

Florida International University
Knight Foundation School of Computing and Information Sciences

Final Deliverable

Hotel Management System

Team Members: Carmine Talarico, Jonathan Gonzalez, Edward Montes, Alberto Enrique Santalo Jr

Product Owner(s): Edward Montes

Mentor(s): Masoud Sadjadi

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2025 *Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

This document presents the complete design, implementation, and evaluation of the Hotel Management System, a web-based application developed for the CIS 4951 Capstone II course. The project addresses the need for a cost-effective, comprehensive Property Management System (PMS) for small to medium-sized hotels. Built on the open-source XAMPP stack (PHP, MySQL, Apache), the system provides a full suite of features including a guest-facing booking engine, multi-role staff management, a real-time administrative dashboard, housekeeping and financial modules, and a complete Night Audit process. This document details the project's user requirements, development plan, system architecture, validation through test cases, and includes a full user and maintenance manual in the appendix.

Table of Contents

Introduction.....	5
Current System.....	5
Purpose of New System	5
User Stories	6
Implemented User Stories	6
Pending User Stories	7
Project Plan	7
Hardware and Software Resources	7
Sprints Plan	8
System Design	9
Architectural Patterns.....	9
System and Subsystem Decomposition	9
Deployment Diagram.....	10
Design Patterns	10
System Validation.....	11
Glossary	13
Appendix.....	14
Appendix A - UML Diagrams	14
Appendix B - User Interface Design.....	15
Appendix C - Sprint Review Reports	16
Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents.....	19
References.....	26

INTRODUCTION

Current System

The problem addressed by this project is the operational challenge faced by small to medium-sized lodging businesses that lack access to an integrated, modern, and cost-effective Property Management System. Many existing solutions are either prohibitively expensive enterprise systems or a patchwork of disconnected tools. This leads to inefficiencies, a higher risk of data entry errors, and a disjointed experience for both staff and guests.

Purpose of New System

The purpose of the Hotel Management System is to provide a centralized, web-based solution that consolidates all core hotel operations into a single application. It is designed to provide the essential functionalities necessary for daily operations, inspired by enterprise-level platforms like Hilton's OnQ. The system supports guest-facing activities such as online booking and registration, as well as a full suite of back-office tools for staff to manage bookings, rooms, rates, housekeeping, financials, and system security.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

- As a Guest, I want to view available rooms and book a stay online so that I can make a reservation without having to call the hotel.
- As a Guest, I want to create an account and view my past and upcoming reservations so that I can keep track of my travel plans.
- As a Hotel Manager, I want to view a real-time dashboard with key metrics like occupancy and revenue so that I can quickly assess the hotel's daily performance.
- As a Hotel Manager, I want to manage staff accounts and roles so that I can control access to sensitive system features.
- As a Hotel Manager, I want to run a Night Audit process so that I can finalize the day's financial transactions and advance the business date.
- As a Front Desk Agent, I want to view today's arrivals and departures so that I can efficiently manage guest check-ins and check-outs.
- As a Front Desk Agent, I want to view a guest's folio and post new charges or payments so that I can keep their bill accurate.
- As a Housekeeper, I want to see a simple list of my assigned rooms for the day so that I can efficiently manage my cleaning schedule.
- As a Housekeeper, I want to mark a room as "Clean" so that the front desk knows it is ready for the next guest.
- As an Accountant, I want to generate revenue and occupancy reports for a specific date range so that I can perform financial analysis.

Pending User Stories

- As a Guest, I want to be able to cancel my own reservation online so that I don't have to call the front desk.
- As a Hotel Manager, I want to create special rate codes or discounts that can be applied during booking.
- As a Front Desk Agent, I want a more advanced search function to find guests by partial name, phone number, or confirmation number.

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all the hardware and software resources that were used in this project:

- Development Environment: XAMPP Server Suite
- Web Server: Apache
- Database: MySQL
- Backend Language: PHP
- Frontend Languages: HTML, CSS, JavaScript
- Version Control: GitHub
- JavaScript Libraries: Chart.js, FullCalendar
- PHP Libraries: PHPMailer

Sprints Plan

Sprint 1 (Week 1-2): Foundation & Core Models.

