

Project Plan



University Project Management System

Group Members

Inam karim 04072212031
Shakir Rabbani 04072312040
Aftab Shaban 04072312046

Software Construction (CS-322)

Submitted to:
DR. ONAIZA MAQBOOL

QUAID-I-AZAM UNIVERSITY ISLAMABAD

DEPARTMENT OF COMPUTER SCIENCE

This is to clarify that the semester project entitled as

UNIVERSITY PROJECT MANAGEMENT SYSTEM

Is the Bonafide record of semester project work done by

INAM KARIM

SHAKIR RABBANI (team lead)

AFTAB SHABAN

of BSCS during Spring-2025

Supervisor

Head of Department

Declaration

We affirm that this Semester project entitled as University Project Management being submitted in fulfillment for the award of 5th semester of **bachelor's in computer science** is the original work carried out by us.

Inam Karim

Shakir Rabbani

Aftab Shaban

Preface

The Department of Computer Science requires a management system to streamline the final semester project process. This project will manage project registrations, facilitate communication between students and supervisors, and provide a structured platform for document submissions and feedback.

The intended users of this project documentation are the faculty members and administrative staff overseeing final semester projects, as well as technical personnel responsible for maintaining the system.

Change History

Version 1.0

Defined Project Scope, Objectives and Summary

Version 1.1

Project Context, References and Definitions

Version 1.2

Project Planning and Risk Management

Version 1.3

Updated the Gantt Chart and Tables

Version 2.0

Draw use case diagram

Version 2.1

Domain model

Use Case text

System Sequence diagram

Table of Contents

1. Project Overview	5
a. Case study	5
b. Project summary	5
i. Purpose	5
ii. Scope	5
iii. Objectives	5
iv. Assumption and Constraints	6
v. Project deliverables.	6
vi. Schedule Summary	7
2. References	8
3. Definition	8
4. Project Context	9
a. Process Model	9
• Iterative Waterfall Model	9
b. Methods, tools and techniques	9
c. Product Acceptance plan	10
5. Project Planning:	10
a. Work activities:	10
b. Schedule Allocation	11
	11
c. Resource Allocation	12
6. Supporting Process Plans	12
a. Risk Management:	12
i. Team members leaving:	13
ii. Loss of Data:	13
iii. Power Breakdown:	13
iv. Technical Challenges:	13
v. Lack of User Adoption:	13
7. Use case diagram	14
8. Domain Model	15
9. Usecase description	16
1. Register for a project: (By Inam karim)	16
2. Upload Work Submission: (By Shakir Rabbani team lead)	16
3. Provide Feedback (By Aftab Shaban)	17
10. System sequence diagram	17
1. Register for a project: (By Inam karim)	17
2. Upload Work Submission: (By Shakir Rabbani)	18

1. Project Overview

a. Case study

The Department of Computer Science wants to develop a project management system. Every semester, the department offers projects to final semester students. These projects are registered by students with supervisors. This project deals with communication between a student and supervisor during the project. It allows all projects to be loaded and supervisors to be defined. Subsequently, throughout the semester, the website can be used for uploading work products (by student) and feedback (by supervisor). The website allows multiple versions of a single work product to be submitted.

b. Project summary

i. Purpose

The purpose of this project is to maintain the complete project information of a department. There are different types of projects that are offered for different programs such as BS, MSc etc. so this project will maintain all types communication, structured work submission, and feedback mechanisms.

ii. Scope

In this project the system will be able to perform the following functionalities

- Student project registration with supervisors.
- Supervisor assignment and project definition.
- Work product uploads and version control.
- Feedback and grading system.
- Secure login and user roles (students, supervisors, admins).

iii. Objectives

- Enhance project collaboration between students and supervisors.
- Enable easy submission and review of work products.
- Provide structured feedback for iterative improvements.
- Maintain a historical record of project versions.

iv. Assumption and Constraints

- **Schedule**

The university required this to be functional at the start of the new semester. So, this project must be completed within this semester.

- **Scope**

The system will enable students to register their projects with assigned supervisors, while supervisors can define projects and provide feedback on student submissions. It will support multiple versions of work product submissions to track progress and revisions. Secure authentication will be implemented for both students and supervisors to ensure data security and access control. It will feature structured messaging and notifications to facilitate clear and organized communication between users.

