Software Requirements Specification

for

Hostel Management System

Version 1.0 approved
Prepared by

Nilita Anil Kumar
Nandini AV

17th january 2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
nandini	17th jan 2019		1
nandini	5th feb		1.1

1.Introduction

1.1Purpose

The purpose of this document is to present a detailed description of the Web Publishing System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed to the Regional Historical Society for its approval.

1.2 Document Conventions

This document uses the following conventions.

- DB Database
- DDB Distributed Database
- ER Entity Relationship
- JS javascript
- SRS – Software Requirements Specifications
- HMS Hostel Management System

1.3 Intended Audience and Reading Suggestions

This project is a prototype for the hostel management system and it is restricted within the college premises. This has been implemented under the guidance of college professors. This project is useful for the hostel management and as well as to the students

1.4 Product Scope

The software product “Hostel Management System” will be an application that will be used for maintaining records in an organised manner and to replace old paper work system. This project aims at automating the hostel management for smooth working of the hostel by automating almost all the activities. Updations and modifications will be easily achievable and all the calculations and accounting work would be more accurate.

1.5References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2.Overall Description

2.1Product Perspective

2.1.1.System Interface
The HMS is a complete web enabled system which can be accessed through web browser.
2.1.2.User Interface

The user interface is as follows:
Screen Name Description
Login Login into system as student or admin
Student Module:
Profile Student can view and update personal details. Apply Room Student applies room by selecting the preferred room, semester. Status of Application Student can check their application status.View history Students can view history of past applications Change Password Student can change his/her password. Logout After the student is done using the system, he/she logs out.
Administrator Module:
Add user Admin can add a new user by assigning new user name and passwords respectively. View Applicants Admin can view the applicants by selecting the year and semester. View Students Admin can view the student'sdetails Change Password Admin can change his/her password.

2.2Product Functions

2.2.1.User Functions
The administrator of HMS shall add new users to the system who is basically the student. After entering the information about the user, the system gives a unique username and password to the user.The administrator shall view applicants and students, and change password.
2.2.2.Student Functions
The student shall view and update their profiles.
The student shall apply a room.
The student shall view the status of application.
The student shall view their history and change their password.

2.3User Classes and Characteristics

The Administrator
This user has to have at least Window 7/Linux OS and Internet browsing skills for administrating HMS user profiles.
The Student
This user has to have at least Window 7/Linux OS and Internet browsing skills to use the system.

2.4Operating Environment

- Server Side

Apache Web server is installed and will enable HMS to interact with its users. PHP is a server-side scripting language, which will be used to code the HMS.

- Client Side

On the client side the required software product is Internet Explorer/Google Chrome/Mozilla Firefox supporting at least HTML version 3.2, java enabled, and any operating system that can run the browsers.

- Communication Interface

The default communication protocol for data transmission between server and the client is Transmission Control Protocol/ Internet Protocol (TCP/IP). At the upper level Hyper Text Transfer Protocol (HTTP) will be used for communication between the web server and client.

- The Administrator

This user has to have at least Window 7/Linux OS and Internet browsing skills for administrating HMS user profiles.

- The Student

This user has to have at least Window 7/Linux OS and Internet browsing skills to use the system.

2.5Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

2.6User 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.>

2.7Assumptions and Dependencies

The system will not store any payment information; rather all payments will be handled by the finance department of the university.

Credit card payment or any other form of payment other than through the finance department is not allowed on the system

3.External Interface Requirements

3.1User Interfaces

The user interface is as follows:
Login Login into system as student or admin
Student Module:

- Profile Student can view and update personal details.
- Apply Room Student applies room by selecting the preferred room, semester.
- Status of Application Student can check their application status.
- View history Students can view history of past applications
- Change Password Student can change his/her password.
- Logout After the student is done using the system, he/she logs out.

Administrator Module:

- Add user Admin can add a new user by assigning new user name and passwords respectively.
- View Applicants Admin can view the applicants by selecting the year and semester.
- View Students Admin can view the Student's Details
- Change Password Admin can change his/her password.
- Add user Admin can add a new user by assigning new user name and passwords respectively.
- View Applicants Admin can view the applicants by selecting the year and semester.
- View Students Admin can view the student's Details.
- Change Password Admin can change his/her password.
- Add user Admin can add a new user by assigning new user name and passwords respectively.
- View Applicants Admin can view the applicants by selecting the year and semester.
- View Students Admin can view the student's Details.
- Change Password Admin can change his/her password.

3.2Hardware Interfaces

Client Side
Any Personal computer, which can support any 7-window or Windows environment with a mouse support, is acceptable.

