Software Requirements Specification

for

GET HOTEL SYSTEM

Version 1.0

Prepared by H21\_GREEN

RIVERFRONT SENTRAL HOTEL

7 March 2024

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Features 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. System Features 3

3.1 System Feature 1 3

3.2 System Feature 2 (and so on) 4

4. External Interface Requirements 4

4.1 User Interfaces 4

4.2 Hardware Interfaces 4

4.3 Software Interfaces 4

4.4 Communications Interfaces 4

5. Other Nonfunctional Requirements 5

5.1 Performance Requirements 5

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 6

Appendix C: Issues List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason for Changes** | **Version** |
| 1. Anaz Azzamuddin 2. Tan Jin Wan 3. Luqman Hazim 4. Nabil Aiman | 19 July 2020 | Initial release | 1.0 |
|  | 26 July 2020 | Formatting Correction | 1.1 |
|  |  |  |  |

# Introduction

## Purpose

The primary goal is to provide customers with a user-friendly platform for browsing, selecting, and booking hotel accommodations efficiently. This system will offer various features such as real-time availability checks, multiple room options and flexible booking modifications. Additionally, it will empower hotel administrators with tools for managing room inventory, processing bookings, generating reports, and maintaining customer records. By automating and optimizing these processes, the "Get Hotel System" will improve operational efficiency, increase revenue opportunities, and enhance overall customer satisfaction.

## Document Conventions

In the context of Get Hotel System Software Requirements Specification (SRS), the following document conventions have been established to ensure clarity and consistency throughout the document.

1. Data Customer:

Priority : High

Reason : Customer’s information are crucial as it involves names, IDs, phone number and etc to make sure the booking process valid. An accurate customer data helps in providing personalized experience and customer satisfaction as it was the primary goal for the Get Hotel System.

1. Data Booking :

Priority : High

Reason : Hence its name, Get Hotel System prioritized bookings information as the system provides a booking service. Aside from customer, the total of rooms booked are the main vision for this system to operate as it stores the monthly income of each room sold and type of rooms that customers preferred. Overtime, the booking data helps client in improving more on managing rooms and services.

1. Data Administration :

Priority : High

Reason : Both staff and admin are one of our priorities as they’ll responsible for managing customers and booking data. In addition, this side of administration in the system focused on high-level security as it blocks any interference form any third parties that can risk the integrity of the data. This system are completed with a robust set of database that helps administration to manage customers personal information securely.

## Intended Audience and Reading Suggestions

This document is intended for a diverse audience, including project managers, developers, designers, testers, business analysts, and stakeholders involved in the development and deployment of the "Get Hotel System." It serves as a primary reference point to ensure alignment with project goals, requirements, and expectations throughout the software development lifecycle.

## Project Scope

The scope of the "Get Hotel System" encompasses a comprehensive range of functionalities and features tailored to meet the needs of both customers and hotel administrators. This includes but is not limited to, user authentication, room search and filtering, booking management, payment processing, real-time availability checks, user profiles, and administrative dashboards.

# Overall Description

## Product Perspective

The Get Hotel System is a standalone software application that operates independently with external systems such as payment gateways. It will be developed using modern programming languages and frameworks, ensuring scalability, flexibility, and maintainability. The system will interact with databases for storing hotel and guest information.

## Product Features

Several major features and significant functions, which can be grouped into the following categories:

* User authentication and authorization
* Reservation management
* Room management
* Guest management
* Billing and invoicing
* Reporting
* Integration with external systems

User need to perform to complete their booking and it related with admin side because admin have to handle the data of the bookings.

## User Classes and Characteristics

Admin

* Manager
* Staff

Users

Customer

* Tourists
* Locals

**ADMINISTATORS**

With highly technical expertise, this role is responsible for daily system management and security, holding privileges that include access to system configuration and user management, and requiring tools for system monitoring, security features, and user management capabilities.

**Managers:**

Characteristics:

May vary in technical expertise, educational level, and experience. Manage their hotel's online presence and orders.

