SLIIT UNI
THE KNOWLEDGE UNIVERSITY

SE2030 – Software Engineering

Department of Information Technology, Faculty of Computing

Year 2 semester 1 (2025)

Group Assignment

Group Assignment Specification (20% of Final Grade)

Assignment Title	Group Assignment
Learning outcomes covered	LO 2 - 4
Assignment Mode	Project
Maximum Marks	100 marks
Contribution to the Final Grade	20%
Week Published	3rd Week
Deadline for Submissions	3rd Week, 10th Week, 14th Week
Mode of Submission	Group Presentation and Report

Assignment Description

Overview

This assignment is designed to help you apply software engineering principles, including SDLC stages, Agile practices, UML modeling, and design patterns, in a practical team-based project. You will work in pre-assigned groups to design and develop a real-world system using Java. Over the course of the semester, your team will go through multiple development phases, create a functional prototype, apply Agile (Scrum), demonstrate your progress, and submit a final working system along with the required documentation.

- Groups: Already formed (6 members each)
- Platform: Java-based web application
- Assignment Weight: 20% of final module grade

Note: In this assignment, students will be assessed both individually and as part of a team across all evaluation components.

Deliverables & Timeline

Phase	Week	Deliverables	Marks
Proposal Evaluation	Week 3	Proposal Presentation & Report	20%
Progress Evaluation	Week 10	Progress Presentation	30%
Final Evaluation and Viva	Week 14	Final Presentation & Report	50%

Project Life Cycle and Deliverables

Phase 1: Project Proposal

- Week: 3
- Weight: 20% of the total project mark
- Components: Presentation and Proposal Report

Proposal Presentation Instructions

- Duration: 10 minutes
- Participation: All group members must actively participate. Each member must present at least their assigned major function.

Proposal Presentation Content

Your presentation should clearly and concisely cover the following:

Presentation Section	Details
1. Project Title and Group Members' Names	Mention the system name and list all group members.
2. Introduction	- Brief overview of the project - Objectives and purpose
3. System Overview Diagram	A visual showing the overall system flow
4. Functional Requirements	Key features and functionalities of the system
5. Non-Functional Requirements	Qualities like performance, usability, security, etc.
6. Major Stakeholders / Users	Who will use or benefit from the system
7. Six Major Functions (one per member)	- Identify the user of each function - Explain the main tasks involved in each function
8. Minor Functions	Supporting features such as login, logout, etc.
9. System Limitations / Constraints	Any known assumptions, technical boundaries, or restrictions
10. Project Timeline	Weekly plan from Week 3 to Week 14 (include key expected activities)
11. Conclusion	Final remarks on the importance, feasibility, and value of the proposed system

Proposal Report Instructions

- Submission Format: Submit the report as a soft copy (PDF format) through the designated platform.
- Length: Maximum 12–15 pages
- Submission: One report per group
- Purpose: This report is a formal document that supports your proposal presentation. It should clearly outline your system idea, user needs, and project plan.

Proposal Report Structure

Section (with Suggested Page Count)	Details
1. Cover Page (1 page)	Include: • Project Title • Group Members' Names and Student IDs • Module Name and Code (e.g., SE2030 – Software Engineering) • Submission Date
2. Introduction (1–2 pages)	Provide a clear overview of the project. Include: • What is the project about? • Objectives or goals • Target users or stakeholders • Scope and limitations
3. System Overview Diagram (0.5–1 page)	Add a high-level diagram showing how your system will work. It can be a simple block or flow diagram.
4. Functional Requirements (1 page)	Clearly list the main functions your system must support. (e.g., user registration, booking, notifications)
5. Non-Functional Requirements (0.5–1 page)	Define quality aspects such as performance, security, usability, or scalability.
6. Major Stakeholders / Users (0.5–1 page)	Identify user roles like admin, customer, staff, suppliers, etc.
7. Six Major Functions (2 pages)	Assign one function to each group member. For each, describe: • Who will use it • What it does • Expected outcome
8. Minor Functions (0.5 - 1 page)	Mention common supportive functions (e.g., login, password reset).

