Software Requirements Specification
Hotel Management System

Prepared by Aleem Maknojia, Ali Asar Khowaja,
Bilal Pervez & Hammad Suleman Khan

SZABIST University

4th January, 2023

1. Introduction

1.1 Purpose

This is our project about HOTEL MANAGEMENT SYSTEM to manage bookings, employees, rooms and hotels efficiently in order to provide the best service. The design of our project is based on SAAS applications where we provide Software as Service. Firstly, we are the super admins of this application and the hotel owners can get our application on a subscription based model. We will be registering the hotels in a database and authenticating them to use this application.

1.2 Document Conventions

We have used a Roboto font style with the size of 11 pts for the text and the size of 12 pts and bold for headings.

1.3 Intended Audience and Reading Suggestions

This software is intended to be used by the owner and the staff of the hotel. Staff includes general staff (receptionist, porter etc.) and manager. Each type of user has different privileges and rights.

This document is meant for the developers, testers, project managers and the client. This document is designed to give the overall description of the requirements for the project. It further elaborates on what the interfaces are used with emphasis on the user and system interface including GUIs and other hardware components. It is recommended to read the document in a linear order and go over each section thoroughly to understand the flow of how the requirements are broken down and addressed.

1.4 Product Scope

The motivation behind the idea is that nowadays the world is getting digitized so in order to remove all manual and paper work we have created system which would help us to manage data more accurately and efficiently. Moreover, this would give an ability to keep records of guests and employees which could be further used for analytics.

We have provided full rights to the user where he can add, update, search and delete information when required. Best database design is used in order to increase efficiency of managing data from various hotels.

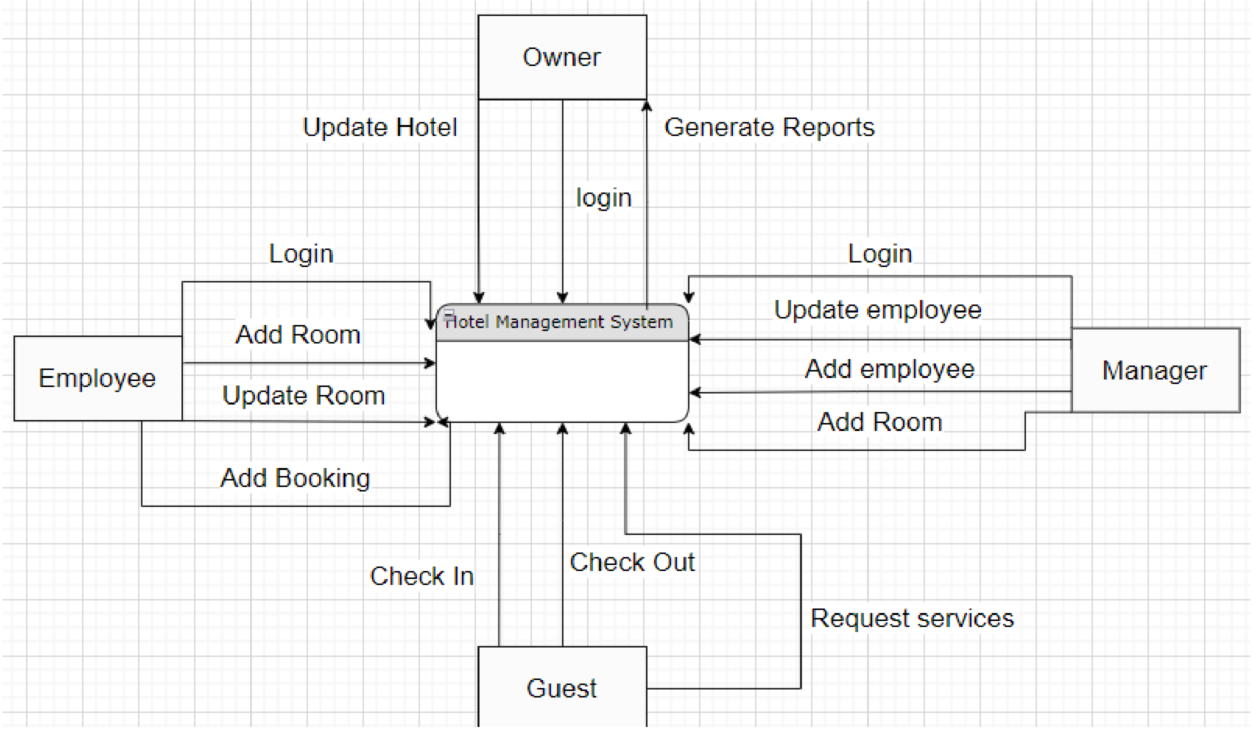
This application is very useful for a long run. This can help hotels manage their staffs and guest with ease just by a single click as this provides functionality of generating reports. Moreover, data is managed efficiently and accurately which would help users to fetch everything easily.