- Setup XAMPP environment and Git repository.
- Design and implement initial database schema.
- Implement basic user registration and login functionality.

Sprint 2 (Week 3-4): Core Booking & Admin Views.

- Develop the public-facing room booking engine.
- Create initial admin pages for viewing users, rooms, and bookings.
- Implement basic Role-Based Access Control in the navigation.

Sprint 3 (Week 5-6): Housekeeping & Financial Modules.

- Develop the housekeeping module for task assignment and status updates.
- Create the payments and reporting pages for the accountant role.
- Implement the room_rates table for dynamic pricing.

Sprint 4 (Week 7-8): Advanced Features & UI Polish.

- Design and implement the real-time Admin Dashboard with KPIs.
- Develop the visual "Upcoming Stays" tape chart.
- Implement the complete Night Audit process.

Sprint 5 (Week 9-10): CRM & Auditing.

- Develop the Guest CRM features, including the guest profile and preferences page.
- Implement the centralized audit logging system and the log viewer.
- Integrate PHPMailer for automated email confirmations and alerts.
- Conduct thorough end-to-end testing.
- Fix bugs and perform final code cleanup.
- Write the initial draft of all project documentation, including the User Manual.

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

The system is built using Multi-Tier Architecture, a standard pattern for web applications that separates concerns into logical layers. This promotes maintainability and scalability.

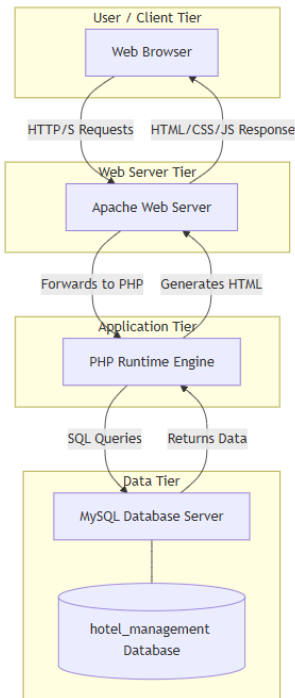
- **Presentation Tier (Client):** The user's web browser, which renders the HTML/CSS/JavaScript interface.
- **Application Tier (Server-Side Logic):** The Apache Web Server and the PHP Runtime Engine. This tier handles business logic, processes user input, and orchestrates data flow between the user and the database.
- **Data Tier (Database):** The MySQL Database Server, which is responsible for the persistent storage and retrieval of all application data.

System and Subsystem Decomposition

The application is decomposed into several logical subsystems:

- **User & Authentication Subsystem:** Manages user registration, login, sessions, and role-based permissions.
- **Booking & Reservation Subsystem:** Handles all guest-facing and staff-facing booking operations, including availability checks, reservation creation, and management.
- **Hotel Operations Subsystem:** Contains modules for managing the hotel's physical and financial assets, including Room Management, Rate Management, and Housekeeping.
- **Financial & Auditing Subsystem:** Manages all financial transactions, including Folio Management, Payments, Reporting, and the Night Audit process. It also includes the system-wide security audit log.

Deployment Diagram



Design Patterns

The project's implementation leverages several established software design patterns to ensure the code is organized, maintainable, and secure.

Template View: This pattern is used throughout the application to separate the business logic (PHP) from the presentation (HTML). The `header.php` and `footer.php` files act as templates. Core PHP scripts perform all necessary data fetching and logic first, and then they require the header and footer files to wrap the content in a consistent HTML structure. This avoids code duplication and makes it easy to update the site-wide layout from a single place.

Front Controller (API variation): The project uses a simplified version of the Front Controller pattern for handling specific actions. For example, the `submit_booking.php` file in the `/api/` directory acts as a single, secure endpoint for all booking form submissions. It doesn't contain the logic itself but is responsible for receiving the request and then including the actual processing script (`book_room.php`). This provides a clean separation of concerns and a single point of entry for that action, which is beneficial for security and logging.