Server Side
HMS will be run on a web server, which is installed into the school server.The school servers have requirements to operate PHP scripts (Apache Web server 1.3.2 with PHP 4.0
modules).

3.3Software Interfaces

Server Side

Apache Web server is installed and will enable HMS to interact with its users. PHP is a server-side scripting language, which will be used to code the HMS.

Client Side

On the client side the required software product is Internet Explorer/Google Chrome/Mozilla Firefox supporting at least HTML version 3.2, java enabled, and any operating system that can run the browsers.

3.4Communications Interfaces

The default communication protocol for data transmission between server and the client is Transmission Control Protocol/ Internet Protocol (TCP/IP). At the upper level Hyper Text Transfer Protocol (HTTP) will be used for communication between the web server and client.

4.System Features

4.1 Description and Priority

The HMS uses the standard input/output devices for a personal computer. This includes the following: Keyboard, Mouse, Monitor and Printer.

4.2 Stimulus/Response Sequences

The HMS operations needed by the users are described below.

Administrator of the system creates and defines the status of users by (Add User). The user will be given a unique username and password. The Admin may change their passwords by (Change Password). The Admin can view applicants and also view the student's details

The student accesses the system by logging in. They can view their profiles and update it (Profile), Apply room, View Status of Application, View history and change their passwords.

4.3 Functional Requirements

4.3.1 Administrator Functions

4.3.1.1 Add User

Introduction: HMS shall enable administrator to add new users to the system.

Input: username and password.

Process: The administrator activates the function and enters the username and password of the new user.

The function will also check the database whether the user already exists or not. According to the results, the system adds the user to the all user list with a confirmation message, or the function displays an error message.

Output: error message or confirmation message.

4.3.1.2 View Applicants

Introduction: HMS shall display all the applicants archived in the system.

Input: none

Process: The administrator selects the semester and year. The function queries the database for the students who have applied for rooms.

Output: All applicants with their respective details (user id, preferred room, and assigned room id) will

be displayed.

4.3.1.3 View Students

Introduction: HMS shall display all the students in the system.

Input: none

Process: When the administrator logon the system, automatically, all student list is displayed. The function queries the database for all the students.

Output: List of all students with their respective details (student id, first name, and last name, and gender, place of residence, phone number, and address) will be displayed.

4.3.1.4 Change Password

Introduction: HMS shall enable administrator to change the password.

Input: old password, new password, confirm password

Process: Administrator activates the function to change the password. The new password and confirm password fields are entered. If they match, the old password will be updated with the new one.

Output: Error or confirmation message will be displayed.

4.3.2 Student Functions

4.3.2.1 Profile

Introduction: HMS shall enable student to view and update their profile.

Input: none

Process: By this function, the database is queried for all the personal information of the student.

Output: All students' personal information is displayed.

4.3.2.2. Apply Room

Introduction: HMS shall enable a student to apply a room.

Input: preferred room, semester, and year.

Process: By this function, the selected information is stored into the database.

Output: All students' application information is stored into the database

4.3.2.3. Status of Application

Introduction: HMS shall enable the student to view the status of their room application.

Input: student id

Process: By this function, the database is queried for all the room application information of the student.

Output: All application status is displayed.

4.3.2.4. View History

Introduction: HMS shall enable the student to view their previous application history.

Input: student id

Process: By this function, the database is queried for all the previous room application information of the
student.

Output: All room application history is displayed.

4.3.2.5. Change Password

Introduction: HMS shall enable student to change the password.

Input: old password, new password, confirm password

Process: student activates the function to change the password. The new password and confirm password
fields are entered. If they match, the old password will be updated with the new one.

Output: Error or confirmation message will be displayed.

5.Other Nonfunctional Requirements

5.1Performance Requirements

Performance requirements define acceptable response times for system functionality. Depending on theuser internet connection speed;
The load time for user interface screens should take no longer than two seconds.
The log in information shall be verified within five seconds.
Queries shall return results within five seconds.

5.2Safety Requirements

Access to the system is protected by username and password by using a user login screen. Maintaining backups ensures the database security. System restores in case of emergencies.

5.3Security Requirements

Nobody should be allowed to tamper with data; Enhanced Security for sensitive data. It should be made sure that only users who are given specific rights can access data and all actions are logged, thus providing an extensive role based authorization.

5.4Software Quality Attributes

Availability:The system shall be available for 24 hours.
Correctness : extent to which program satisfies specifications , fulfills user's mission and objectives

5.5Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

6.Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

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.>