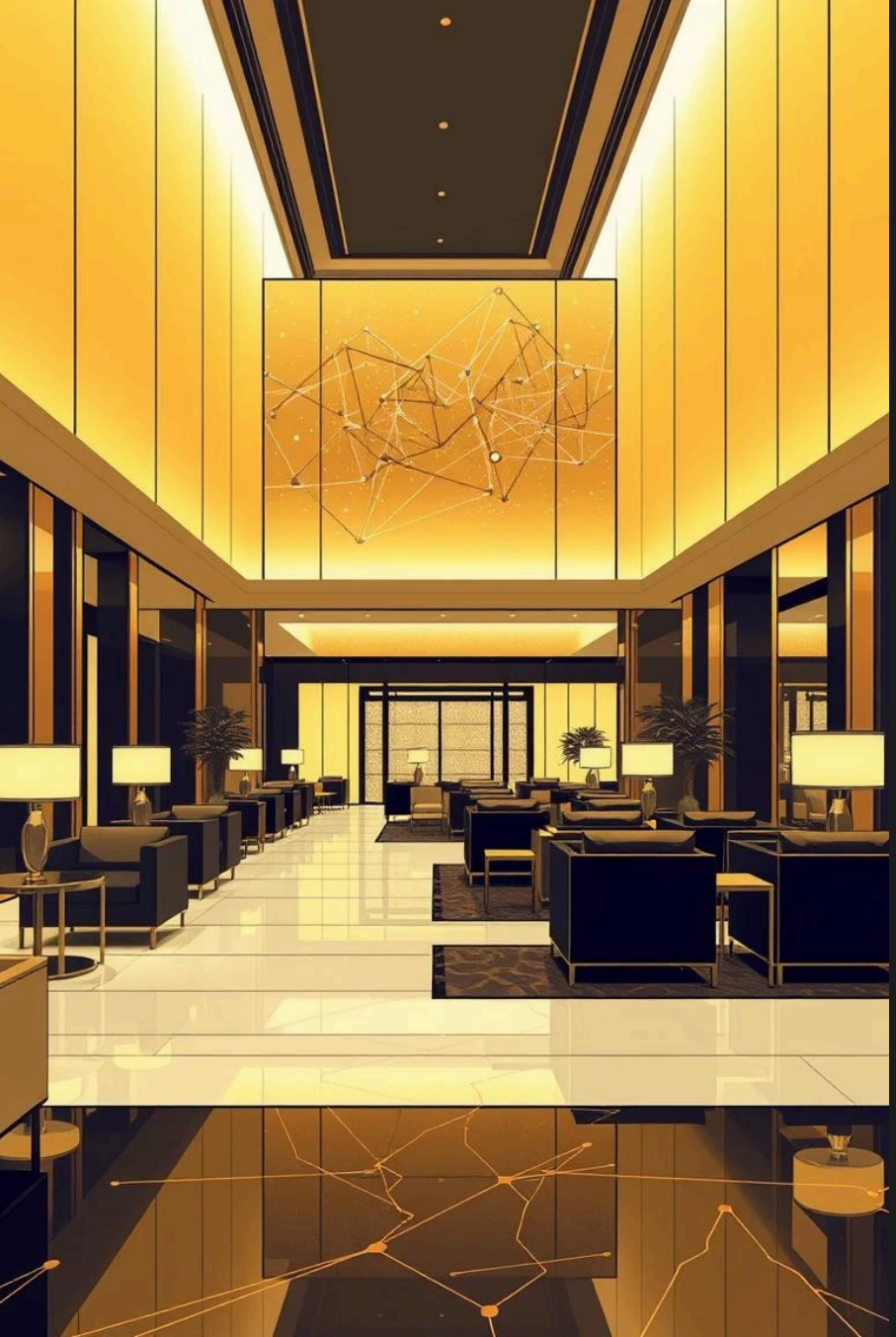


Hotel Management System

Project Overview

This project focuses on creating a robust Hotel Management System.

