Software Requirements Specification

for

Hotel Management

Version 1.0 approved

Prepared by Manali Shah

National Institute of Technology Karnataka, Surathkal

9th January, 2018

Table of Contents

Table of Contents……...2
Revision History……..3
1. Introduction……...4
1.1 Purpose…….4
1.2 Document Conventions……...4
1.3 Intended Audience and Reading Suggestions…….4
1.4 Product Scope…….5
1.5 References……..5
2. Overall Description……...6
2.1 Product Perspective……....6
2.2 Product Functions……...6
2.3 User Classes and Characteristic……..6
2.4 Operating Environment……...7
2.5 Design and Implementation Constraints…….7
2.6 User Documentation……...8
2.7 Assumptions and Dependencies…….8
3. External Interface Requirements…….8
3.1 User Interfaces……8
3.2 Hardware Interfaces……....8
3.3 Software Interfaces……....8
3.4 Communications Interfaces……....8
4. System Features…….9
4.1 Registration……...9
4.2 Login…….....9
4.3 Reservation…….9
4.4 Receptionist Access……....9
4.5 Manager Access…….9
5. Other Nonfunctional Requirements……...10
5.1 Performance Requirements……...10
5.2 Safety Requirements……10
5.3 Security Requirements……..10
5.4 Software Quality Attributes……..10
5.5 Business Rules……..11
6. Other Requirements…….11
Appendix A: Glossary……11
Appendix B: Analysis Models……...11
Appendix C: To Be Determined List……11

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The Software Requirements Specification (SRS) will provide a detailed description of the requirements of the Hotel Management System (HMS). This SRS will allow for a complete understanding of what is to be expected from the newly introduced system which is to be constructed. The clear understanding of the system 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 tested.

This SRS will be used by the system development team which is constructing the HMS and the hotel end users. This will be used to fully understand the expectations of this HMS and construct the appropriate software. The hotel end users will be able to use this SRS as a "test" to see if their needs have been satisfied. If not, they can specify how it is not to their liking and the SRS will be changed to fit the needs.

1.2 Document Conventions

This document is prepared using Microsoft Word 2007 and has used the font type 'Times New Roman'. The font size used is 12pt everywhere except for headings where 14pt and 18pt is used. Line spacing is 1.5. The bold property has been used for headings.

1.3 Intended Audience and Reading Suggestions

This SRS is intended for

- Developers who can review project's capabilities and more easily understand where their efforts should be targeted to improve or add more features to it.
- Project testers who can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized.

- End users of this application (like hotel managers, receptionists and clients) who wish to read about what this project can do.

1.4 Product Scope

The HMS project is intended for the online reservations of rooms in a hotel. The HMS will have three end users: customer, receptionist and hotel manager. Customers will be able to check for availability of rooms, select the rooms and pay for the room. Receptionist will have access to update or modify the booking details. Manager will be able to view the financial report and update room information like cost and category.

The main goal is to simplify the everyday process of hotel booking. The number of hotels are increasing and they need to automate to provide customer ease of access. It will be able to take care of services to customer in a quick manner. This automation will be able to replace the drawbacks of large customer information physical files which were difficult to handle. Various advantages of this are:

- Convenience: Rooms can be booked at home itself, without going anywhere
- Saves time and effort: Time needed to go to the hotel and book rooms is saved.
- Towards a greener planet: Instead of printing receipts and maintaining hard copies, everything is done online so paper is not wasted.
- Improved and optimized service

1.5 References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

2. Overall Description

2.1 Product Perspective

The HMS is a new self contained software product which will be produced in order to overcome the problems that are present in the manual system. This will provide easy access to the system and it will contain user friendly functions. The system will give better options for the problem of handling large scale physical files, errors occurring in calculations and all other required tasks that have been specified by the client. The final outcome will increase the efficiency of almost all the tasks in a more convenient manner.

2.2 Product Functions

- Customer Registration: Before booking, the customer needs to register his/her name on the system.
- Check availability of rooms: The customers can check which rooms are available and of what category.
- Display the rate: The price of each room will be displayed.
- Confirmation of booking: The client can reserve the room online and confirm the booking.
- Set room details: The hotel manager can change the category and cost of rooms.
- Manage booking details: The receptionist can modify or update booking details.
- Generate report: A receipt, or report, containing the room booked, cost and further details is generated.

