College of Natural and Computational Sciences (CNCS)

School of Information Science (SIS)

Student Project Tracking System for SIS

Information Systems Industrial Project I [INSY4121]

February 2023

Group Members

Name	ID
1. Andualem Sebsbe	UGR/7326/12
2. Gelila Tesfaye Gebru	UGR/0903/12
3. Hanan Hussien	UGR/1778/12
4. Huluager Alehegn	UGR/1112/12
5. Ketema Bekele	UGR/0986/12
6. Kasanesh Ayalew	UGR/8580/12

Information Systems Industrial Project I [INSY4121]

Adviser: Mr. Ermias Abebe

Examination board

Adviser name	**Date**	**Signature**
Examiner name	**Date**	**Signature**
Examiner name	**Date**	**Signature**

ACKNOWLEDGMENTS

First, we would like to thank our adviser Mr. Ermias Abebe, for his continuous comments and motivations to contribute or add unique features in our project work. We would like to forward our deepest thanks to all the staff members of SIS who helped us to conduct this project work.

Abstract

It is very tedious to manage and control student's final year projects and course projects using manual or classical processes. The main goal of this project is to build an integrated framework to handle all project activities. Student project tracking system is a system for the management, tracking and supervision of students' project. It is a web-based platform or framework that is useful to students, coordinators and advisers. Firstly, all students must register using the registration form. Then registered students can create account to log in to the system. When the students log into the system, they can submit project titles and get approval from the coordinators and advisers. The coordinator is the key component of the system that assigns advisers and examiners to various projects.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Overview

The student project tracking system is a software solution that is designed to help the School of Information Science to keep track of the student projects. This system will be developed with the goal of improving the efficiency of the project management process, making it easier for students and department members to collaborate, and providing greater transparency into the progress of the projects.

The system will include a variety of features, including a project management dashboard that will allow students and faculty members to see all of the projects that are currently underway, as well as their status and progress. The system will also include tools for scheduling meetings, setting deadlines, and tracking deliverables, making it easier for everyone involved in the project to stay on top of their tasks.

Additionally, the system will include tools for communication and collaboration, including a messaging system that will allow students and faculty members to communicate directly within the system. This will help to streamline the communication process, reducing the need for lengthy email chains and improving overall efficiency.

The student project tracking system will be designed to be user-friendly, with an intuitive interface that will make it easy for students and faculty members to use. It will also be scalable, allowing the system to grow as the School of Information expands and takes on more projects.

The Student project tracking software we develop can help a university in many ways. It can provide transparency around the academic progress of individual students, allowing them to track their progress on coursework and gain greater insight into their skills and achievements. The software can also streamline the communication about student work between teachers, departments and other stakeholders, facilitating better understanding and support for students. Such tracking systems are able to save much of the paperwork associated with student projects by storing documents digitally and increasing collaboration over dispersed networks. Finally, by providing an online platform for submitting student projects, universities are able to reduce their need for physical space allocation and associated costs in order to facilitate project work.

Overall, the student project tracking system will provide the School of Information with a powerful tool for managing student projects, improving collaboration and communication, and ensuring that projects are completed on time and within budget.

1.2 Background of the Organization

The Addis Ababa University School of Information Science was established in 2012 it has been placed under the College of Natural and Computational Sciences (CNCS) while its physical location is at Eshetu Chole Building, FBE campus.

The School of Information Science currently has more than 500 students in all programs, including undergraduate and postgraduate programs. The school has 35 academic staff, 6 support staff, and 3 technical assistants. School of information science (SIS) at Addis Ababa University has a lot of software systems that are developed to perform different tasks in the teaching learning process. However, there are still some problems that the employees of department as well as students faced. One of the challenges faced by both teachers and students is the unavailability of a system to repost the student and staff projects. Such a system could help to store, retrieve and manage project information as well as associated documents. Hence, this project is proposed to develop a web-based system to replace manual paper-based system in place.

1.3 Statement of the Problem

The School of Information Science at Addis Ababa University has identified the need to reform its project and task tracking procedures, and has opted for a web-based platform that would evidently facilitate the control and monitoring of many aspects related to managing, tracking and reporting of projects, tasks and activities. This will bring in an abundance of functions such as task delegation and timeline tracking. While manual paperwork techniques can put organizations in a weaker position when it comes to productivity levels and communication flow, using this system as a replacement won't be a bad alternative. Our initiative also includes generating documents that can easily be accessed from any location. As such, our solution is highly suitable for dealing with the complexities of the School's operations.

All the work is manually done starting from the project title submission to the final evaluation of project. In the current framework, users cannot get the right details at the right time, and users cannot easily handle project progress to achieve their principal objective.

Disadvantages of the current system:

1. Time-Consuming**:** Since the tracking system is paper-based, it is labor intensive and thus can take a great deal of time to document and store data.

2. Inaccurate Data Entry: Paper-based systems are often plagued by human error as data is manually entered into the system. This can lead to inaccuracies in project tracking data, which may complicate efforts to monitor progress or identify issues.

3. Limited Capacity: Traditional paper-based student project tracking systems have limited capabilities when it comes to storage capacity and organization, making them less than ideal for large-scale projects with multiple stakeholders.

4. Difficult to Search and Sort: Manual tracking systems generally require considerable effort on the part of staff in order to locate information or assess performance over time, leading to an inefficient use of time and resources.

5. Modification of student/staff information can be a broad process which can lead to confusion and mistakes.

6. The right information at the right time is not recovered.

1.4 Objective of the Project

1.4.1 General Objective

To develop online student project tracking system for the School of Information Science at Addis Ababa University that will provide an efficient and collaborative platform for managing and monitoring student projects, facilitating communication and collaboration, and enhancing the overall project management process.

1.4.2 Specific Objectives

- To design a user-friendly interface that allows students to easily create and update project timelines, set milestones, assign tasks, and communicate with team members and supervisors in real-time.
- To develop a system that enables faculty members to monitor progress, provide feedback, and evaluate the performance of students on their projects.
- To implement a notification system that will alert students and faculty members of upcoming deadlines and important project updates.
- To ensure the security and privacy of student data by implementing appropriate security measures, such as user authentication and data encryption.
- To provide comprehensive reporting capabilities that allow faculty members to analyze project progress, identify areas of improvement, and generate reports for program evaluation purposes.
- To conduct thorough testing and debugging to ensure the system is reliable and error-free.
- To provide training and support for students and faculty members to ensure they can effectively use the system to manage their projects.
- To continually update and improve the system based on user feedback and changing project management needs.

1.5 Feasibility Study

Feasibility refers to the practicality or viability of a project. The following are the feasibility aspects of the "Student Project Tracking System for School of Information Science" project.

1.5.1 Technical Feasibility

The project should be technically feasible, meaning that it should be possible to design and develop the software within the given timeline and budget. The system is fully web-based so the main technologies and here are the development technologies we use to build the system.

- HTML
- Tailwind css
- JavaScript
- Next JS
- Node JS
- MongoDB and diagram drawing tools

Each of the technologies is freely available and the technical skills required are manageable.

Overall, adapting these cutting-edge technologies as well as robust security models allows us to offer a dependable solution familiarizing with students' current work flow processes & enabling improved response times throughout university projects management operations.

1.5.2 Operational Feasibility

The Student Project Tracking System for the School of Information Science at Addis Ababa University will be designed to streamline the project tracking process and enable better management of both course and industry projects. This system should become an efficient tool for recording, tracking, reporting, and managing all school information science projects while allowing faculty supervisors to monitor progress within each project.

The Student Project Tracking System should have features that include an easily searchable database to store project information such as title, description, developer(s), supervisor(s), deadlines and milestones, assessment criteria, and feedback form. In addition to these basic features, additional functionality includes sending automated reminders for upcoming deadlines and milestone achievements, along with a quick-view portal for all open projects assigned to a particular department/faculty member/group of students. All this data should be securely stored on cloud servers so that access is easy from any device by anyone who has authorization.

Operational feasibility also involves determination of the security and validations. Analyzing user characteristics is an important aspect of any project. It allows us to clearly define and focus on who the end users are for the project. Also, it allows us to check the progress of the project to ensure that we are still developing the system for the end users.

The end users for system are:

- Coordinator will manage entire system and behavior of the system. He has rights to access almost all the rights to access the system.

- End user of the system will be staff, student or any other person who willing to check if the any project has been done before or not.

The users must have following characteristics:

- The user must have basic knowledge about Computer or any smart device.
- User should be comfortable in starting and stopping Web Based Applications. So as most people all over the world are using Web Based Applications, it is not difficult for them to use the system. Thus, the project is well suited to overcome organizational restrictions.

Thus, considering the operational feasibility the development of the proposed system considered as operationally feasible development.

1.5.3 Schedule Feasibility

We have identified that we have approximately one month and a half as a development time in order to apply the prototyping model most effectively. We have planned and divided that the phases will last approximately five weeks, this will give us an overall flexibility of four weeks allowing us to adapt to yet unforeseen problems that may occur during the project.

As per the plan we are hoping to design the full fledge of working website that fulfill the requirements.

1.5.4 Economic Feasibility

It is about the benefits of the proposed system with respect to the cost we spend to complete the system. Being a web-based system SPTS will have associated hosting cost since the system doesn't consist of any multimedia data transfer, bandwidth required for the operation of this system is very low. This system will follow freeware software standards. No cost will be charged from the potential customers. Beside the associated cost there will be many benefits for the department. Especially the extra effort that is associated with storing, searching organizing projects, minutes and sharing letters will be eliminated. Depending on these points it is clear that the project SPMS is economically feasible.

1.6 Significance of the Project

The Student Project Tracking System is a proposed project aimed at improving the efficiency and effectiveness of managing student projects in the School of Information Science. The proposed system will provide a platform for tracking the progress of student projects, enabling students and faculty to monitor the status of their projects in real-time. This paper outlines the significance of the project, its potential impact on the school of information, and the benefits that the system will bring to students, and the school as a whole.

The Student Project Tracking System is significant for several reasons.

Firstly, it will improve the management of student projects in the School of Information. Currently, the school relies on manual processes to manage student projects, which can be time-consuming and prone to errors. The proposed system will automate these processes, making them more efficient and reliable. This will save time for faculty and students, allowing them to focus more on the project itself, rather than on administrative tasks.

Secondly, the Student Project Tracking System will improve communication between students and faculty. The system will provide a platform for students to communicate with their project supervisors, enabling them to get feedback on their work and ask questions. Similarly, faculty will be able to communicate with students more easily, providing guidance and feedback as needed.

Thirdly, the Student Project Tracking System will enable the School of Information to better track the progress of student projects. Currently, there is no centralized system for tracking the progress of student projects, which can make it difficult for faculty to monitor the status of projects. The proposed system will provide a centralized platform for tracking the progress of student projects, enabling faculty to monitor the status of projects in real-time.

Finally, the Student Project Tracking System will improve the overall quality of student projects in the School of Information. By providing a platform for tracking the progress of student projects, the proposed system will enable students to better manage their time and resources, resulting in higher quality projects. Additionally, the system will provide faculty with the tools they need to monitor the progress of projects and provide feedback, which will help students to improve their work.

1.7 Beneficiaries of the project

The Student Project Tracking System is a project that has the potential to benefit a wide range of individuals, institutions, and organizations. The system aims to provide a centralized platform for students in the School of Information to track their project progress, collaborate with their peers and instructors, and access resources and support services. In this article, we will discuss the beneficiaries of this project in detail.

Students:

The primary beneficiaries of the Student Project Tracking System are the students of the School of Information. The system will provide students with a central platform to manage their projects, including scheduling, progress tracking, and resource sharing. The system will facilitate collaboration between students, enabling them to share their work, get feedback from their peers, and work together on group projects. The system will also provide students with access to learning resources and support services, such as tutoring, academic advising, and technical support. Overall, the Student Project Tracking System will empower students to take ownership of their projects, manage their time more effectively, and collaborate more efficiently.

Instructors:

The Student Project Tracking System will also benefit instructors in the School of Information. The system will allow instructors to track student progress, review project submissions, and provide feedback to students. Instructors can use the system to monitor individual and group projects, view progress reports, and assess student performance. The system will also enable instructors to communicate with students, provide guidance, and offer support as needed. Overall, the Student Project Tracking System will help instructors to manage their courses more efficiently, provide more personalized support to students, and improve the overall quality of student projects.

Project Coordinator:

The Student Project Tracking System will be a valuable tool for the coordination of the School of Information Science. The system will provide the administration with a centralized platform to monitor student progress, assess the quality of student projects, and identify areas for improvement. The system will also enable the administration to gather data on student performance, course completion rates, and resource utilization. This information can be used to improve the overall quality of the program, identify trends, and make data-driven decisions. Overall, the Student Project Tracking System will help the administration of the School of Information to monitor and manage the program more effectively.

Employers:

The Student Project Tracking System will also benefit potential employers of the students of the School of Information. The system will provide employers with a centralized platform to view student projects, assess the quality of student work, and evaluate the skills and competencies of students. This information can be used to identify potential candidates for internships, co-op positions, or full-time employment. The system will also provide employers with insight into the types of projects that students are working on, the technologies they are using, and the skills they are developing. This information can be used to identify trends and make informed decisions about hiring and resource allocation. Overall, the Student Project Tracking System will help to bridge the gap between academia and industry, making it easier for employers to identify and recruit top talent.

1.8 Methodology

1.8.1 Data collection

In order to gather appropriate data, we use primary source of data which interview and observation to get firsthand information from sample respondents based on the review of related literature important to the subject of the study. Thus interview and observation designed focusing to grasp the current information on what methods are the School of information sciences(SIS) uses in processing, storing and retrieving projects and also how the staff members' can identify final year projects whether it is done before or not.

We are going through a research and inspection to gather the right amount of data needed to develop this system directly from the client. Some specific methods we will use to collect this data include:

Interview: This is one of data collection method that enables us to gather information from the organization directly in the form of asking question and getting answers for those questions. So, our team uses this method to gather information by asking the school head and instructors in the department.

Observation: This is also another data collecting method. In fact, we have also used this observation method to gather data. This method enables us observing and understanding how the teachers handle industrial project titles so we can understand.

- How the current system works.
- How the teachers check out the project might be done.
- Work overload on staffs.

