

CSL740 Software Engineering Course, IIT Delhi



Software Requirements Specification

Submitted By

Abhishek Srivastava (2011EEY7511)
Anil Kumar (2009CS10180)
Jagjeet Singh Dhaliwal (2008CS50212)
Ierum Shanaya (2010SIY7570)

Supervised By

Dr. S. C. Gupta
National Informatics Centre India

This document gives a detailed requirement specification for a University academic portal. The document follows the IEEE 830 format.

Table of Contents

Table of Figure(s).....	4
1. Introduction	5
1.1 Purpose	5
1.2 Scope.....	5
1.3 Definitions, Acronyms and Abbreviations.....	5
1.4 References.....	6
1.5 Overview	6
2. Overall Description	6
2.1 Product Perspective	6
2.1.1 System Interface	6
2.1.2 User interface	6
2.1.3 Hardware Interface	6
2.1.4 Software Interface	7
2.1.5 Communication Interfaces	7
2.1.6 Memory Constraints	7
2.1.7 Operations	7
2.1.8 Site Adaption Requirements.....	7
2.2 Product functions	8
2.2.1 Context Diagram	8
2.2.2 Use Case Diagrams.....	8
2.2.3 Use case descriptions /Introductions.....	13
2.3 User Characteristics	16
2.3.1 Students.....	16
2.3.2 Professors	16
2.3.3 System Administrators	16
2.4 Constraints	16
2.4.1 User Interface Constraints.....	16
2.4.2 Hardware Constraints	17
2.4.3 Software Constraints.....	17
2.4.4 Data Management Constraints.....	17
2.4.5 Operational Constraints	17

2.4.6 Site Adaptation Constraints.....	17
2.4.7 Design Standards Compliance	17
2.5 Assumptions and dependencies	17
2.6 Apportioning of Requirements	17
Integration of LDAP login might be apportioned to future versions.....	17
3. Specific Requirements	18
3.1 External interface.....	18
3.1.1 Web Server	18
3.1.2 PHP Application	18
3.1.3 MySQL Database	18
3.2 Functional Requirements	18
3.2.1 Use Case Scenario	18
3.3 Performance Requirements	25
3.4 Logical database requirements	25
3.5 Design Constraints	25
3.6 Software System Attributes	25
3.6.1 Reliability	25
3.6.2 Availability	25
3.6.3 Security.....	26
3.6.4 Maintainability.....	26
3.6.5 Portability	26

Table of Figure(s)

Figure 1: Context Diagram	Error! Bookmark not defined.
Figure 2: User Login	8
Figure 3: Discussion Threads.....	9
Figure 4: Content Sharing(Upload).....	9
Figure 5: Search	10
Figure 6: Blogging.....	10
Figure 7: Notification	11
Figure 8: Survey.....	11
Figure 9: Assignment Submissions	12
Figure 10: Common Links.....	12
Figure 11: Process flow diagram of content upload.....	20
Figure 12: Process flow diagram: Blogging.....	22
Figure 13: Process Flow Diagram- Survey.....	24

1. Introduction

1.1 Purpose

The purpose of this document is to capture, in natural language and at a functional level, the description and requirements of an academic portal for any University. The focus here is the IIT Delhi's educational needs. This is a functional description of those features required to address current educational requirements. A short discussion accompanies each requirement, to add the background and framework necessary to explain the functionality. It also describes nonfunctional requirements and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.2 Scope

This software system will be an online academic portal for any university wishing to manage their academic needs online. More specifically to design and develop a simple and intuitive system which shall cater the academic needs of any institute. The system shall provide features to the user of an educational institute to be part of different types of groups, share data and documents amongst different users, have discussion threads, instant notifications via email/message, assignment submissions, blogs, share old papers, availability of books and other useful stuff (buy/sell), online survey etc

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Academic Portal

Academic Portal is software meant to help facilitate electronic classroom management and provide for electronic grading, assignment submission, discussion, and other learning tools.

1.3.2 Course Administrator

Course administrator is a user who can be a professor or system administrator and who has the ability to manage courses and course pages.

1.3.3 Academic Groups

An academic group is a set of users which can belong to the same course, same department, and same year.

1.3.4 System

System refers to the existing academic portal or a new academic portal.

1.4 References

1. IEEE 830 Template
2. Moodle Requirements Brainstorming.pdf
3. University of Melbourne – Student Portal (Getting Started Guide)
4. Stakeholder Requirements for Institutional Portals by Liz Pearce, Leona Carpenter, Ruth Martin
5. Detlor, B. (2000). "The corporate portal as information infrastructure: Towards a framework for portal design", International Journal of Information Management, 20(2) 91-101.

1.5 Overview

This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more refined. This information will serve as a framework for the current definition and future evolution of the University Academic Portal.

2. Overall Description

2.1 Product Perspective

(STUB) is meant to serve as a common platform where management of everyday academic tasks can be carried out conveniently. Our goal is to develop a replacement to the academic portal used at IIT Delhi making it more users friendly and to promote academic networking among the users.

2.1.1 System Interface

Apache will be used as web server. The user inputs data via the web server using HTML forms. The actual program that will perform the operations is written in PHP.

2.1.2 User interface

The new system shall provide a very intuitive and simple interface to the user and the administrator, so that the user can easily navigate through pages, assignments, groups and sub- groups, start discussion threads, blogs, survey, upload assignments, share data, old papers sharing and the administrator can easily manage groups and revoke user permissions.

2.1.3 Hardware Interface

a) Server side

The web application will be hosted on a web server which is listening on the web standard port, port 80.

b) Client side

Monitor screen – the software shall display information to the user via the monitor screen

Mouse – the software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, command buttons and select options from menus.

Keyboard – the software shall interact with the keystrokes of the keyboard. The keyboard will input data into the active area of the database.

2.1.4 Software Interface

a) Server side

An Apache web server will accept all requests from the client and forward it accordingly. A database will be hosted centrally using MySQL.

b) Client side

An OS which is capable of running a modern web browser which supports JavaScript and HTML5.

2.1.5 Communication Interfaces

The HTTP or HTTPS protocol(s) will be used to facilitate communication between the client and server.

2.1.6 Memory Constraints

Memory constraints will come into play when the size of MySQL grows to a considerable size.

2.1.7 Operations

The product shall have operations to protect the database from being corrupted or accidentally altered during a system failure.