2.3 User Classes and Characteristics

There are 3 user levels in out HMS.

1. Hotel Manager
2. Receptionist
3. Customers

Hotel Manager: The manager has access to the hotel system. He is solely responsible for managing hotel resources and staff. Manager can view any report such as financial report, customer information, booking information and room information, analyze them and take the decision accordingly. He is required to have experience on managing a hotel previously and have a basic knowledge of database and application server.

Receptionist: The sole purpose of the hotel receptionist is to provide quality customer service. She will have less access than the manager. She can manage booking details, search for availability of rooms, add customers, confirm the booking and update the booking details. She should have good communication skills and basic IT knowledge.

Customer: They are a vital part of the system. They have access to view the vacant room information and price range. They should be able to confirm the booking and cancel if necessary. The customer should be able to use the interface.

2.4 Operating Environment

Operating Environment for the hotel management system is:

- Distributed Database
- Client/Server System
- Operating System: Windows
- Database: SQL Database
- Platform: PHP

2.5 Design and Implementation Constraints

To maintain the reliability and durability of the system, some design and implementation constraints are applied.

System will need a minimum memory of 512 MB. But it is recommended to have a memory of 1 GB. Language used in the software is English only.

2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2.7 Assumptions and Dependencies

It is assumed that the users of the software can operate and use it and have basic knowledge of how to operate computers. Also, the client should be able to afford the amount of money required to purchase the software. The client should not change his/her decision on the next phases of software development.

3. External Interface Requirements

3.1 User Interface

The user interface for system shall be compatible with web browsers like Mozilla Firefox and Google Chrome.

3.2 Hardware Interfaces

1. Operating System: Supported in Linux
2. Computer: 512 MB+ Ram, keyboard, mouse, monitor
3. Hard drive: Minimum 10 GB of free space.

3.3 Software Interfaces

Software can run on the Linux Operating System. The web server used is Apache and database server used is MySQL. The development end uses HTML, CSS, JavaScript and PHP.

3.4 Communications Interfaces

The system will use HTTP/HTTPS for communication over internet.

4. System Features

4.1 Registration

- The customer should be able to register with the details
- The following customer details should be entered in the database: Name, Email, Contact Number, Password, Address, DOB.

4.2 Logging In

- The system should verify the customer email and password against the member database while logging in
- After login, the member is directed to home screen

4.3 Reservation

- The system should allow the customer to check for availability of rooms
- The system should display rate for all rooms
- The system should allow the customer to confirm or cancel the booking
- The system should record booking details into database

4.4 Receptionist Access

- The receptionist can update, add or delete booking and customer information

4.5 Manager Access

- The manager can view, update, add or delete room information

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The database should be scalable; it must have the capacity to hold large number of users in future.
- The number of connections to the system should not slow down the application to a large degree.
- The data for the analysis will be obtained from the database of users, so the response time for a query from the client side to the database side should not be more than 5 seconds.
- Error handling should be implemented and the application should be able to handle all runtime errors.
- The application should be flexible for future enhancements, for example, the addition of a few more additional features.

5.2 Safety Requirements

Backups must be taken regularly.

5.3 Security Requirements

- Users shall be required to log in to the RRS for their own reservation information and modification with email address and password.
- The system shall permit only authorized members who are in the authorized database.
- The system shall permit customers to view only their own previously placed orders, not orders placed by other customers.
- All data must be stored, protected or protectively marked.

5.4 Software Quality Attributes

- Correctness: It should satisfy the normal regular HMS operations to fulfil end user objectives.

- Efficiency: Resources should be implemented to achieve the particular task efficiency without hassle.
- Flexibility: We should be able to add new features and handle them conveniently.
- Integrity: System should focus on securing customer information and avoid data loss as much as possible.
- Portability: System should run in any Windows or Linux system.
- Maintainability: System should be maintainable.

5.5 Business Rules

None.

6. Other Requirements

When the system is completely developed and submitted to the client, few sessions will be required to make the users of the system understand the functionality and adapt to the system. After these sessions, it is required that a member from the development team should spend some time in the system background for an agreed time period. That time period will be used to identify new bugs.

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>