- **Software**

For frontend HTML, CSS and JavaScript will be used for this system and for backend PHP will be used in order to make this system fully functional.

v. Project deliverables.

- **Project Plan**

In this the whole project plan will be included in document form as a summary of the project schedule design in project libre. The project manager will approve the project plan based on the available resources.

- **Analysis Documentation**

This documentation includes the brief analysis of project plan and deliver to the internal team member for any type of modification in the project plan. A brief instruction will be included in this document that will help to easily trace the project plan part to be updated.

- **Prototype/Low-Fidelity**

Before implementation of the project a low-fidelity prototype will be discussed with the stakeholder in order to make changes for better experience. This design can easily be modified so there will be no need to change in the high-fidelity.

- **Software Requirement Specification**

The SRS will describe the goal of the project and the way users will interact with the system. This also includes how the project will be completed. This will be delivered to the client for agreement and after the approval of this the implementation of the project will start.

- **Fronend of System**

This will deliver to the end user a high-level fidelity without any backend to ensure better user interaction with the system. A manual for interaction will also be provided as a part of documents in this project deliverable part.

- **Database Connected System.**

This deliverable project part provides the system with database connectivity.

vi. Schedule Summary

- **Week 1-2 Planning**

The first week is allocated for project planning which includes the following main point, and its work product will be the project plan.

Project Overview

Project Context

Risk Management

Schedule Allocation

Date 3 March 2025 -14 March 2025

- **Week 3-5 Analysis**

The next three weeks are allocated for analysis, and this includes the following main points, and its work product will SRS.

Use case Diagram and Text.

Data Flow Diagram etc.

Software requirement specification.

FTR for analysis

Date 17 March 2025 - 8 April 2025

- **Week 6-9 Design**

Low-Fidelity Design

High-Fidelity Design

Detailed Design

Date 9 April 2025 - 7 May 2025

- **Week 10 - 13 Development**

Database Connectivity

Refining Front-End

Development of Classes

Date 8 May 2025 – 5 June 2025

- **Week 14 Testing**

Reviewing Software

Presentation

Date 6 June 2025- 12 June 2025

2. References

- [ieee-documentation-template](#)
- Systems and software engineering — Life cycle processes — Project management ISO/IEC/IEEE 16326:2019(E) Second edition 2019-12
- [Project Libre](#)
- Chapter 31-35 Pressman ISO/IEC/IEEE 16326-2019 guideline

3. Definition

- PMP:

Project management plan is the world's leading project management certification.

- SRS:

Software requirement specification is usually called a work product of a project plan. This document required a signature of stakeholders to move toward the development phase.

- IDE:

An integrated development environment (IDE) is an application that facilitates the development of other Applications

- FTR:

Formal Technical Review is a software quality control activity performed by software engineers.

4. Project Context

a. Process Model

- **Iterative Waterfall Model**

Iterative Waterfall Model follows a structured, phase-based approach and is simple and easy to use. We are using this model because we want to use a sequential approach for our project. Secondly, we also have less time and a dedicated portion of time for each phase, so we want to complete one whole portion before going to the next one and the feedback loops enable revisiting and improving previous phases if necessary. This ensures both stability and flexibility of the project.

b. Methods, tools and techniques

Using the Iterative Waterfall model, we are using a sequential and iterative approach. When we work on one phase we are not focusing on any other phase and after moving to the next phase if we want, we can go back to the previous phase as well. This method will help us to use the dedicated portion for each phase effectively and revisit and improve previous phase if necessary.

As we are using a iterative waterfall model for the development of this project, we are using a tool which enables us to proceed from one phase to another in a seamless way to meet the project plan.

- **Programming Languages:** JavaScript (for frontend), PHP (for backend)

JavaScript is used for developing the interactive and dynamic frontend of the application, ensuring a responsive and user-friendly interface. PHP is chosen for backend development as it is well-suited for web applications, providing efficient server-side scripting and database connectivity.

- **Frameworks & Libraries:** React.js (frontend), Laravel (backend)

React.js allows for modular and reusable component-based UI development, improving maintainability and scalability. Laravel is a PHP framework that simplifies backend development with built-in security features, routing, and database management.

- **Database:** (PostgreSQL)

PostgreSQL is used for data storage due to its reliability, scalability, and support for complex queries and transactions.

- **Presentation:** (Microsoft PowerPoint)

At the end of this project, we are required to give presentation of our project and we will definitely use this software widely used for presentation purpose to give our final presentation.