2.1.8 Site Adaption Requirements

Not applicable

2.2 Product functions

2.2.1 Context Diagram

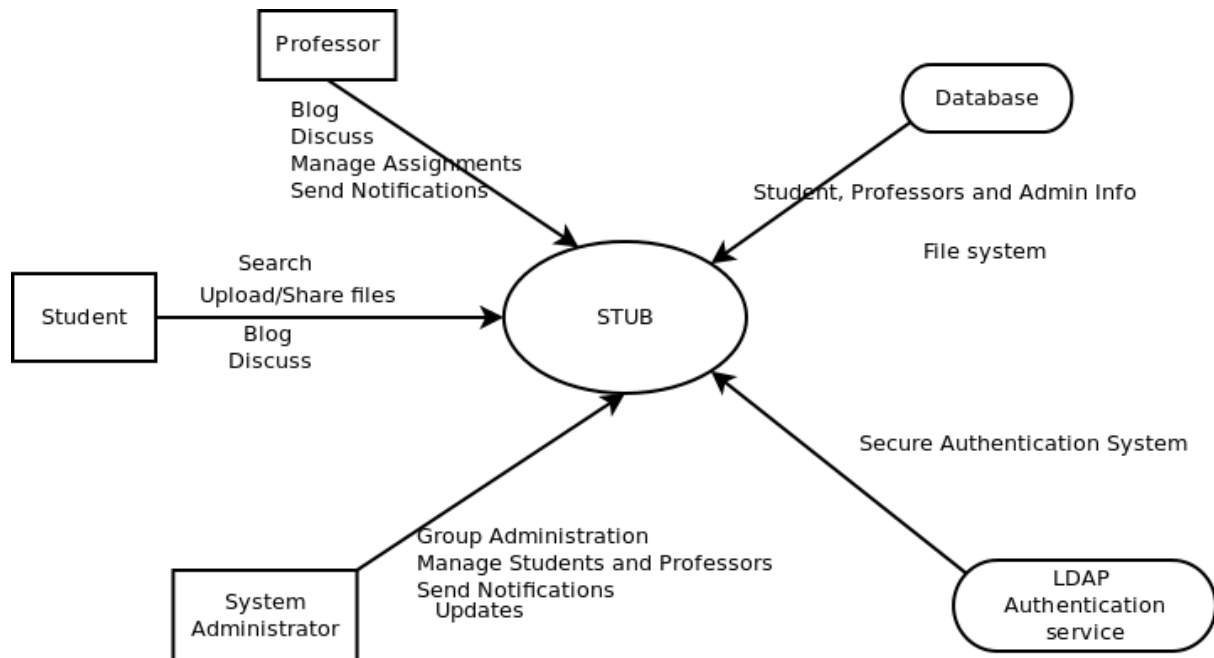


Figure 1: Context Diagram

2.2.2 Use Case Diagrams

2.2.2.1 User Login

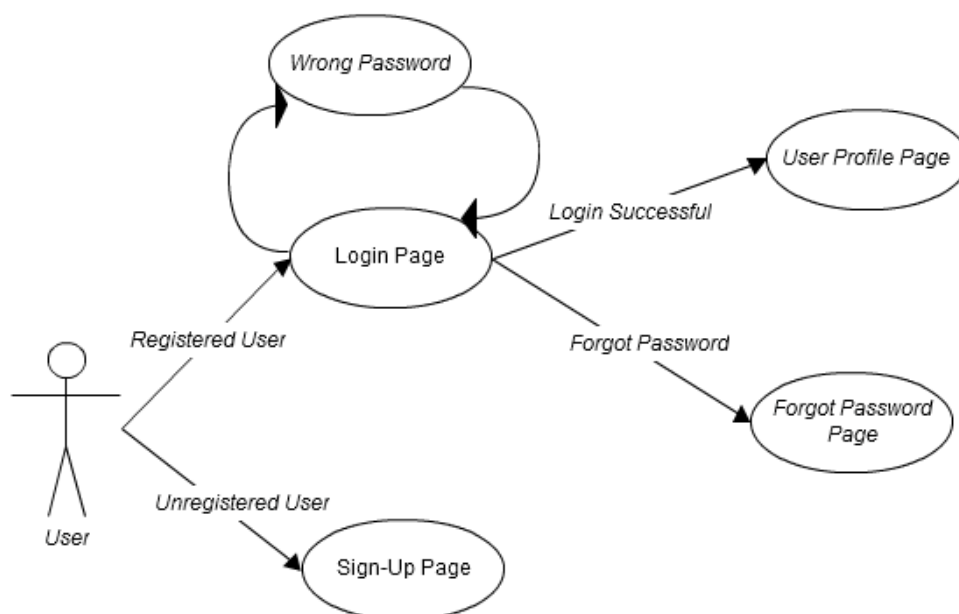


Figure 1: User Login

2.2.2.2 Discussion Threads

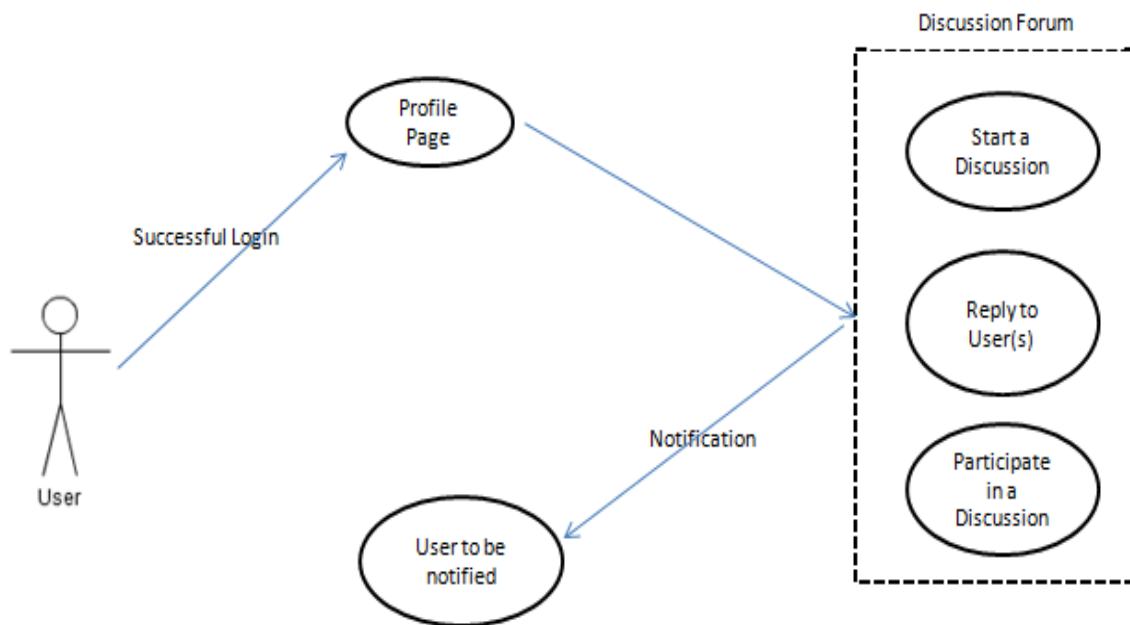


Figure 2: Discussion Threads

2.2.2.3 Content Sharing (file upload/ Old paper sharing)

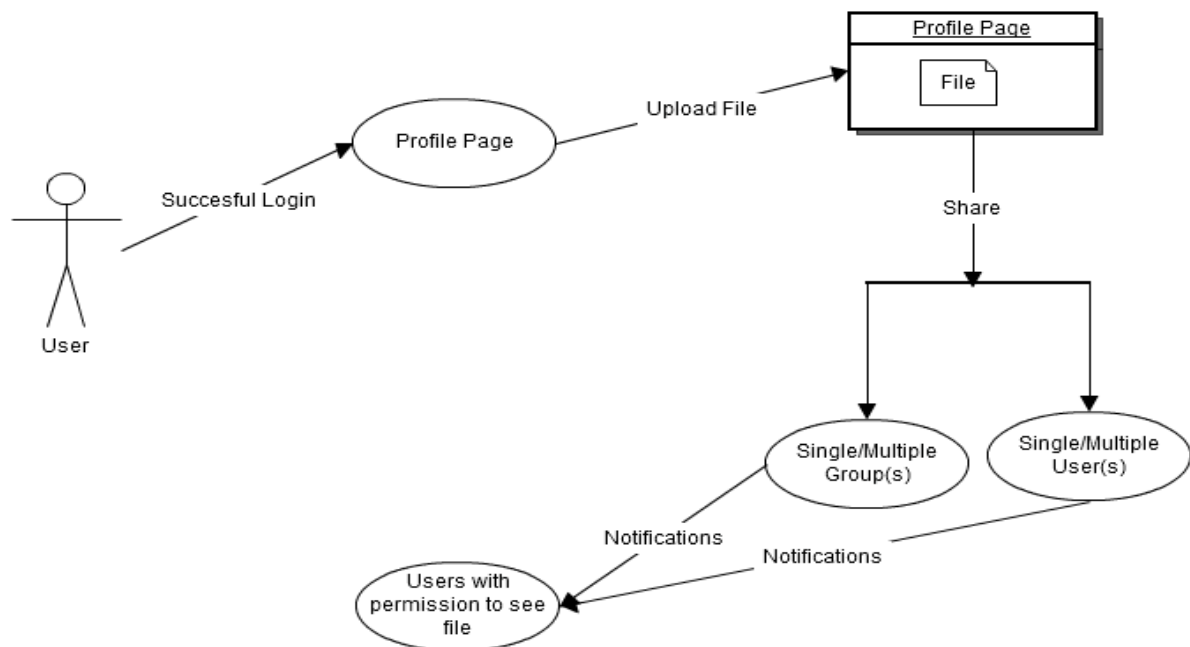


Figure 3: Content Sharing(Upload)

