Software Requirements Specification

for

Academic Information Management System

Oct 5, 2017
version 2.0

Team Members

Narosenla Longkumer	**2015165**
Rambha Sirisha	**2015203**
Devang Mehta	**2015078**
Vivek Shukla	**2015281**

Table of Contents

Revision History

Name	**Date**	**Reason For Changes**	**Version**
Software Requirements Specification for <Academic Information Management System>	September 30, 2017	Modifications in Use Cases and Fonts in the Documentation	1.0

1. INTRODUCTION

1.1 PURPOSE

The purpose of this document is to give a detailed description of the requirements for the "Academic Information Management System" module. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This system is designed to provide detailed information related to Academic programs, curriculum, and courses, Student profiles, Student Grades and CPI, class timetable and exam schedules, academic calendar, list of holidays, Senate, Senate committees, minutes of the Senate meetings.

1.2 DOCUMENT CONVENTIONS

SRS pattern used is based on the standard provided by IEEE .

For important headings, font-size: 18, font style: bold.

For important inline words, font-size: 12, font style: bold.

For other information, font-size: 12, font style: normal .

Font-family: Calibri for all text in document

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

This project is a prototype for the academic information management system and it is restricted within the college premises. This has been implemented under the guidance of college professor Dr. Atul Gupta. This project is useful for user (administrator, students and college staff), developer, tester, project manager or documentation writer that needs to understand the basic academic information system.

Here are the potential uses for each one of the reader types:

- **Developer:** The developer who wants to read, change, modify or add new requirements into the existing system.
- **User:** The user who wants to know wants to review the diagrams and the specifications presented in this document to better understand the system.
- **Tester:** The tester needs this document to validate whether all the requirements are satisfied by this system.

1.4 PROJECT SCOPE

The Objective of this system is to help users to view the academic information of academic programs, curriculum, and courses, student profiles, class timetable and exam schedules, academic calendar, list of holidays , senate minutes.

Furthermore, this system allows the academic staff members to manage time tables,academic calendar,list of holidays,exam Schedule . Allows students to view their profiles ,request for change ,view senate minutes. This system allows faculty members to mark attendance , and request for student profile as well as allows the Student Committee to manage their meetings,maintain attendance ,manage and post senate minutes.

1.5 REFERENCES

- web.iiitdmj.ac.in
- www.iiitdmj.ac.in
- UG Manual
- PG Manual
- Faculty Members
- Academic Staff Member

2. OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

This is a self contained product designed to facilitate the admin to maintain the Academic Information System. It is meant to provide an easy maintainance of the Academic Information.It helps the academic department to manage Time Table, both for the academic year as well as for examinations. The product also aids the administrator to manage issues related to attendance, academic programs and courses & curriculum. It helps in maintaining the profile of students which can be seen by the students as well as the faculty(on request). Other services provided by the software include managing the exam schedules, academic calendar, list of holidays and details related to the Senate.

2.2 PRODUCT FUNCTIONS

2.2.1. Manage Time Table

For ADMIN:

» allows the admin to create a new timetable, update an existing timetable or delete a timetable.

2.2.2. Manage Attendance

For ADMIN:

» allows the admin to manage attendance.

» aids the admin to submit attendance to the accounts section.

2.2.3. Manage Student Profile

For ADMIN:

» allows the to create a new student profile, update an existing profile or delete a student's profile.

» allows to see the profiles if all the students.

2.2.4. Manage Academic Programs

For ADMIN:

» allows the admin to add an academic program, delete an existing program or do update a program.

2.2.5. Manage Curriculum & Courses

For ADMIN:

» allows the admin to add a course, update an existing course or delete a course.

2.2.6. Manage Academic Calendar

For ADMIN:

» helps the admin to create academic calendar, update the existing calendar or delete the academic calendar.

2.2.7. Manage Holidays

For ADMIN:

» facilitates the admin to create a holiday list or update an existing list of holidays.

2.2.8. Manage Senate

For ADMIN:

» allows the admin to add a new member or remove an existing member.

» facilitates the admin to manage the attendance of all the senate members.