1.5 References

1. Database System Concepts - Abraham Silberschatz, Henry F. Korth, S. Sudarshan
2. Database Systems - Ramez Elmasri , Shamkant B. Navathe
3. https://www.javatpoint.com/software-requirement-specifications
4. https://www.geeksforgeeks.org/software-requirement-specification-srs-format/

2. Overall Description

2.1 Product Perspective



2.2 Product Functions

- Adding, updating, deleting and searching Hotels.
- Generate hotel reports based on bookings.
- Adding services.
- Adding employees. (CRUD)
- Adding guests (CRUD)
- Maintaining payments of guests.
- Maintaining revenue of hotels.

2.3 User Classes and Characteristics

- **Owner**

  Owners will Manage the hotel settings and be able to generate reports. The owner is not involved in the day to day usage of the software and will only use it to look at how the business is performing and top level changes.

- **Manager**

Manager will use the system to add employees mostly or update employee information. They will be able to add or remove new services and rooms in case the facility expands or changes.

- **Employee**
  Employees will use the system most of the time. They will be processing bookings for the guests, assigning rooms and updating room status for guests.These guys will be depending on the system for most of their tasks and will need to be properly trained on how to use the system efficiently.

- **Guest**
  Guests will not directly interact with the system however their check-in/check-out information as well as their cost will be processed and displayed by the system .

2.4 Operating Environment

The database will be running on a windows server in the hotel premises and the software application will run on multiple devices on the windows operating system. For example the application will be running on the receptionist’s computer as well in the manager's office.

2.5 Design and Implementation Constraints

We have implemented this project using the concepts of database designing along with frontend using C# and MS Sql Server as per the client’s request and to make our program more efficient. We have provided few features such as CRUD operations for all aspects where required. We have also tried to manage project security and integrity using various constraints and abstracting data. We have also kept in mind to have enough flexibility in our data structure so that we can generate insightful reports and statistics about the hotel.

2.6 User Documentation

The goal of this project is to deliver the required features and along with comprehensive documentation reflecting the process of identifying these requirements and the process of coming up with solutions. This documentation will also serve as a reference for future requirements, design and development as well as usage. This document includes SRS, SDS and STD deliverables as printed word document files as well as a soft pdf copy.

2.7 Assumptions and Dependencies

We have assumed that the user will be computer literate and will receive some form of training to be able to use the software proficiently.
Also this project is way beyond the scope of the limited features that we have defined here. On that account we assume that all the other necessary features are already existing for the scope of designing this SRS.
We have used concepts from our previous project to produce this document in which the language that is used is C# and for IDE Visual Studio is primarily used. Basic concepts of OOP along with the best database design approach is used with MS SQL Server. We have also made use of the UI library GUNA.

3. External Interface Requirements

3.1 User Interfaces

We are using MS Windows GUI standard. Here every screen will have buttons, input text areas and some form of information display section like a data table. There will also be pop ups messages for confirmation or invalid input or response.

3.2 Hardware Interfaces

The system is to be used on a personal computer with the same input output devices such as mouse, keyboard and monitor (ATM card reader if needed). There is also the touch screen feature available that could be used on desktop systems as well as portable windows devices.