1.8.2 System development Methodology

1.8.2.1 Development approach

Object oriented Approach: The method of system development paradigm that we selected is the object-oriented approach because this approach is helpful for us to represent the different phases of the project through many diagrammatic representations such as activity diagrams, use cases, sequence diagrams, class diagrams, etc. Generally, we choose object-oriented development design because of:

1. These techniques have a reusability feature
2. These techniques provide greater opportunities for users to participate in the process
3. Increases flexibility
4. Improves quality
5. It is the latest, powerful, easy and highly usable
6. Increase domain and design reuse

1.8.2.2 Process Model Iterative process model

The process model we used is iterative process model. We select iterative model because, to design our project we are required to review and design in each phase iteratively to meet user requirements

- Flexibility: - this model allows as to make changes at any stage without affecting the scope of the project.
- Improved Quality: - this model allows as to identify and fix problems early in the development process. This leads to improved quality.
- Collaborations: - encourages collaboration between our team members, leading to a better overall understanding of the project goals and requirements.

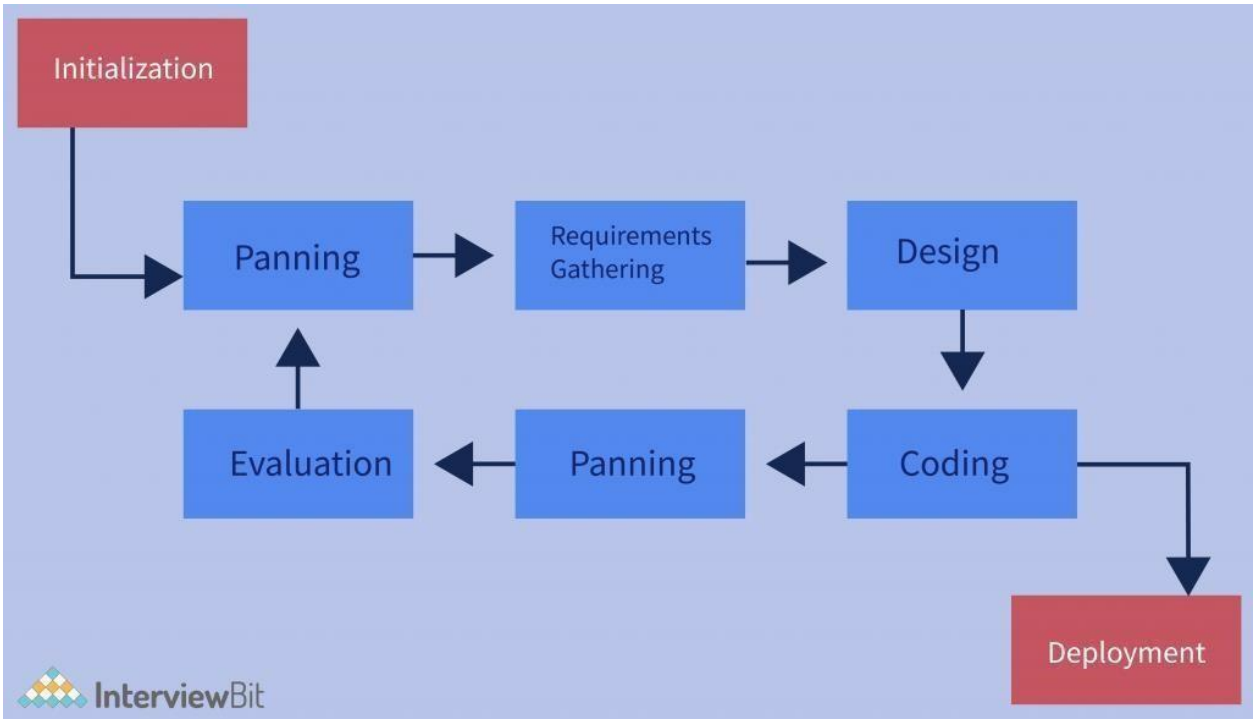


Figure 1-1 Iterative model

1.9 Development Tools and Technologies

Development tools are tools that support the rapid implementation of software applications upon a programming platform. Tools are solutions that programmers utilize to create, edit maintain and debug programmers or applications. Some examples are compilers, code editors, debuggers, visual programming methods, GUI designs and framework generators.

1.9.2 Front End Technologies

Front-end technologies are an essential part of any business that strives to enhance user interaction, efficiency, and the look and feel of their website or application. Development teams use front-end technologies to create a website's design, structure, animation, and everything you see on the screen when you open it.

Next JS

Next JS is one of the recent and flexible react frameworks for creating easily adaptable interactive UI elements. It also handles the tooling and configuration needed for React, and provides additional structure, features, and optimization for applications.HTML and CSS knowledge is required to use Next JS. Also, Frameworks like this can be seamlessly integrated into big projects for front-end development without causing any problems or issues.

1.9.3 Back End Technologies

In this project we will intend to use Node.js as a back-end technologies to the libraries of server-side

languages that are used to create the server configuration of our system. The back-end technologies form the basis of software development. Without specialized back-end developers, fancy designs and intuitive interfaces created using front end coding languages are not sufficient.

Node JS

Node JS is a JavaScript runtime environment that is open-source, cross-platform, and used to create server-side and networking applications. Additionally, Node.js provides a vast library of JavaScript modules, making it considerably simpler to create online apps. Node.js supports a wide range of third-party libraries and APIs. With Node.js, web developers can take advantage of the "JavaScript everywhere" approach (allowing web applications to be developed in a single programming language).

1.9.3 Documentation and Modeling Tools.

This project will use the following modeling tools:

- **MS-word**: we prefer it due to its ease of use plus the fact that it can be used on different operating systems.
- **MS-PowerPoint:** It is a software that we use for our presentation purpose.
- **Lucid**: It provides collaborative online diagramming to make it easy to draw flowcharts, UML, mind maps and more. It offers visual collaboration suite that helps our team see and build the future from idea to reality.
- **Figma**: Figma is a powerful design tool that helps us to create anything: websites, applications, logos, and much more. We prefer to use Figma because it lets us to share design files and collaborate in real-time quickly with our entire team.

1.9.4 Deployment Environment

- Operating System: The system could be deployed on a server running a stable and secure operating system such as window, Linux, and Mac.
- Database: MongoDB
- Web Server: The application will be hosted on an Apache web server, which is the most widely used web server for hosting web applications. Apache is known for its scalability, security, and easy integration with other technologies.
- Frameworks and Libraries: Next JS

1.10 Scope

The scope of the project includes the following features:

Project Title Submission: Students will be able to submit their project title so that it will be approved either by the coordinator or approval committee.

Project title Approval: The Coordinator/approval committee will approve project titles, set deadlines, and assign them to students. The system will also allow the faculty to define the requirements and guidelines for each project.

Project Submission: The students will be able to submit their projects through the system. The system will provide the students with an interface to upload their project files and add comments, if any.

Project Monitoring: The faculty will be able to monitor the progress of the projects. The system will provide the faculty with a dashboard to view the status of each project, such as whether it is in progress or completed.

Feedback Provision: The faculty will be able to provide feedback on the projects submitted by the students. The system will allow the faculty to add comments, suggestions, and grades for each project.

Project Tracking: The students will be able to track their progress through the system. The system will provide the students with a dashboard to view the status of their projects, such as whether it is under review, accepted, or rejected.

Notifications: The system will provide email notifications to the students and department regarding the project status, feedback, and deadlines.

User Management: The system will provide an interface for the department and staff to manage the users. The system will allow the faculty to add, modify, and delete users, such as students and other faculty members.

Reporting: The system will provide reporting functionality for the department and staff. The system will allow them to generate reports on the project status, student progress, and feedback provided by the faculty.

1.11 Risks, assumptions and constraints

Category	Description
Risks	- Technical difficulties or errors that could lead to system downtime or loss of data. - Lack of user adoption and resistance to change among faculty, staff, and students. - Insufficient funding or resources to fully develop and maintain the system. - Security breaches or data leaks due to insufficient data protection measures. - Legal or regulatory compliance issues, such as privacy laws or accessibility requirements.
Assumptions	- The system will be used primarily by faculty and staff members to track student projects and progress. - The system will allow for different levels of access and permissions based on user roles. - The system will integrate with existing school systems and software, such as student portal systems and student information systems. - The system will be customizable and adaptable to the specific needs of the School of Information Science. - The system will improve project management and collaboration among students and faculty members.
Constraints	- The project must be completed within a specified timeframe and budget. - The system must be designed to meet the technical requirements and limitations of the existing school infrastructure. - The system must be scalable and able to accommodate future growth and changes. - The system must comply with relevant data protection and privacy laws and regulations. - The system must be user-friendly and accessible to a diverse user base.

Table 1-1 Risks and assumptions

1.12 Phases and Deliverables of Project

The major phases and deliverables that we undergo through developing this project are:

Phase	Deliverable	Description
Planning	Project Proposal	Document outlining the purpose, scope, goals, objectives, and stakeholders of the project
	Project Plan	Document outlining the timeline, budget, resources, and risks associated with the project
	Requirements Gathering	Document outlining the functional and non-functional requirements for the student project tracking system
Design	System Architecture	Diagram outlining the system components and their interactions
	Data Model	Diagram outlining the database schema for the system
	User Interface Design	Wire frames and mock-ups of the user interface for the system
Development	Source Code	Codebase for the student project tracking system
	Testing Plan	Document outlining the testing strategy and test cases for the system
	User Documentation	Document outlining how to use the system
Testing	Test Results	Document outlining the results of testing the system
	Bug Fixes	Fixes to any bugs or issues discovered during testing
	Acceptance Criteria	Document outlining the criteria for determining if the system meets the requirements
Deployment	Installation Package	Package containing the system code and all necessary dependencies for deployment
	User Training	Training sessions for users of the system
	System Maintenance Plan	Document outlining the plan for maintaining and updating the system

Table 1-2. Deliverables of the project

1.13 Work-break down structure

	Work Break Down	Assigned Member
1	Chapter one: Project Introduction	
1.1	Overview	Kasanesh Ayalew
1.2	Background of the organization and study	Gelila Gebru
1.3	Statement of the problem	Ketema Bekele
1.4	Objective of the project ▪ Specific objective ▪ General objective	Hanan Hussien
1.5	Feasibility Analysis 1. Technical Feasibility 2. Economic Feasibility 3. Operational Feasibility 4. Schedule feasibility 5. Political Feasibility 6. Social Feasibility	Kasanesh Ayalew Huluager Alehegn
1.6	Significance of the project	Andualem Sebsbe
1.7	Beneficiary of the project	Ketema Bekele
1.8	Methodology 1.8 Data collection Method 1.7 Development Methodology	Huluager Alehegn
1.9	Development tools and technologies 1. Frontend technologies 2. Backend technologies 3. Documentation and modeling tools 4. Deployment Environment 5. Project management tools	Hanan Hussien

1.10	Scope	Ketema Bekele
1.11	Assumptions and constraints ▪ Risks ▪ Assumptions ▪ Constraints ▪ Limitation of the project	Gelila Gebru
1.12	Deliverables of the project	Kasanesh Ayalew
1.13	Work-break down structure	Huluager Alehegn
1.14	Project Schedule	Andualem Sebsbe
2	Chapter Two: Business Area Analysis and Definition	
2.1	Business Analysis ▪ Activities/functions of the existing system ▪ Problems of the current system ▪ Forms and reports of the current system ▪ Players of the existing system	Hanan Hussien Gelila Gebru Huluager Alehegn
2.2.	Requirement definition 2.2.1Functional requirement 2.3.1 Non-functional requirement	Andualem Sebsbe Ketema Bekele
3	Chapter Three :Object Oriented Analysis	
3.1 3.2 3.3 3.4 3.5	Introduction System use case i. UI identification ii. Actor identification iii. Designing the use case diagram iv. Use case description Conceptual modeling ▪ Class diagram ▪ Class description Sequence diagramming User interface prototyping	Huluager Alehegn Andualem Sebsbe Kasanesh Ayalew Ketema Bekele
4	Chapter Four: Conclusion of Phase1	Ketema Bekele Hanan Hussien

5	Chapter Five: Object Oriented Design	
5.1	Review of the first phase	Kasanesh Ayalew
5.2	Introduction to the chapter	Gelila Gebru
5.3	Class type Architecture ▪ User interface Layer ▪ Process/controller Layer ▪ Domain/Business Layer ▪ Persistent Layer	Hanan Hussien
5.4	Collaboration Diagram	Andualem Sebsbe
5.5	Component Diagram	Huluager Alehegn
5.6	Deployment Diagram	Hanan Hussien
5.7	Relational Persistent Diagram	Kasanesh Ayalew
5.8	User Interface ▪ User interface flow Diagram ▪ User Interface Design	Andualem Sebsbe
6	Chapter Six: Object Oriented Implementation	
6.1	Introduction	Ketema Bekele
6.2	Implementation technology	Huluager Alehegn
6.3	Testing and testing Procedures ▪ Unit Testing ▪ Integration Testing ▪ System Testing	Gelila Gebru
6.4	Deployment/Installation Process ▪ Hardware and software Acquisitions	All member of the group
7	Chapter seven: Conclusion and Recommendations	
7.1	Conclusion	All members of the group
7.2	Recommendation	All member of the group

Table 1-3. Work breakdown structure

1.14 Project Schedule

A project schedule indicates what needs to be done, which resources must be utilized, and when the project is due. It's a timetable that outlines start and end dates and milestones that must be met for the project to be completed on time. The project schedule is often used in conjunction with a work breakdown structure (WBS) to distribute work among team members. The project schedule should be updated regularly to gain a better understanding of the project's status.

The project will be concluded with a review of the iteration and an evaluation of its success against the initial requirements. As the planned iteration, we are hoping to design the full fledge working website that fulfill all the school program requirements.

