QUAID-I-AZAM UNIVERSITY
ISLAMABAD

STUDENT COURSE REGISTRATION AND TRACKING SYSTEM
PROJECT PLAN

By

PROJECT TEAM LEAD: USMAN MUSTAFA

TEAM MEMBER 01: MAHDI AHRARI

TEAM MEMBER 02: GHULAM MUJTABA

DATE: MARCH 05 , 2025

SUPERVISED BY: DR. ONAIZA MAQBOOL
DEPARTMEWNT OF COMPUTER SCIENCE
Quaid-I-Azam University, Islamabad

SIGNATURE PAGE

Project Team Lead:

Name: Usman Mustafa Signature: __________

Reg#: 04072213047 Date: 05/03/25

Team Member 01:

Name: Mahdi Ahrari Signature:

Reg#: 04072213005 Date : 05/03/25

Team Member 02:

Name: Ghulam Mujtaba Signature:

Reg#: 04072213019 Date: 05/03/25

Project Stakeholder:

Name: Dr. Onaiza Maqbool Signature:

Post: Associate Professor Date:

CHANGE HISTORY

Version	Date	Changed By	Description
1.0	MARCH 05, 25	G13	Initial Draft
1.1			
1.2			

Remarks:

(Signature)

Dr. Onaiza Maqbool

Associate Professor

Preface:

The purpose of this document is to outline the project management plan for the development of the Course Registration Management System. This plan defines the scope, objectives, and methods to be used, following the guidelines of ISO/IEC/IEEE 16326.

Table of Contents

1. Project Overview

1.1 Project Summary

The Computer Science (CS) office requires a system to efficiently manage student course registrations and track their pass/fail status each semester. This system will enforce pre-requisite checking and maintain a detailed record of courses taken, passed, or skipped. The system will provide accessibility to the coordinator and timetable coordinator to ensure streamlined academic management.

1.2 Purpose, Scope, and Objectives

1.2.1 Purpose:

The purpose of this system is to automate and streamline student course registration while ensuring compliance with pre-requisite requirements. Additionally, the system will provide faculty and administration with easy access to student academic progress.

1.2.2 Scope:

- Uploading the scheme of study for each batch.
- Defining courses and their respective pre-requisites.
- Enforcing pre-requisite checks during course registration.
- Providing course history for each student.
- Enabling the coordinator to track academic progress.
- Allowing the timetable coordinator to access student course information.

1.2.3 Objectives:

- Improve efficiency in student course registration.

- Reduce manual errors in pre-requisite checking.
- Provide real-time academic progress tracking.
- Facilitate better course planning and scheduling.

1.3 Assumptions and Constraints

1.3.1 Assumptions:

- Students must register for courses within the designated registration period.
- Each course has a clearly defined set of pre-requisites.
- Faculty and administrative users will have appropriate access levels.
- The system will be web-based and accessible via secure login credentials.

1.3.2 Constraints:

- The system must adhere to university academic policies.
- Student data privacy must be maintained as per institutional guidelines.
- Internet connectivity is required for system access.
- Changes to course structures require administrative approval.

1.4 Project Deliverables

- Project plan documentation
- Software Requirements Specification (SRS)
- System design (Prototype)
- A fully functional course registration system
- Project presentation

1.5 SCHEDULE SUMMARY

PHASE/TASK	**DURATION**	**DELIVERABLES**
Planning	5-03-25 to11-03-25	
Stakeholder meeting		
Development of project plan		Project-Plan Documentation
Analysis	12-03-25 to 17-03-25	
Defining usecase		
Analysis modeling		
SRS development		SRS Documentation
Design	18-3-25 to 16-04-25	
Data Design		
Interface design		
Detailed design		Design (Prototype)
Development	17-04-25 to 15-05-25	
Database Connectivity		
Front-end refinement		
Classes Development		Class Diagram
Refinement of classes and traceability with design		

Testing	16-05-25 to 28-05-25	
Testing of the software System		Student registration system
Deployment	29-05-25 to 02-06-25	
Project Presentation		

2. REFERENCES

- ISO/IEC/IEEE 16326:2019, Systems and software engineering -Life cycle processes- Project management.
- Case Study#3 – Problem description provided by the teacher.

3. DEFINITIONS

- **Coordinator:** The faculty member responsible for managing student course registrations and overseeing academic progress.
- **Timetable Coordinator:** The faculty member who schedules courses and ensures conflict-free timetabling.
- **Pre-requisite:** A course that must be completed before a student can register for a specific subsequent course.
- **Scheme of Study:** The predefined structure of courses that students must follow to complete their degree program.
- **Pass/Fail Status:** The academic standing of a student based on their performance in registered courses.

