

# **HMS Software Requirements Specifications**



**Session: 2021 – 2025**

## **Submitted by:**

Saleem Malik(2021-CS-32)

Mustafa Riaz(2021-CS-39)

Abdur Rehman(2021-CS-44)

## **Supervised by:**

Sir: Atif Warraich

Department of Computer Science

**University of Engineering and Technology**  
**Lahore Pakistan**

## Table of Contents

### Contents

<b>1 Introduction.....</b>	<b>3</b>
1.1 Purpose .....	3
1.2 Scope.....	3
1.3 Vision .....	3
1.4 Overview .....	4
<b>2 The Overall Description .....</b>	<b>4</b>
2.1 Software Interfaces.....	4
2.2 Product Functions.....	4
<b>3 Functional Requirements .....</b>	<b>5</b>
3.1 Reservation/Booking Rooms .....	5
3.2 Food.....	5
3.3 Management.....	5
<b>4 Nonfunctional Requirements .....</b>	<b>6</b>
4.1 Performance Requirements.....	6
4.2 Logical Database Requirements.....	6
4.3 Booking/Reservation System .....	6
4.4 Food Services.....	6
4.5 Design Constraints .....	6
4.6 Availability.....	6
4.7 Security.....	7
4.8 Maintainability.....	7

# **1 Introduction**

The following subsections of the Software Requirements Specifications (SRS) document provide an overview of the entire SRS.

## **1.1 Purpose**

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Hotel Management System (HMS). This SRS will allow for a complete understanding of what is to be expected of the HMS to be constructed. The clear understanding of the HMS and its' functionality will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This SRS will provide the foundation for the project. From this SRS, the HMS can be designed, constructed, and finally tested.

This SRS will be used by the software engineers constructing the HMS and the hotel end users. The software engineers will use the SRS to fully understand the expectations of this HMS to construct the appropriate software. The hotel end users will be able to use this SRS as a "test" to see if the software engineers will be constructing the system to their expectations. If it is not to their expectations the end users can specify how it is not to their liking and the software engineers will change the SRS to fit the end users' needs.

## **1.2 Scope**

The software product to be produced is a Hotel Management System which will automate the major hotel operations. The first subsystem is a Reservation and Booking System to keep track of reservations and room availability. The second subsystem is the Tracking and Selling Food System that charges the current room. The third subsystem is a General Management Services and Automated Tasks System which generates reports to audit all hotel operations and allows modification of subsystem information. These three subsystem's functionalities will be described in detail in section 2-Overall Description.

There are two end users for the HMS. The end users are the hotel staff (customer service representative) and hotel managers. Both user types can access the Reservation and Booking System and the Food Tracking and Selling System. The General Management System will be restricted to management users.

## **1.3 Vision**

The vision of the Hotel Management System (HMS) project is to create a comprehensive and user-friendly software solution that automates major hotel operations. The system aims to enhance efficiency and accuracy in managing reservations, room occupancy, and food services. The project's purpose is outlined in the Software Requirements Specification (SRS), providing a detailed description of the HMS requirements.

The scope includes three main subsystems: Reservation and Booking System, Tracking and Selling Food System, and General Management Services and Automated Tasks System.

The end users, hotel staff, and managers will benefit from functionalities such as reservation recording, room availability tracking, and detailed reporting. The Overall Description section outlines the software interfaces, emphasizing the use of MySQL for databases.

## **1.4 Overview**

The SRS is organized into two main sections. The first is The Overall Description and the second is the Specific Requirements. The Overall Description will describe the requirements of the HMS from a general high level perspective. The Specific Requirements section will describe in detail the requirements of the system

## **2 The Overall Description**

Describes the general factors that affect the product and its requirements. This section does not state specific requirements. Instead, it provides a background for those requirements, which are defined in section 3, and makes them easier to understand.

### **2.1 Software Interfaces**

All databases for the HMS will be configured using MySQL. These databases include hotel rooms and customers information. These can be modified by the end users. The room database will include the room numbers and if they are vacant or occupied. The customers information database will contain all the information of the customer such as first name, last name, number of occupants, assigned room, default room Rate(may be changed), phone number, whether or not the room is guaranteed, credit card number, confirmation number, automatic cancellation date, expected check in date and time.