3.3 Software Interfaces

- **MS SQL**
  As a database storage for convenient data updates and fetching.
- **GUNA**
  A UI Library used to make the UI.
- **C# Libraries (WinForms)**
  We have used many utility functions and classes provided by the built in C# classes and libraries.

3.4 Communications Interfaces

The only communication interfaces and standards involved in the product are related to the MS SQL connection. The data transfer rates, and synchronization mechanisms used are all provided by SQL queries.

4. System Features

4.1 Login

4.1.1 Summary:

The Actors who have access to the system will authenticate into the system using the Login Page.

4.1.2 Actors:

- Hotel Owner
- Manager
- Employee

4.1.3 Preconditions:

- The User is not authenticated.
- User is on the login page

4.1.4 Basic course of events/happy path:

Actor Action	System Response
1. Enters Credentials (username and password)	
2. Press submit button	
	3. Validate Credentials
	4. Upon successful validation goto Main page for relevant user

4.1.5 Alternative path:

- (4) If user Enters incorrect credentials go back to login page and show error

4.1.6 Post condition:

1. User will get redirected to the dashboard for his specific user

4.1.7 Author Name:

Bilal Pervez 3 December

4.2 Hotel Owner - Update Hotel Information

4.2.1 Summary:

Hotel Owner should be able to update hotel information. Such as the name of the hotel and address, etc.

4.2.2 Actors:

- Hotel Owner

4.2.3 Preconditions:

- User is authenticated
- User has role of Hotel Owner
- User has access to Hotel information page

4.2.4 Basic course of events/happy path:

Actor Action	System Response
1. Press edit hotel information button	
	2. Show Edit Hotel Information page, with fields already filled from last values.
3. Update or change field values	
	4. Validate the fields
	5. Update Hotel Information in the database.
	6. Show success message.
	7. Go back to Dashboard

4.2.5 Alternative path:

1. (4) If validation fails show error

4.2.6 Post condition:

1. User will get redirected to the dashboard for his specific user

4.2.7 Author Name:

Bilal Pervez 4 December

4.3 Hotel Owner - Generate Reports

4.3.1 Summary:

Hotel Owner should be able to view statistics and generate reports about things like his

4.3.2 Actors:

- Hotel Owner

4.3.3 Preconditions:

- User is authenticated
- User has role of Hotel Owner
- User has access to Reports Page

4.3.4 Basic course of events/happy path:

Actor Action	System Response
1. Click on Reports Button	
	2. Show parameters for reports
3. User Enters the parameters to generate a report. I.e. Date period selection.	
	4. Check if parameters are valid
	5. Generate the report according to the selected date and other parameters
6. User opens specific report	
	7. Show Report Details

4.3.5 Alternative path:

(4) If validation fails show error

4.3.6 Post condition:

2. User will be on the report details page

4.3.7 Author Name:

Bilal Pervez, 4 December

4.4 Add Employee

4.4.1 Summary :

This use case is used for adding an employee to the system

4.4.2 Actors :

Manager

4.4.3 Preconditions :

Manager is logged in with manager account

4.4.4 Basic course of events/happy path :

Actor Action	System Response
1. Clicks Employee	
	2. Shows Employee menu
3. Fill all the fields and clicks Add	
	4. Validate fields
	5. Add employee to database
	6. Shows employee added
	7. Generates id and password

4.4.5. Alternative path :

(4) If fields are left unfilled shows message all fields are required
(4) If invalid field entered show invalid field entered

4.4.6. Post condition :

1. Employee added
2. Shows Employee page

4.4.7. Author Name :

Hammad Suleman Khan , 1.December.2022

4.5 Update Employee

4.5.1 Summary :

This use case is used for updating an employee to the system

4.5.2 Actors :

Manager

4.5.3 Preconditions :

Manager is logged in with manager account, Employee exists in database

4.5.4 Basic course of events/happy path :

Actor Action	System Response
1. Click Employee	
	2. Shows Employee menu
3. Searches employee (from table or with id)	
	4. Auto fill all the fields with old values
5. Changes desired fields (which are allowed) & presses update	
	6. Validate fields
	7. Updates Employee in database
	8. Shows employee updated