Frequency of Use:

Can range from daily to frequent, depending on the managerial responsibilities and tasks.

Privileges:

Customers management, booking acceptance, and performance analytics.

Requirements:

Easy-to-use hotel dashboard, options for customization based on the specific needs of the business and integration options with their POS systems.

**Staff:**

Characteristics:

Use the system to fulfill their roles in handling customers information, booking fulfillment, customer service, and other operational tasks.

Frequency of Use:

Daily usage as part of their routine tasks, with varying intensity during peak hours.

Privileges:

Access to features related to their specific roles, such as order processing, inventory management, and communication tools.

Requirements:

Easy-to-use hotel dashboard, options for customization based on the specific needs of the business and integration options with their POS systems.

**CUSTOMERS**

Users are focused on tourists, exhibit from various countries, utilizing the system to book a hotel room with a frequency that can range from daily to infrequent, they enjoy privileges like booking placement, customization, and tracking, and their requirements include a user-friendly interface, a diverse range of room choices, booking history, and payment options.

**Tourists:**

Characteristics:

Diverse taste in choosing hotel rooms for a maximum comfortability.

Frequency of Use:

Occasional to frequent, depending on factors such as hotspot for tourists and suggestions on viral places.

Privileges:

Booking placement, customization of rooms, and customer services.

Requirements:

User-friendly interface for quick and efficient booking processing.

**Locals:**

Characteristics:

Search for comfortable room at the reasonable price.

Frequency of Use:

Occasional to frequent, depending on factors such as holidays and suggestions on viral places.

Privileges:

Booking placement, customization of rooms, and customer services.

Requirements:

User-friendly interface for quick and efficient booking processing.

## Operating Environment

The Get Hotel System will operate in a typical hotel environment, which may include desktop computers, tablets, and mobile devices for accessing the system. It will require a stable internet connection for cloud-based functionalities and may run on popular web browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge. The system may be deployed on-premises or hosted on a cloud platform, depending on the hotel's preference.

## Design and Implementation Constraints

Constraints for the Get Hotel System may include compliance with data privacy regulations such as GDPR or CCPA, integration compatibility with existing hardware and software systems used by the hotel, availability of reliable internet connectivity for cloud-based functionalities, and scalability to accommodate future expansion or changes in hotel operations. Additionally, the system should be designed with security best practices to prevent unauthorized access and protect sensitive information.

## User Documentation

The Get Hotel System will provide comprehensive user documentation, including user manuals, tutorials, and troubleshooting guides, to assist hotel staff in using the system effectively. The documentation will cover topics such as system setup, user account management, reservation handling, room allocation, billing processes, report generation, and system maintenance. It will be available in digital format and may include demonstrations for visual learners.

## Assumptions and Dependencies

Assumptions and dependencies for the Get Hotel System may include availability of adequate training and support for hotel staff during system implementation, cooperation from third-party vendors for integration with external systems, availability of hardware infrastructure capable of running the software effectively, and adherence to industry standards and best practices for hotel management software development. Additionally, the system may depend on external services such as payment gateways and online booking platforms for certain functionalities.

# System Features

## Admin Features

### As an admin, I want to login so that I can access the system.

|  |  |
| --- | --- |
| **System Name** | **Get Hotel System** |
| **Team Id** | **H21\_Green** |
| Number # | 1 |
| Title | Login System |
| Description |  |
| As | An admin |
| I want to | Login |
| So that | I can access the system |
| Priority | High |
| Iteration / Sprint | 1 |
| Status  (Planning / In Progress / Complete) | Planning |
| **Acceptance Criteria / Confirmation** | |
| 1. Enter username 2. Enter password | |
| **Task** | |
| 1. Analyze login details 2. Design UI 3. Coding | |

