PROJECT: HOTEL ROOM BOOKING CRUD MANAGER

Identify a Real-World Problem or Need

Hotels need an efficient way to manage room bookings, customer information, and reservation status. Manual systems lead to errors such as double booking, missing records, and difficulty tracking available rooms. A computerized system improves accuracy and access to information.

This project provides a simple CRUD (Create, Read, Update, Delete) software to manage hotel room bookings.

□

2. Objectives and Expected Outcomes

Objectives

- To design a software system that manages hotel room bookings.
- To implement CRUD operations:

➤Add new bookings

➤View booking details

➤Update booking records

➤Delete or cancel bookings

- To apply structured programming and top-down design concepts.
- To use appropriate tools (Python/Java/C++, file handling, or database integration).
- To follow a proper development process from analysis to testing.

Expected Outcomes

- A functional program that allows maintaining booking data.
- A clear interface for performing CRUD operations.
- Accurate storage and retrieval of reservation data.
- Improved understanding of modular programming.

3. Problem Definition

The hotel requires a computer-based system to manage the following tasks:

- Register new customer bookings
- Store room type and check-in/check-out dates
- Display booking details
- Modify booking information if required
- Cancel bookings
- Track room availability

The software should be simple, user-friendly, and capable of running on common hardware.

□

4. Requirement Analysis

4.1 Functional Requirements

- FR1: Add new booking
- FR2: Search/view booking details
- FR3: Update existing booking
- FR4: Delete/cancel booking
- FR5: Display all bookings
- FR6: Validate room availability

4.2 Non-Functional Requirements

- Simple command-line interface
- Reliable and consistent performance
- Data stored using files or a small local database
- Easily maintainable modular code

4.3 System Requirements

- Any OS (Windows/Linux/Mac)
- Programming language (Python/Java/C/C++)
- Storage using text file or SQLite

□

5. Top-Down Design / Modularization

Main System Modules

1. Main Menu Module

- Display options
- Accept user choice

2. Booking Management Module

- add_booking()
- view_booking()
- update_booking()
- delete_booking()

3. Room Availability Module

- check_room_availability()

4. Storage Module

- save_to_file()
- load_from_file()

6. Algorithm Development

Algorithm: Add Booking

1. Start
2. Input customer name, phone, room type, check-in date, check-out date
3. Check if room is available
4. If available → save record
5. Else → display “Room Not Available”
6. End

Algorithm: View Booking

1. Start

2. Enter booking ID
3. Search file/database
4. If found → display details
5. Else → display ″Record Not Found”
6. End

Algorithm: Update Booking

1. Start
2. Enter booking ID
3. Search and load record
4. If found → allow user to modify fields
5. Save updated record
6. End

Algorithm: Delete Booking

1. Start
2. Enter booking ID
3. Remove record from file/database
4. Confirm deletion
5. End