» helps the admin to schedule senate meetings and to see upcoming meetings.

» allows the admin to manage as well as see senate minutes and agendas.

2.2.9 Manage Student Grades And CPI:

For Admin:

- allows the admin to enter the grade details of students and calculate their cpi.
- helps him to see the cpi of the students.

2.3 USER CLASSES AND CHARACTERISTICS

- is used to specify a user.
- is used to specify the functionality corresponding to each user.

The various user classes that will use this product and their functionalities are as follows:

2.3.1- ADMINISTRATOR:

- the administrator manages the whole system.
- checks for update requests and works accordingly.
- responsible for scheduling senate meetings and storing its details.

2.3.2- STUDENT:

- a student can requests for his/her profile update.

2.4 OPERATING ENVIRONMENT

The software would work on the following desktop Browsers like Mozilla Firefox, Google Chrome, Chromium, Safari, etc.

2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS

2.5.1- DESIGN:

- Languages Used: Python, html
- Database: mySQL
- Framework: Django

2.5.2- CONSTRAINTS:

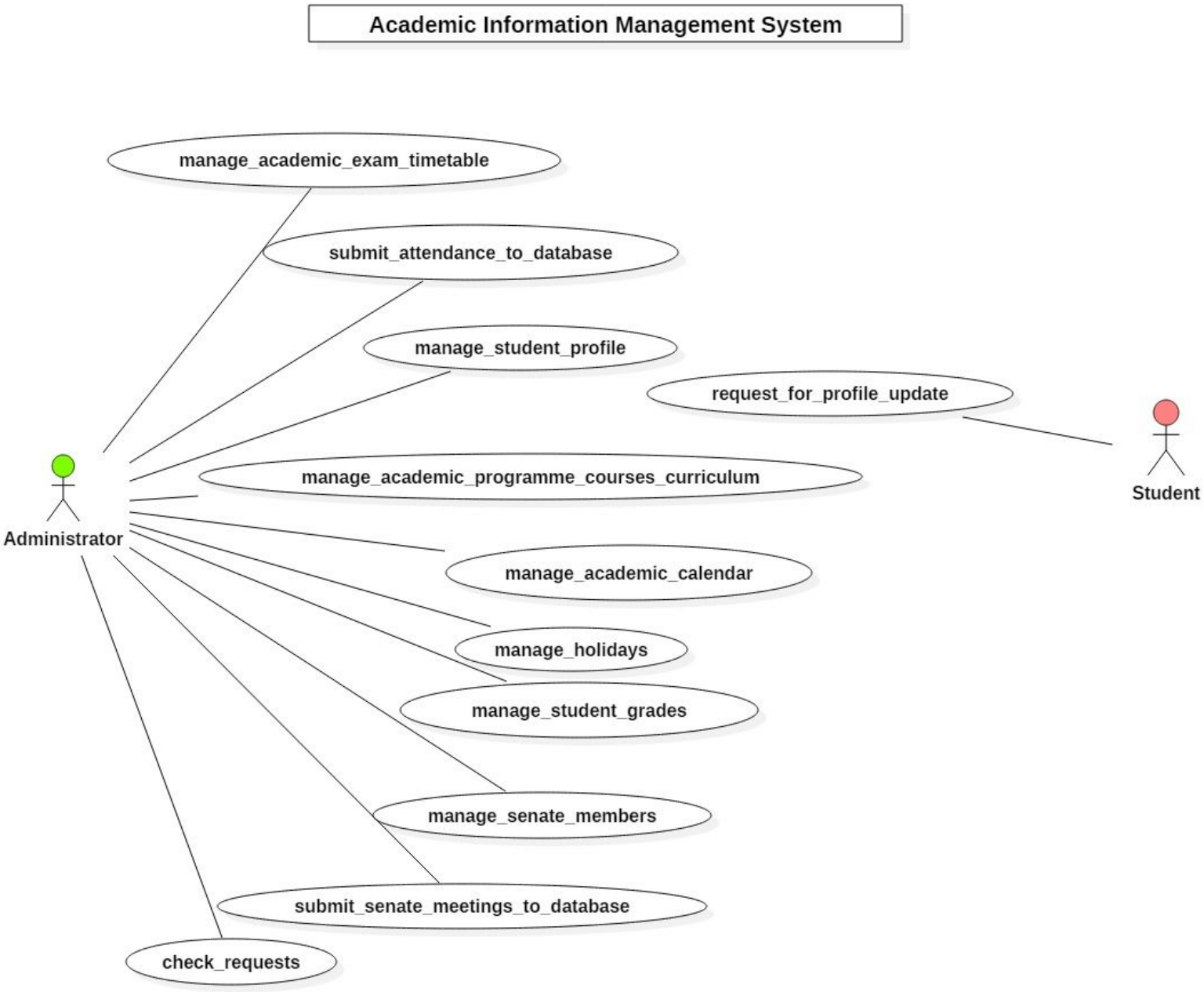
- User must authenticate herself at login