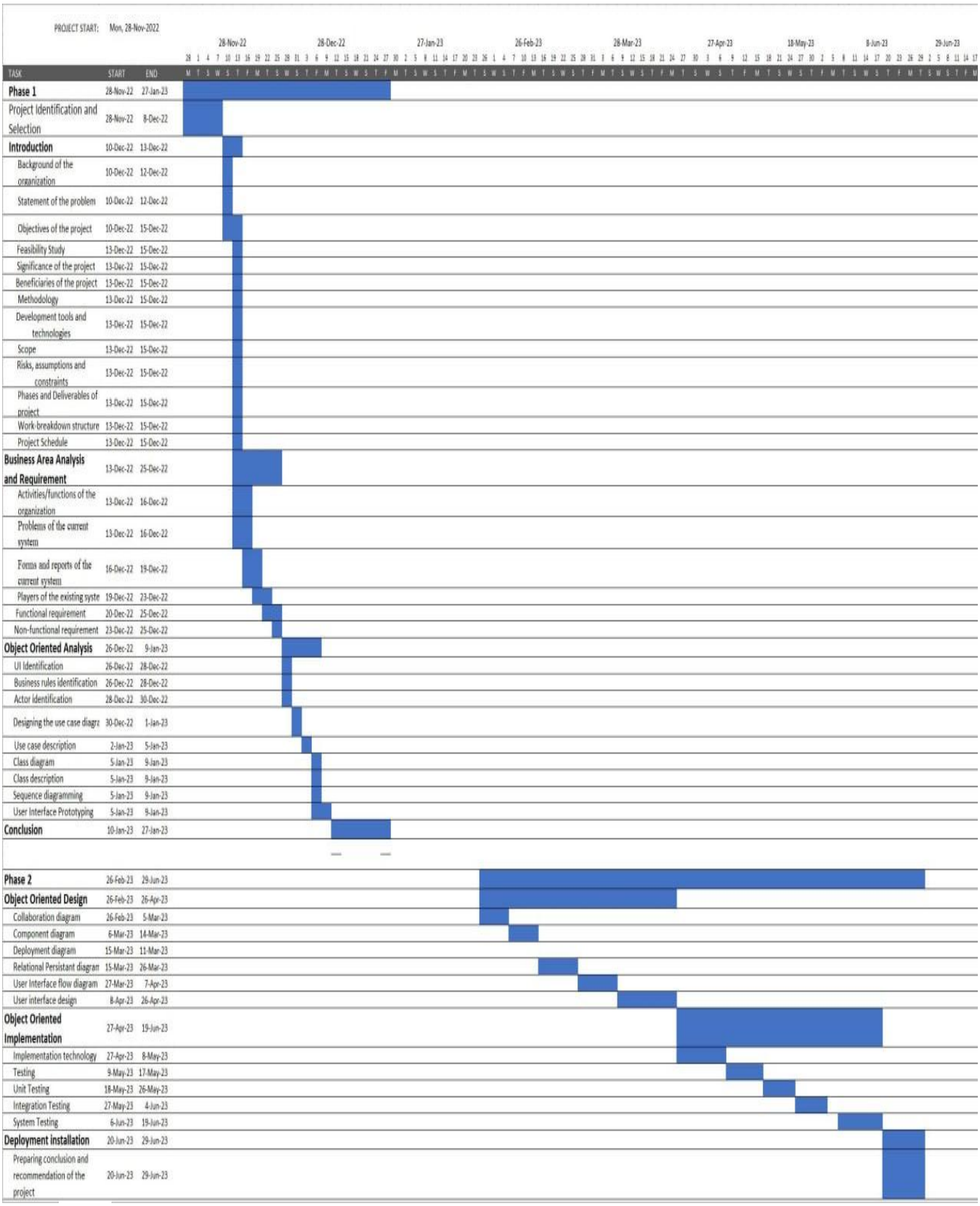


Figure 1-2 Project Schedule

2 BUSINESS AREA ANALYSIS AND REQUIREMENT DEFNATION

2.1.1 Overview

In this chapter, we are focusing on the student project tracking system within the School of Information Science (SIS). We will cover an in-depth look into the operations of the business sector and its current issues, and provide a brief overview of the functional and non-functional requirements for the project. Requirements analysis involves simplifying information and discovering ways to bring value to SIS through validating data, finding solutions that meet their business needs, and assessing potential gain from these activities. With all necessary details provided by Requirements Gathering methods, our Business Area Analysis allows us to begin working on Requirements Analysis. Properly organized and verified criteria from stakeholders establish a solid base for finalizing design plans before proceeding with development of a solution which will bring true benefit to SIS.

2.2 Business Area Analysis

Addis Ababa University is the largest and oldest university in Ethiopia, with over 40,000 undergraduate and graduate students enrolled in various programs. The School of Information Science (SIS) is one of the schools within the university, offering degree programs in information system. SIS has a department of over 35 members and an enrolment of approximately 500, students.

Currently, project management and tracking are done using a combination of paper-based systems, spreadsheets, and email communication. This approach has proven to be time-consuming and inefficient, leading to delays in project completion and difficulties in tracking progress. The SIS administration has identified a need for a more efficient and centralized system to manage and track student projects, streamline communication, and improve collaboration among students and faculty.

The Student Project Tracking System (SPTS) will be a web-based platform that allows faculty and staff members to create and manage project templates, assign projects to students or groups of students, set deadlines, and track progress. The system will also allow students to submit project proposals, collaborate with peers, and submit completed projects for evaluation.

Overall, the SPTS (Student Project Tracking System) is intended to improve project management and collaboration among students and faculty members, streamline communication, and reduce the administrative burden associated with manual project tracking.

Sure, here are some possible problems with the current system that the Student Project Tracking System (SPTS) will address:

Manual Project Management: The current system for managing student projects at the School of Information Science (SIS) is mainly manual and paper-based. This makes it difficult to track the progress of individual students or groups, and also increases the risk of errors and data loss.

Lack of Collaboration: The current system does not provide an effective way for students to collaborate with each other and share project resources. This leads to delays in project completion and missed opportunities for learning and innovation.

Inefficient Communication: Communication among faculty members, staff, and students is mainly through email, which can be time-consuming and prone to errors. This can lead to delays in project approval, submission, and grading.

Difficulty in Tracking Progress: The current system makes it difficult for faculty members and staff to track the progress of individual students or groups, as well as monitor overall project completion rates. This makes it challenging to identify and address potential issues and delays.

Limited Reporting and Analysis: The current system does not provide detailed reporting and analysis capabilities, making it difficult for faculty members and staff to evaluate project performance, identify areas for improvement, and make informed decisions.

Overall, the SPTS (Student Project Tracking System) will address these problems by providing a centralized and efficient system for managing student projects, improving collaboration and communication, and providing detailed reporting and analysis capabilities.

2.2.1 Activities/functions of the organization

Depending on the interview and the observation that we conduct The school of Information science (SIS) has many digital systems for the ease of activities in the department however all undergraduate projects and from different meeting are being processed manually. To understand it in detail, interview is made withdifferent employees and students found in different roles of the department. Having interviewed with staff and students we are able to identify the following key aspects of the current practice:

- The common ways of organizing files.
- The documentation of projects
- The project management.
- The ways to notify staff and students.

So, the student project tracking system for the course and industrial projects in the School of Information Science can serve as an effective tool to track the progress of student projects, provide project supervisors with real-time updates on the projects. To implement such a system, the following are the main activities/functions that the organization should undertake:

Project Title Submission:

The first step in the student project tracking system is to encourage students to form a group and submit project proposals for the course or industrial project. The organization should provide a platform where students can submit their proposals with detailed descriptions of their project ideas, including the project objectives, methodology, expected outcomes, and timelines.

Project Title Approval:

After the proposal submission, the organization needs to select and approve the projects based on their

feasibility, relevance, and potential impact. The selection committee can include the project supervisor, the course instructor, and other relevant stakeholders in the School of Information Science.

Project Monitoring and Tracking:

Once the projects are approved, the organization needs to allocate the projects to the students or the student teams based on their skills, interests, and availability. The allocation process should be fair and transparent to ensure that every student gets an equal opportunity to work on a project.

The organization should establish a system to monitor and track the progress of the projects regularly. The system should include a dashboard that displays the project status, milestones, deadlines, and other essential information. The project coordinator and advisor should have access to the dashboard to monitor the progress of the project and provide guidance and support to the students as

The school should ensure that the students document their project activities and outcomes systematically. The documentation includes project plans, progress reports, and project deliverables. The school should also require the students to submit a final project report that summarizes their project activities, outcomes and lessons learned.

Project Evaluation and Feedback:

Finally, the organization should evaluate the projects and provide feedback to the students. The evaluation can be based on the project outcomes, the quality of the project deliverables, and the students' performance. The organization should also provide constructive feedback to the students to help them improve their skills and knowledge.

In summary, a student project tracking system for the course and industrial projects in the School of Information Science should include project title submission, project selection and approval, project allocation, project monitoring and tracking, documentation and reporting, and project evaluation and feedback. By implementing such a system, the organization can enhance the students' learning experience, improve the quality of the projects, and promote innovation and creativity.

2.2.2 Problems of the current System

The problem of the current system basically is the gap and barriers between the existing papersbased system and proposed system. To point out the current problem the current paper based student project tracking system of SIS we use **PIECES framework**. It is a problem-solving framework. In other words, the PIECES framework is a problem-solving framework that is used for the identification of problems within an existing system and equally suited for analyzing both manualand computerized systems and applications. It contains the checklist for identifying problems.

1. **Performance problem:** as it is a manual system, projects which are done by seniors are not adequately accessible to students who need some references for their academic works. Software projects usually stored in unorganized manner. They may not accurately match with its documentation. Searching is mainly done by file name only.
2. **Information: -** the current system involves both the input and output data problems
   - Lack of necessary information about the project since they aren't well formatted.
   - It is difficult to share letters timely and to its subsequent use.
   - Projects are not accurately captured based on categories.
   - Difficulty to identify projects which are done before this tends to redundancy.
3. **Economic problem:** we can see that the existing system from time and cost of storage area perspectives
4. **Control /security**:-
   - File is not secure from accident or vandalism.
   - Projects are not well organized.
   - Not easy to meet new information needs from stored data

5. **Efficiency**: - we valued the current system from searching time and perspective Effort required for managing document is excessive.
6. **Service:-** problems of the current manual project management system in **SIS**

   - Students cannot access projects to use as a reference.
   - Retrieving projects is often as a last resort and require asking people to deliver it.

2.2.3 Forms and Reports of the current system

Since there is no system and project (document) organizing methods being used in the school of Information science (SIS), we could not be able to find Forms and Reports of the current system.

2.2.4 Players of the existing system

There is no system developed in AAU school of information system to store documents and projects. SIS manage its documents with paper and file storage.

-The major players of these were:

- Coordinator
- Project Adviser
- Student
- Teacher/Instructors
- Project examiners
- Approval Committee

2.3 Requirements Definitions

Requirements definitions are the set of specifications and criteria that describe what a system, product, or service must do or possess to meet the needs of its users or stakeholders. It is an essential part of the software development process, and it outlines the goals, objectives, and expectations of the project.

2.3.1 Functional Requirement

Functional Requirements are a set of detailed descriptions of what a software system or application should do in order to fulfill a specific business or user need. These requirements describe the system's behavior, features, and functionality, including how the system should respond to inputs, process data, and interact with users. Functional requirements are typically documented as part of the software development life cycle, and serve as a blueprint for the system's design, development, and testing phases. They help to ensure that the system meets the needs and expectations of the users, stakeholders, and business requirements.

Functional Requirements for Student Project Tracking System:

1. **User Authentication & Registration:** The system should support authentication by means of passwords allowing the users to have access to personalized areas once identified successfully those users can also register with information like an username name, email address, course number etc..
2. **Submit Title:** Students should be able to submit their project titles through the system. The system should validate the input and ensure that the title is not already taken. Once submitted, the title should be visible to the faculty members for review.
3. **Project Tracking & Management:** Ability to track existing projects assigned to students andfaculties at any given point of time; ability to create a project plan outline; ability to create task plans including deliverables & deadlines associated with each task; ability to set milestones & associated goals; ability provide timely notifications on deadlines etc…
4. **Custom Designations:** With custom designations instructor can customize someprivileges/access levels between different roles within one single project.
5. **Reports & Analytics:** System should generate reports related with every completed tasks, status reports, rewards / penalty analysis specific related with each courses and departments automatic graphical visualization charts interpreted namely pie charts, bar diagrams etc..
6. The system should allow users with appropriate access permission mark a project has been completed by marking it with a check or other symbol.
7. It should provide feature for past project data search.
8. It should enable lecturers/supervisors assign tasks to students/researchers by due dates.
9. The system should generate project records as reports such as a list of projects running late or according to certain criteria like discipline wise etc.
10. It should provide feature for past project data search.
11. The system should be able to store the project title, the deadline for completion, tasks assigned to determinate people, researcher's name and in-charge staff.
12. Ability to report on task completion status in real-time.
13. Automated notifications for students about upcoming deadlines.

2.3.2 Non-functional Requirement

Non-functional needs are those that have little to do with the system's functions. Certain restrictions apply to the system's services or functions. They include time constraints, development process constraints, and standard constraints. Non-functional requirements are frequently used to refer to the entire system as opposed to specific system features or services.

Non-functional requirements for the proposed system include the following:

- Usability: The software must be easy to use and understand for both instructors and students. It should also facilitate communication between the different stakeholders involved in the project management process. That is the system must be user friendly with an intuitive graphical user interface (GUI).
- Security: The system needs to be protected against unauthorized access and manipulation of data stored in the database. This includes both technical protection measures such as authentication, authorization, encryption and auditing as well as administrative mechanisms such as user accesses regulations. Growth: As more projects get added to the system, it should grow accordingly providing a flexible architecture that scales with growing demand.
- Performance: The tracking software must respond quickly when retrieving data from or updating records in the database, so that information processing times do not impact on user experience negatively.
- Maintainability: As new versions of software are released regularly maintenance operations must be done easily without disruption of service or costly resources allocation.
- Reliability: The software must operate reliably and accurately regardless of changes in data volume or usage patterns for long periods of time without failure or unavailability issues.

3 OBJECT ORIENTED ANALYSIS

3.1 Overview

In this project, we plan to develop a student project tracking system for the School of Information Science (SIS). In this chapter we present see about use case modeling which shows the way users interact with the system. As such, the use case model defines the user's objective, the interactions between the system and the user, and the system's behavior required to meet these objectives. It acts as an integrated thread in the development of the entire system.

This use-case model is used like the main specification of the system functional requirements as the basis for design and analysis, as the basis for user documentation, as the basis of defining test cases, and as an input to iteration planning. It includes user identification, Business rule identification and UI identification.

Under the conceptual modeling the class diagram describes the structure of a system by showing the system's classes, their attributes, operations (or models) and the relationship among objects.