4.PROJECT CONTEXT

4.1 PROCESS MODEL

The project will follow the Agile Methodology, specifically **SCRUM**. The project involves multiple functionalities (student course registration, prerequisite checking, academic tracking, etc.) which can be developed, tested, and improved in iterations known as sprints. This approach is suitable for the dynamic nature of software development, ensuring flexibility and adaptability to changing requirements. Given the team's need for an evolving system, Agile will also facilitate incremental learning and improvement. Each development phase will comprise requirements gathering, design, implementation, testing, and evaluation before moving on to the next iteration.

This model also emphasizes:

- Regular team meetings in each sprint to identify risks early.
- Adapting development strategies in case new requirements arise.

4.2 METHODS, TOOLS, AND TECHNIQUES

Development Tools:

- **Eclipse IDE:** For coding and debugging.
- **Django or Spring Boot:** For server-side development.
- **PostgreSQL/MySQL:** For database management.
- **HTML/CSS, JavaScript:** For front-end design and interaction.
- **MVC Architecture:** For separating logic, presentation, and data layers.

Programming Language:

- JavaScript

Project Management Tools:

- Project Libre

4.3 PRODUCT ACCEPTANCE PLAN

The Product Acceptance Plan will involve:

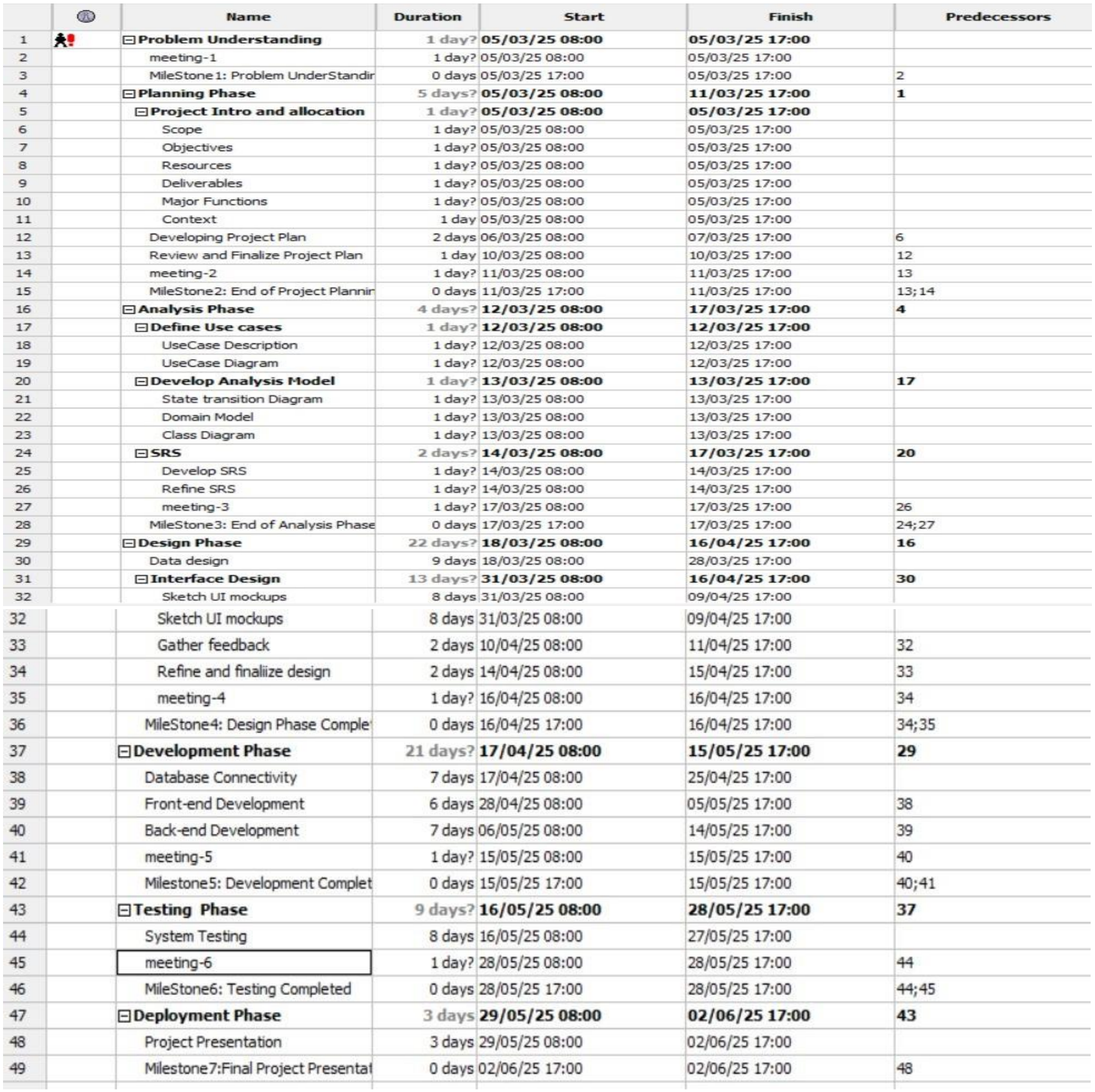
- **User Acceptance Testing (UAT):** Conducted by faculty coordinators and students to validate functionality and usability.
- **Performance Testing:** Ensuring smooth system operations under high student registration loads.
- **Security Testing:** Verification of secure student data storage and controlled access.
- **Acceptance Criteria:** Defined for each module, ensuring alignment with student registration and academic tracking requirements.
- **Final Approval:** Deployment upon successful testing and feedback from academic staff.