SYSTEM VALIDATION

System validation was conducted based on a formal testing plan with defined success metrics and detailed test cases for all critical workflows.

Feature Success Metrics

- **User Authentication:** Success is when a registered user can log in and access resources appropriate to their role, while an unregistered or unauthorized user is denied access. 100% of login/logout actions must be recorded in the audit log.
- **Guest Booking Flow:** Success is when a guest can select dates, choose an available room, enter their details, and receive a booking confirmation. The new booking must be accurately reflected in the database and the admin's "Manage Bookings" view.
- **Housekeeping Flow:** Success is a 100% match between the status updated by a housekeeper and the status reflected in the manager's "Master List" view.
- **Night Audit:** Success is the transactional completion of all audit steps without error. The business date must advance by exactly one day, daily charges must be posted to all checked-in guest folios, and no-shows must be cancelled.
- **Financial Reporting:** Success is when the generated report for a given date range accurately calculates and displays Total Revenue and Occupancy Rate that matches a manual verification against the database.

Test Case 1: Successful Booking by New Guest

- Objective: Verify that a new, unregistered user can complete the booking process, which should automatically create a guest account.
- Steps:
 1. Open the website in a private/incognito browser window.
 2. Navigate to the "Book a Room" page.
 3. Select a valid check-in and check-out date.
 4. Click "Check Availability" and select a room.
 5. In the guest details form, enter a unique Full Name and Email.
 6. Click the "Complete Booking" button.
- Expected Result: The user is redirected to the "Booking Confirmed!" page. A new users entry (role guest) and a corresponding bookings entry are created in the database.

Test Case 2: Successful Night Audit Execution

- Objective: Verify that the Night Audit correctly processes daily transactions and advances the business date.
- Steps:
 1. Log in as an admin.
 2. Navigate to the "Night Audit" > "Run Audit" page.
 3. Click the "RUN NIGHT AUDIT" button and confirm.
- Expected Result: A success message is displayed. The business_date in the settings table is updated to 2025-07-23. The no-show booking is changed to cancelled. A new "Room & Tax" charge is added to the folio of the checked-in guest.

Test Case 3: End-to-End Housekeeping Workflow

- Objective: Verify the complete lifecycle of a cleaning task.
- Steps:
 1. Manager: Log in, navigate to "Housekeeping" > "Assign Tasks", and assign a dirty room to a housekeeper.
 2. Housekeeper: Log in, navigate to "Housekeeping" > "My Tasks", and click "Mark as Clean" for the assigned room.
 3. Manager: Log back in and navigate to "Housekeeping" > "Master List".
- Expected Result: After step 2, the housekeeping_tasks status becomes completed and the rooms status becomes clean. The manager's view in step 3 accurately reflects this "Clean & Ready" status.

GLOSSARY

ADR (Average Daily Rate): A KPI representing the average revenue earned per paid occupied room.

Folio: A guest's bill or account, containing a detailed list of all charges and payments made during their stay.

KPI (Key Performance Indicator): A measurable value that demonstrates how effectively the hotel is achieving key business objectives.

Night Audit: An end-of-day process that reconciles and finalizes a hotel's daily financial activities.