The sequence diagram shows the process interactions arranged in time sequence. It depicts the process involved and the sequence of messages exchanged between the processes needed to carry out the functionality.

This chapter also includes user interface prototyping which helps us to test out the user interface design including its usability before the real development starts.

3.2 Use case modeling

A Use Case Model describes the proposed functionality of a new system. A Use Case represents a discrete unit of interaction between a user and the system. This interaction is a single unit of meaningful work, such as Create Account or View Account Data.

Use cases add value because they help explain how the system should behave and, in the process, they also help brainstorm what could go wrong. They provide a list of goals, and this list can be used to establish the cost and complexity of the system.

3.2.1 UI Identification for Student Project Tracking System:

UI Element	Description
Login Screen	Allows users to enter their login credentials to access the system.
Submit Project title	Allows users to enter project details such as title, description, objectives, etc.
Submit Project Screen	Allows Students to submit their project.
Feedback	Is the Interface where the Advisor or Instructor gives comment on course or final year projects?
Evaluate Project Screen	Allows Instructors or Examiners to evaluate the submitted project based on a set of criteria and provide feedback to the student.
Coordinator Dashboard	Is the Interface which allows the coordinator to view and approve submitted titles, Assign Advisors, Manage Monitoring Plan, Assign Examiner and Manage Instructors?
Instructor Dashboard	Is the Interface which allows the Instructor to Managing courses by viewing, evaluating and giving feedback to Course Projects?
Advisor Dashboard	Allows the Advisor to view, evaluate and give feedback to Active Final year Projects.
Manage Students	Allows the Coordinator to manage students, including adding, editing, and deleting student profiles.
Manage Courses	Allows Instructor to manage courses, including adding, editing, and deleting course details.
Manage Monitoring	Allows authorized users to manage the monitoring process, including preparing monitoring plan and setting deadlines for course and final project.
Project Details Screen	Is the interface that displays the details of a specific project, including the project title, description, objectives, methodology, and feedback/comments from the instructor/examiner to students.

UI Element	Description
Projects Screen	Allows users to search, view and download submitted projects.
Profile Screen	Allows users to view and edit their own profile details, including name, email, password, and contact information.

Table 3-1User interface identification

3.2.2 Business Rules Identification

Project Submissions

All student project submissions must be made through the SPTS. Students must log in to the system using their credentials to access the submission form. The submission form will require the student to provide information about the project, including the project title, abstract, and the name of the faculty member who will be supervising the project. The student must also upload a copy of their project in PDF or other approved formats.

Submission Deadline

The deadline for project submission will be determined by the School of Information, and it will be communicated to all students at the beginning of the semester. Students must submit their projects before the deadline. Any submission made after the deadline will not be accepted.

Project Review and Evaluation

All projects submitted through the SPTS will be reviewed by the faculty member assigned to supervise the project. The faculty member will provide feedback to the student on the project, and the student will be required to make any necessary changes before the final submission deadline. The final version of the project must be submitted through the SPTS before the deadline.

Project Ownership

The student will retain ownership of their project. However, by submitting the project through the SPTS, the student grants the School of Information permission to use their project for educational and research purposes. The School of Information may use the project in its coursework or for research purposes, provided that appropriate credit is given to the student.

Data Security and Confidentiality

The SPTS will ensure that all student project submissions are stored securely and confidentially. Access to student project submissions will be restricted to authorized personnel only. The system

will use industry-standard encryption technologies to ensure that all data stored in the system is secure and protected from unauthorized access or tampering.

System Availability and Support

The SPTS will be available 24/7 for students and faculty to access and submit projects. Technical support will be available during regular business hours to assist users with any issues they may encounter while using the system.

Compliance with School Policies

All users of the SPTS must comply with the policies and regulations of the School of Information. Failure to comply with the policies and regulations may result in disciplinary action, including but not limited to revocation of access to the SPTS.

3.2.3 Actor Identification

An actor is someone or something outside the system that interacts with the main system. An actor can be human being or another system. An actor is represented by a stick figure. In our system the main actors are:-

- Student
- Instructor
- Project coordinator
- Advisor
- Title approval committee
- Examiners

Each actor performs a specific role in the system.

3.2.4 Designing the Use case diagram

Use case diagram is a behavioural UML diagram type and frequently used to analyse various systems. They enable you to visualize the different types of roles in a system and how those roles interact with the system. It mainly had four parts.

Actor: represents an external entity that interacts with a system
Use case: A use case represents a function or an action within the system. It's drawn as an oval and named with the function.
System: The system is used to define the scope of the use case and drawn as a rectangle.
Relationship: The relationship between actor and the system and it can be.

List of use case

- Submit title
- Submit final project's
- Approve title
- Assign advisers
- Feedback
- Assign examiners
- Evaluate project's
- Submit project's
- Manage instructors
- Manage students
- Manage course
- Manage monitoring
- login and logout

3-1 Use case Diagram

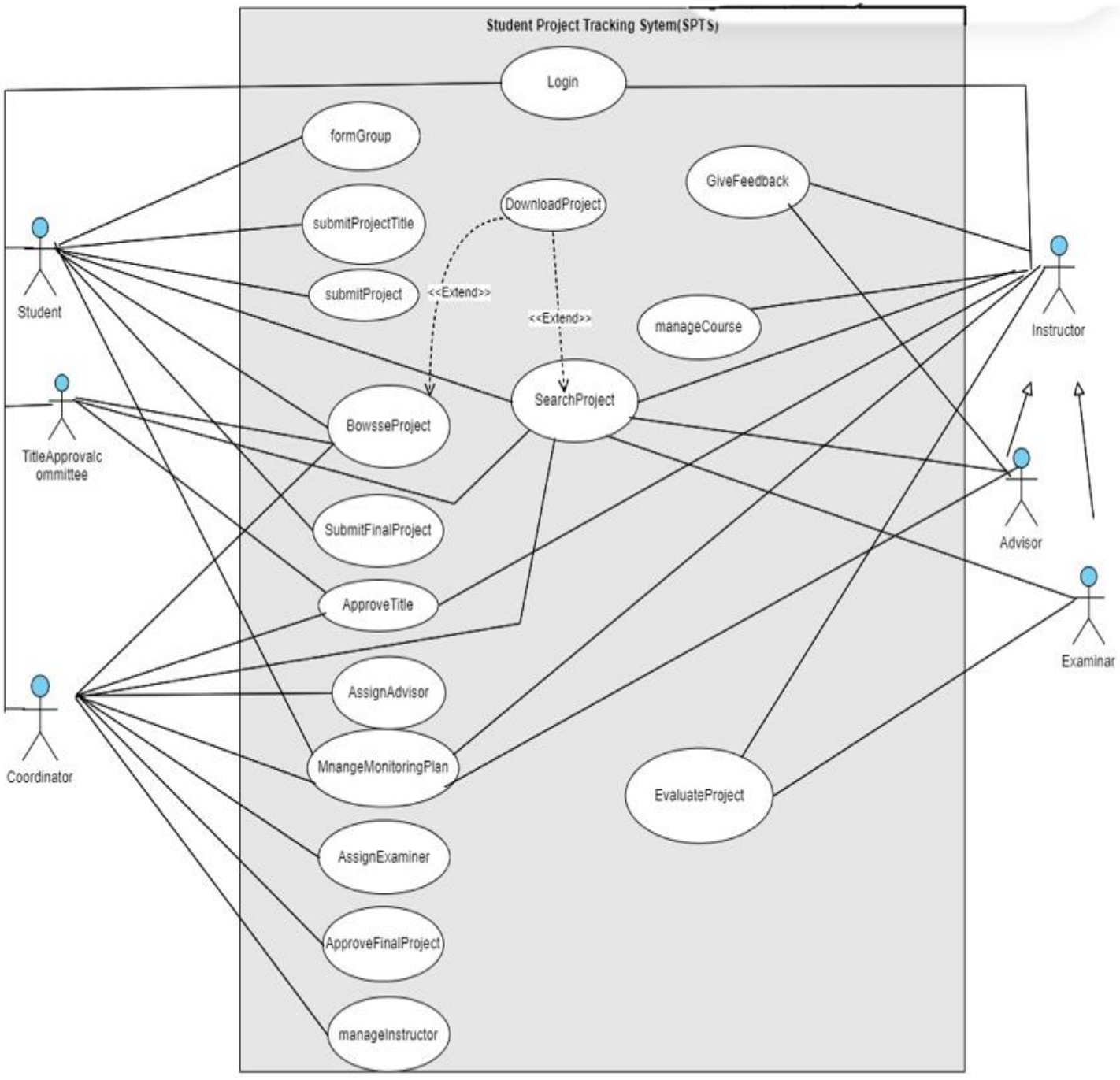


Figure 3-1 Use case diagram

3.2.5 Use case descriptions

Use case Name	**Login**
Use case Id	ST001
Actors	Student, coordinator,adviser, examiner,title approval committees
Description	This use case describes the process of a user logging into the system.
Precondition	The user has registered an account with the system and has valid login credentials.
Postconditio n	The user is logged into the system and is presented with their dashboard.
Main Flow	1. The user navigates to the login page. 2. The system displays the login form. 3. The user enters their username and password. 4. The system verifies the user's credentials. 5. If the credentials are valid, the system logs the user in and presents their dashboard. 6. If the credentials are invalid, the system displays an error message and prompts the user to enter valid credentials.

Table 3-2 Use case description for Login

Use Case Name	**Forming Groups**
Use case Id	**ST002**
Actors	Student
Description	This use case describes the process of a student forming a group with other students.
Precondition	The student must logged into the system to access the group formation feature.
Postconditio n	The student's group is created and stored in the system.
Main Flow	1. The student navigates to the group formation page. 2. The system displays a list of available students to form a group with. 3. The student selects other students to form a group with. 4. The system creates the group and assigns a group ID.
Alternate Flow	1. If the student cannot find other students to form a group with, they can select the "Find a Group" option to join an existing group.

Table 3-3 Use case description for forming Group

Use Case Name	**Submit Title**
Use case Id	ST003
Actors	Student
Description	This use case describes the process of a student submitting a title for their final project to the coordinator.
Precondition	The student is registered in the course and has logged into the system.
Postcondition	The title is submitted and stored in the system.
Main Flow	1. The student selects the "Submit Title" option from their dashboard. 2. The system prompts the student to enter a title. 3. The student enters the title and submits it. 4. The system confirms the submission and stores the title in the database.
Exception Flow	1. If the student enters an invalid title, the system displays an error message and prompts the student to enter a valid title.
Alternate Flow	Title is not unique; student is prompted to enter a different title.
Exception Flow	User authentication fails or system errors occur.
Special Needs	The system must be able to verify the uniqueness of project titles.

Table 3-4 Use case description for submit title

Use Case Name	**Approve Title**
Actors	Coordinator,Instructor
Use case Id	ST004
Description	This use case describes the process of a coordinator approving a student's final project title.
Precondition	The student has submitted their project title for approval.
Postcondition	The title is approved or rejected by the coordinator.
Main Flow	1. The coordinator selects the "Approve Title" option from manage project section of the dashboard 2. The system displays a list of submitted project titles. 3. The coordinator selects a title to review. 4. The coordinator approves or rejects the title. 5. The system updates the status of the title in the database as Accepted, Accepted with Modifications or Rejected and sends it to the submitter.
Alternate Flow	Coordinator disapproves title, system prompts student to submit another or approve with modification. title.

Exception Flow	User authentication fails or system errors occur.

Table 3-5. Use case description to Approve title

Use case Name	**Submit Project**
Use case Id	ST005
Actors	Student
Description	This use case describes the process of submission of a progress of the project by the students to their Advisors based on submission dates set by their coordinator.
Precondition	The project title must be approved
Post condition	The advisors and Instructors gives feedback to the submitted project progress
Main flow	1. The student selects the "Submit Project Progress" option from the student dashboard. 2. The system prompts the student to enter the status their final project progress. 3. The student chooses the project file and uploads and finally hits the submit Button. 4. The system confirms the submission and stores the document in the database. 5. use case ends

Table 3-6 Use case description for Submit project

Use Case Name	**Assign Advisers**
Actors	Coordinator
Use case Id	ST006
Description	This use case describes the process of a coordinator assigning advisers to students for their final project.
Precondition	The coordinator has access to a list of available advisers and a list of registered students.
Postcondition	Advisers are assigned to the students' final projects.
Main Flow	1. The coordinator selects the "Assign Advisers" option from the available Instructors list 3. The coordinator selects a student to assign an adviser to. 4. The system displays a list of available advisers. 5. The coordinator selects an adviser to assign to the student. 6. The system updates the adviser assignment in the database.

Table 3-7 Use case description for Assign Adviser

Use Case Name	**Give Feedback**
Actors	Adviser
Use case Id	ST007
Description	This use case describes the process of an adviser providing feedback to a student on their final project.
Precondition	The adviser has been assigned to the student's final project.
Postcondition	Feedback is provided to the student on their final project.
Main Flow	1. The adviser selects the "Feedback" option found next to the submitted project. 2. The system prompts the advisor to enter a list of comments for the progress of the final Project. 3. Use case ends

Table 3-8 Use case description for feedback

Use Case Name	**Submit Final Project**
Use Case ID	ST008
Actors	Students
Description	Allows students to submit their final project.
Precondition	User has access to the system and has logged in, project title has been approved.

Post condition	The system stores the submitted project and notifies the adviser.
Main Flow	1 Student selects their approved project title. 2 Student uploads their final project. 3 System notifies adviser.
Alternate Flow	None
Exception Flow	User authentication fails or system errors occur.
Special Needs	The system must be able to detect plagiarism in submitted projects.

Table 3-9 Use case description for Submit final project

Use Case Name	**Assign Examiners**
Actors	Coordinator
Use case Id	ST009
Description	This use case describes the process of a coordinator assigning examiners to evaluate a student's final project.
Precondition	The coordinator has access to a list of available examiners and a list of registered students who have submitted their final project.
Postconditio n	Examiners are assigned to evaluate the students' final projects.
Main Flow	1. The coordinator selects the "Assign Examiners" option from manage Examiners section the coordinators dashboard. 2. The system displays a list of students who have submitted their final project. 3. The coordinator selects a student to assign examiners to. 4. The system displays a list of available examiners. 5. The coordinator selects examiners to assign to the student. 6. The system updates the examiner assignment in the database.