5. PROJECT PLANNING

5.1 PROJECT WORKPLANS

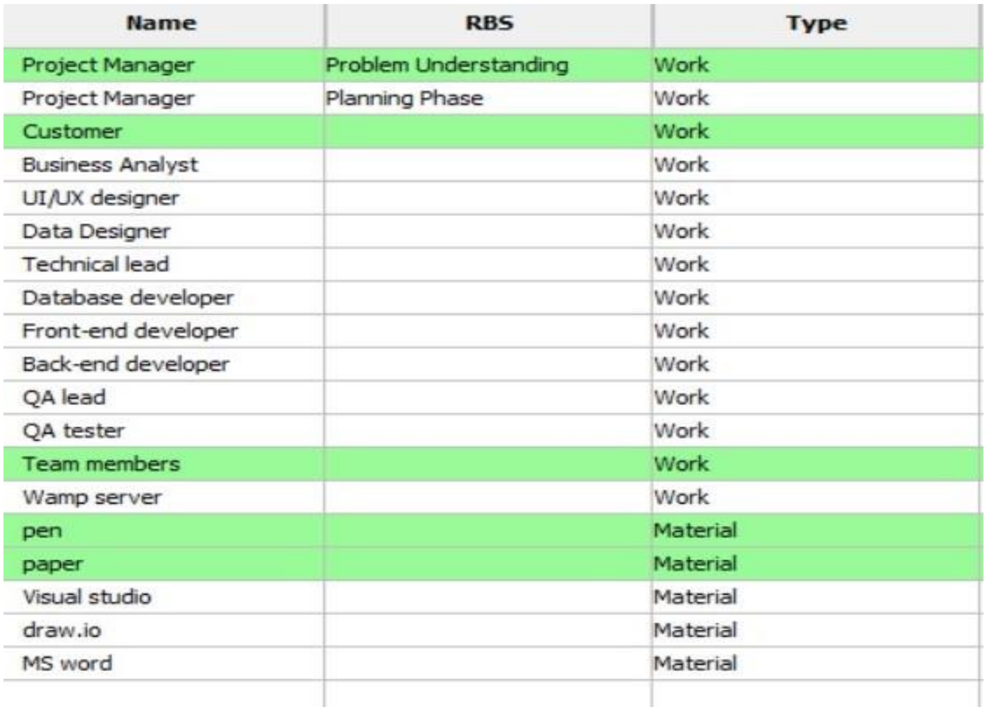
5.2 WORK ACTIVIYIES

5.3 SCHEDULE ALLOCATION



		Name	Duration	Start	Finish	Predecessors
1		**⊟Problem Understanding**	1 day?	**05/03/25 08:00**	**05/03/25 17:00**	
2		meeting-1	1 day?	05/03/25 08:00	05/03/25 17:00	
3		MileStone1: Problem UnderStandir	0 days	05/03/25 17:00	05/03/25 17:00	2
4		**⊟Planning Phase**	5 days?	**05/03/25 08:00**	**11/03/25 17:00**	**1**
5		**⊟Project Intro and allocation**	1 day?	**05/03/25 08:00**	**05/03/25 17:00**	
6		Scope	1 day?	05/03/25 08:00	05/03/25 17:00	
7		Objectives	1 day?	05/03/25 08:00	05/03/25 17:00	
8		Resources	1 day?	05/03/25 08:00	05/03/25 17:00	
9		Deliverables	1 day?	05/03/25 08:00	05/03/25 17:00	
10		Major Functions	1 day?	05/03/25 08:00	05/03/25 17:00	
11		Context	1 day	05/03/25 08:00	05/03/25 17:00	
12		Developing Project Plan	2 days	06/03/25 08:00	07/03/25 17:00	6
13		Review and Finalize Project Plan	1 day	10/03/25 08:00	10/03/25 17:00	12
14		meeting-2	1 day?	11/03/25 08:00	11/03/25 17:00	13
15		MileStone2: End of Project Plannir	0 days	11/03/25 17:00	11/03/25 17:00	13;14
16		**⊟Analysis Phase**	4 days?	**12/03/25 08:00**	**17/03/25 17:00**	**4**
17		**⊟Define Use cases**	1 day?	**12/03/25 08:00**	**12/03/25 17:00**	
18		UseCase Description	1 day?	12/03/25 08:00	12/03/25 17:00	
19		UseCase Diagram	1 day?	12/03/25 08:00	12/03/25 17:00	
20		**⊟Develop Analysis Model**	1 day?	**13/03/25 08:00**	**13/03/25 17:00**	**17**
21		State transition Diagram	1 day?	13/03/25 08:00	13/03/25 17:00	
22		Domain Model	1 day?	13/03/25 08:00	13/03/25 17:00	
23		Class Diagram	1 day?	13/03/25 08:00	13/03/25 17:00	
24		**⊟SRS**	2 days?	**14/03/25 08:00**	**17/03/25 17:00**	**20**
25		Develop SRS	1 day?	14/03/25 08:00	14/03/25 17:00	
26		Refine SRS	1 day?	14/03/25 08:00	14/03/25 17:00	
27		meeting-3	1 day?	17/03/25 08:00	17/03/25 17:00	26
28		MileStone3: End of Analysis Phase	0 days	17/03/25 17:00	17/03/25 17:00	24;27
29		**⊟Design Phase**	22 days?	**18/03/25 08:00**	**16/04/25 17:00**	**16**
30		Data design	9 days	18/03/25 08:00	28/03/25 17:00	
31		**⊟Interface Design**	13 days?	**31/03/25 08:00**	**16/04/25 17:00**	**30**
32		Sketch UI mockups	8 days	31/03/25 08:00	09/04/25 17:00	
32		Sketch UI mockups	8 days	31/03/25 08:00	09/04/25 17:00	
33		Gather feedback	2 days	10/04/25 08:00	11/04/25 17:00	32
34		Refine and finaliize design	2 days	14/04/25 08:00	15/04/25 17:00	33
35		meeting-4	1 day?	16/04/25 08:00	16/04/25 17:00	34
