Namal University, Mianwali

Department of Computer Science

CSC-225 – Software Engineering

Complex Computing Problem – Milestone 1

Submission Deadline: 9th November, 2025 This milestone is worth **<3%>** of total marks

Instructions:

You have to work in group of 3, however marks will be assigned individually based on understanding.

Use of AI tools is allowed for understanding and ideas generation. Report must be written by yourselves and should not be generated by AI tool.

Objective

The goal of this milestone is to initiate your semester-long software engineering project by meeting a real-world user, identifying a real-world problem, forming a project team, and developing a **Project Proposal**.

Task Overview

In this milestone, you will:

1. Work in group which is already formed by the instructor and the list is provided.
2. Identify and meet a real requirement provider (RP), a person who will act as the *primary user* or *client* for your system.
3. Conduct your first meeting with the RP.
4. Select a real-world project idea that addresses a genuine problem or need.
5. Develop and submit a detailed project proposal
6. Create and maintain meeting minutes, documenting all discussions with your RP throughout the semester.

Requirement Provider (RP) Guidelines

Your RP can be:

- A faculty or staff member of Namal University,
- A Final Year Computer Science, or
- An industry professional (working in any organization).

You must meet your RP (Physically or Virtually) once every two weeks to discuss requirements, progress, and feedback. These meetings are essential after the successful submission of project proposal in every milestone.

Meeting Minutes and Video Recording

You will maintain meeting records using Google Sheet and upload a video recording of your first meeting.

1. **Google Sheet Meeting Minutes Template:**

The following columns should be included in your google sheet meeting minutes and every meeting record should be maintained.

- Meeting Date
- Attendees
- Agenda
- Key Discussion Points
- Decisions Made
- Action Items
- Next Meeting Date

The Google Sheet must be shared with all stake holders and the instructor.
Add the link to this document in your GitHub repository's README file.

2. **Video Recording:**

- A video showing your interaction with the RP about the introduction, project idea discussion, and user needs.
- Upload the video to your GitHub repository under a folder named Meeting_Videos/.
- Ensure clear audio and visibility of both student(s) and RP.

Project Proposal Document Structure

Your proposal must include the following sections:

1. **Title Page**
   - Project title
   - Subject title
   - Team members (names, roll numbers, and emails)
   - Submission date
   - Department and University Name

2. **Requirement Provider Agreement**

A formal agreement between the student team and their RP confirming collaboration. You are allowed to use any template for this agreement.

3. Table of Content

Provide a clear, automatically generated list of all the major sections and subsections in your proposal.

4. Introduction

Give an overview of your proposed project. Explain the background or context that led to identifying the problem, and describe the general area your system belongs to (for example, education, healthcare, or campus services). The introduction should help the reader understand what your project is about and why it matters.

5. Problem Statement

Describe the specific problem your project aims to solve. Focus on the pain point or inefficiency that currently exists and who is affected by it. The goal is to help the reader quickly grasp what issue your system will address and why solving it is important.

6. Project Objective:

List 3–5 specific goals that your project aims to achieve. Objectives act as a checklist for success when your project is complete. Each objective should be clear, measurable, and achievable. For example:

- To automate the process of student attendance using QR codes.
- To develop an easy-to-use platform for managing faculty schedules.

7. Stakeholder Identification:

Identify all the people or groups who will interact with or be affected by your system. This usually includes the end users, system administrators, managers, or clients. For each stakeholder, mention their role and how they are related to the system. This section shows that you understand who your system serves.

8. Software Development Methodology:

Specify which software development methodology you will use for your project. Briefly explain why this methodology suits your project considering factors like project size, requirement clarity, or user involvement. Assume a development schedule of one year. Then, provide a tentative schedule showing how your work will be divided according to the chosen methodology.

9. Tools and Technologies:

List the primary software tools, programming languages, frameworks, and platforms you intend to use for system development (e.g., MySQL, Figma, Python, etc.). If you are uncertain about the appropriate tools, conduct research to identify and select the most suitable technologies for your project's development requirements.