4.5.5 Alternative path :

3. If id does not exist shows incorrect value for id
6. If invalid field entered show invalid field entered

4.5.6 Post condition :

Employee updates; shows Employee page

4.5.7 Author Name :

Hammad Suleman Khan , 1.December.2022

4.6 Add Room

4.6.1 Summary :

This use case is used for adding a room to the system

4.6.2 Actors :

Manager

4.6.3 Preconditions :

Manager is logged in with manager account, Room does not already exists in database

4.6.4 Basic course of events/happy path :

Actor Action	System Response
	1. Shows Room fields
2. Fills all the fields	
3. Presses add room	
	4. Validate fields
	5. Add room to database
	6. Shows room added

4.6.5 Alternative path :

6. If invalid field entered show invalid field entered
11. If invalid field entered show invalid field entered

4.6.6 Post condition :

Employee updates; shows Employee page

4.6.7 Author Name:

Hammad Suleman Khan , 1.December.2022

4.7 Update Room

4.7.1 Summary :

This use case is used to update rooms availability and details.

4.7.2 Actors :

Employee

4.7.3 Preconditions :

Employees must be logged in the system and the room which needs to be updated must exist in the database.

4.7.4 Basic course of events/happy path :

Actor Action	System Response
1. Select Room panel on dashboard.	
	2. Show room panel menu.
3. Search or select a room which needs to be updated. (by id)	
	4. Load Room fields data from database.
5. Fill the fields that need to be updated.	
6. Press update button	
	7. Validate fields.
	8. Update database.
	9. Show success message.

4.7.5 Alternative path :

3. If the room doesn't exist, show a NOT FOUND message.
7. If invalid field is entered, show invalid field entered for every individual error.
9. In case of any error show a failure message.

4.7.6 Post condition :

Rooms updated; Show the tables where all rooms exist on the same page.

4.7.7 Author Name:

Ali Asar Khowaja, 7-Dec-2022

4.8 Add Booking

4.8.1 Summary :

This use case is used to add bookings.

4.8.2 Actors :

Employee

4.8.3 Preconditions :

Employees must be logged in the system.

4.8.4 Basic course of events/happy path :

Actor Action	System Response
1. Select Booking panel on dashboard.	
	2. Show booking panel menu.
	3. Shows Booking fields
4. Fills all the fields	
5. Click the calculate amount button to get total payable payment.	
	6. Validate fields
	7. Generate Booking Id
8. Press the add booking button.	
	9. Validate fields
	10. Add booking to database
	11. Show initial status as checked in.
	12. Shows booking added.

4.8.5 Alternative path :

5. If the total amount is not calculated, booking cannot be added.
6. If invalid field is entered, show invalid field entered for every individual error.
12. In case of any error show a failure message.

4.8.6 Post condition :

Bookings updated; Show the tables where all new booking is added on the same booking page.

4.8.7 Author Name:

Ali Asar Khowaja, 7-Dec-2022

4.9 Update Booking

4.9.1 Summary :

This use case is used to update bookings details.

4.9.2 Actors :

Employee

4.9.3 Preconditions :

Employees must be logged in the system and the booking which needs to be updated must exist in the database.

4.9.4 Basic course of events/happy path :

Actor Action	System Response
1. Select Booking panel on dashboard.	
	2. Show booking panel menu.
3. Search or select a booking which needs to be updated. (by id)	
	4. Load Booking fields data from database.
5. Fill the fields that need to be updated.	
6. Compute payment again if new services or new room types are added.	
7. Press the update button.	
	8. Validate fields.
	9. Update database for same booking id.
	10. Show success message.

4.9.5 Alternative path :

3. If the booking doesn’t exist, show a NOT FOUND message.
6. Payment must be computed first then update can be done.
8. If invalid field is entered, show invalid field entered for every individual error.