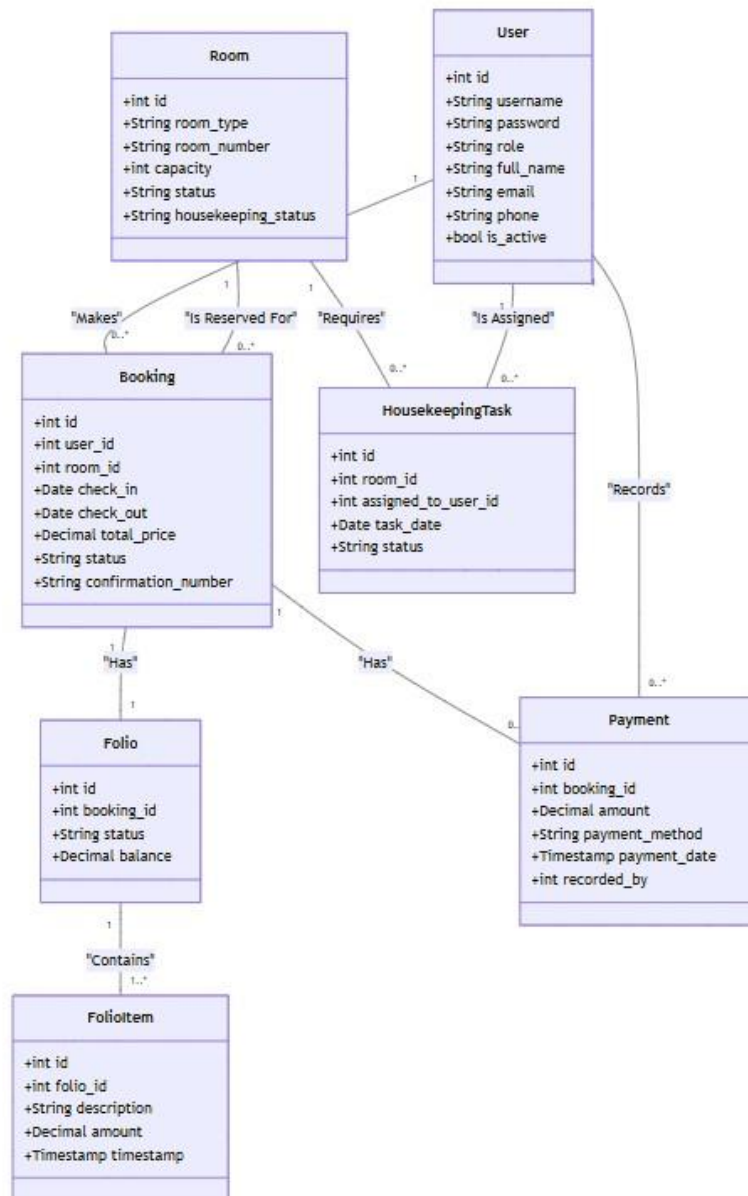
PMS (Property Management System): A software application for the operations of hospitality accommodations.

RBAC (Role-Based Access Control): A security model that restricts system access to authorized users based on their job function.

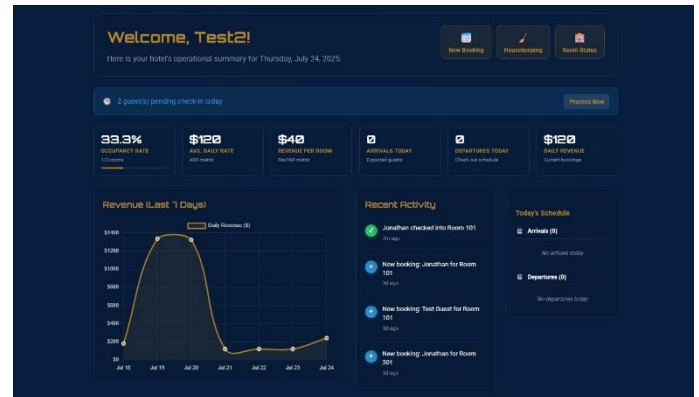
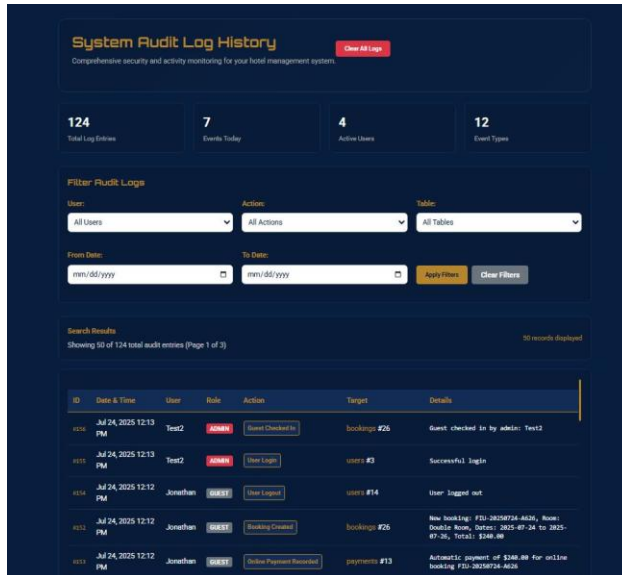
RevPAR (Revenue Per Available Room): A KPI representing the average revenue generated per available room, regardless of occupancy.

