

ASSESSMENT FORM

Course: COMP6100001 – Software Engineering

Method of Assessment: Project

Semester/Academic Year : Even/2024-2025

Name of Lecturer :
Date :
Class :
Topic : **Final Project Presentation and Course Review**

Group Members :	1
	2
	3
	4
	5
	6
	7
	8

Student Outcomes:

SO 3 - Mampu berkomunikasi secara efektif dalam berbagai konteks profesional

Able to communicate effectively in a variety of professional contexts

SO 4 - Mampu mengenali tanggung jawab secara profesional dan membuat justifikasi yang terverifikasi dalam praktik komputasi berdasarkan prinsip hukum dan etika

Able to recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles

SO 5 - Mampu menjalankan peran secara efektif sebagai anggota atau pemimpin dalam tim dalam melakukan aktivitas yang sesuai dengan ilmu komputer

Able to function effectively as a member or leader of a team engaged in activities appropriate to computer science

Learning Objectives:

LObj 3.1 - Mampu mengidentifikasi informasi yang dibutuhkan dalam berbagai konteks profesional

Able to identify information required in a variety of professional contexts

LObj 4.2 - Mampu membuat penilaian yang tepat dalam praktik komputasi berdasarkan prinsip hukum dan etika

Able to make informed judgments in computing practice based on legal and ethical principles

LObj 5.1 - Mampu melakukan kerja tim yang efektif dalam praktik komputasi

Able to perform effective teamwork in computing practice

Learning Outcomes :

LO 1: Understand the fundamentals of software engineering and various software development process models by defining the key principles and the concepts

LO 2: Apply project management and software design concepts for the proposed potential business project idea by using requirement engineering and architectural models

LO 3: Analyze software quality, risks, and security to ensure software reliability and address security vulnerabilities in software engineering

LO 4: Utilize modern tools and methodologies to enhance software development process and project management efficiency

LO 5: Present and demonstrate software solutions by applying software engineering concepts to develop the proposed potential business project idea, while addressing ethical concerns and trends in software engineering

No	Related LO-LOBJ-SO	Assessment criteria	Weight	Excellent (85 - 100)	Good (75-84)	Average (65-74)	Poor (0 - 64)	Score	(Score x Weight)
1	LO1-LObj 3.1-SO3	Understanding Software Engineering Fundamentals :Ability to define software engineering principles, explain various software	20%	Clearly defines key concepts of software engineering, including process models (e.g., Waterfall, Agile, Iterative), provides detailed explanations, and gives relevant real-world examples.	Provides clear definitions of the key concepts and process models with examples, but lacks some depth.	Provides basic definitions of key concepts and process models with minimal examples.	Provides incomplete or incorrect definitions and fails to demonstrate an understanding of key concepts.		

No	Related LO- LOBJ-SO	Assessment criteria	Weight	Excellent (85 - 100)	Good (75-84)	Average (65-74)	Poor (0 - 64)	Score	(Score x Weight)
		development process models, and demonstrate an understanding of key concepts in software engineering.							
2	LO2-LObj 4.2-SO3	Project Management and Software Design: Ability to create software project planning using project management tools and develop system designs using architectural models	25%	Creates comprehensive project plans using tools like Gantt charts or Trello, clearly identifying tasks, timelines, and responsibilities; develops detailed system architecture models (e.g., UML diagrams) that align with the project's goals.	Creates a clear project plan covering most tasks and timelines; develops architectural models, though some details may be missing.	Produces a basic project plan with limited details; architectural models are present but lack clarity.	Fails to create an effective project plan; architectural models are incomplete or missing.		
3	LO3-LObj 4.2-SO4	Risk Analysis, Security, and Testing: Ability to conduct risk analysis, address security issues, and perform	20%	Accurately identifies risks, proposes appropriate mitigation strategies, identifies potential security vulnerabilities, and integrates comprehensive	Identifies most risks and security vulnerabilities with reasonable mitigation strategies; performs essential testing activities with detailed test cases and reports.	Identifies some risks and vulnerabilities but lacks detailed mitigation strategies; performs basic testing activities with limited documentation.	Fails to identify key risks or vulnerabilities; testing activities are minimal or absent, and no clear test documentation is provided.		