2.6 ASSUMPTIONS AND DEPENDENCIES

- the product in specifically designed for IIITDM Jabalpur.

3. SYSTEM FEATURES

3.1 USE CASE MODEL



3.2 USE CASES

UC#1 **manage_academic_exam_timetable**

UC#2 **submit_attendance_to_database**

UC#3 **manage_student_profile**

UC#4 **manage_academic_programme_courses_curriculum**

UC#5 **manage_academic_calendar**

UC#6 **manage_holidays**

UC#7 **manage_student_grades**

UC#8 **manage_senate_member**

UC#9 **submit_senate_meetings_to_database**

UC#10 **check_requests**

UC#11 **request_for_profile_update**

3.3 USE CASES DOCUMENTATIONS

3.3.1 Use Case #1

UC ID	UC #1	
Use case Name	manage_academic_exam_timetable	
Description	This use case represents the interaction between the administrator and the system in order to manage the timetable of academic as well as exams.	
Actor	Administrator	
Precondition	Administrator must be logged into the system.	
Main Flow	1	The admin selects one of the option of managing exam or academic timetable.
	2.	On both the cases, the timetables will be displayed along with the options to add,edit or delete the timetable.
	3	Based on his selection the corresponding web page appears. [S1][S2][S3]
	3	The admin is asked for confirmation.
	4	The admin clicks the confirmation button[A1].

	5	An acknowledgement message is displayed on the screen.
	6	The admin is redirected to the timetable main page.
Post Condition	The information is successfully received by the system and is reflected in the database.	
Alternate Flows	A1	The admin chooses not to confirm.
		Post-Condition : The system goes back to the main page.
Sub flow	S1	If the admin chooses the 'add timetable' option, a form is displayed . The admin fills in the form and submits it.
	S2	If the admin chooses the 'update timetable' option, a form is displayed where old contents can be edited. The admin updates it and submits it
	S3	If the admin selects the 'delete timetable' option, a list of existing timetables is displayed. The admin selects the timetables to be deleted, and clicks the 'delete' button.
Global Alternate Flow	GA1	The admin can go back to the dashboard at any time.
		Post-Condition : The system returns to the admin dashboard.

3.3.2 Use Case #2

UC ID	UC #2	
Use case Name	submit_attendance_to_database	
Description	This use case facilitates the interaction between the admin and the system in order to submit the marked attendance.	
Actor	Admin	
Precondition	Admin must be logged into the system.	
Main Flow	1	The Admin clicks on 'submit attendance' button.
	2	The system displays a list of current batches for which the attendance can be managed.
	3	The admin selects the batch for which the attendance has to be filled in.
	4	The system displays a form having a list of roll number of the students enrolled in that particular batch.
	5	The admin enters the attendance details into the form.
	6	The admin clicks the 'submit' button.

	7	The admin is asked for confirmation.
	8	The admin clicks the confirmation button[A1].
	9	An acknowledgement message is displayed.
	10	The system returns to the admin dashboard.
Post Condition	The information is successfully received by the system and is reflected in the database.	
Alternate Flows	A1	The admin chooses not to confirm.
		Post-condition : The system goes back to the submit page.
Sub flow	NIL	
Global Alternate Flow	GA1	The admin can go back to the dashboard at any time..
		Post-condition : The system returns to the admin dashboard.