- **Version Control:** (GitHub)

GitHub is used for version control, enabling collaborative development, tracking changes, and ensuring code integrity.

- **Communication Tools:** Microsoft Teams, Email

Microsoft Teams is used for team discussions, file sharing, and virtual meetings, while email ensures formal communication and documentation.

- **Development Tools:** Visual Studio Code, PHPStorm, Postman (API testing)
Visual Studio Code and PHPStorm provide efficient coding environments with debugging and syntax highlighting features. Postman is used for API testing to ensure smooth backend functionality.
- **Project Management Tools:** Project Libre
Project Libre is used for project scheduling and timeline management. It provides a range of tools and functionalities which we are looking for to complete our project plan.

c. Product Acceptance plan

As we follow the Iterative Water Fall Model in our project, we provide deliverables to our client for review periodically after two weeks. To do this, we use Formal Technical Reviews (FTR), during which our users can dually check our work and provide guidance on the next steps. We will ensure a high-quality product is a part of our product acceptance plan, which calls for holding weekly meetings to review current products and discuss the next steps

The system will undergo rigorous testing before deployment. The acceptance process will include:

1. **Unit Testing:** Developers will test individual components to ensure they work as expected.
2. **Integration Testing:** Verify that different modules work together correctly.
3. **User Acceptance Testing (UAT):** Students and supervisors will test the system to confirm it meets requirements.
4. **Final Approval:** The project will be validated by the department before deployment.

5. Project Planning:

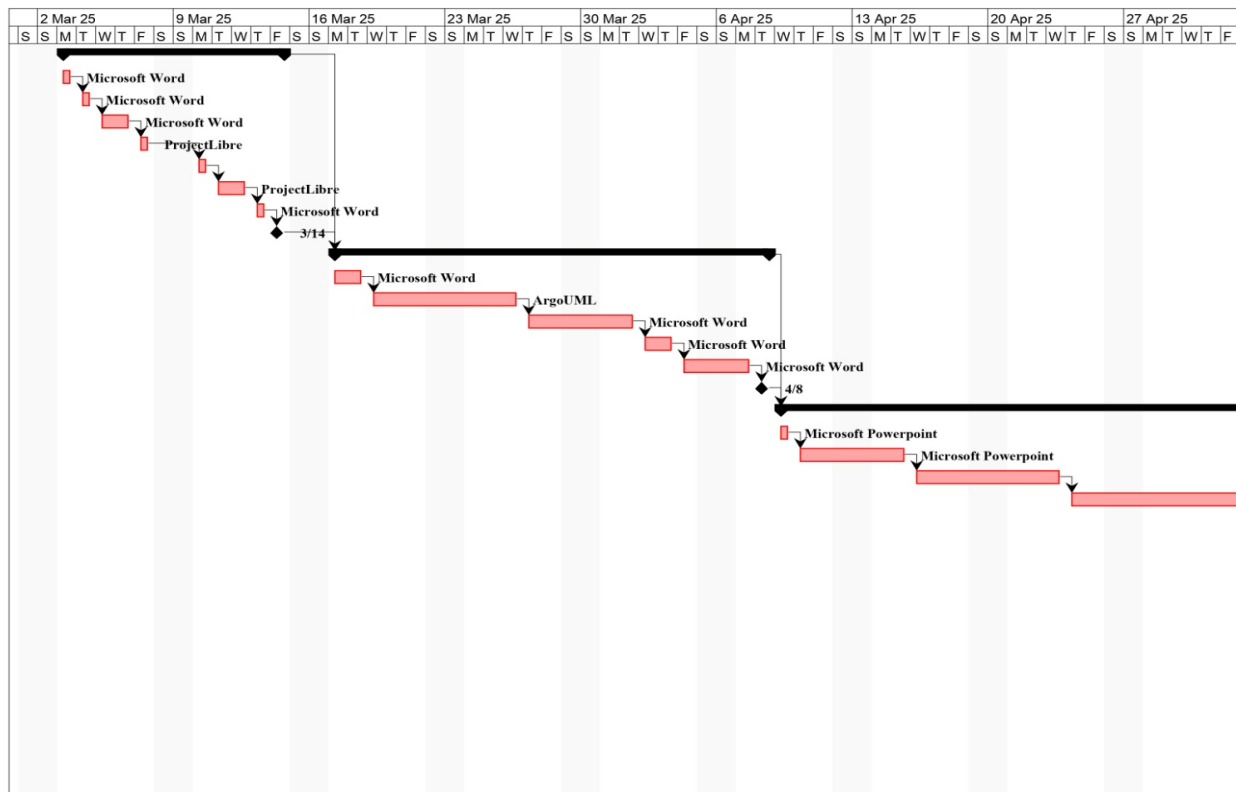
a. Work activities:

The work activities should be represented using a work breakdown structure where an accurate determination of the resources needed and the time frame for each work activity is possible. It contains the timeframe, resources allotted to the project, and all of the tasks required for each project phase.

b. Schedule Allocation

		Name	Duration	Start	Finish	Predecessors	Resource Names
1		planning	10 days?	3/3/25 8:00 AM	3/14/25 5:00 PM		
2		project overview	1 day	3/3/25 8:00 AM	3/3/25 5:00 PM		Microsoft Word
3		Project Frount Mater	1 day?	3/4/25 8:00 AM	3/4/25 5:00 PM	2	Microsoft Word
4		Defining resources	2 days	3/5/25 8:00 AM	3/6/25 5:00 PM	3	Microsoft Word
5		Budget planning	1 day?	3/7/25 8:00 AM	3/7/25 5:00 PM	4	ProjectLibre
6		Time planning	1 day	3/10/25 8:00 AM	3/10/25 5:00 PM	5	
7		define project context	2 days	3/11/25 8:00 AM	3/12/25 5:00 PM	6	ProjectLibre
8		Risk Management Plan	1 day?	3/13/25 8:00 AM	3/13/25 5:00 PM	7	Microsoft Word
9		Plan Featured	1 day?	3/14/25 8:00 AM	3/14/25 5:00 PM	8	
10		Analysis	17 days?	3/17/25 8:00 AM	4/8/25 5:00 PM	1;9	
11		Define Usecases	2 days	3/17/25 8:00 AM	3/18/25 5:00 PM		Microsoft Word
12		Develop Analysis Model	6 days	3/19/25 8:00 AM	3/26/25 5:00 PM	11	ArgoUML
13		Develop SRS	4 days	3/27/25 8:00 AM	4/1/25 5:00 PM	12	Microsoft Word
14		Review SRS	2 days	4/2/25 8:00 AM	4/3/25 5:00 PM	13	Microsoft Word
15		Refine SRS	2 days	4/4/25 8:00 AM	4/7/25 5:00 PM	14	Microsoft Word
16		Reviewed and Refined SRS	1 day?	4/8/25 8:00 AM	4/8/25 5:00 PM	15	Microsoft Word
17		Design	21 days?	4/9/25 8:00 AM	5/7/25 5:00 PM	10;16	
18		Discussing Design Principle	1 day	4/9/25 8:00 AM	4/9/25 5:00 PM		Microsoft Powerpoint
19		Data Design	4 days	4/10/25 8:00 AM	4/15/25 5:00 PM	18	Microsoft Powerpoint
20		Interface Design	6 days	4/16/25 8:00 AM	4/23/25 5:00 PM	19	
21		Detailed Desing	9 days	4/24/25 8:00 AM	5/6/25 5:00 PM	20	
22		Final Product Desing	1 day?	5/7/25 8:00 AM	5/7/25 5:00 PM	21	
23		Development	21 days?	5/8/25 8:00 AM	6/5/25 5:00 PM	17;22	
24		Database connectivity	5 days	5/8/25 8:00 AM	5/14/25 5:00 PM		Visual Studio Code
25		Refining Front end	5 days	5/15/25 8:00 AM	5/21/25 5:00 PM	24	Visual Studio Code
26		Develop Classes	5 days	5/22/25 8:00 AM	5/28/25 5:00 PM	25	Visual Studio Code
27		Refining Classes	2 days	5/29/25 8:00 AM	5/30/25 5:00 PM	26	Visual Studio Code
28		Tracing classes with desi...	3 days	6/2/25 8:00 AM	6/4/25 5:00 PM	27	Microsoft Powerpoint
29		Working Product	1 day?	6/5/25 8:00 AM	6/5/25 5:00 PM	28	
30		Testing	5 days?	6/6/25 8:00 AM	6/12/25 5:00 PM	23;29	
31		Reviewing the Develop...	1 day?	6/6/25 8:00 AM	6/6/25 5:00 PM		Visual Studio Code;Testers
32		Removing Errors	4 days	6/9/25 8:00 AM	6/12/25 5:00 PM	31	Visual Studio Code
33		Final Testing Product	1 day	6/13/25 8:00 AM	6/13/25 5:00 PM	30;32	
34		Presentations	10 days	6/13/25 8:00 AM	6/26/25 5:00 PM	30	Microsoft Powerpoint

assignment gantt chart - page1



assignment gantt chart - page2

i. Team members leaving:

- If any member is leaving, who has a significant role in software management, provides him with a more feasible environment or provide him with the desired package if he deserves this.
- Moreover, keep at least 2 employees at the back end so if anyone leaves you must have a backup person to lessen your loss

ii. Loss of Data:

- To avoid data loss, one must have at least 2 backup plans. One in the hard form and one in the soft form. So, if in any case your data is lost one must already know how to cope with it.

iii. Power Breakdown:

- If due to any reason power breakdown occur and your work get deleted unsaved one must have remedies defined for them as generators even work manually we have to switch on them so as to avoid this the whole system should be connected to solar plates so that if power supply get off within less than a second it is connected to solar continuing the work at its best.

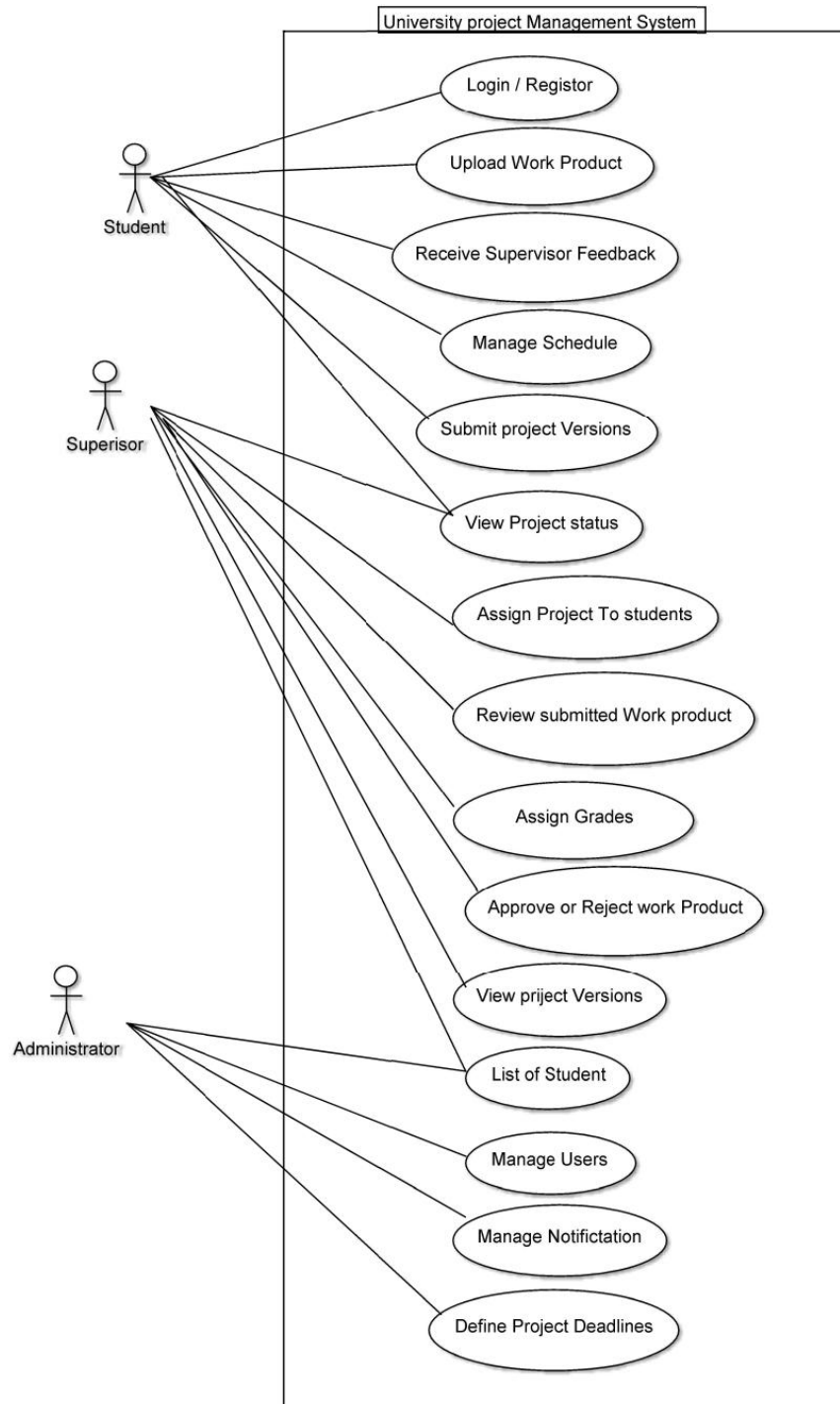
iv. Technical Challenges:

- Integration issues between frontend (React.js) and backend (Laravel) so, conduct regular integration testing and maintain proper documentation.

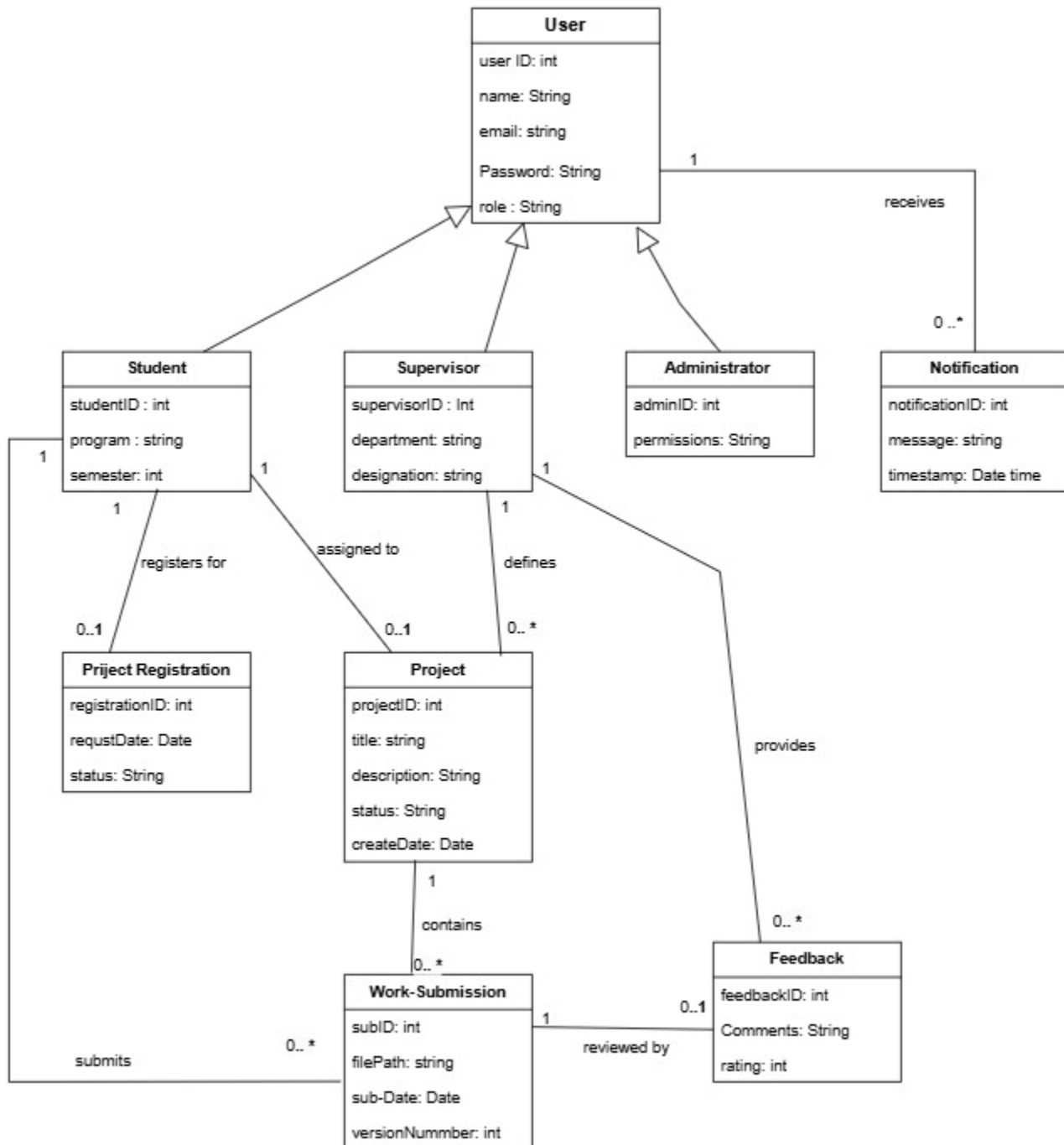
v. Lack of User Adoption:

- Students and supervisors may not use the system as expected so, we can provide user training and detailed documentation to ease adoption.

