Software Requirements Specification

for

Academic Affairs System

Prepared by : Group 11

Tables of Content

1. Introduction

1. 1 Purpose

The purpose of this document is to provide a clean understanding of the requirements for the creation of the academic affairs system. This document will act as a foundation for software design, development and testing.

1.2 Intended audience and Reading Suggestions

The intended audience for this document include software developers, testers, project managers, administrators, faculty and students as well as the other stakeholders who are involved in the development, design and implementation of this system. This document will ensure that all the people involved have a clear,concise and shared understanding about the system requirements, product goals and its deliverables.

1.3 Product Scope

The purpose of the Academic Affairs System is to provide a one stop solution for students, faculty and administration to manage all the academic aspects in a university. This system will create a comprehensive centralized system through which academic activities like tracking grades, sending broadcasts etc can be done easily. This system will streamline the academic process and enhance efficiency and accuracy in academic operations.

2. Overall Description

2.1 Product Perspective

This Academic Affairs System is an essential component of any educational institution. Many institutions use multiple systems in order to cater to different users in its institutions. This system aims to provide a centralized system to cater to each and every user in the institution. This is done by creating a web application which can be accessed within the institution and which would greatly increase the efficiency and ease of doing operations.

2.2 Product Functions

As the system is designed to be able to cater to different users within the academic institution it should have the following features:

i) User Friendly Interface : The interface of the web application should be minimalistic, simple and easy to use and should be functional enough to provide detailed information in a clean manner all while being efficient.

ii) Student Management: The web application should provide the students a platform through which access to their academic information like grades, attendance etc become easily accessible. Also it should provide students with the ability to provide feedback.

iii) Faculty Management: The web application should provide the faculty members with necessary functionalities like maintaining course resources, generating statistical reports etc.

iv) Administration Management: The web application should provide the administration with tools to manage all the academic activities of the institution. This includes broadcasting important information, maintaining student and faculty information etc.

v) Security and Privacy: As the system is being used by multiple users with different authority levels it should be safe and robust enough to distinguish between the roles of different users. Also it should be secure in terms of holding personal and sensitive information.

2.3 User Classes and Characteristics

The system has 3 major users with each having different functionalities and authoritative levels:

i) <u>Student user</u>

a) Enrolled in the institution and studying certain subjects
b) Needs a simple and intuitive system to perform everyday academic activities.
c) Use the application to access academic records, course resources and fees reports.
d) May use the system to provide feedback about certain courses.

ii) <u>Faculty user</u>

a) Employed by the institution
b) Needs a simple system that allows them to quickly manage and access information.
c) Use the application in order to manage course information and resources, generate reports and access attendance information.

iii) <u>Administrative user</u>

a) Responsible for managing all academic affairs of the institution.
b) Needs a comprehensive reporting and analytic system to monitor the performances of the institution.
c) Use the application to access student and faculty information and add or remove them from the system and read feedback about different courses.
d) May use the system to broadcast info to the students and faculty.

2.4 Operating Environment

The following operating environment is used for the academic affairs system:

- No SQL Database
- Platform: HTML/CSS/Node JS
- OS independent Web Application

2.5 Assumptions and Dependencies:

For some of the functionalities in the system like adding new students/faculty to the system it is assumed that all the changes will be directly made in the database rather than using the system and the details have to be verified manually.

3. External Interface Requirements

3.1 User Interface

The system will be having the following user interfaces :-
i) Frontend - A GUI based web application which uses Html,Css and Javascript.

ii) Backend - For connecting the database to the frontend Node.JS is used and for the NoSQL database MongoDB is used.

iii) Main Lander Page: The homepage for the system for each user accessing the system will be a lander page which will have different buttons to access the login page for each user.

iv) Login Page: Each user will be having its own login page through which the user can access the system by validating the credentials.

v) Hamburger Menu Overlays: For every user, after login each page will have a hamburger menu which will be activated after clicking. Once Activated, an overlay will be displayed which will contain links to access all the different functionalities available for the current user.

vi) Logout Button: Each user will be able to easily logout from the session using the logout button provided on each and every page.

vii) Error Messages: The error messages will be popped up when the system fails to validate the user or if it fails to perform a certain function.

3.2 Hardware Interfaces

- The system is only supported on a desktop and not created for mobile devices
- In order to access the system a web browser which supports Html, Css and Javascript is required.
- The browser should support HTTPS and various other protocols.

3.3 Software Interfaces

The following Software Interfaces were used for the system:

Sr No	Technology	Reason
1	MongoDB	It's a NoSQL Database model and is very easy to use and scale
2	Html, CSS	For the Frontend we used these technologies in order to create a static website.
3	Node.JS	For connecting with the backend of the system we used Node.JS. It has a large number of frameworks to perform different functionalities.
4	Express	It's a Node.JS framework and was used to create routing between web pages.
5	Passport	A node JS library used for authentication of users

3.4 Communication Interfaces

The system only uses simple forms in order to collect and fetch data. It supports all types of web browsers
For the new users, the password is sent using the email address

4. System Features

4.1 Student Management

4.1.1 Description and Priority

Students need to access their academic records like grade, attendance. They also have to access fees reports and provide feedback. Occasionally they might receive certain important events or information via feedback.All the above function should be accessible with secure login. In the current system, Student Management has the highest priority as this system is most essential for the students.