Table 3-10 Use case description for Assign Examiner

Use Case Name	**Search project**
Use case Id	ST010
Actors	Coordinator,Instructor, Adviser, Student
Description	This use case describes the process of searching published projects with their keys.
Preconditio n	The user browse the student project tracking system
Postconditi	The user can download a project document or simply view the project.

on	
Main Flow	1. The use case starts when the system user wishes to search project. 2. The user query the project searching key on the search bar and press search icon. 3. The system retrieves published and stored projects with that key. 4. Use case ends
Exception	1. If there no project that matches the key the system returns nothing.
Extends	Download project

Table 3-11 Use case description for search project

Use Case Name	**Evaluate Project's**
Actors	Examiner
Use case Id	ST011
Description	This use case describes the process of an examiner evaluating a student's final project.
Precondition	The examiner has been assigned to evaluate the student's final project.
Postcondition	The student's final project has been evaluated and feedback has been provided.
Main Flow	1. The examiner selects the "Evaluate Project's" option from their dashboard. 2. The system displays a list of assigned projects to evaluate. 3. The examiner selects the student's project to evaluate. 4. The examiner evaluates the project based on predetermined criteria. 5. The examiner provides feedback on the project. 6. The system stores the evaluation and feedback in the database.

Table 3-12 Use case description to evaluate project

Use Case Name	**Manage Instructors**
Actors	Coordinator
Use case Id	ST012
Description	This use case describes the process of a coordinator managing instructors who are assigned to teach the course.
Precondition	The coordinator has access to a list of registered instructors and the course schedule.
Postcondition	The instructors' assignments are updated in the system.
Main Flow	1. The coordinator selects the "Manage Instructors" option from their dashboard. 2. The system displays a list of registered instructors.

	3. The coordinator selects an instructor to manage. 4. The system displays the instructor's current schedule. 5. The coordinator updates the instructor's schedule as necessary. 6. The system updates the instructor's schedule in the database.

Table 3-13 use case description for manage Instructors

Use Case Name	**Manage Students**
Actors	Coordinator
Use case Id	ST013
Description	This use case describes the process of a coordinator managing registered students in the course.
Precondition	The coordinator has access to a list of registered students.
Postcondition	The students' information and status are updated in the system.
Main Flow	1. The cordinater selects the "Manage Students" option from their dashboard. 2. The system displays a list of existing student records.

Table 3-14 use case description for manage Student

Use Case Name	**Manage Course list**
Actors	Instructor
Use case Id	ST014
Description	This use case describes the process of a coordinator managing the course details, such as course syllabus, course schedule, and course materials.
Precondition	The Instructor has access to the course management system and the necessary permissions to manage course details.
Postconditio n	The course details are updated in the system.
Main Flow	1. The Instructor selects the "Manage Course list" option. 2. The system displays a list of available course management options. 3. The Instructor selects the desired option, such as updating the course syllabus or schedule. 4. The system displays the corresponding form to update the selected option 5. The Instructor updates the course details as necessary. 6. The system updates the course details in the database.

Table 3-15 use case description for manage course

Use Case Name	**Manage Monitoring plan**
Actors	Coordinator, Instructor, Advisor,
Use case Id	ST015
Description	This use case describes the process of a coordinator monitoring the progress of the course, including student participation, instructor performance, and course outcomes.
Precondition	The coordinator has access to the course management system and the necessary permissions to monitor course progress.
Postconditio n	The course progress data is updated and stored in the system.
Main Flow	1. The coordinator or Instructor selects the "Manage Monitoring" option from their dashboard. 2. The system displays a list of available monitoring options, such as viewing student participation or instructor performance. 3. The coordinator selects the desired option. 4. The system displays the corresponding data or report. 5. The coordinator analyzes the data or report to monitor course progress. 6. The system stores the course progress data in the database.

Table 3-16 use case description for manage monitoring plan

Use Case Name	**Approve Final Project Document Submission**
Actors	Coordinator,Instructor
Description	This use case describes the process of a coordinator approving final project document submissions from students and Instructor for the course project.
Precondition	The coordinator is logged into the system and has access to the final project document submission feature.
Postconditio n	The final project document is approved and stored in the system.
Main Flow	1. The coordinator receives a notification of a final project document submission. 2. The coordinator navigates to the final project document submission page. 3. The system displays a list of submitted final project documents. 4. The coordinator selects the document to approve. 5. The system updates the document status and stores the document in the database.
Alternate Flow	If the coordinator or the instructor cannot approve the final project document, they can reject the document and provide feedback to the student.

Table 3-12 use case description for Approve Final project submission

3.3 Conceptual Modeling

3.3.1 Class Diagram

Class diagram is an essential part of the unified modeling language to show the object class in the system and associations between these classes. Class diagrams are most useful in illustrating relationships between classes and interfaces. Generalizations, aggregations, and associations are all valuable in reflecting inheritance, composition or usage, and connections respectively.

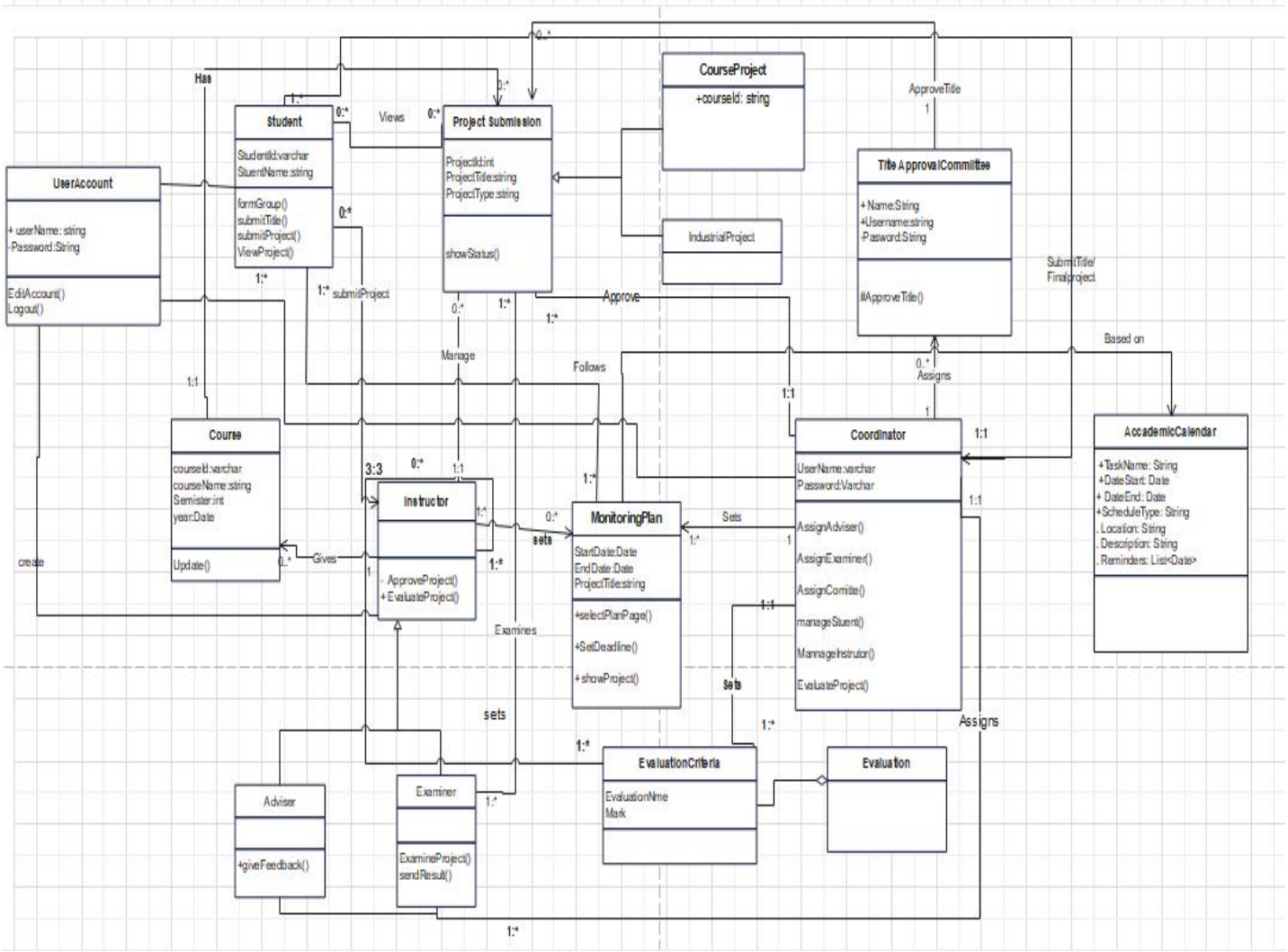


Figure 3-2 Class diagram

3.3.2 Class description

Class Name	Attributes	Data Type	Methods	Description
Course	course_id, course_name, Semister year	String, String, String, Instructor	upateCourse()	A course that is offered in the program. It has a course code, name, description, and an instructor who teaches the course. Students can be added or removed from the course.
Student	student_id, student_name, student_email,	String, String, String,	formGroup(), submit_project(project), view_project()	A student who is enrolled in the program. They have a student ID, name, and email address. They can submit a project and view their Project. Can submit projects.
Approval Committee	User_name, password, Name	String,string,string	approveTitle()	Approval committee approves titles submitted by students.
Instructor	instructor_id, instructor_name, instructor_email password	String, String, String String	Approve_project (p roject, student), evaluate_project (project)	An instructor who teaches a course. They have an instructor ID, name, and email address. They can assign projects to students and grade them.
Project	project_id, project_name, project_description, status	String, String, String, String	update_status(status), get_status()	A project that a student is working on. It has a project ID, name, description, and status. The status can be updated, and the current status can be retrieved.
Coordinator	coordinator_id, coordinator_name, coordinator_email	String, String, String	add_advisor(advisor), remove_advisor(advisor)	A coordinator who oversees the program. They have a coordinator ID, name, and email address. They can add or remove advisors for students.
Examiner	examiner_id, examiner_name, examiner_email	String, String, String	evaluate_project (project, evaluation_criter	An examiner who evaluates a project. They have an examiner ID, name, and email

			ia)	address. They can evaluate a project using evaluation criteria.
Advisor	advisor_id, advisor_name, advisor_email	String, String, String	create_monitoring_plan(), update_monitoring_plan()	An advisor who advises a student on their project. They have an advisor ID, name, and email address. They can create and update a monitoring plan for the student's project.
Monitoring Plan	plan_id, plan_name, plan_description, tasks, due_date	String, String, String, List, Date	add_task(task), remove_task(task), update_due_date(date)	A monitoring plan for a student's project. It has a plan ID, name, description, tasks, and a due date. Tasks can be added or removed, and the due date can be updated.
Evaluation Criteria	criteria_name, Mark	String, String,	add_subcriteria(subcriteria), remove_subcriteria(subcriteria)	Evaluation criteria used to evaluate a project. It has a criteria ID, name, and description. Sub-criteria can be added or removed.
Academic Calendar	task_name, start_date,end-date,schedule_time, location, desc, reminder	String,Date,Date, String, String, String, Date	getInstance()	The calendar class can help identify tasks and milestones for the monitoring plan, while the monitoring plan can provide actionable steps for completion of those tasks within the set timeframe.
Course Project	course_project_id, course_project_name, course_project_description, course_id	String, String, String, String	view_course_projects()	A project that is assigned as part of a course. It has a project ID, name, description, and a course ID. Course projects can be viewed.

Table 3-13 class description table

3.4 Sequence diagram

UML offers a large set of constructs for each of its diagram. UML Sequence Diagrams are interaction diagrams that detail how operations are carried out. Sequence diagram shows interaction between objects over specific of time. One sequence diagram typically represents a single Use Case development or flow of events. It shows the flow of messages from one object to another, and as such correspond to the methods and events supported by a class or object. The diagram illustrated below shows major sequence diagrams for SPMS, with the user or actor on the left initiating a flow of events and messages that correspond to the Use Case scenario. The messages that pass between objects will become class operations in the final model.