No	Related LO- LOBJ-SO	Assessment criteria	Weight	Excellent (85 - 100)	Good (75-84)	Average (65-74)	Poor (0 - 64)	Score	(Score x Weight)
		testing activities throughout the project.		testing activities (unit, integration, system, and acceptance tests) into the development process. Test reports are detailed, demonstrating effective use of testing tools and techniques.					
4	LO4-LObj 5.1-SO5	Version Control and Team Collaboration : Ability to use version control systems (VCS) effectively for team collaboration and code management.	20%	Proficiently uses VCS (e.g., Git) to track changes, manage branches, merge code, and handle conflicts effectively; ensures that all team members contribute consistently throughout the project.	Uses VCS for tracking changes and merging code with some team collaboration; handles basic conflicts, but some inefficiencies are observed.	Uses VCS minimally with limited tracking of changes; some team members contribute inconsistently.	Fails to use VCS effectively; lacks proper version control and collaboration practices, and few team members contribute consistently.		
5	LO5-LObj 5.1-SO5	Presentation and Demonstration of Software Solutions: Ability to present and demonstrate the completed software project,	15%	Delivers a clear, well-organized presentation and demonstration, effectively communicating the solution's features, benefits, and alignment with software engineering trends. Ethical	Provides a clear presentation covering most key aspects of the project, though some areas need more elaboration; discusses ethical concerns but lacks depth.	Presents the project with basic clarity, lacking in-depth coverage of features or trends; ethical concerns are mentioned but not explained.	Presents the project with basic clarity, lacking in-depth coverage of features or trends; ethical concerns are mentioned but not explained.		

No	Related LO- LOBJ-SO	Assessment criteria	Weight	Excellent (85 - 100)	Good (75-84)	Average (65-74)	Poor (0 - 64)	Score	(Score x Weight)
		addressing ethical concerns and trends in software engineering.		concerns are thoroughly discussed, with references to industry standards and practices.					
Total Score: $\sum(\text{Score} \times \text{Weight})$									

Remarks:

ASSESSMENT METHOD

Project

Instructions:

1. Students are divided into 3-5 people per group. Group should have already been determined at the latest Week 2.
2. Students are expected to be able to develop software following appropriate SDLC rules based on the creative business idea. Project themes can be varied from: Foods, Energy, Health, Transportation, Maritime, Social Humanities, Education, Art and Culture, Multidiscipline and Intersectoral, etc. Idea should be discussed by Week 3.
3. Students should choose one preferred model in SDLC and have to implement the methodology throughout the project.
4. Students should use any project management tools for tracking their progress.

5. Students should use any Version Control System platforms (Github, Gitlab, Bitbucket, or etc.) to collaboratively develop the project, ensuring all team members make contributions.
6. Perform comprehensive testing (unit, integration, system, and acceptance) during development, and document the test results.
7. Students should create the report in Project Portfolio form or Notion, as complete as it is (Project Introduction, SDLC model used, each phases of the models are explained, mention how tools that you use support each phases, the explanation of the requirements, design, development, testing, research, and conclusions) and the result.
8. Students have to develop the system or application in form of web, mobile, or any applications as long as they are not only prototype. At least one main feature can be used.
9. Students should present the project results in slides or videos that showcase all aspects of the project.

Project Outputs:

1. **Application:** Completed project link in Version Control System platform (e.g., GitHub).
2. **Project Report:** in Project Portfolio form or Notion,
3. **Presentation:** Key aspects of projects, such as the problem, solution, design, testing, and team contributions

Note for Lecturers:

1. Verify the feasibility and relevance of proposed project ideas in Week 3.
2. Track each group's progress during scheduled consultations in Week 7 and Week 11, focusing on project management and testing practices.
3. Assess project artifacts (code, report, presentation) at the end of the semester, with a particular focus on testing documentation, risk analysis, and effective use of VCS.