3.3.3 Use Case #3

UC ID	UC #3	
Use case Name	manage_student_profile	
Description	This use case facilitates the interaction between the admin and the system in order to manage the profiles of students.	
Actor	Admin	
Precondition	Admin must be logged into the system.	
Main Flow	1	The admin selects one of the options from add student / delete student / update student profile.
	2	Based on his selection, the corresponding webpage is opened .[S1][S2][S3]
	3	The admin is asked for confirmation.
	4	The admin clicks the confirmation button . [A1]
	5	An acknowledgement message is displayed on the screen.
	6	The system returns to the admin dashboard.
Post Condition	The information is successfully received by the system and is reflected in the database.	

Alternate Flows	A1	The admin chooses not to confirm.
		Post-condition : The system goes back to the submit page.
	A2	No student matches the filled details in the database.
		Post-Condition: The System shows the admin a message ‘student not found’.
Sub flow	S1	If the admin selects ‘add student’ option, a form is displayed. The admin fills in the details of the student to be added. The admin clicks the 'submit' button.
	S2	If the admin selects the ‘update’ button, he searches for the student based on his roll number, a form will be displayed and the admin enters the details to be updated. The admin clicks the 'submit' button.[A2]
	S3	If the admin selects the ‘delete student’ option, a form will appear asking the roll number and name of the student to be deleted. The admin enters the details clicks the 'submit' button.[A2]
Global Alternate Flow	GA1	The admin can go back to the dashboard at any time.
		Post-condition : The system returns to the admin dashboard.

3.3.4 Use Case #4

UC ID	UC #4	
Use case Name	manage_academic_programme_courses_curriculum	
Description	This use case represents the interaction of admin with the system to manage academic programs, courses and curriculum. For this, admin is provided with option of edit, add ,delete and view each academic program. This use case shows how system assists admin to edit, add, delete academic programs.	
Actor	Admin	
Precondition	Admin must be logged in the system.	
Main Flow	1	The admin selects manage academic programs out of the list of choices on dashboard and directed to page showing academic programs.
	2	Admin is provided with 4 options : add program, edit program, delete program, view program details Along with the list of the programs..
	3	Admin selects edit academic program, and selects an academic program[S1][S2] .A new page appears showing all disciplines of that program. Option of edit, add, delete for each discipline is provided.
	4	Academic selects a particular discipline and clicks edit [S3][S4]. A new page appears showing various courses in that discipline semester wise appears. Admin is provided with four option – add course, delete course, edit course , view course curriculum.
	5	Admin selects edit option for a particular course [S5][S6][A2] and directed to form page of that course with old course details written.
	6	Admin modifies any of field and clicks save changes button [A1] .
	7	Admin gets acknowledgement message and redirected to course page.

Post Condition	All changes made by admin are saved , database updated and admin is redirected to dashboard.	
Alternate Flows	A1	Admin clicks cancel button .
		Post-condition : The admin remains on that page only and system does not saves changes in the database.
	A2	Admin clicks view courses/curriculum.
		Post-condition : Admin directed to page showing details of selected course .
Sub flow	S1(a)	Admin selects add program, and a page appears having fields for entering program details.
	S1(b)	Admin fills details and clicks Submit. [A1]
		Post-condition :Admin gets acknowledgement message and redirected to academic program page.
	S2(a)	Admin selects delete option for a particular academic program . A warning message pops up .Two options are provided .Admin selects yes .[A1]
		Post-condition : Admin gets acknowledgement message and is directed to main academic program page showing list of academic programs.
	S3(a)	Admin selects add academic discipline, and a page appears having fields for entering program details.
	S3(b)	Admin fills details and clicks Submit. [A1]
		Post-condition :Admin gets acknowledgement message and redirected to page showing all disciplines of that academic program user selected.