APPENDIX

Appendix A - UML Diagrams



Appendix B - User Interface Design



Appendix C - Sprint Review Reports

C.1 Sprint 1 Review Report (June 1, 2025)

Sprint: 1 (Foundation & Core Models)

Sprint Dates: May 19, 2025 – June 1, 2025

- **Sprint Goal:** To establish the project's foundational architecture by setting up the development environment, designing the initial database schema, and implementing basic user registration and secure login functionality.
- **Completed User Stories:** The implementation of the following foundational user stories was demonstrated and accepted:
 - As a Guest, I want to create an account and view my past and upcoming reservations so that I can keep track of my travel plans.
 - As a Hotel Manager, I want to manage staff accounts and roles so that I can control access to sensitive system features.
- **Backlog Adjustments:** All planned tasks for the sprint were completed. No user stories were moved back to the backlog.

C.2 Sprint 2 Review Report (June 15, 2025)

Sprint: 2 (Core Booking & Admin Views)

Sprint Dates: June 2, 2025 – June 15, 2025

- **Sprint Goal:** To develop the public-facing room booking engine, create the initial administrative pages for viewing system data, and implement basic Role-Based Access Control (RBAC).
- **Completed User Stories:** The implementation of the following user stories was demonstrated and accepted:
 - As a Guest, I want to view available rooms and book a stay online so that I can make a reservation without having to call the hotel.
 - As a Front Desk Agent, I want to view today's arrivals and departures so that I can efficiently manage guest check-ins and check-outs.
- **Backlog Adjustments:** All planned tasks for the sprint were completed. No user stories were moved back to the backlog.

C.3 Sprint 3 Review Report (June 29, 2025)

Sprint: 3 (Housekeeping & Financial Modules)

Sprint Dates: June 16, 2025 – June 29, 2025

- **Sprint Goal:** To build the core operational modules, including the housekeeping system for assigning and tracking cleaning tasks, and the financial pages for payments and reporting.
- **Completed User Stories:** The implementation of the following user stories was demonstrated and accepted:
 - As a Front Desk Agent, I want to view a guest's folio and post new charges or payments so that I can keep their bill accurate.
 - As a Housekeeper, I want to see a simple list of my assigned rooms for the day so that I can efficiently manage my cleaning schedule.
 - As a Housekeeper, I want to mark a room as "Clean" so that the front desk knows it is ready for the next guest.
 - As an Accountant, I want to generate revenue and occupancy reports for a specific date range so that I can perform financial analysis.
- **Backlog Adjustments:** All planned tasks for the sprint were completed. No user stories were moved back to the backlog.

C.4 Sprint 4 Review Report (July 13, 2025)

Sprint: 4 (Advanced Features & UI Polish)

Sprint Dates: June 30, 2025 – July 13, 2025

- **Sprint Goal:** To implement advanced, high-impact features including the real-time administrative dashboard, the visual "Upcoming Stays" tape chart, and the complete end-to-end Night Audit process.
- **Completed User Stories:** The implementation of the following user stories was demonstrated and accepted:
 - As a Hotel Manager, I want to run a Night Audit process so that I can finalize the day's financial transactions and advance the business date.
- **Backlog Adjustments:** All planned tasks for the sprint were completed. No user stories were moved back to the backlog.