2.2.2.4 Search

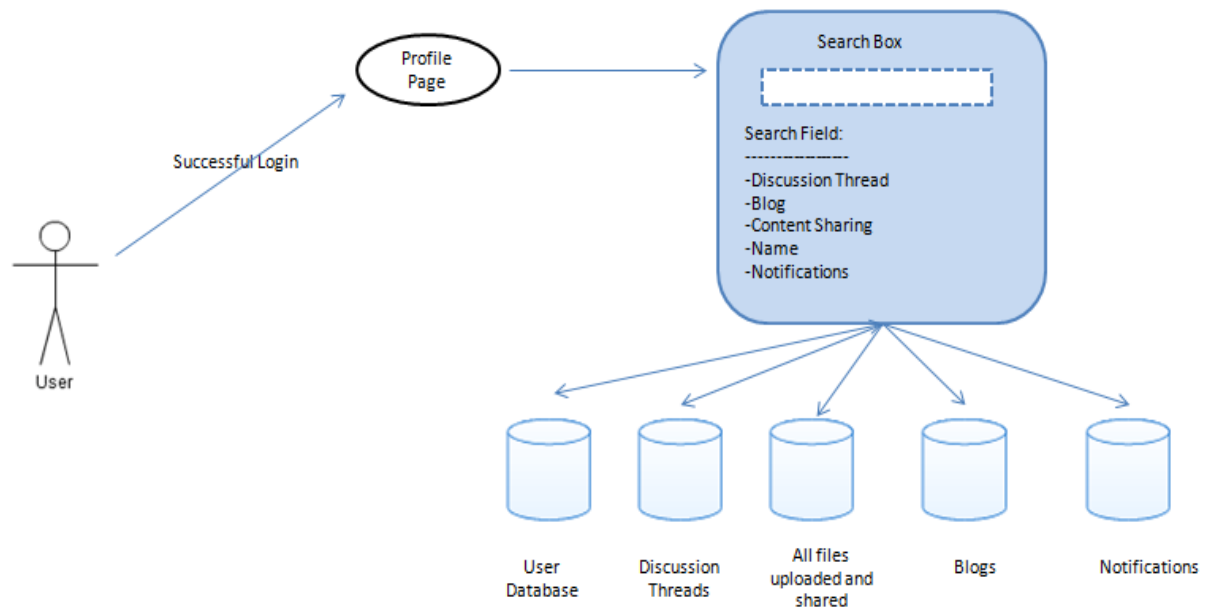


Figure 4: Search

2.2.2.5 Blogging

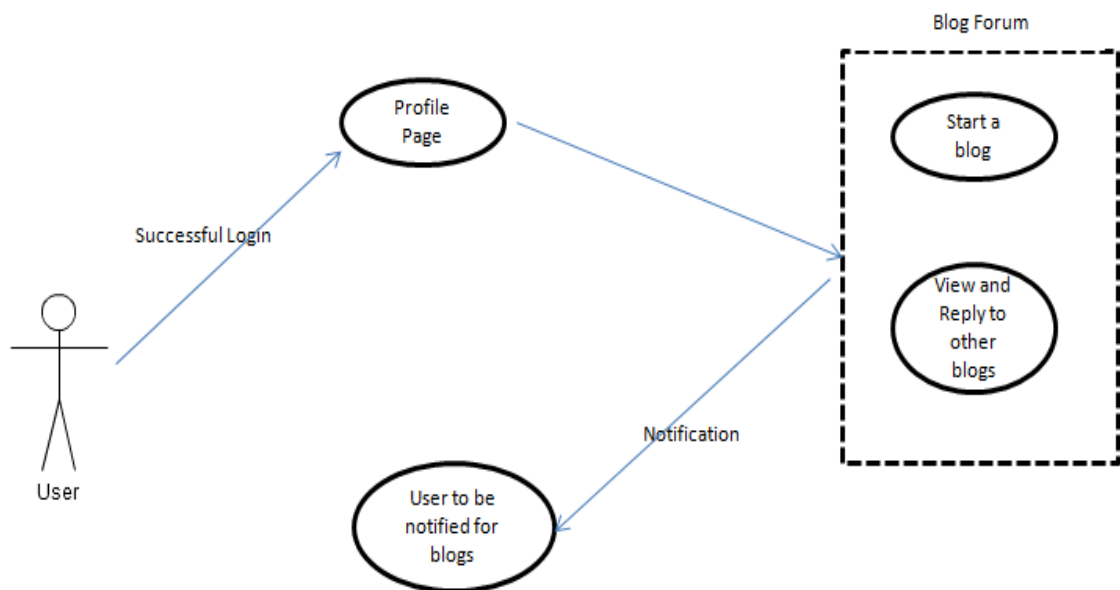


Figure 5: Blogging

2.2.2.6 Notifications

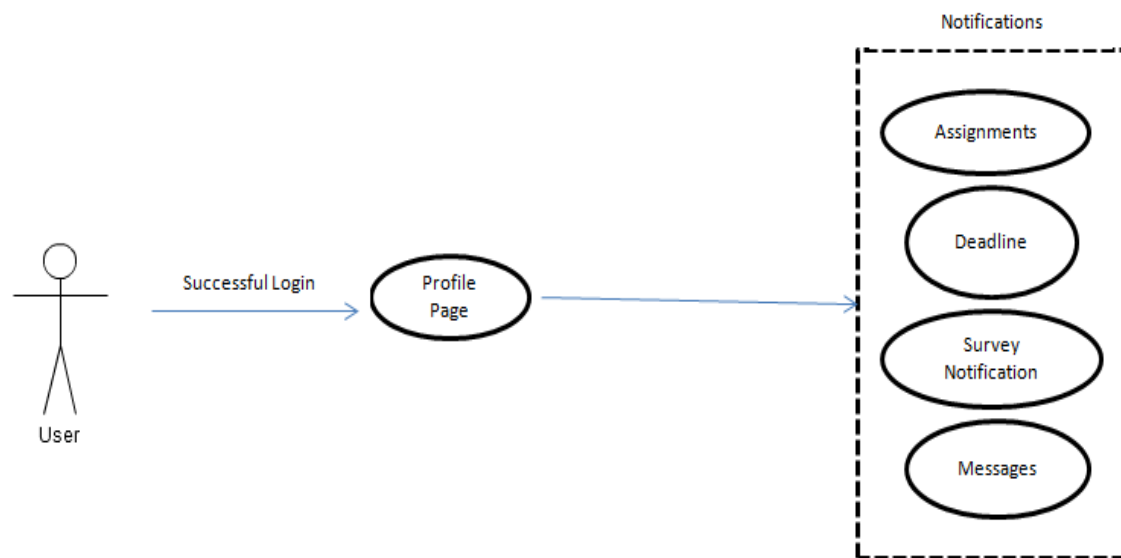


Figure 6: Notification

2.2.2.7 Survey

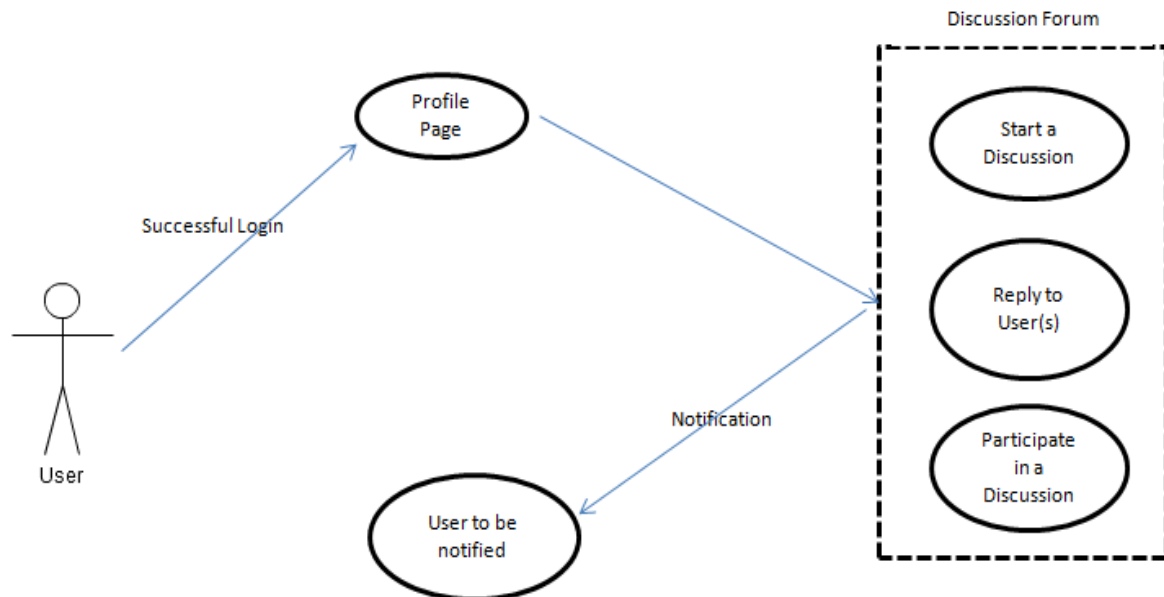


Figure 7: Survey

2.2.2.8 Assignment Submissions

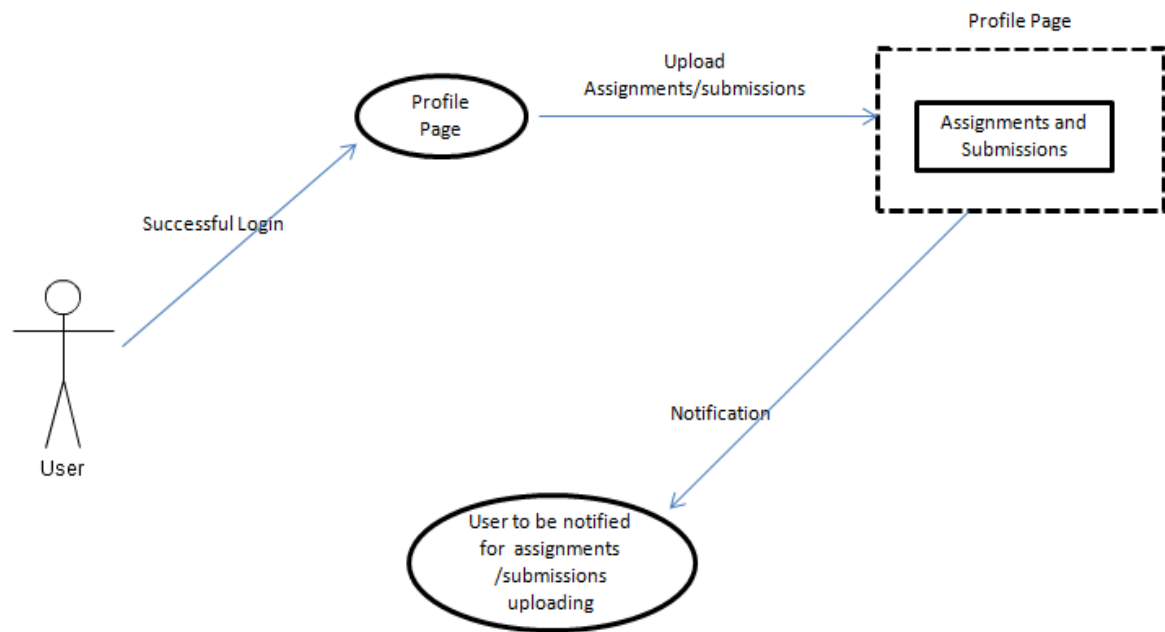


Figure 8: Assignment Submissions

2.2.2.9 Common Links

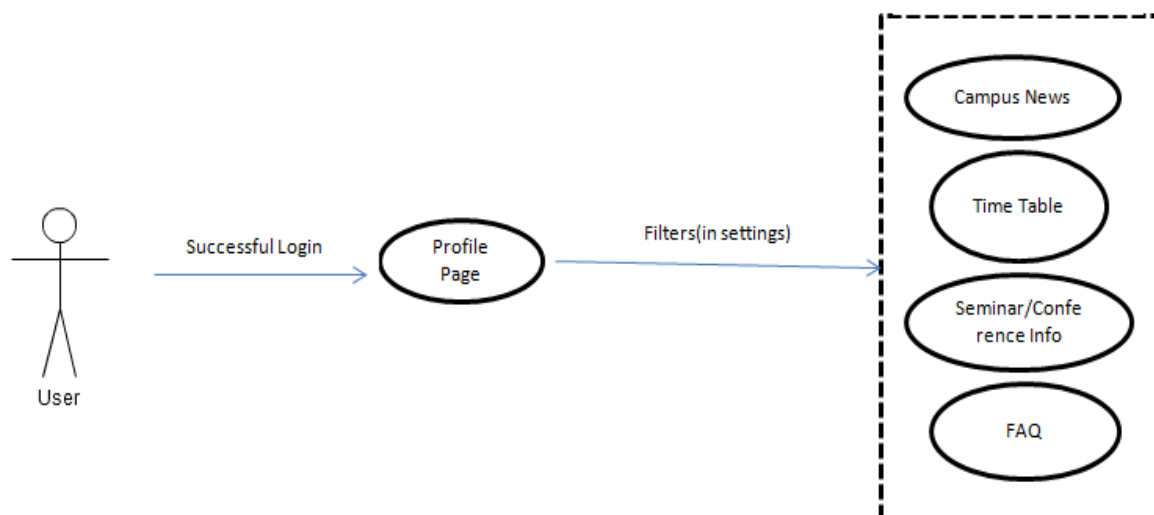


Figure 9: Common Links

2.2.3 Use case descriptions /Introductions

2.2.3.1 Groups

The system shall provide the administrator ability to create and manage new groups of different kinds like a course group or a department group. Here a user can be part of multiple groups.

Course Group

Students belonging to same course (e.g. Database Course) along with the professors and teaching assistant of the course will be part of the particular course group. Here the professors along with the teaching assistants (*conditional) will be the course administrators with the students being the users of the group.

Department Group

Students belonging to same department (e.g. Computer Science, Electricals) along with the professors of the department will be part of the particular department group. Here an assigned professor will be the course administrator with the students and the other professors being the users of the group.

2.2.3.2 Discussion Threads

