Business Requirements Document

Hotel Room Booking Management System

Authors: Mark Christian L. Balagot

Ivhan Patrick D. De Leon

Version: 0.1

Date: March 29, 2023

Table of Contents

[1](#_gjdgxs) Executive Summary 3

[2](#_30j0zll) Project Description 3

[3](#_1fob9te) Project Scope 3

[3.1](#_3znysh7) In Scope 3

[3.2](#_2et92p0) Out of Scope 4

[4](#_tyjcwt) Current Process 4

[5](#_3dy6vkm) Functional Requirements 4

[5.1](#_1t3h5sf) Priority 4

[6](#_4d34og8) Glossary 6

[7](#_2s8eyo1) References 6

[8](#_17dp8vu) Appendix 8

[9](#_3rdcrjn) Document History 9

# Executive Summary

The Hotel Room Booking Management System (HRBMS) capstone project aims to develop a modern and user-friendly web application that streamlines the hotel room booking process for customers and improves operational efficiency for hotel staff. The system will provide a platform for customers to search for available rooms while allowing hotel staff to make reservations, manage guest bookings, manage room inventory, view booking details, and process payments. The project will involve developing the frontend user interface, backend server logic, and database schema, as well as integrating with third-party tools and payment gateways.

Overall, the HRBMS capstone project aims to create a scalable, reliable, and secure platform for managing hotel room bookings and inventory, while providing an intuitive and efficient user experience for the hotel staff.

# Project Description

Functional Requirements:

* The system will allow users to book hotels by selecting check-in and check-out dates, room type, and other options.
* The system will provide users with real-time information on hotel availability and pricing.
* The system will allow users to book, modify and cancel their reservations.

Non-functional Requirements:

* The system will be designed to handle a large number of concurrent users.
* The system will be secured with Hashing algorithm encryption to protect user data. (Subject to change)
* The system will be designed to be scalable and flexible to accommodate future enhancements and changes.

# Project Scope

The HRBMS will be a web-based application that allows customers to search for available rooms while allowing hotel staff to make reservations, manage guest bookings, manage room inventory, view booking details. customers to make room bookings, view availability, and manage their reservations. The system will also provide hotel staff or admin with a centralized platform to manage bookings, room inventory, and customer data. The project will involve developing the frontend user interface, backend server logic, and database schema.

## In Scope

The following areas are in the scope for this project:

Web Services Rest API’s (Staff Side)

* Login/Logout
  + Profile Credentials
  + Identity JWT enabled
* List of Rooms
* List of Categories
* List of Invoices
* List of Bookings
* List of Users

Web Application

* Login/Logout
  + Profile
  + Identity JWT enabled
  + Change password

Admin Dashboard

* Login/Logout
* List of Booked Rooms
  + Manipulation of Room Availability (CRUD)
* List of User Credentials
  + Manipulation of User Credentials (CRUD)
    - Guests Credentials
    - Staff Credentials
* List of Room Categories
  + Manipulation of Room Details (CRUD)
* Booking Application
  + Ability to Book Rooms for Users (CRUD)

Staff Dashboard

* Login/Logout
* List of Booked Rooms
  + Manipulation of Room Availability (CRUD)
* List of User Credentials
  + Manipulation of User Credentials (CRUD)
    - Guests Credentials
* Booking Application
  + Ability to Book Rooms for Users (CRUD)

Guest Dashboard

* Login/Logout
* Booking Application
  + Ability to Book Rooms(CRUD)
* Invoice

## Out of Scope

The following areas are out of scope for this project:

* Integrating with external third-party tools for managing the remainders like, google calendar or outlook calendar, etc.
* Integrating with external third-party tools for managing payments like Gcash, Maya Wallet, Paypal, etc.
* Allows guest user to book and cancel rooms with certain conditions

# Current Process

The hotel is experiencing rapid growth in the number of customers seeking accommodation services. The existing manual system of booking hotel rooms is becoming inefficient, cumbersome, and time-consuming, leading to a lot of errors, double bookings, and customer dissatisfaction. This calls for the need to develop a web-based hotel room booking application that provides a reliable and user-friendly platform for customers to search, book, and manage their hotel reservations easily and efficiently.

The proposed solution should address the following key challenges:

* Provide a user-friendly interface that enables customers to search and book rooms according to their preferences, such as room type, price range, and availability.
* Allow guests to view real-time room availability, prices, and amenities.
* Enable guests to view their booking history, modify or cancel their reservations with the help of staff, and receive notifications and alerts related to their bookings.
* Provide hotel staff with a user-friendly dashboard that allows them to manage or modify bookings, reservations,generate invoice and view guest data in real time.
* Provide administrators with a user-friendly dashboard that allows them to manage room availability, rates, reservations,manage bookings and manage guest and staff data in real time.