C.5 Sprint 5 Review Report (July 27, 2025)

Sprint: 5 (CRM & Auditing)

Sprint Dates: July 14, 2025 – July 27, 2025

- **Sprint Goal:** To develop the Guest CRM features for managing profiles and preferences, implement the centralized security audit logging system, and integrate PHPMailer for automated email notifications.
- **Completed System Features:** The following system features and enhancements were demonstrated and accepted:
 - **Guest CRM:** A comprehensive guest profile page was developed, allowing staff to view booking history and manage guest preferences.
 - **Audit Logging:** A centralized logging system was implemented to record all significant user actions, with a searchable log viewer for administrators.
 - **Email Integration:** The PHPMailer library was integrated to handle automated booking confirmations and system alerts.
- **Backlog Adjustments:** All planned tasks for the sprint were completed. No user stories were moved back to the backlog.

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

1. Guest & Public-Facing Features

1.1. Creating an Account

New guests can create an account to manage their bookings.

- **Procedure:**

1. From any page, click the **"Register"** link in the main navigation.
2. Fill out the registration form with your Username, Password, Full Name, and Email. A phone number is optional.
3. Click the **"Register"** button.
4. Upon successful registration, you'll be automatically logged in and taken to the Welcome page. The system creates your account with a "guest" role.

1.2. Booking a Room

Guests can book a room through a multi-step process.

- **Procedure from the Rooms Page:**

1. Navigate to **"Hotel Info" > "Rooms & Suites"** to see the available room types and their standard nightly rates.
2. Click the **"Book Now"** button for your desired room. This will take you to the booking page with that room type pre-selected.
3. Select your desired Check-in and Check-out dates using the calendar. The calendar will automatically disable and block dates where the selected room type is fully booked.
4. Click **"Check Availability"**. The system will confirm availability and show you the total price for your stay.
5. Fill in your Full Name, Email, and optional Phone number. If you're logged in, this information will be auto-populated from your profile.
6. Click **"Complete Booking"** to finalize your reservation. A simulated online payment is automatically recorded.

- **General Booking Procedure:**

1. Navigate to **"Hotel Info" > "Book a Room"**.
 2. Select your Check-in and Check-out dates and click **"Check Availability"**.
 3. The system will display cards for all available room types for those dates, showing the total price for the stay.
 4. Click **"Select Room"** on your desired option.
 5. The guest details form will appear. Fill in your information and click **"Complete Booking"** to finalize.
- **Confirmation:** After booking, you're taken to a confirmation screen and an email is sent to both you and the hotel management.

1.3. Managing Your Reservations

Logged-in guests can view and manage their reservations.

- **Procedure:**

1. Log in to your account.
2. Click on the **"My Reservations"** link in the navigation.
3. The page displays your bookings, which you can filter between **"Current & Upcoming"** and **"Past"** reservations using the dropdown menu.
4. Each reservation card shows the room type, dates, duration, total price, and status.
5. Click **"View Details"** on any reservation to see a more comprehensive summary, including your confirmation number and an image of the room.

2. Staff Login & Welcome Portal

2.1. Staff Login

- **Procedure:**

1. Navigate to the **"Staff Login"** page.
2. Enter the Username and Password provided by the administrator.
3. Upon successful login, the system logs the event for security auditing and directs you to a welcome page. Failed login attempts are also logged.

2.2. Role-Based Welcome Page

After logging in, staff are presented with a welcome page tailored to their role. This page provides a brief overview of their permissions and quick links to their most relevant tools.

3. Administrative & Management Roles (Admin, Manager)

3.1. Admin Dashboard

The dashboard is the central hub for daily operations.

- **Key Performance Indicators (KPIs):**
 - **Rooms Occupied:** A live count of all guests currently checked-in. Click this card to open a modal with a detailed list of occupied rooms.
 - **Avg. Daily Rate (ADR):** The average revenue earned per paid occupied room.
 - **Revenue / Room (RevPAR):** The average revenue generated per available room, regardless of whether it's occupied.
 - **Arrivals Today:** The number of guests scheduled to check in today. Click this card to see a list of arriving guests.
- **Revenue Chart:** A line graph displaying the hotel's revenue over the last 7 days.
- **Action Lists:**
 - **Today's Arrivals:** A list of guests scheduled to arrive, with a direct link to their booking detail page for quick check-in.
 - **Today's Departures:** A list of guests scheduled to check out today.