The system shall provide the functionality to start a discussion thread on any topic which is relevant to that context and allows the user to reply to any discussion thread on a group level or a personal level.

Reply to a thread on a group level

The user shall be provided with the functionality to post a reply to a particular discussion thread which can be read by entire group.

Reply to a thread on an individual level

The user shall be provided with the functionality to post a reply to a particular person who is the part of that discussion thread.

2.2.3.3 Messaging

The system shall provide the functionality to send a personal/selective message to any of the group member(s).

Personal Message

The user shall be provided with the functionality to send message to any other user belonging to the same group.

Selective Group Message

The user shall be provided with the functionality to send message to any subset of users belonging to the same group.

2.2.3.4 Content Sharing

The System shall provide the ability to the user to share content on a group level or individual level i.e. one can share any data (books/pdfs) with the entire group or with a particular individual.

The users along with the course administrators will be able to upload various types of files (PowerPoint presentation, Portable document format, AV Clips, Spread sheets, Documents, Programs etc.) for others to download/view.

Individual Content Sharing

The user shall be provided with the functionality to share a particular piece of data with any other user belonging to the same group.

Group Content Sharing

The user shall be provided with the functionality to share a particular piece of data with any subset of users belonging to the same group.

2.2.3.5 Blogging

The System shall provide the ability to the user to start a Blog and share the Blog content on a group level or individual level

The users will be able to start a blog on any topic and upload it on the portal for other users to view and give their comments

2.2.3.6 Survey

The System shall provide the ability to the user to start a survey at group level.

The users along with the course administrators will be able to start a survey on various topics at the individual or group level and the consolidated data can be shared at the individual or group level

Group Survey Request

The user shall be provided with the functionality to start a survey on a particular piece of topic and request for survey for any subset of users belonging to the same group or different group.

2.2.3.7 Assignment Submission

The system shall provide assignment submission facility i.e. assignments could be submitted via the portal similar to how it happens on Moodle.

- The assignment submission page would have an active timer to record the time of submission in order to penalize the students for late submission. The Submission page shall also provide the administrator with the option to automatically close the submission link once past the deadline.
- Only one submission per assignment per user is allowed, so only the latest submitted file shall be considered for evaluation.