36		MileStone4: Design Phase Comple	0 days	16/04/25 17:00	16/04/25 17:00	34;35
37		**⊟Development Phase**	21 days?	**17/04/25 08:00**	**15/05/25 17:00**	**29**
38		Database Connectivity	7 days	17/04/25 08:00	25/04/25 17:00	
39		Front-end Development	6 days	28/04/25 08:00	05/05/25 17:00	38
40		Back-end Development	7 days	06/05/25 08:00	14/05/25 17:00	39
41		meeting-5	1 day?	15/05/25 08:00	15/05/25 17:00	40
42		Milestone5: Development Complet	0 days	15/05/25 17:00	15/05/25 17:00	40;41
43		**⊟Testing Phase**	9 days?	**16/05/25 08:00**	**28/05/25 17:00**	**37**
44		System Testing	8 days	16/05/25 08:00	27/05/25 17:00	
45		meeting-6	1 day?	28/05/25 08:00	28/05/25 17:00	44
46		MileStone6: Testing Completed	0 days	28/05/25 17:00	28/05/25 17:00	44;45
47		**⊟Deployment Phase**	3 days	**29/05/25 08:00**	**02/06/25 17:00**	**43**
48		Project Presentation	3 days	29/05/25 08:00	02/06/25 17:00	
49		Milestone7:Final Project Presentat	0 days	02/06/25 17:00	02/06/25 17:00	48

5.4 RESOURCE ALLOCATION



Name	RBS	Type
Project Manager	Problem Understanding	Work
Project Manager	Planning Phase	Work
Customer		Work
Business Analyst		Work
UI/UX designer		Work
Data Designer		Work
Technical lead		Work
Database developer		Work
Front-end developer		Work
Back-end developer		Work
QA lead		Work
QA tester		Work
Team members		Work
Wamp server		Work
pen		Material
paper		Material
Visual studio		Material
draw.io		Material
MS word		Material

8. SUPORTING PROCESS PLANS

8.1 RISK MANAGEMENT

Risk Identification:

- **Technical Risks**: Issues related to learning and implementing Java, integration problems, data security.
- **Operational Risks**: Delays in development due to the learning curve, resource availability, changes in project scope.

Risk Analysis:

- **Probability and Impact Assessment**: Each identified risk will be assessed for its likelihood and potential impact on the project.
- **Risk Prioritization**: Risks will be prioritized based on their severity and likelihood.

Risk Mitigation Strategies:

- **Technical Risks**: Regular code reviews, pair programming, testing, robust security measures.
- **Operational Risks**: Detailed project planning, regular progress reviews, additional training sessions for Java.

Risk Monitoring and control:

- **Regular Risk Reviews**: Will be conducted during Project Meetings to monitor identified risks and identify new risks.
- **Risk Register**: Will be maintained to track all risks, their status, and mitigation actions.