# A

# PYTHON PROJECT REPORT

# On

# " HOTEL MANAGEMENT SYSTEM "

Submitted in partial fulfillment of the requirements for the award of the degree of

**BACHELOR OF TECHNOLOGY**

# in

**CSE (AI&DS)**

## by

22BH1A7208 G.DIVYA

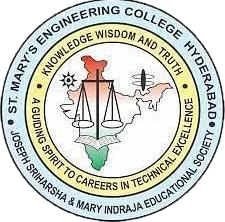
23BH5A7201 CH.POOJA

22BH1A7210 G.ARUN

23BH5A7206 P.ESHWAR

22BH1A7214 M.ASHOK

**UNDER THE GUIDENCE OF**



DEPARTMENT OF CSE(AI&DS)

**St. Mary’s Engineering College**

**(Approved by AICTE, NEW DELHI. & Affiliated to JNTU-HYDERABAD, Accredited by**

**NAAC)** Deshmukh (V), Pochampally (M), Yadadri Bhuvanagiri (D), Telangana -508284

[2023-24]

**CERTIFICATE**

This is to certify that project report entitled "HOTEL MANAGEMENT SYSTEM” is bonafide work carried out in the II/II semester by G.DIVYA(22BH1A7208),CH.POOJA(23BH5A7201),G.ARUN (22BH1A7210), P.ESHWAR (23BH5A7206),M.ASHOK(22BH1A7214) in partial fulfillment award of Bachelor of Technology in CSE (AI&DS) from St. Mary's Engineering College during the academic year 2023-2026.

INTERNAL GUIDE HEAD OF THE DEPARTMENT

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this project would be incomplete without the mention of the people who made it possible. We consider it as a privilege to express our gratitude and respect to all those who guided us in the completion of the project.

We are thankful to our internal guide PRIYANKU, in the St. Mary's Engineering College for having been of a source encouragement and for insisting on vigour to do this project work.

We are obliged to Dr. K. Suresh Kumar, Head of the Department of CSE(AI&DS), St. Mary's Engineering College for his guidance and suggestions throughout the project work.

We take this opportunity to express a deep sense of gratitude to Dr. T.N. Srinivas Rao, Principal of St. Mary's Engineering College for allowing us to do this project and for this affectionate encouragement in presenting this project work.

We convey our sincere thanks to Sri Rev. K.V.K RAO, Chairman of St. Mary's Engineering College for giving us a learning environment to grow out self personally as well as professionally.

We would like to express our thanks to all staff members who have helped us directly and indirectly in accomplishing this project work. We also extended our sincere thanks to our parents and friends for their moral support throughout the project work. Above all, we thank god almighty for his manifold mercies in carrying out this project work successfully.

22BH1A7208 G.DIVYA

23BH5A7201 CH.POOJA

22BH1A7210 G.ARUN

23BH5A7206 P.ESHWAR

22BH1A7214 M.ASHOK

**DECLARATION**

This is to certify that the work report in the thesis titled, " HOTEL MANAGEMENT SYSTEM ", submitted to the Department of CSE (AI&DS), St. Mary's Engineering College in fulfillment of degree for the award of Bachelor of Technology, is a bonafide work done by us. No part of the thesis is copied from books, journals, or the internet and wherever the portion is taken, the same has been duly referred in the text. The reported results are based on the project work entirely done by us and not copied from any other sources. Also, we declare that the matter embedded in this thesis has not been submitted by us in full or partially there for the award of any degree from any other institution or university previously

G.DIVYA -22BH1A7208

CH.POOJA -23BH5A7201

G.ARUN -22BH1A7210

P.ESHWAR -23BH5A7206

M.ASHOK -22BH1A7214

**Abstract**

The Hotel Management system HMS is a comprehensive software solution designed to streamline and automate the daily operations of hotels,resorts,and other hospitality establishment.

The system manager room reservations, guest information,front desk operations, housekeeping,billing,and revenue management in a single, intuitive platform.

Key features including:

. Room booking and reservation management

.Guest check -in/check -out and billing

. Housekeeping and room assignment management

. Reportin and analytics for revenue and occupancy

The HMS aims to improve efficiency,reduce errors,and enhance the Ave guest experience,making it an essential tool for hoteliers and hospitality professionals.