10. In case of any error show a failure message.

4.9.6 Post condition :

Booking updated; Show the tables where all bookings exist on the same booking page.

4.9.7 Author Name:

Ali Asar Khowaja, 7-Dec-2022

4.10 Check in

4.10.1 Summary :

This use case is used to check into the room.

4.10.2 Actors :

- Guest

4.10.3 Preconditions :

- The user must have valid booking ID

4.10.4 Basic course of events/happy path :

Actor Action	System Response
1. Enters Booking ID	
2. Press submit button	
	3. Validate Booking ID
	4. Upon successful validation update the booking details with room ID
	5. Show success and go to main page

4.10.5 Alternative path :

4. If user Enters incorrect Booking ID go back to main page and show error

4.10.6 Post condition :

User will be shown a “check in successful” message and will be directed to the main page

4.10.7 Author Name:

Aleem Maknojia, 27-Dec-2022

4.11 Check out

4.11.1 Summary :

This use case is used to check out from the room.

4.11.2 Actors :

- Guest

4.11.3 Preconditions :

- The user must have valid booking ID
- The booking ID must have a valid room ID linked to it

4.11.4 Basic course of events/happy path :

Actor Action	System Response
1. Enters Booking ID	
2. Press submit button	
	3. Validate Booking ID
	4. Upon successful validation remove the linked room ID from the Booking ID
	5. Update Booking status as complete
	6. Show success and go to main page

4.11.5 Alternative path :

4. If user Enters incorrect Booking ID go back to main page and show error

4. If there is no room linked to the booking ID in database go back to main page show error message

4.11.6 Post condition :

User will be shown a “check out successful” message and will be directed to the main page

4.11.7 Author Name:

Aleem Maknojia, 27-Dec-2022

4.12 Request Service

4.12.1 Summary :

This use case is used to request room services.

4.12.2 Actors :

- Guest

4.12.3 Preconditions :

- The user must have valid room ID

4.12.4 Basic course of events/happy path :

Actor Action	System Response
1. Enters Room ID	
2. Press submit button	
	3. Validate room ID
	4. Show list of room services
5. User choose services	
	6. Calculate the charges and update on the database.
	7. Show the total charges and show success message
	8. Go to main page

4.12.5 Alternative path :

3. If validation fails show “Invalid Id”.
5. Request services that are available.

4.12.6 Post condition :

User will be shown the total and a “room service request successful” message and will be directed to the main page

4.12.7 Author Name:

Aleem Maknojia, 27-Dec-2022

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The system should be efficient enough to handle several requests at the same time. The software will be used by a hotel which would have multiple bookings, therefore it must be structured to encounter those requests simultaneously. The speed of response time must be high in order to provide hassle free experience for hotel owners. Storage capacity and storage time must be managed systematically.

5.2 Safety Requirements

The customer data should be backed up so that the valuable data stored in the database can be recovered in case of an accident. UPS systems should be installed to stabilize electrical disruptions that may cause harm to the data. Network security protocols should be followed under the ISO/IEC 27033-6:2016 to prevent unauthorized access .

5.3 Security Requirements

The system will have sensitive information such as customer personal information and credit information. To ensure that none of the data is misused, data should be encrypted and strong password authentication for every user should be used. Also sensitive data handling privileges should reside with the manager and owner only. Credit information should not be accessible to anyone other than for verification purposes.

5.4 Software Quality Attributes

This system needs to be reliable and fast as it will be used at the reception and if the system is slow or if the system is unavailable it will affect customer satisfaction.
The system will also hold data that will be viable for the day to day and long term operation to ensure that the data does not get corrupted.
The system also needs to be portable as it would be used on many different devices in many places in the hotel (e.g. reception, managers office, different places for the employees on each floor).

5.5 Business Rules

Customers cannot directly access the system. Bookings would only be made by the system operation that is receptionist. Customers cannot get access to the room without payment.