	S4(a)	Admin selects delete option for a particular academic discipline . A warning message pops up .Two options are provided .Admin selects yes .[A1]
		Post-condition : Admin gets acknowledgement message and redirected to page showing all disciplines of that academic program user selected.
	S5(a)	Admin selects add course [A2] and a form appears asking for details of new course such as course code, course title, contact hours, credit, prerequisites, semester, evaluation scheme, brief details of course content ,and text/reference books
	S5(b)	Admin fills the above information and clicks submit button.[A1]
		Postcondition : Admin gets acknowledgement message and redirected to course page.
	S6(a)	Admin clicks delete option for a particular course.[A2]
	S6(b)	Admin gets warning message and admin selects "yes , i understand and i wish to continue with my choice" .[A1]
		Post-condition : Admin is directed to course page showing various course of that discipline semester wise.
Global Alternate Flow	GA1	The Admin can 'cancel' the procedure at any time by exercising such an option
		Post-condition : The system returns to the employee 'Dashboard' – initial screen.

3.3.5 Use Case #5

UC ID	UC #5	
Use case Name	manage_academic_calendar	
Description	This use case involves the management of the college's academic calendar. It includes editing, deletion, viewing and creation of the calendar by the administrator in charge.	
Main Flow	1	The admin selects the academic calendar option out of the list of options mentioned on dashboard and gets directed to the academic calendar page.
	2	The admin can view the current academic calendar, along with edit or delete buttons adjacent to the days and a create button in the corner of the page. The admin manages the calendar according to the options. [S1-S3]
	3	After the following tasks, a final confirmation dialog box appears and the admin click on OK. [A1]
	4	Admin gets acknowledgement message and redirected to academic calendar page and see the new changes.
Post Condition	All changes made by admin are saved and the database is updated.	
Alternate Flows	A1	Admin clicks cancel button.
		Post-condition: The admin remains on the main page of the academic calendar without any changes.
Sub flow	S1	When the edit option is selected, a dialog box appears asking for the changes of the selected day(s). The admin makes the following changes and presses save.[A1]
	S2	When the delete option is selected, a dialog box appears asking if we want to delete the info of the selected day(s.[A1]
	S2	When the create option is selected, a dialog box appears with a form to make the new day(s). The admin makes it and presses create.[A1]

Global Alternate Subflow	GA1	The admin can go back to the dashboard at any time
		Post-condition : The system returns to the admin dashboard.

3.3.6 Use Case #6

UC ID	UC #6	
Use case Name	manage_holidays	
Description	This use case involves the management of the college's holiday list. It includes editing, deletion, viewing and creation of the restricted and unrestricted holidays by the administrator incharge.	
Actor	Administrator	
Precondition	The administrator is authenticated.	
Main Flow	1	The admin selects the holidays option out of the list of options mentioned on dashboard and gets directed to the holiday page.
	2	The admin can view the restricted and unrestricted holidays, along with edit or delete buttons adjacent to the days and a create button in the corner of the page. The admin manages the holidays according to the options. [S1-S3]
	3	After the following tasks, a final confirmation dialog box appears and the admin click on OK. [A1]
	4	Admin gets acknowledgement message and redirected to academic calendar page and see the new changes.
Post Condition	All changes made by admin are saved and the database is updated.	
Alternate Flows	A1	Admin clicks cancel button.

		Post-condition: The admin remains on the holiday page without any changes.
Sub flow	S1	When the edit option is selected, a dialog box appears asking for the changes of the selected day(s). The admin makes the following changes and presses save.[A1]
	S2	When the delete option is selected, a dialog box appears asking if we want to delete the info of the selected day(s).[A1]
	S3	When the create option is selected, a dialog box appears with a form to make the new day(s). The admin makes it and presses create.[A1]
Global Alternate Subflow	GA1	The admin can go back to the dashboard at any time
		Post-condition : The system returns to the admin dashboard.

3.3.7 Use Case #7

UC ID	UC #7	
Use case Name	manage_student_grades	
Description	This use case involves the management of the student's grades, spis and cpi who are studying in the college. It includes editing and viewing of the grades, spis and cpi by the administrator incharge.	
Actor	Administrator	
Precondition	The administrator is authenticated.	
Main Flow	1	The admin selects the student's grades and cpi option out of the list of options mentioned on dashboard and gets directed to their page.
	2	The admin searches for the particular student's grade according to the roll no.[A1] The admin finds it and selects the student.