**Introduction**

The Hotel Management System project in Python is a console-based application designed to streamline and automate essential hotel management tasks. This system aims to provide a user-friendly interface for hotel administrators to efficiently manage rooms, guests, and bookings. By leveraging Python's capabilities, the application offers functionalities such as adding and updating room details including room number, type, and price. It also facilitates guest registration, capturing crucial information such as guest names, ages, check-in dates, and allocated room numbers.

This project is particularly beneficial for smaller hotels or educational purposes, offering a foundational understanding of how programming can optimize daily hotel operations. The console-based approach ensures simplicity and ease of use, allowing staff to quickly access and update information related to rooms and guests. Additionally, the system supports real-time updates on room availability, aiding in effective room allocation and maximizing occupancy rates.

Through this project, users can gain insights into basic hotel management practices while honing their skills in Python programming. Future enhancements could include advanced features such as check-out management, billing systems, and reporting functionalities to cater to more comprehensive hotel management needs.

**Problem Statement**

Problem Statement: Hotel Management System

The objective of this Python-based console application is to develop a Hotel Management System that simplifies and enhances the management of rooms and guests within a hotel environment. The system aims to automate core hotel operations and provide an efficient platform for hotel administrators to handle various tasks seamlessly.

Key Features to Address:

Room Management: Implement functionalities to add new rooms with details such as room number, type (standard, deluxe, suite), and price per night. Enable updates to room availability status as rooms are occupied or vacated.

Guest Management: Provide capabilities for registering guests by capturing essential information including name, age, check-in date, and assigned room. Ensure that the system can handle multiple guests checking in and out simultaneously.

Real-time Updates: Enable real-time updates on room availability to facilitate quick allocation of rooms to incoming guests. Ensure that room availability status is accurately reflected and updated as operations proceed.

Data Integrity: Implement mechanisms to maintain data integrity by using timestamping for guest check-in dates and ensuring that guest and room information is stored and retrieved accurately.

User Interface: Design a user-friendly console interface with clear menu options for functionalities like adding rooms, registering guests, viewing room and guest details, and managing system operations.

Error Handling: Incorporate robust error handling mechanisms to manage scenarios such as room availability checks, guest registration validations, and user input errors effectively.

By addressing these features, the Hotel Management System aims to streamline operations, improve efficiency in room allocation and guest management, and enhance overall customer satisfaction through better service delivery. The project also serves as an educational tool for understanding basic hotel management principles and applying Python programming concepts in a practical setting.

**Software Requirements**

**Python 3.8+**: The core programming language for developing the application.

**Software Environment**

**Operating System**: The application should be compatible with major operating systems, including Windows, macOS, and Linux.

**Integrated Development Environment (IDE)**: Recommended IDEs include PyCharm, VS Code, or any text editor that supports Python development, such as Sublime Text or Atom.

**Python Package Manager (pip)**: For installing and managing Python packages and dependencies.

**Version Control System (Git)**: For tracking changes, collaborating with others, and maintaining the project’s source code.

**Code**  
import datetime

class HotelManagementSystem:

    def \_\_init\_\_(self):

        self.rooms = {}  # Stores room info with room number as key

        self.guests = {}  # Stores guest info with guest ID as key

        self.current\_guest\_id = 0

    def add\_room(self, room\_number, room\_type, price):

        room\_info = {

            "room\_number": room\_number,

            "room\_type": room\_type,

            "price": price,

            "available": True

        }

        self.rooms[room\_number] = room\_info

        return f"Room {room\_number} added successfully."

    def add\_guest(self, name, age, room\_number):

        if room\_number in self.rooms and self.rooms[room\_number]['available']:

            self.current\_guest\_id += 1

            guest\_info = {

                "id": self.current\_guest\_id,

                "name": name,

                "age": age,

                "room\_number": room\_number,

                "check\_in\_date": datetime.datetime.now()

            }

            self.guests[self.current\_guest\_id] = guest\_info

            self.rooms[room\_number]['available'] = False

            return f"Guest {name} added successfully with ID {self.current\_guest\_id}."

        else:

            return "Room not available."

