

Online Academic Registration

CS4089 Project

Software Requirements Specification (SRS)

Rohit Garlapati (B110316CS)
Debesh Baidya (B111001CS)
Bakki Ebenezer Jayakar (B110822CS)
Setty Prasanna Kumar (B110724CS)

Guided By: Ms. POURNAMI P N

December 11, 2015

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Document conventions	2
1.3	Target Audience	2
1.4	Project Scope	2
1.5	References	2
2	Software Description	2
2.1	Perspective	2
2.2	Features	2
2.3	Operating Environment	3
2.4	Documentation	3
3	Functional Requirements	3
3.1	Inputs	3
3.2	Actions	4
3.3	Outputs	4
3.4	Use case diagrams	4
3.5	Error Avoidance	5
3.6	Requirements	5
4	External Interface Requirements	5
4.1	User Interfaces	5
4.2	Hardware Interfaces	5
4.3	Software Interfaces	5
4.4	Communication Interfaces	5
5	Nonfunctional Requirements	6
5.1	Performance Requirements	6
5.2	Safety Requirements	6
5.3	Security Requirements	6

1 Introduction

This section provides an initial detail of several functionalities and services provided by the web application. It reflects the current requirements of the project according to the user needs, as observed by the team mates. Developing a software that meets SRS is a challenging task.

1.1 Purpose

Purpose of this project is to provide an online application to smoothen the process of academic registration, by converting the procedures done offline to online.

1.2 Document conventions

No conventions are followed in this project.

1.3 Target Audience

This application is intended to be used by various people in our college according to their respective category. Students, Faculty, Hostel office and other college staff are the primary users who can easily navigate through the web application, to get the registration process done smoothly.

1.4 Project Scope

The project scope is mentioned in final report of the project.

1.5 References

This SRS document has followed the template and guidelines of Karl Wiegers, author of Software Requirements.

2 Software Description

2.1 Perspective

A future version of website design maybe needed according to the needs of the students and other staff members. The sole purpose of the website is to provide a user-friendly way to complete the registration process. The application should be 'quick to learn' model of an interactive website for any type of user.

2.2 Features

Important features of the web application so far designed are,

1. Student login page, and registration option which involves step by step process of finishing tasks required.

2. Faculty login page.
3. Hostel records management and displaying hostel dues of individual student.
4. Library dues verification.
5. Uploading documents, which will be verified by a team of students who are members of SAC.
6. Registration by registration desk.

2.3 Operating Environment

The web application will be run on college servers.

2.4 Documentation

There are no specific user documentations provided with this software.

3 Functional Requirements

Before looking into the functional requirements of this project, let us see the inputs and outputs required for the web application,

3.1 Inputs

- A student is required to enter his user id or password and set the mode to student.
- A faculty advisor is required to set the mode to Faculty advisor and log in with his user id and password.
- The hostel staff, library staff are required to enter their respective modes, user names and passwords to log in to the web application.
- The hostel staff is supposed to enter the dues of all the students into the forms provided.
- Hostel staff can alter, update the data in the hostel dues table.
- Student has the option to enter payment details in case of dues.
- The library staff will be provided with a form to enter the dues of students.
- The library staff has the option to delete a student name from dues list by entering his roll number.
- The fa has an option to ask a student to meet him with a message.
- The student can request an appointment with FA.
- The student can request documents verification from SAC.
- The registration desk has the form which takes in a student id to display his check list, and register the student if verification has been done.

3.2 Actions

- When a user logs in, the user id and password are verified with the data in the database, and is directed to his/her homepage after successful login.
- The data entered by the hostel staff is stored in the hostel_dues entity.
- Temporary payment details of the student are stored in the unverified payments table.
- The data entered by the library staff is stored in library dues entity.
- When library staff deletes a student id from dues list the library_dues table gets altered.
- When a FA messages student, the message gets stored in student notifications table.
- When a student requests appointment with FA, his id gets stored in faculty notifications table.
- If a student requests document verification, his id is saved in sac notifications table.
- The checklist is updated for every completed task.
- The student registration is completed by updating the checklist to the maximum 5.

3.3 Outputs

- When a student successfully logs in he is shown all his dues from the database.
- Student will be able to proceed to each step one after the other.
- The hostel staff will be able to see the temporary payment details uploaded by the student.
- Library staff will have no specific output to see.
- As soon as a FA logs into his account he'll be shown notifications from faculty notifications table based on his user id.
- Similarly students and SAC members are shown their respective notifications.
- Registration desk will be able to see if the student has completed all the required tasks.

3.4 Use case diagrams

The Use cases and Use case diagrams have been explained in detail in design document provided with this document separately.

3.5 Error Avoidance

The most important problem a database can face in the time of retrieval and processing information is the wrong inputs. The forms that user fills will be validated at each step to avoid any unnecessary data.

3.6 Requirements

The website shall require users, of different type to register to access the features.

A constant internet connection is necessary.

4 External Interface Requirements

4.1 User Interfaces

These are the following user interfaces provided by the web application,

1. Interface for student
2. Interface for Faculty Advisors
3. Interface for Sac members
4. Interface for Hostel staff
5. Interface for Library staff
6. Interface for Registration desk

4.2 Hardware Interfaces

The hardware interfaces of the website will be handled by the operating system, in this projects case, Ubuntu Server 14.04.3 LTS.

4.3 Software Interfaces

As the software is designed to be economically feasible, all the software requirements are nominal.

4.4 Communication Interfaces

The web application will require internet connection to display content from the databases remotely.

5 Nonfunctional Requirements

5.1 Performance Requirements

- 64-bit Processor
- Min. of 2GB RAM
- 500 GB HDD
- Pentium IV or greater
- Any 64-bit OS
- Works with most of the latest browsers & IE9 or above.
- Constant and fast internet access.

5.2 Saftey Requirements

Saftey requirements are not required for this project. SQL injection will be detected as the data entered will be sanitized. The input forms will be validated continuously.

5.3 Security Requirements

- A application required user to enter his id/pwd to access the site.
- User will have no other option unless he logs into his account except to change his password or register.