4.1.2 Stimulus/Response Sequence

Stimulus: User wants to view their grades.
Response: System displays a page with the information about the grades of the user.

Stimulus: User wants to check their attendance in the courses
Response: System displays a page with the information about the users attendance and the total attendance in all the subjects of the current semester.

Stimulus: User wants to view and modify their personal information

Response: System displays a page with the personal information of the user and gives a button through which the user can modify the shown details.

Stimulus: User wants to generate a course report about a particular course he/she is enrolled in.
Response: System displays a course selector page through which user can choose the course he/she is enrolled in and then the system redirects to display a page with all the marks details of the user in the selected course.

Stimulus: User wants to generate transcript
Response: System displays a page with the user's transcript.

Stimulus: User wants to view all the fee receipts
Response: System displays a table with all the semesters listed and download links for all the receipts.

Stimulus: User wants to access resources of a particular course
Response: System displays a course selector page through which the user can select the course for which resources need to be accessed. Then the system redirects to a page which has download links for all the resources of the selected course.

Stimulus: User wants to view all the broadcasts.
Response: System displays a page with all the broadcasts made by the admin with the date of the broadcast.

Stimulus: User wants to share a feedback for a course
Response: The system provides a form to the user with a course selector drop down menu with the options of the user’s enrolled

course. It also provides an input text area where users can write the feedback and submit it.

Stimulus: User wants to login into the system
Response: System displays a login screen with inputs for username and password and after the user enters the details the system validates the details and let user enter the system.

Stimulus: User wants to enter as new user
Response: The system displays the user a form with input for ID and birthdate. The user enters the details and submits the form. The system sends the user a new password for accessing the system on the registered email.

4.1.3 Functional Requirements

Feature Name	Description
Login	Displays the login page, validates the user and gives the access to the system
New user	Displays a form with email and DOB, the user enters the details and the system sends new password to the user via email
Personal Information	Displays a page with the personal information of the user
Course Report	Displays a page with the course reports
Grade Tracking	Displays the grade tracking page
Transcript	Displays a page with the Transcript information
Attendance	Displays the page with the attendance details of all the subjects the user is enrolled in.

Fees Receipts	Displays the page with download links for all fees receipts
Feedback	Displays a page with a feedback form
Resources	Displays a page with the table of all resources with their download links.
Broadcasts	Displays all the broadcasts made by the admin

4.2 Faculty Management

4.2.1 Description and Priority

Faculty want to maintain their course resources as well as want to access the course statistics of the course they are teaching. Occasionally they want to check the broadcasts made by the admin. All of these operations should be performed on a secure system. In the current system the priority of the Faculty Management is Medium.

4.2.2 Stimulus/Response Sequence

Stimulus: User wants to login into the system
Response: System displays a login screen with inputs for username and password and after the user enters the details the system validates the details and lets the user enter the system.

Stimulus: User wants to view and modify their personal information
Response: System displays a page with the personal information of the user and gives a button through which the user can modify the shown details.

Stimulus: User wants to access the course reports of the subject he teaches
Response: System displays a page with a menu where the user chooses the course he wishes to find the report for. The system then redirects to display the page with all the necessary course reports.

Stimulus: User wants to maintain the course information.
Response: System displays a page with a menu where the user chooses the course he wishes to maintain the information for. The system then redirects to display the page with all the necessary resources and links.

Stimulus: User wants to view all the broadcasts.
Response: System displays a page with all the broadcasts made by the admin with the date of the broadcast.

4.2.3 Functional Requirements

Feature Name	Description
Login	Displays the login page, validates the user and gives the access to the system
Broadcasts	Displays all the broadcasts made by the admin
Maintain course information and resources	Displays a page with all the resources and their download links and a button to modify them
Personal Information	Displays a page with the personal information of the user
Course Report	Displays a page with the course reports

4.3 Admin Management

4.3.1 Description and Priority

The Administrator has to maintain the information of students and faculty currently in the institution. Also they have to view the feedback given by the students and occasionally send out important information via broadcasts. All of these functions have to be on a secure system protected via authentication. The priority of administrator management for the current system is High.

4.3.2 Stimulus/Response Sequence

Stimulus: User wants to login into the system
Response: System displays a login screen with inputs for username and password and after the user enters the details the system validates the details and lets the user enter the system.

Stimulus: User wants to access information of all the students in the institution
Response: The system prompts the user to select the batch in order to fetch the student information. The system then displays a page with the student details.

Stimulus: User wants to access information of all the faculty in the institution
Response: The system then displays a page with the faculty details.

Stimulus: User wants to see the feedback of the students for the courses.
Response: The system displays a page with all the feedbacks from the students

Stimulus: User wants to broadcast some information
Response: The system displays a page with all the broadcasts that are visible along with a form to fill out the new broadcast and send it.