2.2.3.8 Search

The system must be able to use search functionality as a way to navigate pages instead of using hierarchical links. Academic Portals can be difficult to navigate and requires too many clicks to be efficiently used. Too many steps to complete basic actions—such as submitting an assignment—lead to frustration on the part of the student. A search utility enables students to find what they are looking for quickly in addition to having a hierarchical approach to finding functions of the academic portal.

1. A user must be able to search through pages in a course.

Users should be able to search for pages of the academic portal, since most students will be searching for material given in courses. The search feature should be present in the following way.

It should be present in the form of a search box and a “Search” button.. When a user clicks a search result link, they should be taken to the page corresponding to the search result. In addition, the search page described above might allow for the possibility of searching the World Wide Web using a standard search engine such as Google.

2. The system must display a search box on every page after a user has logged in. Users should be able to search from any page.

2.2.3.9 FAQ, Time Table and Conference/Seminar Info

This utility enables students to find out the Time Table, or have a look at the FAQ to get the common questions answered or view the Conference/Seminar Info that is going to take place

The system must display the time table, FAQ or Conference/Seminar Ingo based on the user request after a user has logged in. Users should be able to see the requested info .

2.2.3.10 Notifications

Currently, there is no system that allows users to receive SMS or email notifications of changes to Portal pages (such as assignments or announcements).

1. The system must provide both e-mail and SMS notifications for pages.

For this requirement, page refers to any distinct page (e.g., an assignment, a forum post, etc.). When a page is created, the user in the role of course administrator should be able to toggle whether notifications are turned on. By default, they should be turned on for announcements. If notifications are turned off, users in the capacity of student should be able to subscribe to notifications.

2.3 User Characteristics

2.3.1 Students

Students are the primary consumers of an academic portal. They are accessing information posted by professors, uploading assignments and project files, and discussing concepts.

2.3.2 Professors

Professors are the primary content administrators of an academic portal. They are uploading files, links, and multimedia, and grading assignments in addition to creating new places for students to discuss and collaborate.

2.3.3 System Administrators

System administrators are primarily responsible for maintaining the academic portal. They contribute minimally to the courses themselves, but spend more time modifying the system's configuration and making appropriate updates.

2.4 Constraints

2.4.1 User Interface Constraints

Using this system is fairly simple and intuitive. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.

2.4.2 Hardware Constraints

The system should work on most home desktop and laptop computers which support JavaScript and HTML5.

2.4.3 Software Constraints

The system will be intended to run on Firefox 4 and above, Google Chrome 10 and above and Internet Explorer 8 and above.

2.4.4 Data Management Constraints

System shall be able to interface with other components according to their specifications.

2.4.5 Operational Constraints

The system is limited by its operating server in terms of the maximum number of users it can support at a given time.

2.4.6 Site Adaptation Constraints

The component will be adapted to the overarching system at the conclusion of the system creation.

2.4.7 Design Standards Compliance

The system shall be implemented in PHP.

2.5 Assumptions and dependencies

Most of the academic portals have a lot of redundant features which are rarely used in an academic sessions. Our new system focuses on the features which are most important to the users of an academic institute along with introduction of some new features which other portals lacks.

2.6 Apportioning of Requirements

Integration of LDAP login might be apportioned to future versions.

3. Specific Requirements

3.1 External interface

3.1.1 Web Server

- Apache will be used as web server:
- The user inputs data via the web server using HTML forms
- The web server executes the PHP as a module and PHP script retrieves the post data if available.
- The web server receives information back from the PHP script.
- The web server displays a HTML page as result to the end-user.

3.1.2 PHP Application

The actual program that will perform the operations is written in PHP. All data will be stored in a database.

3.1.3 MySQL Database

It's an open source SQL database to store all data which communicates with the application on the server.

3.2 Functional Requirements

3.2.1 Use Case Scenario

3.2.1.1 Use Case Scenario 1 – User Login

User Login

Purpose	User logs in to system using existing profile.
User	A user with an existing profile.
Input Data	Profile username and password.
Output Data	Corresponding page data.
Invariants	Profile table data and user information.
Pre-conditions	User is not logged in to a profile, input profile exists in data base, user password matches profile

Post-conditions	User's computer has been supplied with appropriate cookie, page data is appropriate for selected profile
Basic Flow:	Webpage looks up profile data and returns the matching cookie. Webpage is updated to match new user data.
Alternative Flow(s):	Invalid password, invalid username, or mismatched username and password redirect to error message and previous page.
Business Rules:	This allows users to log in to their profile from anywhere.

3.2.1.2 Use Case Scenario 2 – Content Sharing (Upload Files)

A user logs into the system and is on any page and wants to share some content on a group/individual level.

Content Sharing (Upload Files)

Purpose	A user wants to share some data (pdf, ppt etc).
User	A legitimate user logged into the system
Input Data	The file to be shared.
Output Data	File ready to download by other users.
Invariants	The file.
Pre-conditions	User is Logged in; file exists on user's computer.
Post-conditions	Any other person to whom the content was made available is able to download it.
Basic Flow:	The user uploads a file to be shared using the upload box and selects a subset of other users of the group with whom the user wants to share the file. The file then gets uploaded to the server and desired users are able to download it after logging in.

