

Software Requirements Specifications

AIM: TO WRITE THE SRS FOR UNIVERSITY REGISTRATION PORTAL.

1. INTRODUCTION.

- 1.1. PURPOSE
- 1.2. SCOPE
- 1.3. DEFINITIONS, ACRONYM, ABBREVIATIONS
- 1.4. REFERENCES
- 1.5. OVERVIEW

2. OVERALL DESCRIPTION

- 2.1. PRODUCT PERSPECTIVE
 - 2.1.1. *User interfaces*
 - 2.1.2. *Hardware Interfaces*
 - 2.1.3. *Software Interfaces*
 - 2.1.4. *Communication Interfaces*
 - 2.1.5. *Memory Constraints*
 - 2.1.6. *Site adaptation requirements*
- 2.2. PRODUCT FUNCTIONS
- 2.3. USER CHARACTERISTICS
- 2.4. CONSTRAINTS
- 2.5. ASSUMPTIONS AND DEPENDENCIES
- 2.6. APPORTIONING OF REQUIREMENTS

3. SPECIFIC REQUIREMENTS

- 3.1. EXTERNAL INTERFACE REQUIREMENTS
 - 3.1.1. *User Interfaces*
 - Login Screen*
 - User Profile*
 - Sign-up screen*
 - Course Detail screen*
 - Registration Screen*
 - Registration Confirmation screen*
 - Lists generation and viewing screen*
- 3.2. PRODUCT FUNCTIONS
 - 3.2.1. *Sign-up*
 - 3.2.2. *Login*
 - 3.2.3. *View Profile*
 - 3.2.4. *View Courses*
 - 3.2.5. *Register*

Practical - 6

- 3.2.6. *Manage Database*
- 3.2.7. *Maintain Course Details*
- 3.2.8. *Generate Lists*
- 3.2.9. *Update Profile*
- 3.2.10. *View Enrollments*
- 3.3. PERFORMANCE REQUIREMENTS
- 3.4. DESIGN CONSTRAINTS
- 3.5. SYSTEM ATTRIBUTES
 - 3.5.1. *Security*
 - 3.5.2. *maintainability*
 - 3.5.3. *Portability*
- 3.6. LOGICAL DATABASE REQUIREMENTS
- 3.7. OTHER REQUIREMENTS

1. Introduction.

This document aims at defining the overall software requirements for “UNIVERSITY REGISTRATION PORTAL”. Efforts have been made to define these requirements exhaustively and accurately. The final product will be having only features/functionalities mentioned in this document and assumptions for any functionality/feature should not be made by any of the parties involved in the development/testing/implementing/using of this product. In case it is required to have some additional features, a formal change request will need to be raised and subsequently a new release of this document and/or product will be produced.

1.1.Purpose

This specification document describes the capabilities that will be provided by the software application “UNIVERSITY REGISTRATION PORTAL”. It also states the various required constraints by which the system will abide. The intended audience for this document are the development team, testing team and the end user of the product.

1.2.Scope

The product “UNIVERSITY REGISTRATION PORTAL” will be web portal that will be used for the registration of B. Tech Program of a university. The application will manage the information about the students, the courses, and the enrollments of different students in various courses along with the information of professors assigned to said courses. Printable documents regarding the section/department/course student lists and the registration confirmation form/document will be generated.

The application will greatly simplify the speed of the registration process at the start of a semester.

1.3.Definitions, Acronym, Abbreviations

Following abbreviations have been used in the document:

B. Tech: Bachelors of Technology

DBA: Database Administrator

DEO: Data Entry Operator.

Practical - 6

1.4.References

- (i). University website: For information on courses
- (ii). IEEE Recommended Practice for Software Requirement Specification-IEEE Std. 830-1998

1.5.Overview

Rest of this document describes the various system requirements, interfaces features and functionalities in detail.