3.2. User Management (Staff & Guests)

The "**Staff**" link in the navigation leads to the comprehensive user management page, which is split into two sections.

- **Staff Management:**
 1. **Create Staff:** Use the form at the top to create a new staff member. You must provide a username, password, and role. Name, email, and phone are optional.
 2. **Edit Staff:** In the "Active Staff Members" table, click "**Edit**" next to a user's name.
 3. **Deactivate Staff:** Click "**Deactivate**" next to a user's name. This performs a soft delete, preventing the user from logging in but preserving their records in the system.

- **Guest Relationship Management (CRM):**

1. **Search Guests:** Use the search bar in this section to find any guest by their name, email, or username.
2. **View Guest List:** The table displays all guest accounts with CRM data points like total bookings, total lifetime spending, and dates of first and last stays.
3. **View Guest Profile:** Click "**View Profile**" to open the detailed guest profile page. Here you can see their full booking history and manage their preferences, such as preferred room type, special requests, and dietary needs. All updates to preferences are logged in the audit trail.

3.3. Booking Management

- **Manage Bookings Table:**

- Navigate to "**Bookings**" > "**Manage Bookings**" to see a list of all bookings in the system.
- The table includes guest name, room number, dates, price, and status. It also shows the room's current housekeeping status.

- **Booking Details & Actions:**

- Click "**View Details**" on any booking to open its detail page.
- **Check-In:** For a "Confirmed" booking, click the "**Check-in Guest**" button. This updates the booking status to 'checked-in' and the room's housekeeping status to 'occupied'.
- **Check-Out:** For a "Checked-in" booking, click "**Check-out Guest**". This updates the booking status to 'checked-out', sets the housekeeping status to 'dirty', and closes the guest's folio.
- **Edit Booking:** Click "**Edit Booking Dates**" to change the assigned room, check-in/out dates, or manually change the status (e.g., to "Cancelled").

- **Upcoming Stays (Room Plan):**

- Navigate to "**Bookings**" > "**Upcoming Stays**" to view the visual room plan (tape chart).
- This grid shows all rooms and their occupancy status over a 7-day period. Bookings are color-coded by status (Checked-in vs. Confirmed) and vacant cells are colored by housekeeping status (Clean vs. Dirty).
- You can drag-and-drop a booking to a different vacant cell to change its room and/or start date.

3.4. Room Management

- **Procedure:**

1. Navigate to **"Staff"** in the main navigation and select the appropriate link for managing rooms.
2. The table lists all rooms with their number, type, capacity, and status ('available' or 'maintenance').
3. Click **"Manage"** to open the room detail page. Here you can edit the room's properties.
4. The detail page also includes a booking calendar for that specific room, showing all its past and future reservations.

3.5. Rate Management

- **Procedure:**

1. Navigate to **"Staff" > "Rate Management"**.
2. Use the form to create new pricing rules. Each rule requires a name, room type, price per night, and a start/end date range.
3. The table on the right lists all existing rate rules. Click **"Edit"** to populate the form with a rule's details for updating, or **"Delete"** to remove it.

3.6. Housekeeping Management

- **Master List:**

- Navigate to **"Housekeeping" > "Master List"**.
- This page displays a comprehensive list of all rooms. Admin/Manager can filter by status: All, Occupied, Needs Cleaning, Clean & Ready, or Maintenance.
- The status of any non-occupied room can be manually changed from this screen.

- **Assign Tasks:**

- Navigate to **"Housekeeping" > "Assign Tasks"**.
- The left panel shows all unassigned 'dirty' rooms. Select a housekeeper, check the rooms you want to assign, and click **"Assign Selected Rooms"**.
- The right panel shows all cleaning tasks currently pending for the day and who they are assigned to.

- **Daily Tasks:**

- The **"Daily Tasks"** page shows a view of all pending tasks grouped by the assigned housekeeper, allowing for easy progress monitoring.