7. Use case diagram



8. Domain Model



9. Usecase description

1. Register for a project: (By Inam karim)

Actor: Student

I. Preconditions:

- The student must be logged into the system.
- Projects must be available by the department for selection.
- Student must be in his final year.

II. Main Success Scenario:

1. The student logs into the system.
2. The system displays a list of available projects.
3. The student selects a project and submits a registration request.
4. The system records the request and notifies the supervisor.
5. The supervisor reviews and approves/rejects the request.
6. The student is notified of the decision.

III. Alternative Flow:

- 3) If the project is already assigned, the system prevents registration.
- 5) If the supervisor rejects the request, the student is asked to select another project.

IV. Postconditions:

- If approved, the student is assigned to the project.
- If rejected, the student must choose another project.

2. Upload Work Submission: (By Shakir Rabbani team lead)

Actor: Student

I. Preconditions:

- The student must be assigned a project.

II. Main Success Scenario:

1. The student logs into the system.
2. The system displays the assigned project.
3. The student uploads a work product file.
4. The system stores the file and notifies the supervisor.
5. The student sees a confirmation message.

III. Alternative Flow:

- 3) If the file format is incorrect, the system rejects the upload.
- If the file exceeds the allowed size, the system rejects the upload.

IV. Postconditions:

- The work product is stored in the system.
- The supervisor can review the submission.
- The submission version is incremented and stored.

3. Provide Feedback (By Aftab Shaban)

Actor: Supervisor

I. Preconditions:

- A student must have submitted a work product.

II. Main Success Scenario:

1. The supervisor logs in.
2. The system displays student submissions.
3. The supervisor selects a submission and adds feedback.
4. The system saves the feedback and notifies the student.

III. Alternative Flow:

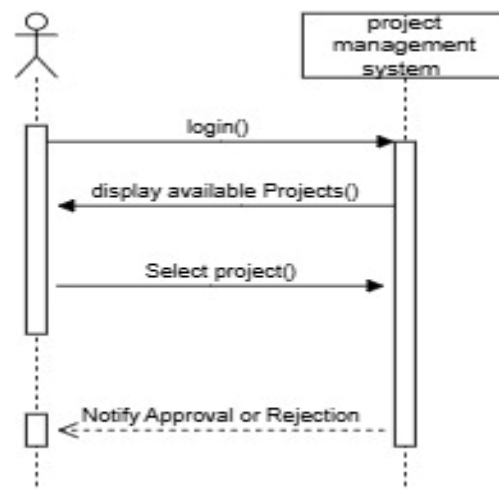
- If the feedback is missing, the system prompts the supervisor to enter a comment.

IV. Postconditions:

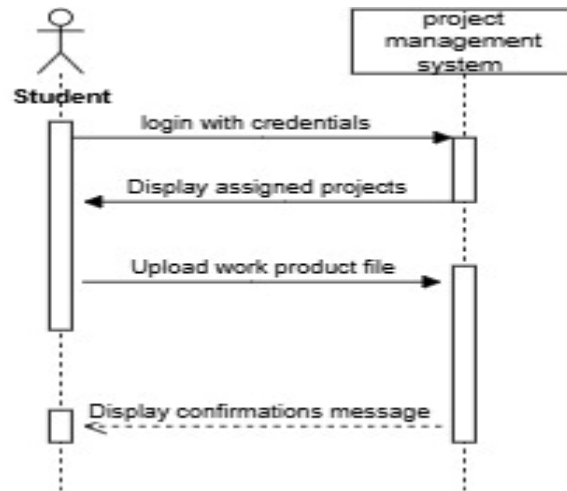
- The student is notified of the feedback.
- The feedback is stored for future reference.

10. System sequence diagram

1. Register for a project: (By Inam karim)



2. Upload Work Submission: (By Shakir Rabbani)



3. Provide Feedback (By Aftab Shaban)