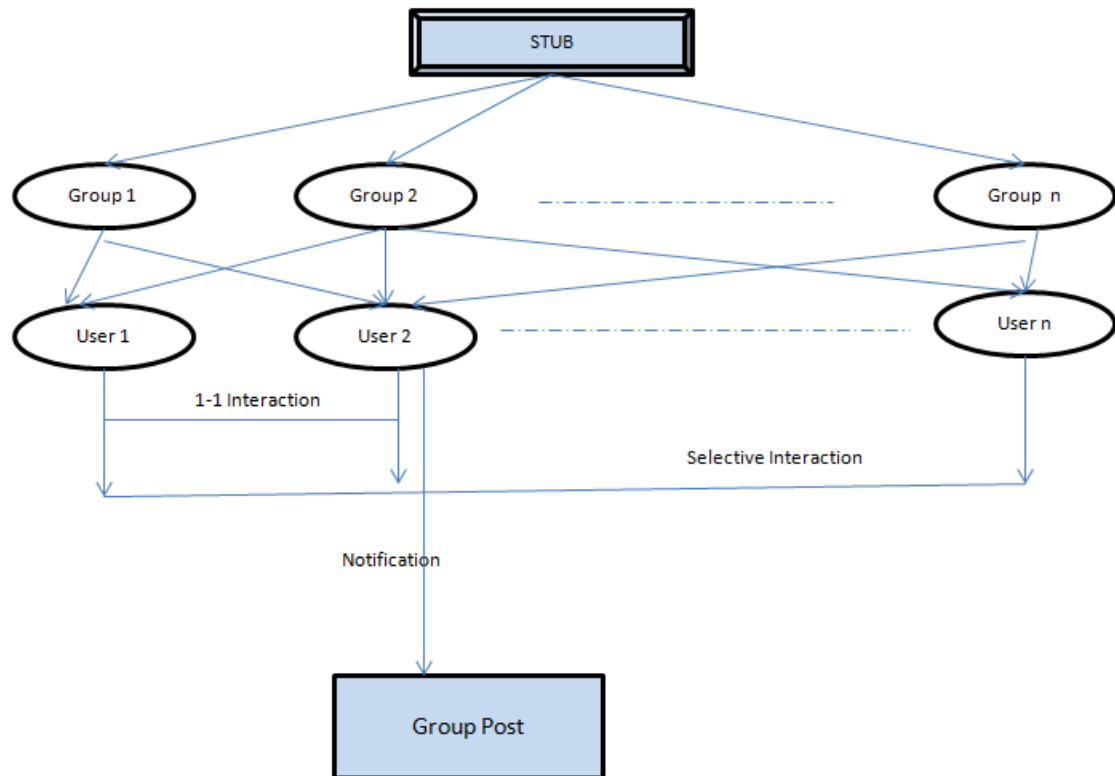


Figure 10: Process flow diagram of content upload

3.2.1.3 Use Case Scenario 3 – Discussion Thread

A user starts a discussion thread in a particular which, to which the group members are able to respond.

Discussion Thread

Purpose	A user wants to start a discussion on a particular topic.
User	Any user of the academic portal.
Input Data	Title and body of the discussion and the group in which the user wants to have the discussion.
Output Data	A discussion thread to which other users are able to view and comment.
Invariants	The discussion data.
Pre-conditions	User is logged in and is a member of the group in which he wants to post.
Post-conditions	A discussion thread to which other users are able to view and comment.
Basic Flow:	User logs in, selects a group and submits the title and body of the discussion topic through a submission form. The data is then

	stored on the server and other group members will be able to view and respond.
--	--

3.2.1.4 Use Case Scenario 4 – Content Sharing

A user wants to download a particular file.

Content Sharing (Download Files)

Purpose	A user wants download a particular file.
User	Any user of the academic portal.
Input Data	Request to download a particular file.
Output Data	File is downloaded on user's computer.
Invariants	The user and the file.
Pre-conditions	User is logged in and the file must be shared with him or with the group which he is a member of.
Post-conditions	The user has downloaded the file successfully.
Basic Flow:	User logs in, selects the file which he wants to download. The file is then transferred from the server to the user's computer.

3.2.1.5 Use Case Scenario 5 – Search Result

A user wants to search for a topic using a keyword.

Search Result

Purpose	A user wants to search for a particular keyword.
User	Any user of the academic portal.
Input Data	The keyword.
Output Data	Search Results.
Invariants	The user and the portal itself.
Pre-conditions	User is logged in.
Post-conditions	Search results.
Basic Flow:	User logs in, Enters the keyword in the search box, clicks the search button and gets the search results.

3.2.1.6 Use Case Scenario 6 – Blog Thread

A user starts a blog in a particular subject, to which the members of the portal are able to respond.

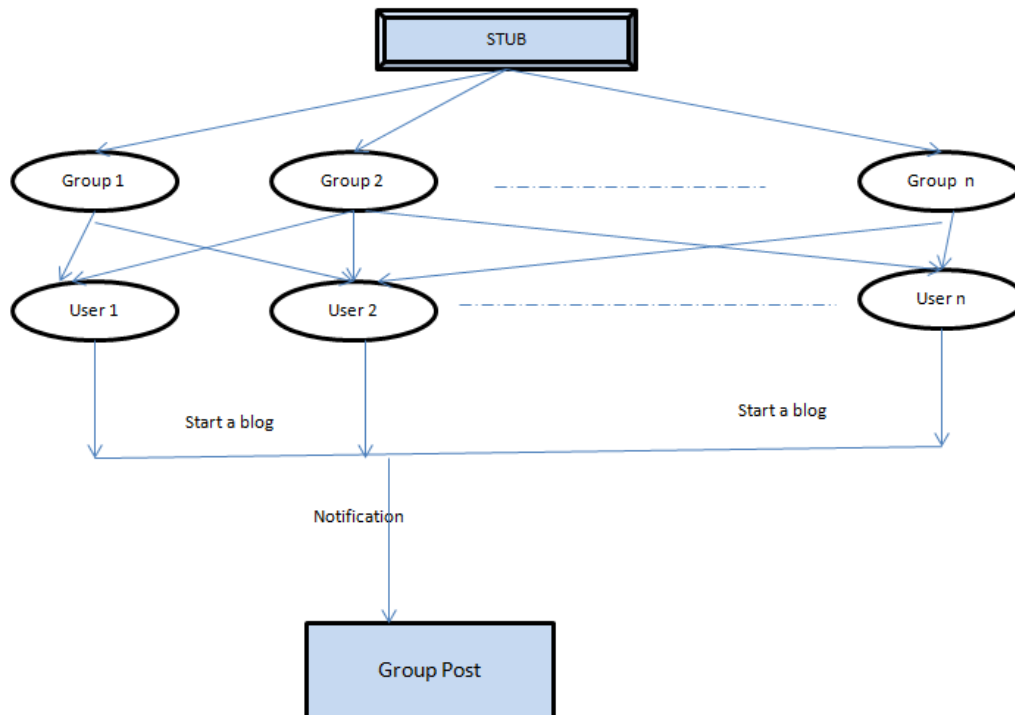


Figure 11: Process flow diagram: Blogging

Blog Thread

Purpose	A user wants to start a Blog on a particular topic.
User	Any user of the academic portal.
Input Data	Title and body of the Blog and the forum in which the user wants to have the blog.
Output Data	Blog to which other users are able to view and comment.
Invariants	The Blog data.
Pre-conditions	User is logged in and is a member of the group in which he wants to post.
Post-conditions	A Blog to which other users are able to view and comment.
Basic Flow:	User logs in, selects a group and submits the title and body of the blog topic through a submission form. The data is then stored on the server and other group members will be able to view and respond.

3.2.1.7 Use Case Scenario 7 - Notifications

A user wants to search for a topic using a keyword.