	3	The admin can view the respective student's grade sheet. There is option to edit it. The admin manages the grade sheet according to the option. [S1]
	4	After the following tasks, a final confirmation dialog box appears and the admin click on OK. [A1]
	5	Admin gets acknowledgement message and redirected to the student's grade sheet with the new changes.
Post Condition	All changes made by admin are saved and the database is updated.	
Alternate Flows	A1	Admin clicks cancel button.
		Post-condition: The admin remains on the student grade sheet page without any changes.
Sub flow	S1	When the edit option is selected, a dialog appears where the grades, spis and cpi is edited. The admin makes the following changes and presses save.[A1]
Global Alternate Subflow	GA1	The admin can go back to the dashboard at any time
		Post-condition : The system returns to the admin dashboard.

3.3.8 Use Case #8

UC ID	UC #8
Use case Name	manage_senate_members
Description	This use case represent the interaction of an admin with the system to manage senate's information. The admin removes, edit and cane view the senate datas.
Actor	Administrator

Precondition	Admin must be logged in the system.	
Main Flow	1	The admin selects manage senate members out of the list of choices on dashboard and directed to page showing all the senate members.
	2	Admin views all senate members and Admin is provided with 3 options : add senate member, edit senate member, delete senate member. Admin selects any of the options[S1][S2][S3]
	3	The admin clicks OK/SAVE. [A1]
	4	Admin gets another confirmation request. The admin clicks YES. [A1]
	5	The admin is redirected to the senate page with the updated content.
Post Condition	All changes made by admin are saved , database updated and admin is redirected to dashboard.	
Alternate Flows	A1	Admin clicks cancel button .
		Post-condition : The admin goes back to the main senate page and system does not saves changes in the database.
Sub flow	**S1**. When admin selects add senate members option, a form will be displayed and the admin enter the details of the new member.	
	S2. When admin selects edit senate members option, a dialog box for editing details of selected senate members will be displayed. The admins edits the datas.	
	S3. When admin selects delete senate members option, a dialog box will appear confirming the deletion.	
Global Alternate Flow	GA1	The Admin can 'cancel' the procedure at any time by exercising such an option
		Post-condition : The system returns to the admin 'Dashboard' – initial screen.

3.3.9 Use Case #9

UC ID	UC #9	
Use case Name	submit_senate_meetings_to_database	
Description	This use case represent the interaction of an admin with the system to submit the details of senate meetings and its agendas and minutes. The admin can also edit the details.	
Actor	Admin	
Precondition	Admin must be logged in the system.	
Main Flow	1	The admin selects submit senate meetings details out of the list of choices on dashboard and directed to page showing senate meetings along with their minutes and agenda
	2	The admin get displayed with options submit meeting,delete meeting and edit meetings. [S1][S2][S3]
	3	The admin clicks save/ok button [A1]
	4	The system ask for a confirmation
	5	The employee confirm for submission [A1]
	6	Admin gets acknowledgement message
	7	The system is directed to admin dash -board
Post Condition	All changes made by admin are saved , database updated and admin is redirected to dashboard.	
Alternate Flows	A1	Admin clicks cancel button .
		Post-condition : The admin remains on the main page and system does not saves changes in the database.

Sub flows	S1. When admin selects submit meeting option he will be displayed a form to fill about details about the meeting and for uploading minutes,agenda and attendance of that meeting. S2. When admin selects delete meeting option the will be asked to confirm it. S3. When the admin selects the edit meeting options, a form will come containing the details of that meeting in edit mode. The admin edits those datas.	
Global Alternate Flow	GA1	The Admin can 'cancel' the procedure at any time by exercising such an option
		Post-condition : The system returns to the admin 'Dashboard' – initial screen.

