COMPUTER SCIENCE & ENGINEERING

COLLEGE : K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI, 591201

ADDRESS : Chikodi

AGGREGATE : 90.80%

Career Objectives:-

I am seeking for an opportunity to start my career with your organization which will make me to grow competitively, which will enhance my growth by meeting social concern.

Educational Qualification:-

a) Diploma in Mechatronics Engineering:-

SEM	INSTITUTE/BOARD	YEAR OF PASSING	CLASS OBTAINED	PERCENTAGE %/SGPA
6 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	WATING	WATING	WATING
5 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2024-25	DISTINCTION	10.00
4 TH	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	10.00
3 RD	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2023-24	DISTINCTION	10.00
2 ND	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2022-23	DISTINCTION	9.64
1 ST	339 –[K L E SOCIETY'S C B KORE POLYTECHNIC CHIKODI]	2022-23	DISTINCTION	9.56

b) Prior Education:-

	Institute	Year of passing	Marks obtained	Percentage %
SSLC	CBSE	2022	314	62.80%

Software efficiency:-

Language	HTML, CSS, PYTHON, JAVA, DBMS
----------	-------------------------------

Personal profile:-

Name : KISHAN NINAGANGAUDA MANGASULE
Father name : NINAGANGAUDA MANGASULE
Mother name : LALITA NINAGANGAUDA MANGASULE
Nationality : Indian
DOB : 13/11/2006
Hobbies : CRICKET
Language known : KANNADA, ENGLISH, HINDI, MARATHI
Marital status : SINGAL
Alternate contact no 9591057263
Address : D K S S K FACTORY LAXMI NAGAR NANADI, DIST: Belgaum, Karnataka
Email :kishanmangsule21@gmail.com

Declaration:-

I hereby informed that above mentioned details are true the best of my knowledge.

Place:- D K S S K FACTORY LAXMI NAGAR NANADI
Date:

kishan mangasule
(student name.)

NAME :- ISHWAR RAJENDRA GANDH

DIPLOMA BRANCH :- COMPUTER SCIENCE ENGINEERING

E-MAIL:- ishwarcs0011@gmail.com

COLLEGE:- C.B. Kore polytechnic college ,Chikodi

CONTACT NO:- 9972854419

ADDRESS:- CHIKODI

Career Objectives:-

I am seeking for an opportunity to start my career with your organization which will make me to grow competitively, which will enhance my growth by meeting social concern.

Educational Qualification:-

a) Diploma in Mechatronics Engineering:-

SE M	INSTITUTE/BOARD	YEAR OF PASSIN G	CLASS OBTAIN ED	PERCENTAGE%/ SGPA
6 TH	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2024-25	WAITIN G	WAITING
5 TH	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2024-25	DISTINC TION	8.1
4 TH	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2023-24	DISTINC TION	8.48
3 RD	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2023-24	DISTINC TION	8.62
2 ND	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2022-23	DISTINC TION	9.27
1 ST	339[KLESOCIETY'S CB.KORE.POLYTE CHNIC CHIKODI]	2022-23	DISTINC TION	7.56

b) Prior Education:-

	Institute	Year of passing	Class obtained	Percentage %
SSLC	CBSE	2022	472	79.85

Software efficiency:-

Language	HTML,CSS,PYTHON,JAVA,DBMS
----------	---------------------------

INTERNSHIP PROJECT PROFILE

TITLE:- HOSPITAL MANAGEMENT SYSTEM & HOTEL MANAGEMENT SYSTEM

Personal profile:-

Name :- ISHWAR GANDH

Father name :- RAJENDRA GANDH

Mother name :-JAYASHREE GANDH

Nationality :-REPUBLIC OF INDIAN

DOB:-16-11-2005

Hobbies:-FOOTBALL ,CRICKET ,CARROM ,etc.

Language known :-HINDI ,KANNADA ,MARATHI ,ENGLISH.

Marital status:-SINGLE

Alternate contact no :-9448941709

Declaration:-

I hereby informed that above mentioned details are true the best of my knowledge.

Place:- Basweshwar Circle ,Sadurga

Date:-

ISHWAR .R. GANDH
(student name.)

APPENDICES

Appendix No	Title	Description
Appendix A	Source code	Contains the complete source code used in the project.
Appendix B	Project Screenshots	Screenshots of various modules and interface developed.
Appendix C	Survey/Questionnaire	Blank and filled survey forms used for data collection
Appendix D	User Manual / Installation Guide	Step-by-step instructions for installation and usage of the application.
Appendix E	Additional Charts / Graphs	Extra data visualizations or performance results.