Figure 3-3 Sequence diagram for login

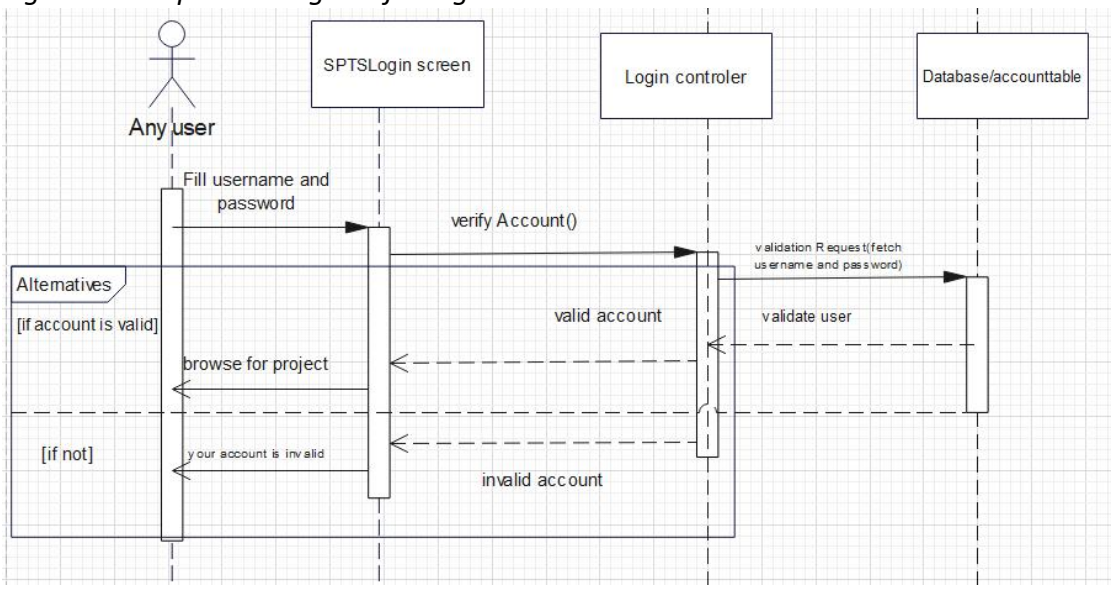
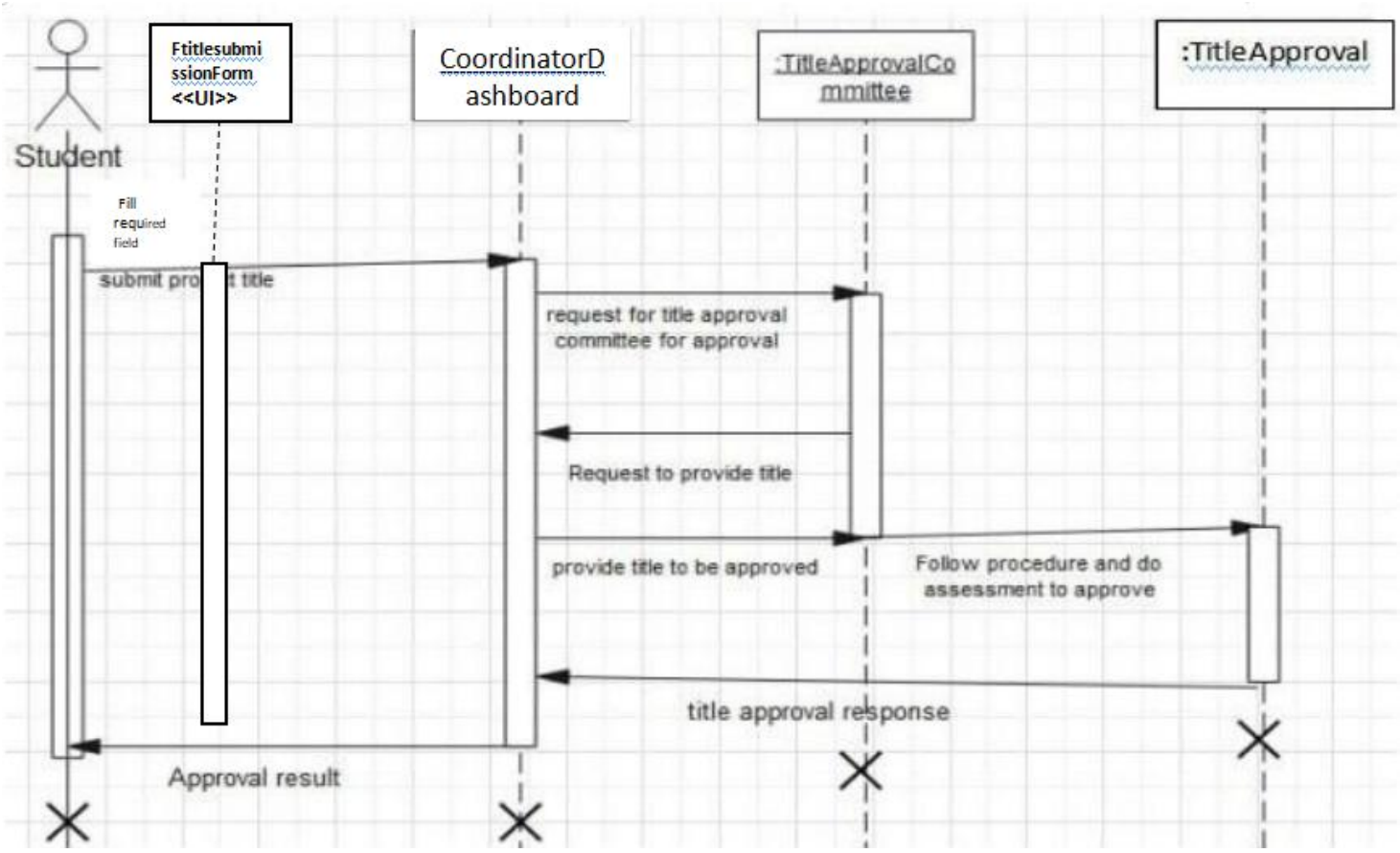


Figure 3-4 Sequence diagram for Approve title


Student
Ftitlesubmi
ssionForm
<<UI>>
CoordinatorD
ashboard
:TitleApprovalCo
mmittee
:TitleApproval
Fill
required
field
request for title approval
committee for approval
Request to provide title
provide title to be approved
Follow procedure and do
assessment to approve
title approval response
Approval result

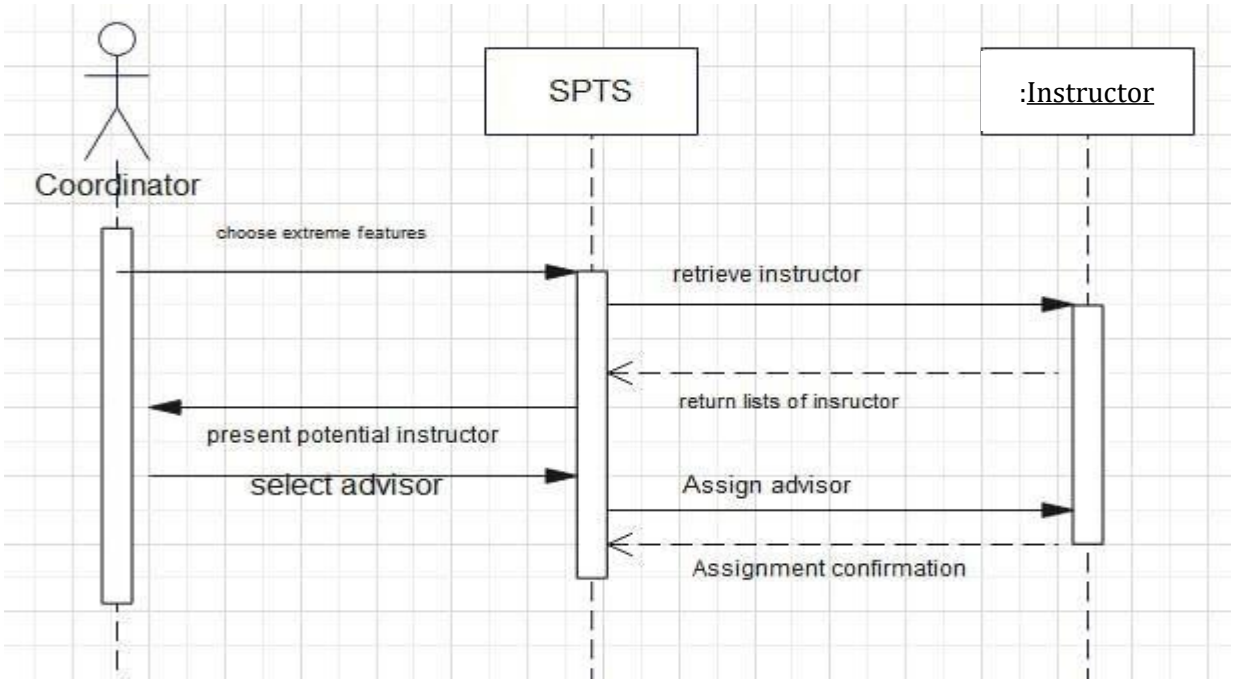


Figure 3-7 Sequence diagram for Assign Adviser

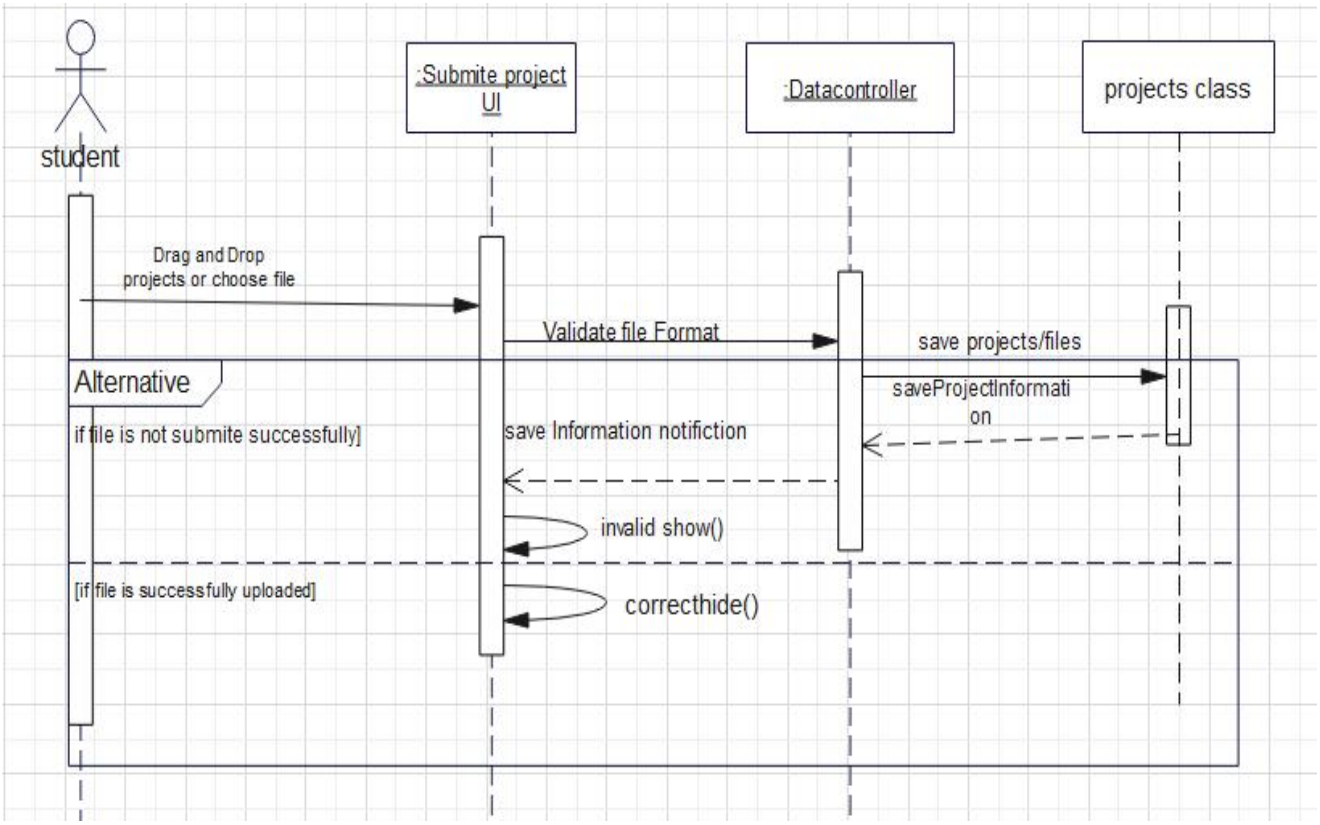


Figure 3-6 Sequence diagram for submit project

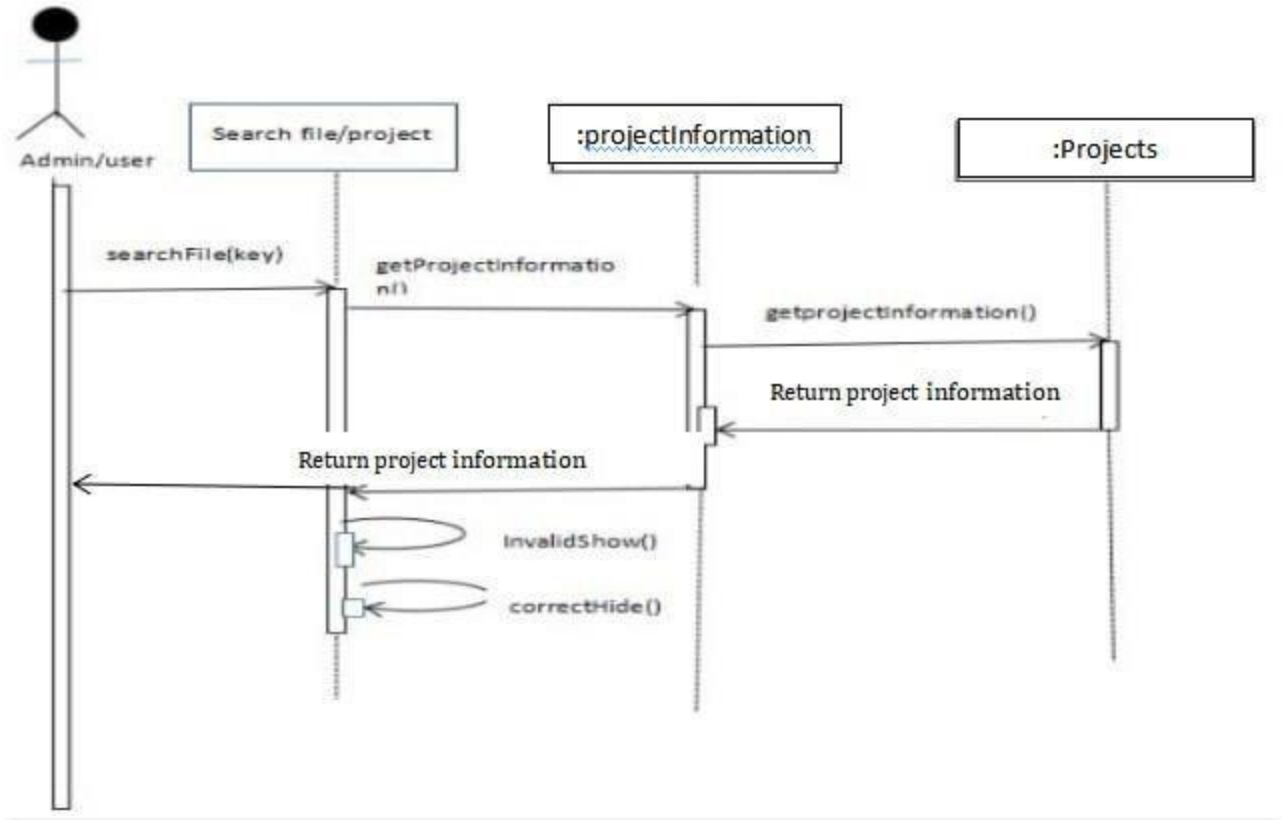
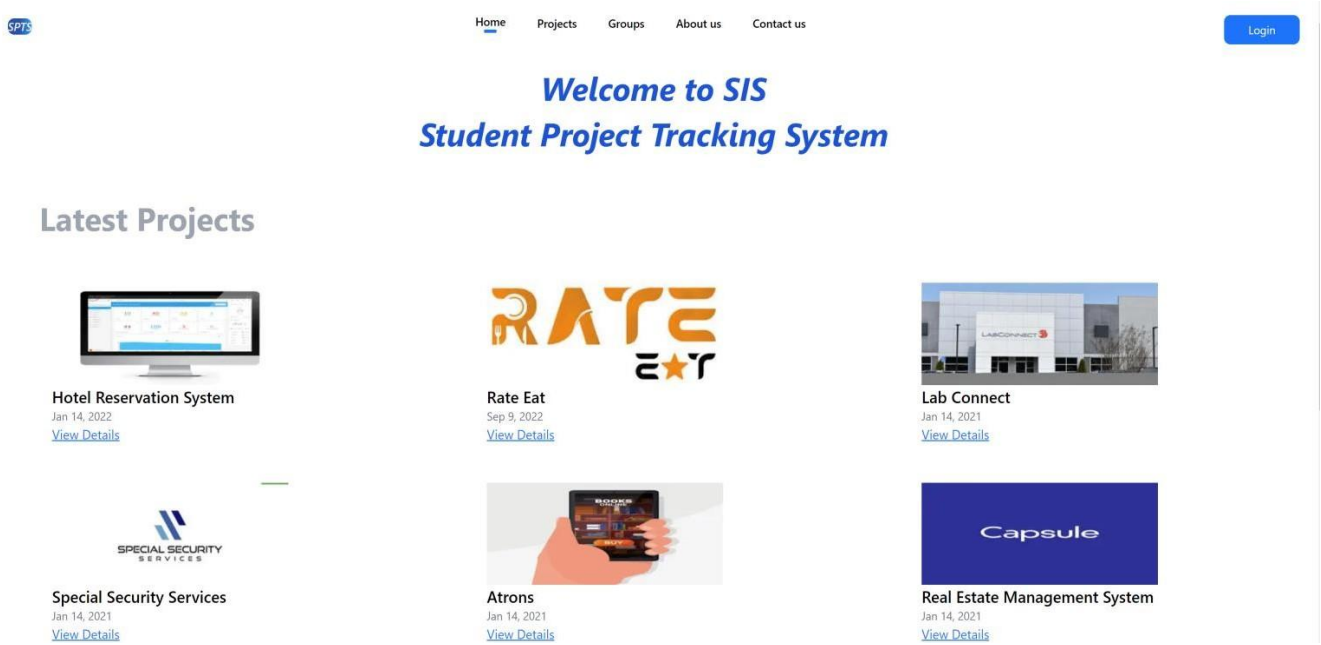