- **Presented by:** Vaishnavi Gupta

Python & PyCharm

Core logic and development environment.

MySQL Workbench

Database management and structure.

Tkinter

Graphical User Interface toolkit.

Introduction: Revolutionising Hotel Operations

The primary objective of this project was to develop a **desktop-based hotel management system** designed to streamline operations.

- It empowers hotel staff to efficiently manage **customer information**, **room allocations**, and **booking processes**.
- This application serves as a comprehensive tool for **hotel reception** and **administrative personnel**, enhancing their daily workflows.

Technology Stack

Our Hotel Management System is built upon a robust and well-integrated set of technologies.

Python

The **core programming language** for all backend logic and system functionalities.

MySQL

A powerful relational database management system, used to securely **store and manage** all customer, room, and booking data.

Tkinter

Python's standard GUI library, chosen for its simplicity and effectiveness in **building the intuitive user interface**.

PyCharm

The integrated development environment (IDE) that facilitated **efficient coding, debugging**, and project management throughout the development cycle.

System Features: GUI Overview

The application's main Graphical User Interface is designed for ease of navigation, featuring a clear menu with distinct sections:

1

Customer

Dedicated module for **adding, updating, and deleting customer details**, ensuring accurate guest records.

2

Room

Manages all **room-related information**, including type, availability status, and pricing, vital for inventory control.

3

Booking

Facilitates the **creation of new bookings** and provides an overview of all existing reservations, optimising occupancy.