    def view\_rooms(self):

        for room\_number, room\_info in self.rooms.items():

            print(f"Room Number: {room\_number}, Type: {room\_info['room\_type']}, Price: {room\_info['price']}, Available: {room\_info['available']}")

    def view\_guests(self):

        for guest\_id, guest\_info in self.guests.items():

            print(f"ID: {guest\_id}, Name: {guest\_info['name']}, Age: {guest\_info['age']}, Room Number: {guest\_info['room\_number']}, Check-in Date: {guest\_info['check\_in\_date']}")

def main():

    hms = HotelManagementSystem()

    while True:

        print("\nHotel Management System")

        print("1. Add Room")

        print("2. Add Guest")

        print("3. View Rooms")

        print("4. View Guests")

        print("5. Exit")

        choice = input("Enter your choice: ")

        if choice == '1':

            room\_number = input("Enter room number: ")

            room\_type = input("Enter room type: ")

            price = float(input("Enter room price: "))

            print(hms.add\_room(room\_number, room\_type, price))

        elif choice == '2':

            name = input("Enter guest name: ")

            age = int(input("Enter guest age: "))

            room\_number = input("Enter room number: ")

            print(hms.add\_guest(name, age, room\_number))

        elif choice == '3':

            hms.view\_rooms()

        elif choice == '4':

            hms.view\_guests()

        elif choice == '5':

            print("Exiting...")

            break

        else:

            print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

    main()  
  
  
  
  
  
**Output**  
Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 1

Enter room number: 101

Enter room type: Standard

Enter room price: 100

Room 101 added successfully.

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 1

Enter room number: 102

Enter room type: Suite

Enter room price: 200

Room 102 added successfully.

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 2

Enter guest name: Alice Smith

Enter guest age: 30

Enter room number: 101

Guest Alice Smith added successfully with ID 1.

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 2

Enter guest name: Bob Johnson

Enter guest age: 25

Enter room number: 102

Guest Bob Johnson added successfully with ID 2.

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 3

Room Number: 101, Type: Standard, Price: 100.0, Available: False

Room Number: 102, Type: Suite, Price: 200.0, Available: False

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 4

ID: 1, Name: Alice Smith, Age: 30, Room Number: 101, Check-in Date: 2024-06-18 12:00:00.000000

ID: 2, Name: Bob Johnson, Age: 25, Room Number: 102, Check-in Date: 2024-06-18 12:01:00.000000

Hotel Management System

1. Add Room

2. Add Guest

3. View Rooms

4. View Guests

5. Exit

Enter your choice: 5

Exiting…

**Conclusion**

The console-based Hotel Management System developed in Python provides a practical solution for efficiently managing rooms and guest information within a hotel setting. This project showcases the capability of Python to create a straightforward yet effective system that supports essential hotel operations including room management, guest registration, and real-time status updates.

Key features and benefits of the Hotel Management System include:

**Room Management**: The system allows hotel administrators to add and maintain detailed information about rooms, including room number, type, price, and availability status. This ensures that room allocation and pricing are accurately managed.

**Guest Management**: Hotel staff can seamlessly register guests, capturing essential details such as name, age, check-in date, and allocated room number. This helps streamline guest services and ensures efficient handling of guest information.

**Real-time Updates**: The system provides real-time updates on room availability, ensuring that staff can quickly identify and allocate available rooms to incoming guests. This helps in maximizing room occupancy and optimizing revenue generation.

**User-Friendly Interface**: Designed with a simple console interface, the system offers intuitive menu options that make it easy for hotel staff to perform tasks such as adding rooms, registering guests, and viewing room and guest information.

**Data Integrity**: Utilizes Python's datetime module for accurate timestamping of guest check-ins, ensuring that records are maintained with precision and integrity.

This project serves as a foundational tool for small to medium-sized hotels looking to automate and streamline their operations, without the complexity of larger-scale hotel management systems. By automating routine tasks and centralizing data management, the Hotel Management System enhances operational efficiency, improves guest service delivery, and contributes to overall customer satisfaction.

In conclusion, the Python-based console application demonstrates the practical application of programming in enhancing hotel management operations. Future enhancements could include additional features such as check-out management, billing and invoicing, reporting capabilities, and integration with external systems to further enhance functionality and meet evolving hotel management needs.