Figure 3-7 Sequence diagram for search project

3.5 User Interface Prototype

1. Home Page



Login page

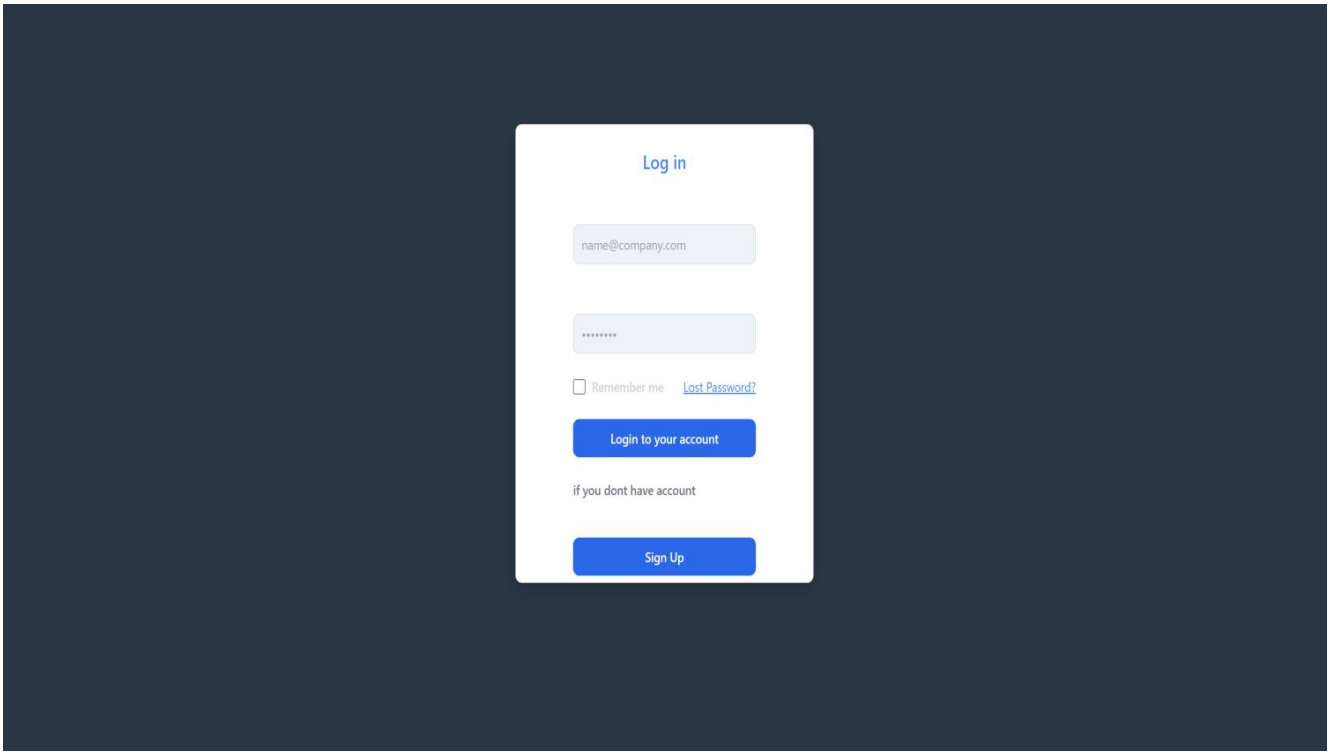


Figure 3-8 UI for Login Page

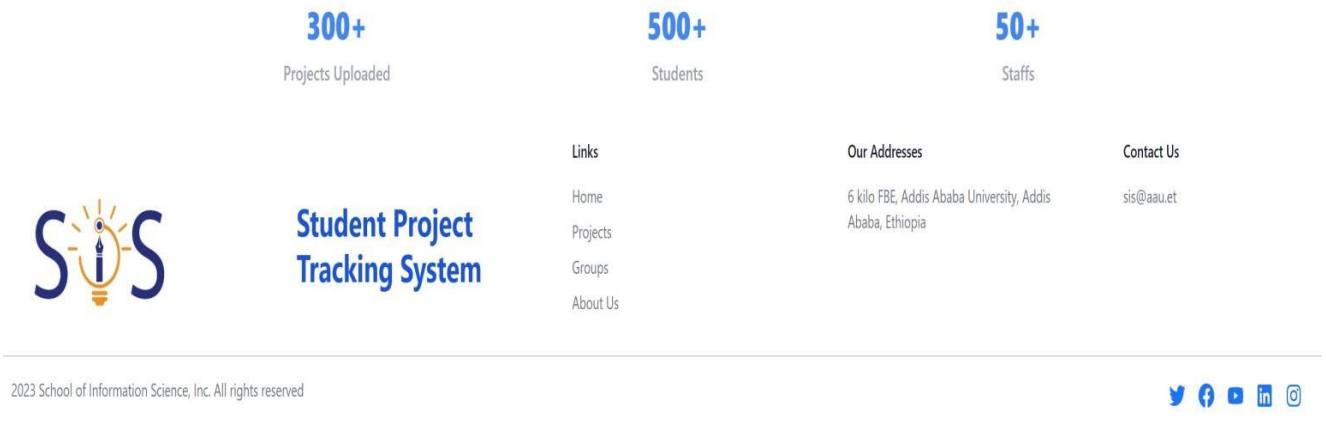


Figure 3-8 UI for Home Page

2. Coordinator Dashboard

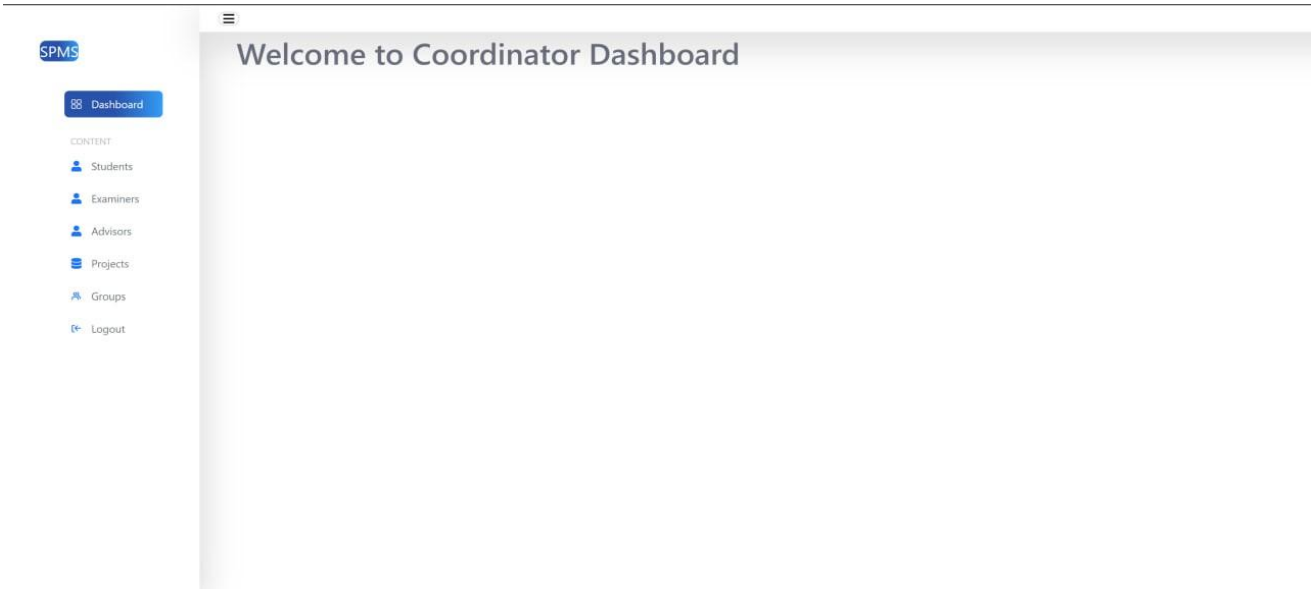


Figure 3-10 UI for Coordinator Dashboard

3. Add Student

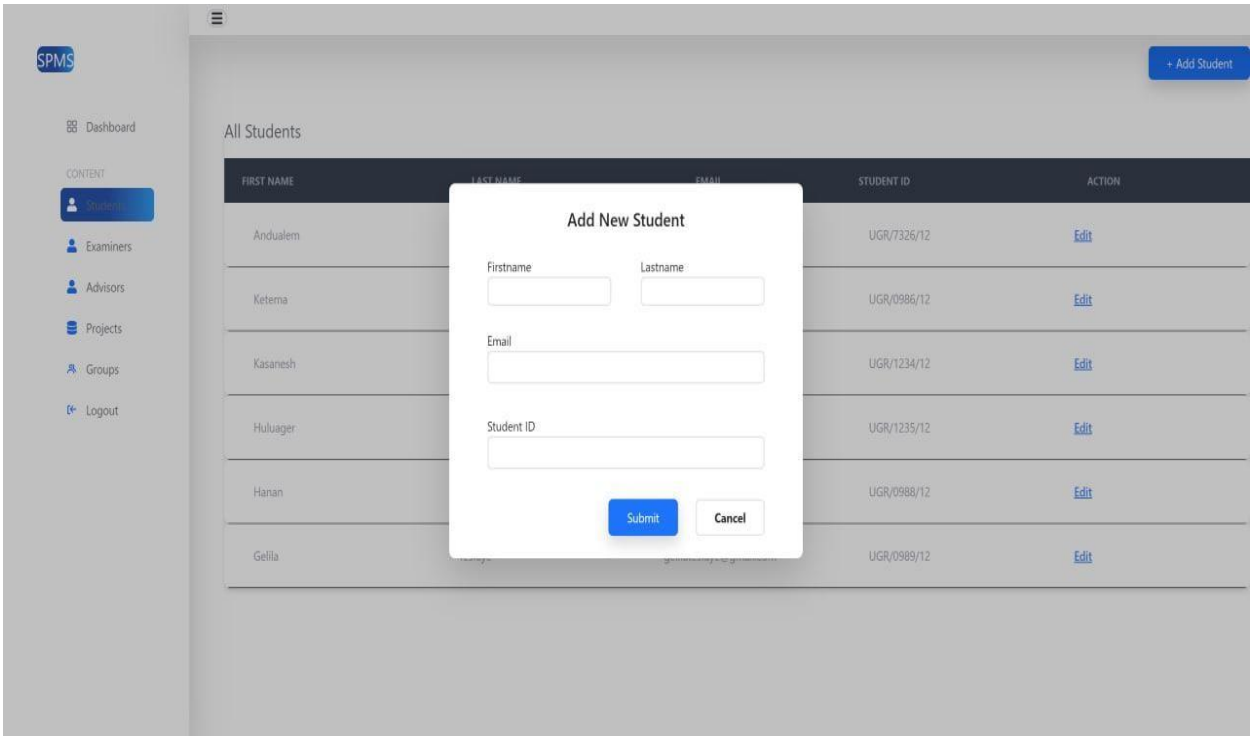


Figure 3-11 UI to ad student

4. Students List

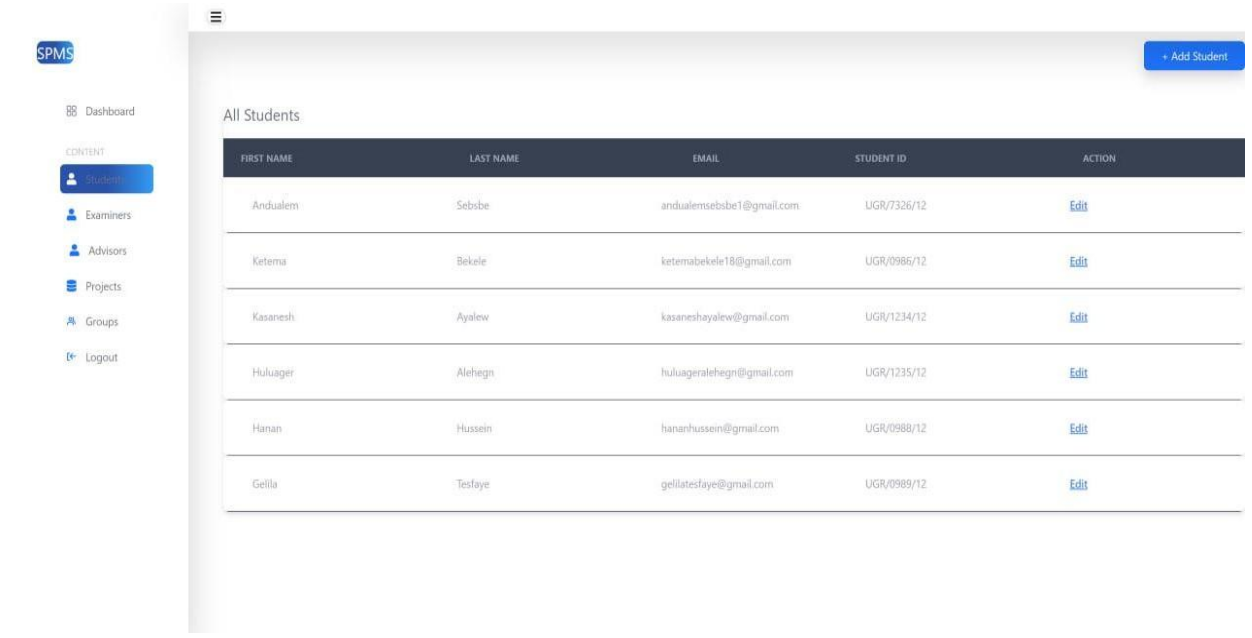


Figure 3-12 UI for students list

5. Add Project

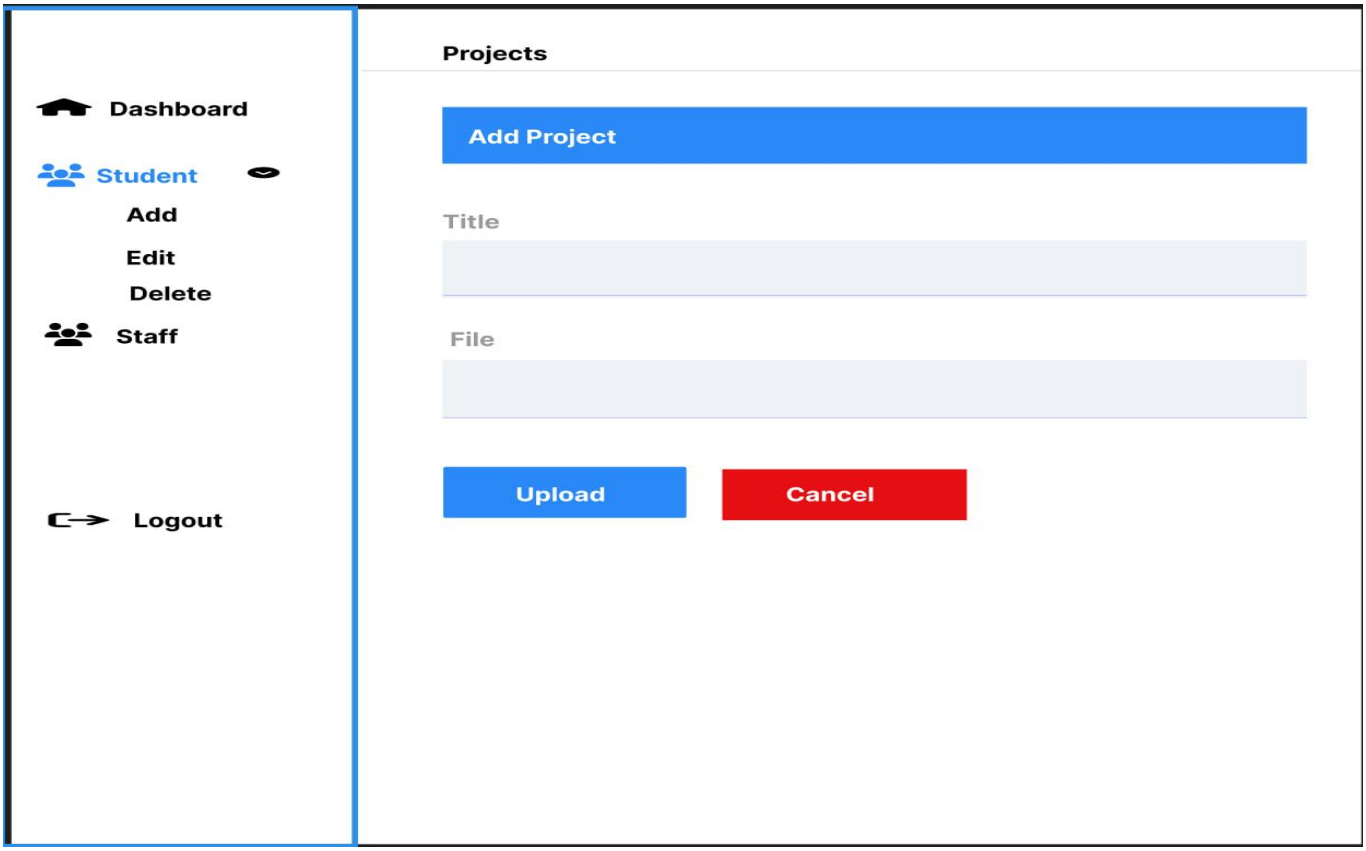


Figure 3-13 UI to add project

6. Projects List

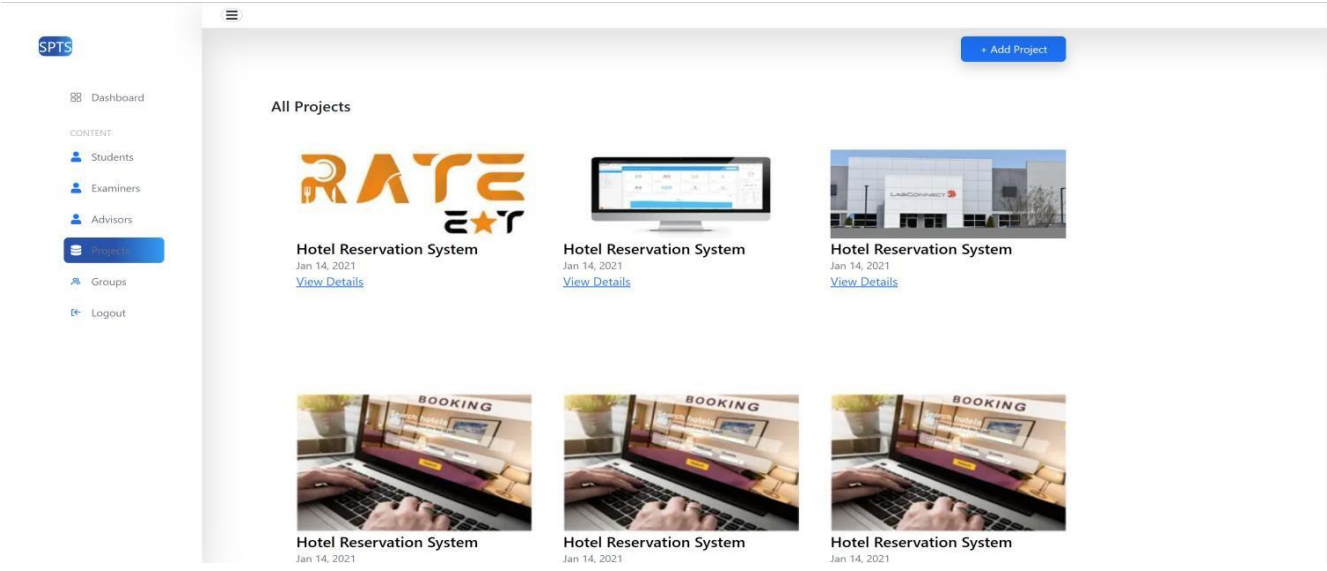


Figure 3-14 UI for project list

7. Project Detail

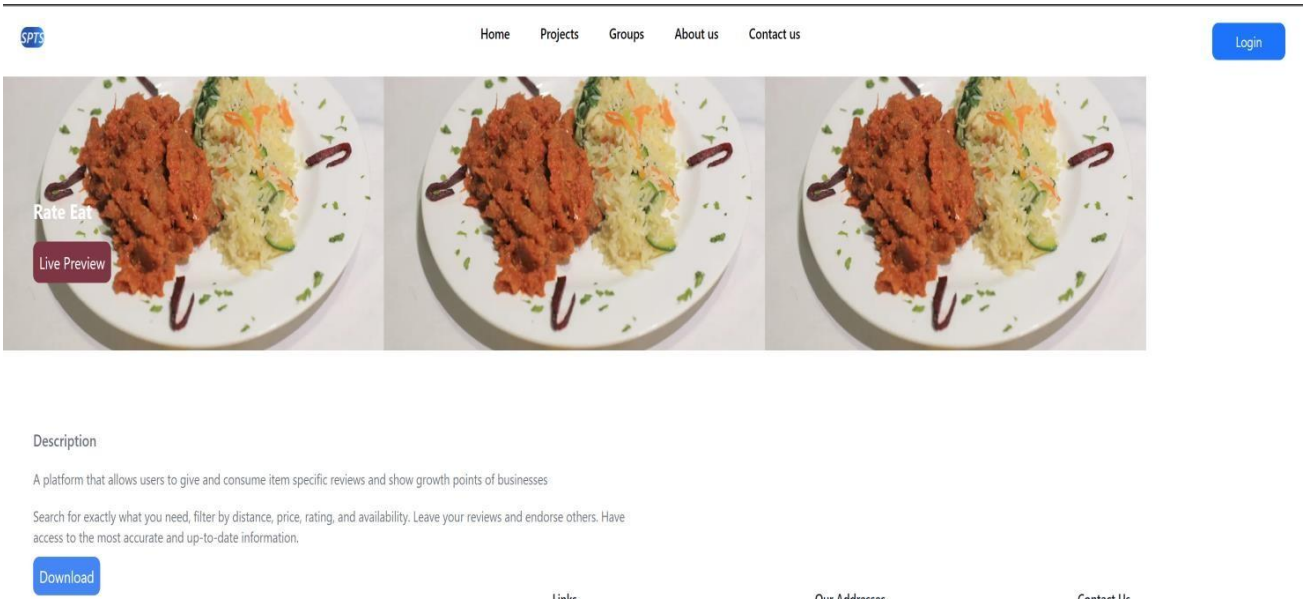


Figure 3-15 UI for project detail

8. About us

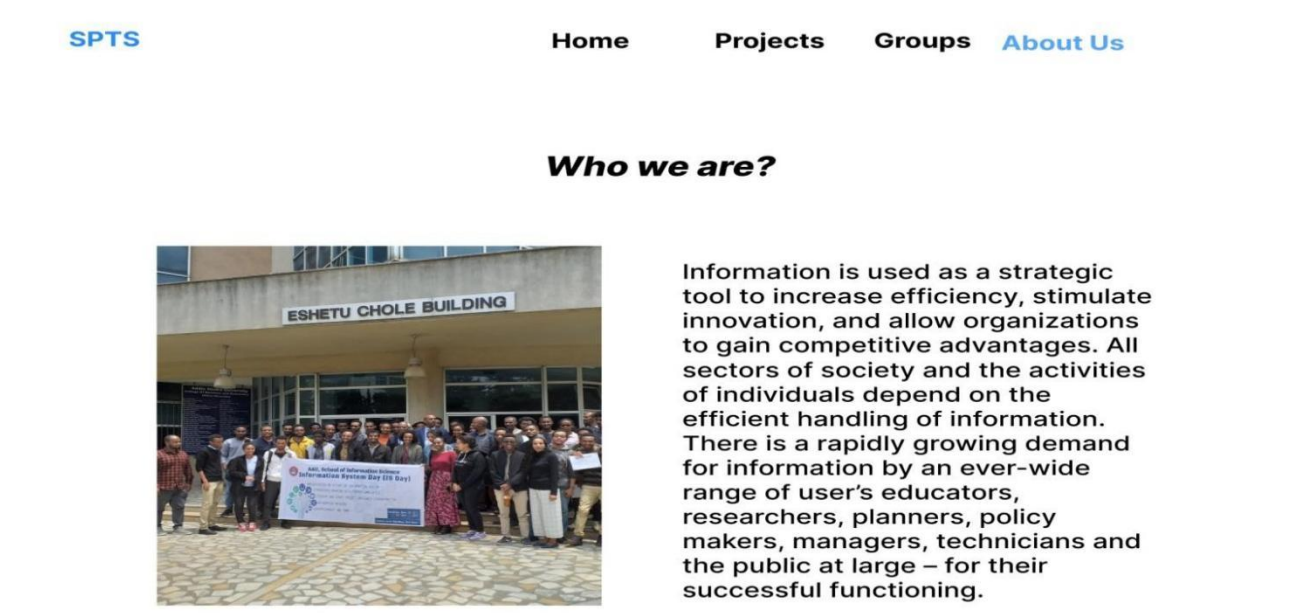


Figure 3-16 UI for about us

4 Conclusions and Recommendations

4.1 Conclusions

This paper shows the initial step for developing a web based student project tracking system for SIS at Addis Ababa University. It provides an electronic form of creating, storing, retrieving and controlling projects. Moving from a current manual work system to a web based system helps the SIS to reduce costs, automating processes increased project security and minimizing errors. The main information collection method was interview; the nature of the interview was semi- structure with a basic guideline. During the interview numerous discussions with different employees made and challenges of the manual recording process are identified. Generally this School Program Management system will benefit the SIS Employee's, and the students. It reduces the

Time spent looking for project approval and submission to ensure that the system users will find important information timely. This will enhance and improve their personal productivities.

4.2 Recommendations

To enhance the efficiency of student project tracking system the following recommendations are made for the future works.
To facilitate easy information access in relation to projects and other to stakeholders in the department thesis and project portal system need to be integrated with the student project tracking system(SPTS) .

References

[1] T.Leikums, "a study on electronic document management system integration needs in Public sector," International Journal of Advances in Engineering & Technology, vol. 5, No. 1, pp. 194-205, 2012.

K.Latham, L. W and T. Gorichanaz, "A Discussion on Document Conceptualization," The Document academy, vol. 3, no. 2, 2016.