### **2.2 Product Functions**

Reservation and Booking System

- Allows for typing in customer information
- Has a default room rate that is adjustable
- When a customer checks in, the room number will be changed to occupied in the database
- Ability to modify a reservation
- When no rooms are available and a customer would like to extend their reservation, their information will be placed in a database and when there are rooms available the first customer on the list will have the room
- When a customer checks out the amount owed is displayed
- Records that room is vacant
- Records payment
- Allows for space to write customer's feedback
- Reports generated to audit hotel occupancy, future occupancy, room revenue, and food revenue
- Exception reports listing exceptions to the normal cost
- Allows addition, deletion and modification of information on rooms and rates, menu items and prices, user profiles
- Creation of users and assigning passwords.

### **3 Functional Requirements**

Functional requirements define the fundamental actions that system must Perform. And We Divided it in three main categories Reservation/Booking, Food, and Management.

#### **3.1 Reservation/Booking Rooms**

- 1.1. The system shall record reservations.
- 1.2. The system shall record the customer's first name and last Name.
- 1.3. The system shall record the number of occupants in one Room.
- 1.4. The system shall record the room number.
- 1.5. The system shall record the customer's phone number.
- 1.6. The system shall display whether or not the room is guaranteed.
- 1.7. The system shall generate a unique confirmation number for each reservation.
- 1.8. The system shall record the expected check-in date and time.
- 1.9. The system shall record the expected checkout date and time.
- 1.10. The system shall check-in customers.
- 1.11. The system shall checkout customers.
- 1.12. The system shall mark guaranteed rooms as "must pay" after 6:00 pm on the check-in date.
- 1.13. The system shall record customer feedback.

#### **3.2 Food**

- 1.14. The system shall track all meals purchased in the hotel (restaurant and room service).
- 1.15. The system shall record payment and payment type for meals.
- 1.16. The system shall bill the current room if payment is not made at time of service.
- 1.17. The system shall accept reservations for the restaurant and room service.

#### **3.3 Management**

- 1.18. The system shall display projected occupancy for a period of time (days).
- 1.19. The system shall display room revenue for a specified period of time (days).
- 1.20. The system shall display food revenue for a specified period of time (days).
- 1.21. The system shall display an exception report, showing where default room and food prices have been overridden.
- 1.22. The system shall allow for the addition of information, regarding rooms, rates, menu items, prices, and user profiles.
- 1.23. The system shall allow for the deletion of information, regarding rooms, rates, menu items, prices, and user profiles.
- 1.24. The system shall allow for the modification of information, regarding rooms, rates, menu items, prices, and user profiles.
- 1.25. The system shall allow managers to assign user passwords.
- 1.26. The system allow Manager to delete any servant account with reason.

## **4 Nonfunctional Requirements**

Non-Functional requirements define the needs in terms of performance, logical database requirements, design constraints, standards compliance, reliability, availability, security, maintainability.

### **4.1 Performance Requirements**

Performance requirements define acceptable response times for system functionality.

- The load time for user interface screens shall take no longer than two seconds.
- The log in information shall be verified within five seconds.

### **4.2 Logical Database Requirements**

The logical database requirements include the retention of the following data elements. This list is not a complete list and is designed as a starting point for development.

### **4.3 Booking/Reservation System**

- Customer first name
- Customer last name
- Customer address
- Customer phone number
- Number of occupants
- Assigned room
- Default room rate
- Rate description
- Guaranteed room (yes/no)
- Credit card number
- Confirmation number
- Total Bill

### **4.4 Food Services**

- Meal
- Meal type
- Meal item
- Meal order
- Meal payment Type (Bill to room/Credit/Check/Cash)

### **4.5 Design Constraints**

The Hotel Management System shall be a stand-alone system running in a Windows environment and system should be Desktop Application in Python

### **4.6 Availability**

The system shall be available during normal hotel operating hours.

## **4.7 Security**

Securing a hotel management system is crucial to protect sensitive guest information. There for following feature should be added in the System.

- Extra Login Checks
- Safe Website Connection
- Keep Important Information Secret
- Check for Computer Safety Often
- Record and Watch for Strange Actions
- Follow Credit Card Safety Rules
- Use Strong Password Rules
- Protect Computers with Virtual Walls
- Keep Computers in Safe Places
- Save Important Info in More Than One Place
- Teach Workers About Keeping Things Safe

## **4.8 Maintainability**

The Hotel Management System is being developed in Python and it used the OOP Concept it