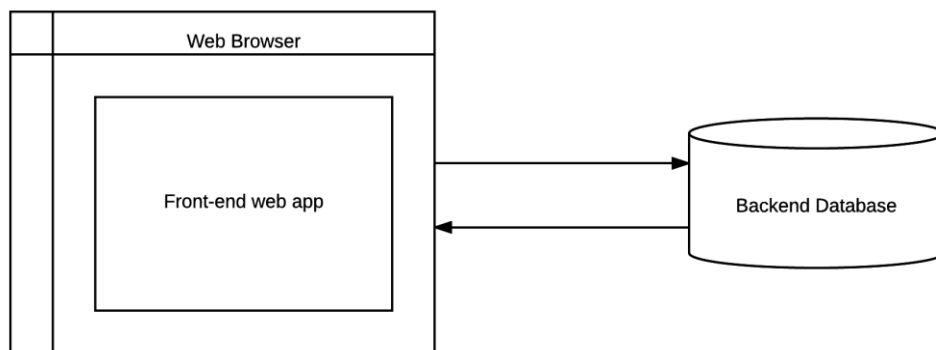
2. Overall Description

B. Tech is an 8-semester course. The students are offered between 4-6 subjects per semester along with the practical (where applicable). In the first four semesters the students follow a fixed department syllabus set by the university. In the last 4 semesters the students have the option to choose up-to 3 electives per semester from within or outside the branch/department. Each semester the student must obtain at least 24 credits to move on-to the next semester.

The “UNIVERSITY REGISTRATION PORTAL” will have the capability to maintain the student information of 2nd to 8th semester students and allows 1st semester students to sign up for the system providing them the facility to register for the next semester by selecting the required courses. The software will also generate a registration verification document and the department/course/section wise lists of the enrolled students.

2.1.Product perspective

The software will be web based, device independent and self-contained.



2.1.1. User interfaces

The application will have a user-friendly interface based on modern design standards following screens will be provided

- 2.1.1.1. A login in screen, for entering ID and Password.
- 2.1.1.2. A sign-up screen for 1st semester students.
- 2.1.1.3. A user profile screen for the students and professors to view and edit their information.
- 2.1.1.4. A courses screen showing all courses/departments.
- 2.1.1.5. A registration screen for the registration process.

Practical - 6

- 2.1.1.6. An admin panel for the management and admin to generate lists and reports.
- 2.1.1.7. A Registration Confirmation page for printing the registration form.
- 2.1.1.8. A screen to show various lists.

The following documents will be generated:

- 2.1.1.9. Student list for each subject/course.
- 2.1.1.10. Student list for each department.
- 2.1.1.11. Student Section list for the section assigned to each student.
- 2.1.1.12. A registration confirmation document.

2.1.2. Hardware Interfaces

- a) Screen resolution of at least 800x600 for proper viewing of the web-interface.
- b) Support for a printer for printing for generated documents

2.1.3. Software Interfaces

- a) Any Windows/Mac-OS/Linux Based system capable of running a web-browser.
- b) Oracle/MySQL/SQLite as the database client.
- c) An IDE (atom/Visual Studio/X-code)

B and C are only required during the development process.

2.1.4. Communication Interfaces

A working and stable internet connection preferably broadband.

2.1.5. Memory Constraints

Node, only those required by the browser to run on the system.

2.1.6. Site adaptation requirements

At the university a server to host the web service is required or at a hosting service like AWS or Azure.

2.2.Product Functions

The system will only allow access to users with valid logins and the functionalities will differ based on the roles.

Broadly the major functions are:

1. A sign-up facility for new students.
2. A login facility to all users.
3. A facility for students to view their and the professor's profiles
4. A facility for the students to browse the courses offered in the current semester.
5. A facility for the student to register for the upcoming semester and print generated documents.
6. A facility for the DBA to manage the database.
7. A facility for the DEO to update course details
8. A facility for the management to generate the various lists
9. A facility for students/professors/DEO to update the profile details
10. A facility for the DEO to update the registration details
11. A facility for the management and professors to view the enrolled students.

Practical - 6

2.3. User Characteristics

Education level: The DEO/Management staff should be comfortable with English language

Expertise: The DEO/Management staff should be well versed about the course structure.