3.3.10 Use Case #10

UC ID	UC #10	
Use case Name	check_requests	
Description	The use cases lets the administrator check the request of updating the institute user's profile sent by the them.	
Actor	Administrator	
Pre condition	The administrator is authenticated.	
Main Flow	1	The admin clicks on the option "Check requests" on the dashboard.
	2	A list of requests will be displayed. [A1] The admin selects any one of them.
	3	The admin will check the following request. If the conditions satisfy, he will click on update button. [A2].
	4	Another dialog box will appear asking for final confirmation. The admin selects OK. [A2]

	5	Admin gets acknowledgement message of successfully updated and redirected to the dashboard.
Post Condition	All changes made by admin are saved and the database is updated.	
Alternate Flows	A1	There are no requests pending.
		Post-condition: The admin remains on the dashboard but the list will be empty.
	A2	The admin presses cancel
		Post-condition: The admin remains on the dashboard with no changes.
Sub flow	NIL	
Global Alternate Flow	GA1	The admin can go back to the home dashboard anytime.
		Post-condition: The admin remains on the initial dashboard with no changes.

3.3.11 Use Case #11

UC ID	UC #11
Use case Name	request_for_profile_update
Description	This use case facilitates the interaction between the institute user and the administrator in order to request for an updation in the user's profile.
Actor	Institute User
Precondition	The institute user must be logged in to the system.
Main Flow	1. The user clicks on the 'request for profile update' button.
	2. An Update form is displayed on the screen.
	3. The user enters his new details into the form.
	4. The user clicks the 'submit' button.
	5. The user is asked for confirmation.
	6. The user clicks the confirmation button. [A1]
	7. A message is shown that the request for updation is sent to the administrator.
	8. The user is redirected to the profile page.

Post Condition	The user's request is successfully sent to the administrator and is reflected in the database.	
Alternate Flow	A1.	The student chooses not to confirm.
	Post-Condition: The system display the form with the data filled in so far.	
Sub flow	NIL	
Global Alternate Flow	GA1	The student can 'cancel' the procedure at any time by exercising such an option
		Post-condition : The system returns to the student's home page.

4. OTHER NONFUNCTIONAL REQUIREMENTS

4.1 PERFORMANCE REQUIREMENTS

Response Time

The system shall take as less time as possible to provide service to the administrator/user.Average response time shall be less than 3 seconds.

Throughput

It should process 500 requests/queries per 5 minutes.

Recovery Time

In case of failure,redundant system shall resume operations within 30 seconds. Average repair time shall be less than 1 hour.

Start-up/shutdown time

The system shall be operational within 1 minute of starting-up.

Capacity

The system should handle the simultaneous usage of almost 1000 users at a time.

Utilization of Resources

The system shall store in the database no more than one million.If the database grows over the limit, old data should be backed up and deleted from the operational database.

4.2 SAFETY REQUIREMENTS

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

4.3 SECURITY REQUIREMENTS

Firewall Protection

The academic information management system shall run inside a firewall.

Support different roles

The system shall support different roles for users, such as students, administrative staff, faculty, the user logged in with given role should only be allowed access consistent with that role.

4.4 SOFTWARE QUALITY ATTRIBUTES

↠ Easy to learn how to operate.

Adaptability

Can be used on any browser on any operating system. It can be applied for time table of any organization.

Availability

Based on the network connection of browser and availability of databases.

Correctness:

Always return valid results and does not allow clashes while adding or updating time slots.

Reliability

It always provides correct data and results. Only authorized personnel are allowed to update the time table.

Robustness

Calculation of valid output based on the input query of user.

Usability

The system shall allow the users to access the system from the Internet using HTML or its derivative technologies. The system uses a web browser as an interface. Since all users are familiar with the general usage of browsers, no specific training isrequired. The system is user friendly and self-explanatory.

Scalability

Scaling the system to large number of users.

5. OTHER REQUIREMENTS

Platform/Browser independence

The system should be able to work on any of the modern browsers like Firefox / Explorer / Opera /Chrome, and any of the common Operating Systems like Linux, Windows and Mac OS.

Backup

There should be an easy back-up feature for the entire data, to prevent losing any data.

Data migration

There should be an easy way to migrate data from the current system to a new system.