Notifications

Purpose	To get notifications of assignments, deadlines and other important messages.
User	Any user of the academic portal.
Input Data	User enables the notifications from settings.
Output Data	Notifications.
Invariants	The user and the portal itself.
Pre-conditions	User has chosen to receive notifications.
Post-conditions	User receives the desired notifications.
Basic Flow:	User enables the desired notifications from settings and receives notifications.

3.2.1.8 Use Case Scenario 8 - Survey

A user wants to start a survey for a topic

Survey

Purpose	To start a survey for a particular topic
User	Any user of the academic portal.
Input Data	Title and body of the survey and the forum in which the user wants to have the blog.
Output Data	Survey to which other users are able to do the polling and see the results.
Invariants	The survey data.
Pre-conditions	User is logged in and is a member of the group in which he wants to do the survey.
Post-conditions	A survey to which other users are able to do the polling and view the data.
Basic Flow:	User logs in, selects a group and submits the title and body of the survey topic through a submission form. The data is then stored on the

	server and other group members will be able to view and respond.
--	--

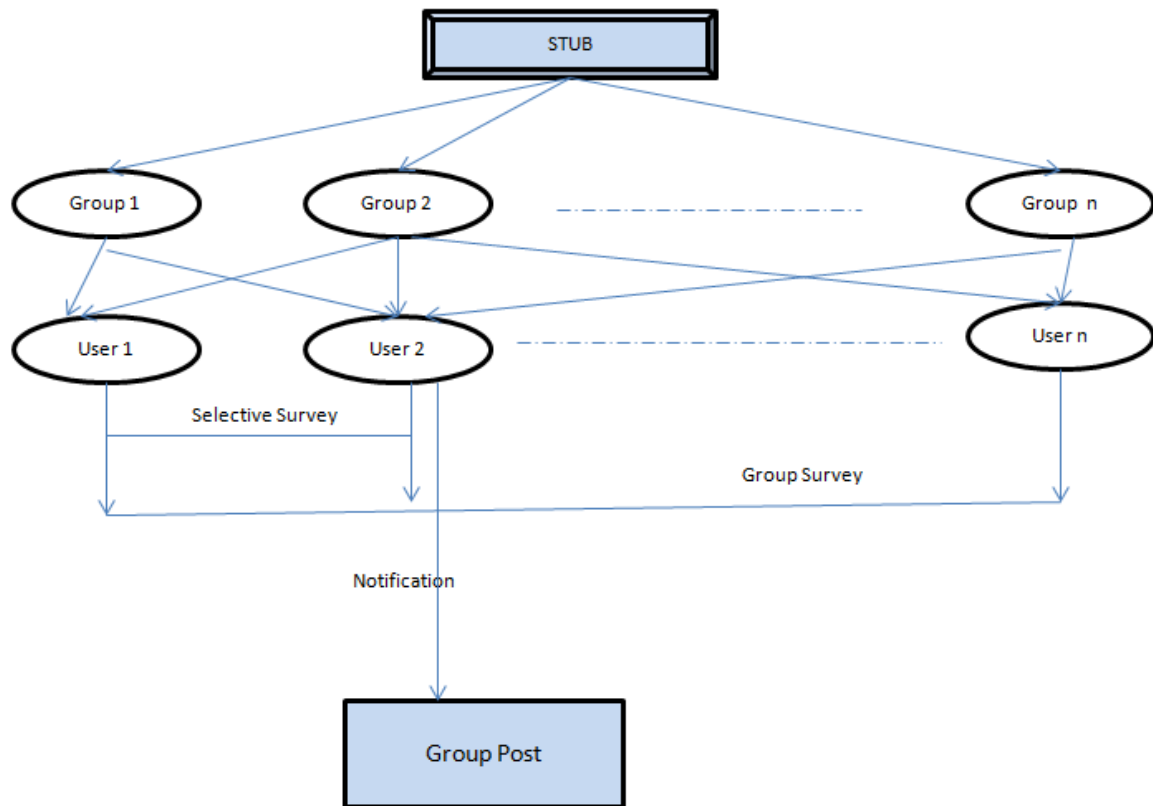


Figure 12: Process Flow Diagram- Survey

3.2.1.9 Use Case Scenario 9 – FAQ, Time Table and Conference/Seminar Info

A user wants to look at the FAQ, Time Table, and Conference/Seminar Info.

FAQ, Time Table and Conference/Seminar Info

Purpose	A user wants to look at the time table, FAQ, conference/Seminar Info.
User	Any user of the academic portal.
Input Data	The keyword.
Output Data	FAQ, Time Table, Conference/Seminar Info.
Invariants	The user and the portal itself.
Pre-conditions	User is logged in.
Post-conditions	FAQ, Time Table, Conference/Seminar Info Data.
Basic Flow:	User logs in, Selects the FAQ, Time Table or Conference/Seminar Info button and gets the results.

3.3 Performance Requirements

The system should support at least 200 concurrent users.

This statement provides a general sense of reliability when the system is under load. It is important that a substantial number of users be able to access the system at the same time, since an academic portal is important to the courses that employ it. The times when the system will be under the most stress are likely during assignment submissions. Therefore, it must be able to handle at least 200 concurrent users.

3.4 Logical database requirements

All data will be saved in the database: user accounts and profiles, discussion data, messages etc. (except files which are stored on the disk.) The database allows concurrent access and will be kept consistent at all times, requiring a good database design.

3.5 Design Constraints

1. The communication between the portal software and the database will be in SQL.
2. The portal layout will be produced with HTML/CSS.
3. The product will be written in PHP.
4. The output must be compatible with W3C XHTML 1.0
5. The source code must follow the coding conventions of PHP.
6. System administrators must have access to comprehensive documentation.

3.6 Software System Attributes

The software consists of the following elements:

1. The apache web server
2. The PHP application
3. The MySQL database
4. The database should remain consistent at all times in case of an error.

3.6.1 Reliability

The reliability of the overall program depends on the reliability of the separate components.

3.6.2 Availability

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system

runs. In case of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved with the MySQL server and saved by the administrator.

3.6.3 Security

1. Passwords will be saved encrypted in the database in order to ensure the user's privacy.
2. The user's IP will be logged.
3. The system will be protected against vulnerabilities such as SQL injection attacks.

3.6.4 Maintainability

MySQL is used for maintaining the database and the Apache server takes care of the site. In case of a failure, a re-initialization of the program is recommended.

3.6.5 Portability

The application is Linux-based and should be compatible with other systems. Apache, PHP and MySQL programs are practically independent of the OS-system which they communicate with. The end-user part is fully portable and any system using any web browser should be able to use the features of the application.