Technical Expertise: The DEO/Management staff should be comfortable in using web browsers and general-purpose applications.

2.4. Constraints

Facility of database audits will be provided due to limited capabilities of MySQL and SQLite.

University will have to implement their own safe guarding the server from external threats and data backups.

2.5. Assumptions and Dependencies

The number of courses for each semester remains the same.

The subject type does not change.

The number of semester of B. Tech Program does not change.

2.6. Apportioning of Requirements

Not required.

3. Specific Requirements

This section contains the software requirements to a level of detail sufficient to enable designer to design the system, and testers to test the system.

3.1. External Interface Requirements

3.1.1. User Interfaces

The following screens will be provided

Login Screen

This is the first screen displayed to the users, it allows the user to log into the system to use it. It'll have the following fields:

User ID: email/Employee_ID.

Password: Alphanumeric, at least 8 characters.

User Profile

It will display the users Name, Department, Course, Degree, Photo, personal details. It will also show an option to edit the details. (Professors only) The option to view their sections.

Sign-up screen

It will display a form containing the following fields:

Name

Parents Names

Complete Address

Previous results

Practical - 6

Photo

Course Detail screen

This is accessible by students only, it allows the student to view the subject info and the professors who've taken the course. The screen will provide the following details:

Subject Code

Subject Name

Subject Credits

Syllabus

Prerequisites

Professors who've taken the course

Subject Department.

Registration Screen

This is the registration screen it provides the user (student) with a form to fill in their choices of subjects for the semester this also includes area for cores and electives separately to avoid confusion.

Registration Confirmation screen

The shows the student details along with the enrolled courses, as well as, the option to print it.

Lists generation and viewing screen

This shows the generated lists to the management and the professors, this includes the sectional/departmental/course wise lists for the semester.

3.2.Product Functions

The functions can be studied from the following use cases:

3.2.1. Sign-up

Firstly, the Student needs to meet the eligibility set by the university (new students) or have valid college IDs.

After successful signing up the students are redirected to their profiles.

In case of failure the student is prompted to re-enter the data/ contact admin.

1. The student is presented a form to be filled containing basic details and results.
2. The student then sets a Login and Password.
3. The form is validated, and the student is signed up.

In case of Invalid student ID/ Missing Data: student is prompted to correct the ID/ fill in data.

Practical - 6

Validity Check

The student needs to meet minimum age and marks.

The Login ID and Passwords must be valid.

Error Check

In the use case.

3.2.2. Login

Firstly, all Users must have a Login ID and passwords created for them in the system.

On successfully Logging in, the student and Professors are redirected to their profiles.

The management, Admin, and DEO are redirected to the management site.

1. The user is prompted to enter their Login and Passwords.
2. The user then enters their Login and Passwords.
3. The data is the validated and the user logged in.

In case of Invalid student ID/ Missing Data: student is prompted to correct the ID/ fill in data.

Validity Check

The Login ID and Password must match with the ones in the database.

Error Check

In the use case.

3.2.3. View Profile

Firstly, the user must have successfully logged into the system.

Upon successfully logging in, the user is presented their generated profiles with their information on it.

1. The user logs in and is the redirected to their profiles.

Validity Check

N/A.

Error Check

N/A.

Practical - 6

3.2.4. View Courses

Firstly, the user must have successfully logged into the system.

Then the user is redirected to the courses list page.

1. The user clicks on the courses menu on their profile window.
2. The user is presented with the list of courses.
3. The user then selects the course of their choice and are redirected to the course details page.

Validity Check

N/A.

Error Check

N/A.

3.2.5. Register

Firstly, the student must have successfully logged into the system.

After successfully registering for the semester the student is redirected to the confirmation windows showing the courses they've signed up for.

In case of failure the student is prompted to re-enter the data/ contact admin.

1. The student selects the REGISTER option from their profile.
2. The student is then presented with a list of courses they can take in the semester (Cores and electives)
3. The student selected the minimum number of courses required and additional (if they chose to do so up to a maximum of 6 courses).
4. The student then submits their selections.
5. The selections are validated, and the student is redirected to the confirmation page, where they can print the registration forms.