3.7. Financial Management

- **Payments:**
 - Navigate to "**Accounting**" > "**Payments**".
 - To record a payment, enter the Booking ID and click "**Fetch Details**" to auto-populate guest name and amount. Fill in the payment method and click "**Record Payment**".
 - A history of all transactions is displayed in the table below.
- **Reports:**
 - Navigate to "**Accounting**" > "**Reports**".
 - Select a start and end date and click "**Generate Report**".
 - The system will display Total Revenue, Total Bookings, and Occupancy Rate for the period, along with breakdowns of bookings by status and revenue by room type.

3.8. Night Audit

The Night Audit is a critical, irreversible process that finalizes the day's business.

- **Procedure:**
 1. Navigate to "**Night Audit**" > "**Run Audit**".
 2. Review the **Pre-Audit Checklist**. This shows pending departures and potential no-shows that should be resolved before running the audit.
 3. Once all daily tasks are reconciled, click the "**RUN NIGHT AUDIT**" button and confirm the prompt.
 4. The audit will automatically:
 - Change the status of any no-show bookings to "cancelled".
 - Post the daily room and tax charges to the folios of all checked-in guests.
 - Advance the hotel's official business date by one day.
 - Log the completion of the audit in the system log.

3.9. System Monitoring

- **Procedure (Admin Only):**
 1. Navigate to "**Night Audit**" > "**Audit Logs**".
 2. This page provides a complete, searchable log of every significant action taken in the system.
 3. Use the filters at the top to search for events by User, Action Type, affected

Database Table, or a specific Date Range.

4. For extreme cases, the **"Clear All Logs"** button will permanently delete all log history. This action is protected by a confirmation modal and should be used with extreme caution.

4. Specialized Staff Roles

4.1. Front Desk

Front Desk staff have access to a subset of management tools focused on guest interactions.

- **Permissions:** Access to the Admin Dashboard, Bookings (View/Check-in/Check-out), and Upcoming Stays (Room Plan).
- **Primary Duties:**
 - Monitor arrivals and departures via the **Dashboard**.
 - Check guests in and out via the **Manage Bookings** detail page.
 - View room availability and occupancy using the **Upcoming Stays** grid.

4.2. Accountant

The accountant role is focused solely on financial data.

- **Permissions:** Access to the Accounting module (Payments and Reports).
- **Primary Duties:**
 - Record and review all guest payments in the **Payments** section.
 - Generate financial reports on revenue and occupancy in the **Reports** section.

4.3. Housekeeping Staff

Housekeeping staff have a simplified interface to manage their daily tasks.

- **Permissions:** Access to the Housekeeping module (Master List and My Tasks).
- **Primary Duties:**
 1. Log in and navigate to **"Housekeeping" > "My Tasks"** to see their assigned rooms for the day.
 2. After cleaning a room, find it on their list and click the **"Mark as Clean"** button.

3. This action automatically updates the room's status in the system and marks the task as complete.

5. Automated System Processes

The system includes automated scripts designed to run on a daily schedule.

5.1. Automated Email Reminders

- The system automatically sends friendly reminder emails to guests with upcoming confirmed bookings. Reminders are sent out 3 days before check-in and again 1 day before check-in.

5.2. Low Inventory Alerts

- The system automatically checks the number of available rooms for each room type daily. If the number of available rooms falls below a preset threshold, an alert email is sent to the hotel administrator.

REFERENCES

Inspiration: The system's design and operational workflow were inspired by industry-leading Property Management Systems, specifically Hilton's OnQ platform.

The Apache Software Foundation. (2025). *Apache HTTP Server version 2.4 documentation*. Retrieved July 24, 2025, from <https://httpd.apache.org/docs/2.4/>

The PHP Group. (2025). *PHP 8 manual*. Retrieved July 24, 2025, from <https://www.php.net/manual/en/>

Oracle Corporation. (2025). *MySQL 8.0 reference manual*. Retrieved July 24, 2025, from <https://dev.mysql.com/doc/refman/8.0/en/>