10. References (if any):

Include all sources referenced in your proposal, such as websites, research articles, or existing software systems. Follow the IEEE citation style for consistency and professionalism. Additionally, provide a list of any AI-generated prompts or queries you used during the preparation of this proposal.

Submission Guidelines:

- Create a GitHub repository named after your project; it should be maintained by the group lead, with all members contributing regularly.
- Upload all project materials to this repository in an organized manner.
- Create a folder named “Meeting Minutes” and upload all meeting records there.
- Create a folder named “Meeting Videos” and upload all recorded meetings there.
- Write the Project Proposal using LaTeX and upload both the .tex and .pdf files to the repository.
- Include a README file containing the project summary, group details, meeting minutes link, and video link.
- Submit the link to your GitHub repository on QOBE for evaluation.

Evaluation Rubric

Viva	**Excellent** **1.0**	**Good** **0.7-0.9**	**Satisfactory** **0.5-0.6**	**Unsatisfactory** **0.2-0.4**	**Poor** **0.0-0.1**
	Answered all questions correctly	Answered most questions correctly	Answered some questions correctly	Answered very few questions correctly	Answered no questions correctly

		Weight	**Excellent** **90% - 100%**	**Good** **70% - 89%**	**Satisfactory** **50% - 69%**	**Unsatisfactory** **20% - 49%**	**Poor** **< 20%**
Project Proposal (CLO 1) 35 marks	Introduction	3	Provides a strong foundation for understanding the problem.	Introduction is relevant and mostly clear with minor gaps in context or motivation.	Gives a basic overview but lacks depth or connection to the problem.	Introduction is vague or unfocused; little context provided.	Missing or irrelevant introduction.
	Problem Statement & Objectives	3	Clearly defined problem and well-articulated objectives.	Problem and objectives are mostly clear.	Problem and objectives are somewhat vague.	Unclear or poorly defined problem and objectives.	Missing or irrelevant problem statement and objectives.
	Requirement Provider (RP) Engagement	5	Real RP identified and documented; RP agreement signed; first meeting held and recorded.	RP identified with signed agreement; minor gaps in meeting evidence.	RP identified but agreement or video missing.	RP mentioned without proof of interaction.	No RP engagement shown.
	Meeting Minutes	4	Meeting minutes are complete, detailed, and	Meeting minutes provided but missing minor	Basic meeting records available but lack structure or depth.	Incomplete or unclear meeting minutes.	Meeting minutes missing.

			well-documented.	details or consistency.			
	Software Development Methodology	5	Methodology is well-chosen, justified, and supported with a detailed, logical plan.	Methodology appropriate with a reasonable timeline.	Methodology stated but lacks justification or scheduling.	Methodology unclear or unrealistic plan.	Missing methodology and plan.
	Tools & Technologies	3	Tools and technologies are well-researched, justified, and suitable for the system.	Tools listed appropriately with partial reasoning.	Basic tools mentioned but unclear suitability.	Tools poorly chosen or unjustified.	Missing or irrelevant tools.
	References & AI Usage	2	Properly cited references and AI tool usage documented.	Mostly proper citations with minor omissions.	Some references provided, but incomplete.	References and AI tool usage are missing or unclear.	No references or AI usage details provided.
	Use of LaTeX and Professional Formatting	5	Document demonstrates excellent LaTeX usage with proper formatting.	Mostly correct LaTeX usage with minor formatting or styling issues.	Adequate LaTeX structure but contains several formatting inconsistencies.	Limited use of LaTeX features; document lacks professional presentation.	Poorly formatted document or not written in LaTeX.
	GitHub Repository & Organization	5	Repository is well-structured with required folders and consistent commit history.	Repository complete with minor structural issues.	Repository created but lacks clear organization or some folders.	Incomplete or disorganized repository.	No valid repository submitted.

***All marks will be scaled based on Viva/ Understanding**