In case of a validity failure:

1. Invalid student ID/ Missing Data: student is prompted to correct the data.
2. Minimum courses not met: student is prompted to select more courses.
3. Maximum courses exceeded: Student is prompted to deselect some courses.

Practical - 6

Validity Check

The Student ID must be valid

The student must select the allowed subjects only meeting a minimum required quantity.

Error Check

In the use case.

3.2.6. Manage Database

These are administrative functions designated only to the DBA. The operation of these is as follows:

1. The Admin logs in and is the redirected to the admin window.
2. The admin then executes maintenance tasks like:
 - a. Add/Remove tables
 - b. Alter tables
 - c. Reset system.

Validity Check

All operations must be performed by DBA only, any other update is invalid.

Error Check

N/A.

3.2.7. Maintain Course Details

This is designated to the DEO; the operations are as follows:

1. The DEO logs in and is the redirected to the Maintenance window.
2. The DEO then executes required tasks like:
 - a. Add course.
 - b. Remove course.
 - c. Alter existing course.

Validity Check

All updates must be done by the DEO only, all other updates are invalid

Error Check

In the use case.

Practical - 6

3.2.8. Generate Lists

The following lists are generated:

Section wise student list for each subject.

This is assigned to the Management staff.

1. The user logs in and is the redirected to their management window.
2. The user then selects a course.
3. The user then selects the option to generate sections defining the maximum number of students per section.
4. Upon completion the user is presented the generated section list.

Student list for each course irrespective of the section

This is also assigned to the Management staff.

1. The user logs in and is the redirected to their management window.
2. The user then selects a course/department.
3. The user then selects the option to generate student list.
4. Upon completion the user is presented the generated student list.

Validity Check

The lists must only be generated by assigned Management staff.

Error Check

Generation command given before completion of registration process, reset database to clear invalid records.

3.2.9. Update Profile

This option is available to the students, professors and the DEO.

1. The user selects the update profile option from their window.
2. The user is then redirected to the updation form.
3. The user enters the updated information and submits it.
4. The data is then validated and on success the profile is updated.

The validity checks fail under the following case:

1. Protected data: On updation of fields like name, date of birth, parent names the process fails, and the user is prompted to bring proof of documents to the administration before the updation is completed by the DEO.

Practical - 6

Validity Check

The updates must only be done by Student/Professor/DEO.

Updates to protected data are invalid as mentioned in use case.

Error Check

In use case.

3.2.10. View Enrollments

To view enrollment lists, the user must have successfully logged into the system. Additionally, the list must've been generated by the management. When both the conditions are met the user is given access to the list.

The process is as follows:

1. The user selects a course.
2. The user then selects the option to view lists this can be:
3. View section list.
4. View Department list.
5. View Course list.

The process fails in the following case:

1. Lists not generated: In case the lists are not generated then the user is presented with the list not generated page, contact admin. The management is presented with the option to generate lists.

Validity Check

N/A

Error Check

In the use case.

3.3.Performance Requirements

None.

3.4.Design Constraints

None.

Practical - 6

3.5.System Attributes

3.5.1. Security

The application is password protected for all types of users, no one can access the system unless they've validated themselves.

3.5.2. maintainability

The application is designed in a maintainable manner. It will be easy to incorporate new requirements in the individual modules.

3.5.3. Portability

As the application is browser bases it is universally portable.

3.6.Logical Database Requirements

The following will be stored in the database:

Student info: Name, Parent Names, Address, Dept, S_ID, DOB and Degree.

Professor info: Name, Address, Dept, P_ID, DOB and Specializations.

Course info: C_ID, Name, Dept, Semester, Type, Credits and taken_by.

Enrollments: S_ID, C_ID, P_ID, En_ID.

Staff Info: Name, Address, Post, Dept, DOB, Emp_ID.

The table for Professor info can be merged with the Staff info table with some modifications if needed.

3.7.Other Requirements

None.

Learnings

We have made the SRS for university registration portal.