The proposed hotel room booking application should aim to improve the overall user experience, reduce the workload of hotel staff, and increase the revenue and profitability of the hotel by maximizing room occupancy and reducing booking errors and cancellations.

# Functional Requirements

Guest Portal

The Guest portal should have the following features:

* An easy-to-use interface for searching and booking available rooms
* The option to view and modify bookings, including cancellations and amendments
* Secure payment processing using industry-standard encryption and authentication protocols
* Integration with popular payment gateways such as PayPal, Stripe, and Authorize.net

Hotel Staff Portal

The hotel staff portal should have the following features:

* A dashboard showing the current room inventory, and revenue
* The ability to add, modify, and delete room types, rates, and availability
* The option to view and manage customer bookings, including cancellations and amendments.

Admin Portal

The admin portal should have the following features:

* User management, including the ability to add, modify, and delete user accounts with different levels of access
* System configuration, including the ability to set up room types, rates, and availability rules
* Payment gateway integration, including the ability to configure and manage payment gateway accounts
* Reporting functionality to track system usage, user activity, and other key metrics

# Non-Functional Requirements

Performance

The HRBMS should be able to handle a large number of concurrent users and provide a responsive user experience. The system should be able to process bookings and payments quickly and efficiently.

The system will be secured with Hashing algorithm encryption to protect user data. (Subject to change)

The system will be designed to be scalable and flexible to accommodate future enhancements and changes.

## Priority

The requirements in this document are divided into the following categories:

Admin Portal

| **Item** | **Rating** | **Description** |
| --- | --- | --- |
| Login/Logout | High | Admin will be able to log in with his provisioned account details. |
| Dashboard for Admin | High | Dashboard screens lists User Data, Staff Data, and Room Data |
| List Users | High | List of Users which can be used for managing the CRUD operations. |
| List Staff | High | List of Staff which can be used for managing the CRUD operations. |
| List Booked Rooms | High | List of Booked Rooms which can be used for managing the CRUD operations |

Hotel Staff Portal

| **Item** | **Rating** | **Description** |
| --- | --- | --- |
| Login/Logout | High | Hotel Staff will be able to log in with their provisioned account details. |
| Dashboard for Staff | High | Dashboard screens lists User Data and Room Data |
| List Users | High | List of Users which can be used for managing the CRUD operations |
| List Booked Rooms | High | List of Booked Rooms which can be used for managing the CRUD operations |

User Web Portal

| **Item** | **Rating** | **Description** |
| --- | --- | --- |
| Login/Logout/Register | High | Users will be able to register for an account as well as log in with their registered login details |
| Dashboard for Users | High | Dashboard screens with Room Details, Calendar and user input boxes |
| List Booked Rooms | High | List of Booked Rooms for canceling booked rooms and viewing |

Non-Functional Requirements

| **Item** | Rating | Description |
| --- | --- | --- |
| External 3rd party Application API | Low | Allow Hotel Web Application to expose API’s specific to 3rd party tools to consume and user data, room data and staff data with full security |

# Glossary

*This section explains all of the terms and abbreviations that were used in this document, for those who are unfamiliar with them. Not everybody who reads this document will understand all of the terms, so this section is helpful.*

| **Term** | **Explanation** |
| --- | --- |
| CRUD | CRUD stands for Create, Read, Update, and Delete, which are the four basic functions used in persistent storage. CRUD operations are commonly used in database applications, but they can also be applied to file systems, spreadsheets, and other types of data storage systems.  Create: This operation involves creating a new record or entry in the database or data storage system.  Read: This operation involves retrieving or reading data from the database or data storage system. It is also known as the "Retrieve" or "Select" operation.  Update: This operation involves modifying or updating an existing record or entry in the database or data storage system.  Delete: This operation involves deleting or removing a record or entry from the database or data storage system.  CRUD operations are fundamental building blocks for database and software development and are used to manage data effectively and efficiently. |
| Pagination | refers to the process of dividing large amounts of content or data into separate pages or sections. This is commonly seen on websites where the content is too lengthy to be displayed on a single page, and users must navigate through multiple pages to access all of the content. |
| Schema | refers to the overall organization or blueprint for a database, including the tables, fields, relationships, and constraints. |
| API | API stands for "Application Programming Interface." It is a set of protocols, routines, and tools for building software applications. In simple terms, an API defines how different software components should interact with each other.  An API acts as an intermediary between different software programs, allowing them to communicate with each other and share data. APIs can be used to access services provided by other software applications, such as retrieving data from a database, performing calculations, or sending and receiving messages. |

# Document History

*This section details the history of the document at each version. It’s good to know what has changed in each version, by who, and when it happened.*

| **Version** | **Date** | **Changes** | **Author** |
| --- | --- | --- | --- |
| 0.1 |  |  |  |
| 0.3 |  |  |  |
| 0.4 |  |  |  |
| 0.5 |  |  |  |
|  |  |  |  |