4.3.3 Functional Requirements

Feature Name	Description
Login	Displays the login page, validates the user and gives the access to the system
Broadcasts	Displays all the broadcasts and gives a form to send out new broadcasts
Student info	Displays a page with all the student details
Faculty Info	Displays a page with all the faculty details
Feedback	Displays a page with all the feedbacks

5. Other Non-Functional Requirements

5.1 Performance Requirements

i) The system should be able to perform all the desired functions in a very short period of time.
ii) The system should be able to handle a large number of simultaneous users.
iii) It should be able to handle and process multiple requests simultaneously.

5.2 Security

i) The system should validate if the user is a member of the system or not as it contains personal as well as academic information.
ii) The system should be able to distinguish between different types of users.

5.3 Software Quality Attributes

i) Accessibility and Ease of Use

a) A Web application is required for the system with simple and interactive user experience for better usability.

ii) Reliability

a) In case of a server failure, the system should recover in a short span of time without losing the previously held data.

b) The System should perform all the operations as expected from them with great accuracy.

iii) Scalability

a) The system should be able to handle a large number of data and it should be able to maintain a lossless database.

iv) Maintainability

a) It should be easy for the administrator to maintain the system as it will contain a lot of database entries.

v) Testability

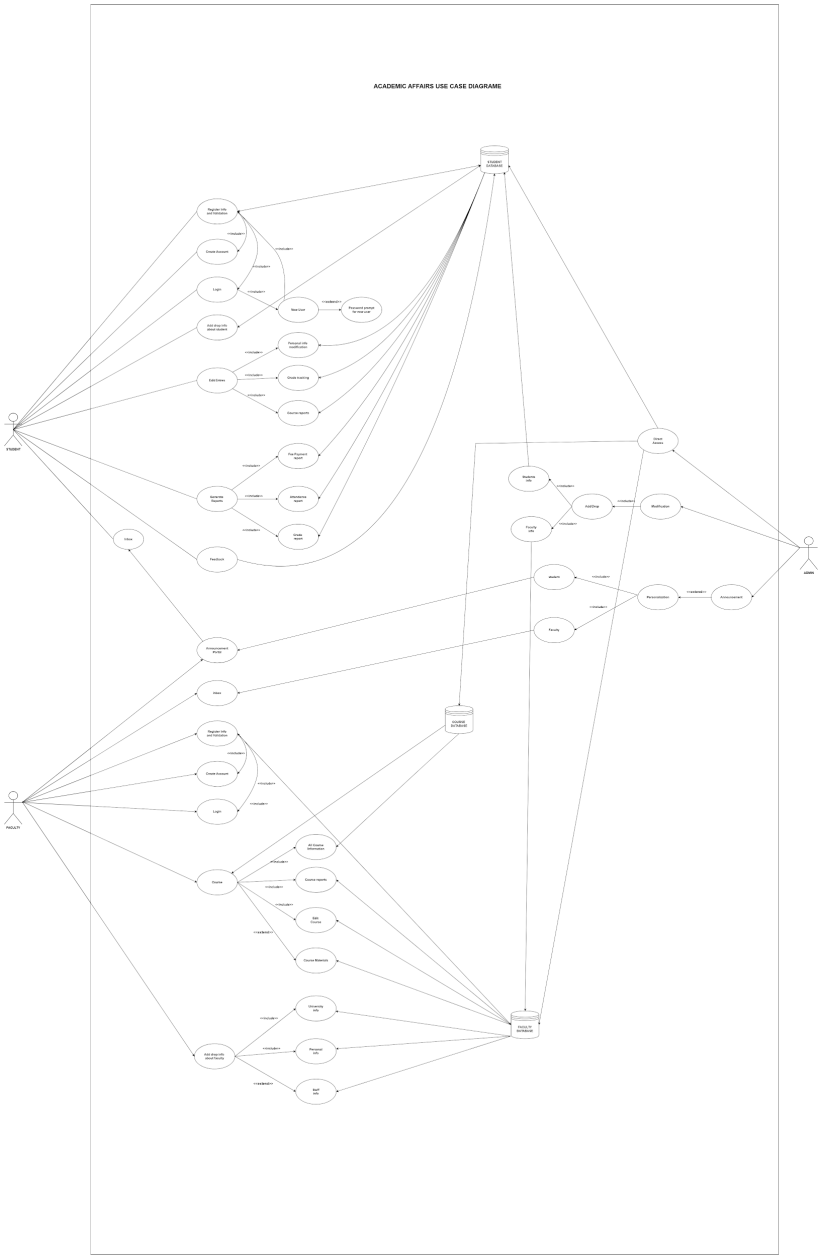
a) It should be easy to test the system and find errors in the system and fix them.

vi) Reusability

i) The code for the system should be reusable so that it can be utilized easily in future versions of the system or different systems.

APPENDIX A : ANALYSIS MODELS

A.1 Use Case Diagram



A.2 Class Diagram