### As an admin, I want to access the booking list system so that I can see the latest booking history.

|  |  |
| --- | --- |
| **System Name** | **Get Hotel System** |
| **Team Id** | **H21\_Green** |
| Number # | 1 |
| Title | Booking List System |
| Description |  |
| As | An admin |
| I want to | Access the booking list |
| So that | I can see the latest booking history |
| Priority | High |
| Iteration / Sprint | 1 |
| Status  (Planning / In Progress / Complete) | Planning |
| **Acceptance Criteria / Confirmation** | |
| 1. Enter the booking list  2. Add booking order  3. Delete booking order  4. Clear booking order | |
| **Task** | |
| 1. Analyze order list details  2. Design UI  3. Coding | |

## Customer Feature

### As a customer, I want to login so that I can access the system

|  |  |
| --- | --- |
| **System Name** | **Get Hotel System** |
| **Team Id** | **H21\_GREEN** |
| Number # | 1 |
| Title | Login System |
| Description |  |
| As | A customer |
| I want to | Login |
| So that | I can access the system |
| Priority | High |
| Iteration / Sprint | 1 |
| Status  (Planning / In Progress / Complete) | Complete |
| **Acceptance Criteria / Confirmation** | |
| 1. Enter username 2. Enter password | |
| **Task** | |
| 1. Analyze login details 2. Design UI 3. Coding | |

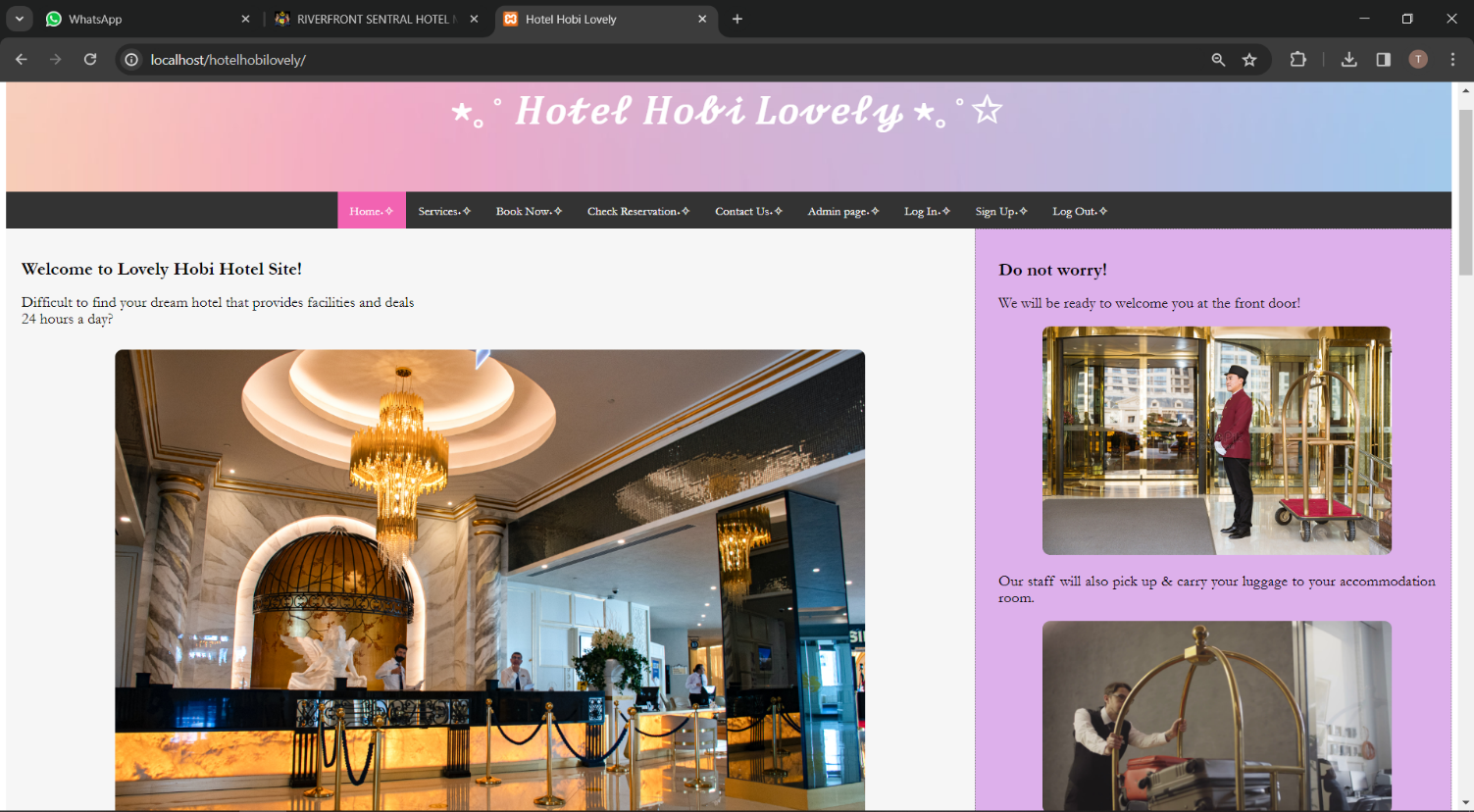
### As a new customer, I want to sign up so that I can access the system