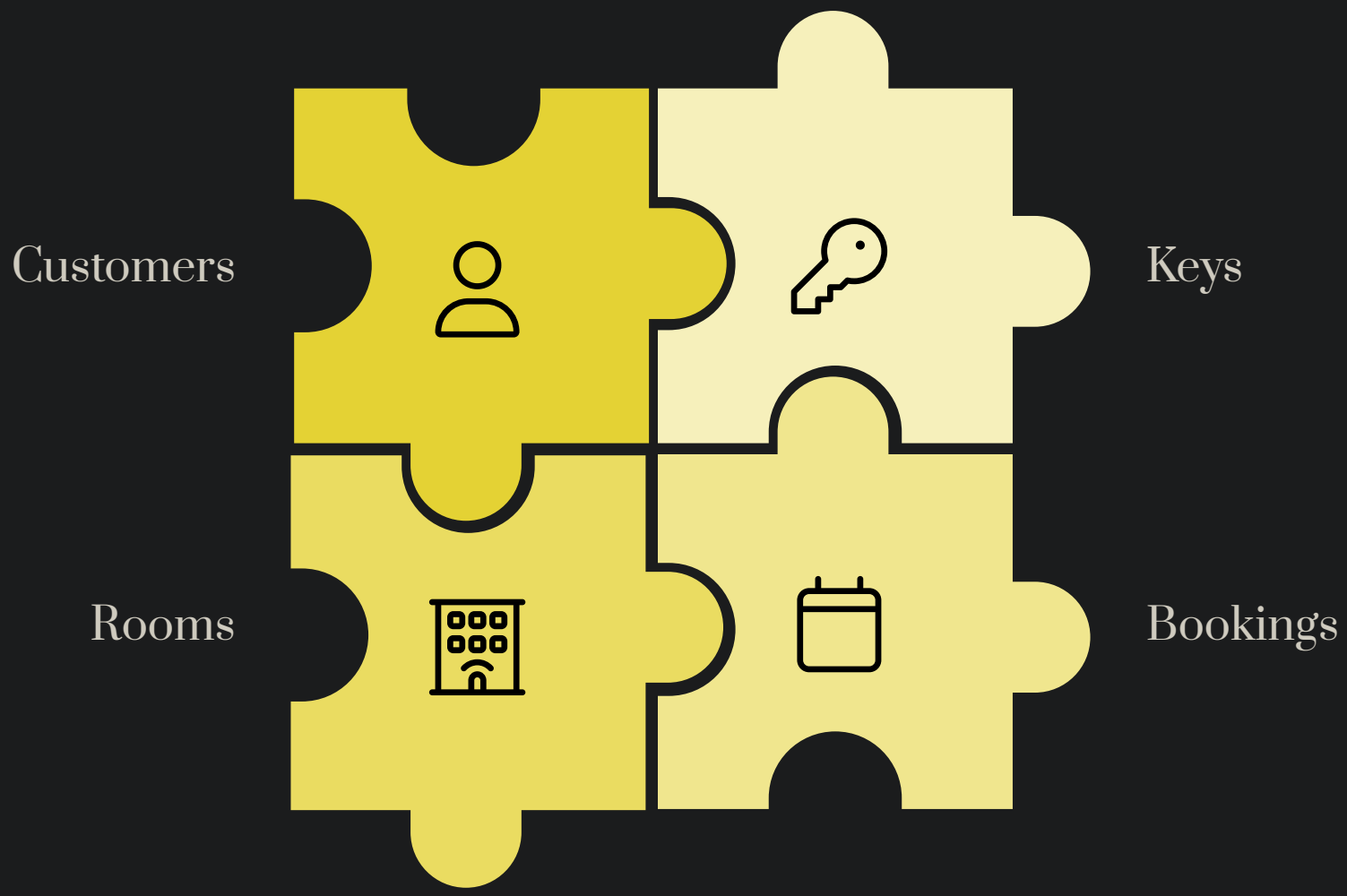
4

Logout

Provides a secure method to **exit the application** or log out of the current user session.

Database Architecture

The system's backbone is a meticulously designed MySQL database, ensuring data integrity and efficient retrieval. Key tables include:



- **Customers Table:** Stores guest personal details. (e.g. `customer_id` PRIMARY KEY)
- **Rooms Table:** Manages room types, rates, and current availability. (e.g. `room_number` PRIMARY KEY)
- **Bookings Table:** Records reservation details, linking customers to rooms. (e.g. `booking_id` PRIMARY KEY, `customer_id` FOREIGN KEY, `room_number` FOREIGN KEY)

System Workflow: A Step-by-Step Guide

The user interaction within the system follows a logical progression, ensuring a smooth and intuitive experience:

01

User Login

Secure authentication to access the system functionalities.

02

Menu Navigation

Users select from Customer, Room, or Booking modules based on their task.

03

Data Management

Perform CRUD (Create, Read, Update, Delete) operations on relevant data.

04

MySQL Storage

All data changes are instantly and securely committed to the MySQL database.

05

Logout/Exit

Users can safely log out, preserving data integrity.

Challenges & Key Learnings

Key Challenges Encountered:

- **Python-MySQL Connectivity:** Ensuring seamless and reliable data exchange between the application and the database.
- **Tkinter GUI Design:** Crafting an intuitive and visually appealing user interface within Tkinter's framework.
- **CRUD Operations:** Implementing robust Create, Read, Update, and Delete functionalities for all data types.

Valuable Learnings:

- **GUI Design Principles:** Gained practical experience in creating user-friendly interfaces.
- **Database Handling:** Mastered efficient data storage, retrieval, and manipulation techniques.
- **Project Structuring:** Developed skills in organising a complex application for maintainability and scalability.

Conclusion & Future Directions

This project successfully delivered a functional Hotel Management System.

Summary of Achievement

- A **fully functional desktop application** integrating GUI with database backend.
- Proven utility for **streamlining hotel operations**.
- Demonstrates practical application of Python, Tkinter, and MySQL in a **real-world scenario**.

Future Enhancements

- **Advanced Login System:** Incorporate multi-user roles and secure authentication.
- **Booking Reports:** Generate detailed analytical reports for occupancy and revenue.
- **GUI Framework Migration:** Explore alternative GUI frameworks (e.g., PyQt, Kivy) for enhanced aesthetics and features.