|  |  |
| --- | --- |
| **System Name** | **Get Hotel System** |
| **Team Id** | **H21\_GREEN** |
| Number # | 1 |
| Title | Sign Up |
| Description |  |
| As | A customer |
| I want to | Sign up |
| So that | I can access enter the website |
| Priority | High |
| Iteration / Sprint | 1 |
| Status  (Planning / In Progress / Complete) | Complete |
| **Acceptance Criteria / Confirmation** | |
| 1. Enter username  2. Enter password  3. Enter confirm password | |
| **Task** | |
| 1. Analyze sign up details  2. Design UI  3. Coding | |

# External Interface Requirements

## User Interfaces

### Home page



A screenshot of a computer

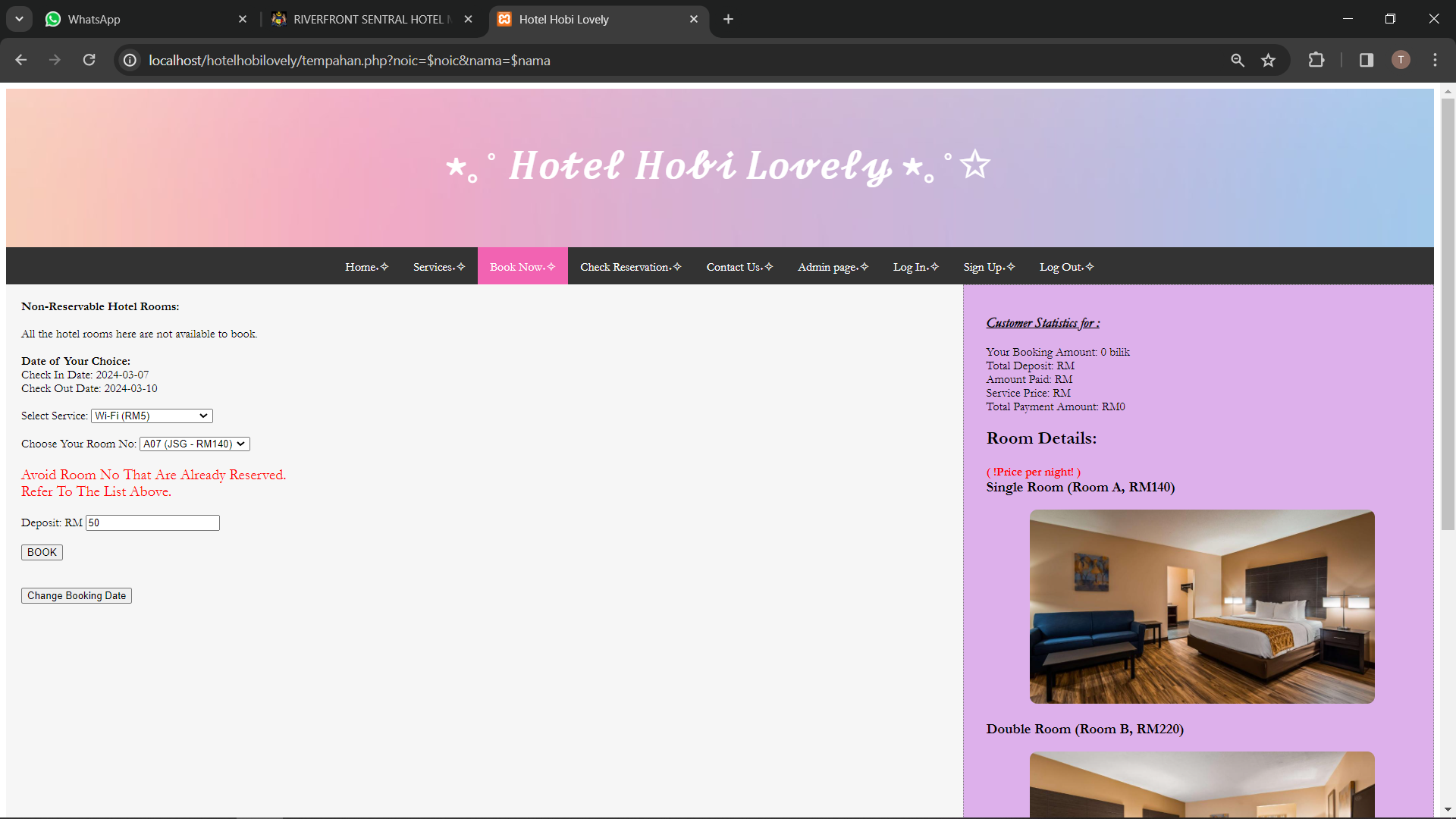
Description automatically generated

### Hotel Service Page



A screenshot of a computer

Description automatically generated



### User Registration page

### User Log In page

## Hardware Interfaces

User can use the system in PC, laptop, and tablet.

## Software Interfaces

Name: Get Hotel System

Version: 1.0

Database: Local Host (PHPMYADMIN)

Operating System: Windows

Tools: HTML, CSS, JavaScript, and PHP

Libraries: Development (XAMPP and Notepad++)

## Communications Interfaces

In the system, we have link between our system with bank website and credit card. Our system using

HTTP.

# Other Nonfunctional Requirements

## Performance Requirements

* Response Time: The system shall respond to user requests within 2 seconds under normal load conditions.
* Concurrency: The system shall support a minimum of 100 concurrent users without any noticeable degradation in performance.
* Scalability: The system shall be scalable to accommodate an increase in user load. It should handle a 50% increase in user traffic without any significant performance issues.

## Safety Requirements

* Data Integrity: The system shall ensure the integrity of user data through appropriate encryption methods during transmission and storage.
* Error Handling: The system should handle errors gracefully to prevent any data loss or corruption. Error messages should be clear and user-friendly.

## Security Requirements

* Authentication: The system shall enforce user authentication using secure methods such as username/password or multi-factor authentication.
* Authorization: Access to sensitive information shall be restricted based on user roles and permissions.
* Data Protection: The system shall implement measures to protect user data from unauthorized access, modification, or deletion. This includes encryption of sensitive data and regular backups.
* Secure Communication: All communication between the client and server shall be encrypted using industry-standard protocols such as HTTPS.

## Software Quality Attributes

* Reliability: The system shall be reliable, with an uptime of at least 99.9%. It should minimize the occurrence of system failures and errors.
* Maintainability: The system shall be designed and implemented in a modular and well-documented manner to facilitate ease of maintenance and future enhancements.
* Usability: The system shall be user-friendly, with intuitive interfaces and clear navigation paths. It should require minimal training for users to effectively use the system.
* Performance Efficiency: The system shall optimize resource utilization to ensure efficient performance and minimize response times.
* Portability: The system shall be designed to be platform-independent, allowing it to run on various operating systems and environments without modification.