9. System Limitations / Constraints (0.5 – 1 page)	Mention known constraints such as platform limitations, expected device type, or assumptions.
10. Simple Timeline (0.5–1 page)	Create a basic timeline table from Week 3 to Week 14. For each key week, include what your group *plans to complete* during that time.
11. Conclusion (0.5–1 page)	Summarize your project idea and emphasize why it's worth implementing. Mention how your team plans to successfully complete it.

Phase 2: Progress Presentation

- Week: 10
- Weight: 30% of the total project mark (30 Marks)
- Type: Demo-based team presentation
- Duration: 15 minutes
- Participation: All group members must contribute and explain their progress, especially on their assigned major function.

Progress Presentation Content

Your team should demonstrate and explain the following:

Component	Details
1. Working Prototype	• User Interface (UI) • Partial Backend Functionality
2. Progress on Each Major Function	• Clearly show what each team member has completed
3. Agile Sprint Summaries	• Provide a summary of sprints completed so far • Mention key deliverables and activities per sprint
4. Updated Use Case Diagram	• Reflect any changes based on the progress
5. Class Diagram	• Show object-oriented structure and key relationships
6. Activity Diagrams	• One Activity Diagram for each major function
7. Ethical Considerations	• Highlight any ethical concerns and how they are addressed • Example: data privacy, user consent, accessibility

Phase 3 - Final Presentation

- Week: 14
- Duration: 15 minutes
- Type: Team-based presentation and demonstration
- Expectations:
    - o Demonstrate a fully working system
    - o Each team member should present part of the system, especially their assigned major function
    - o Explain how the system meets the defined requirements and objectives
    - o Highlight key features, technologies used, and any improvements made based on earlier feedback

Final Report Instructions

- File Type: PDF
- Length: Maximum 25 pages
- Submission Mode: Submit soft copy through the course platform

Final Report Structure

Section	Details
1. Cover Page *(1 page)*	Include the following details on the first page of your report: • Project Title • Group Members' Names and Student IDs • Module Name and Code (e.g., SE2030 – Software Engineering) • Submission Date
2. Introduction *(1-2 pages)*	Provide a clear overview of the project. Include the following: • **Project Overview**: Briefly describe what your project is about. • **Objectives**: State the main goals you aim to achieve through the system. • **Target Users or Stakeholders**: Identify who will use the system or who is affected by it. • **Scope and Limitations**: Clearly define what the system will include and what is out of scope.

3. Requirements *(2-3 pages)*	Present the gathered requirements for the system in a structured format. Include: • **Functional Requirements**: What the system must do (e.g., user login, search functionality). • **Non-functional Requirements**: Quality aspects such as performance, usability, and security. • **Constraints or Assumptions**: Any limitations or assumptions made (e.g., must run on Android, assumes users have internet access).
4. Design *(4 - 6 pages)*	Describe how the system is planned and structured before coding begins. Include: • **Use Case Diagram**: A visual representation of user interactions with the system. • **Database Design / ER Diagram**: Show how data is structured and related. • **UI Sketches or Description**: Provide screenshots, wireframes, or written descriptions of the user interface. • **System Architecture**: Outline the overall system structure (e.g., MVC, client-server, layered).
5. Implementation *(2 - 3 pages)*	Explain how the system was built. Include: • **Tools and Technologies Used**: Mention programming languages, IDEs, libraries, frameworks, etc. • **Key Features Developed**: List core features that were implemented. • **Screenshots of Core Functions**: Add images showing working parts of your system.
6. Project Management *(1–2 pages)*	Describe how the team organized the development process. Include: • **Agile Approach and Sprint Summary**: Show how sprints were planned and executed. • **Task Distribution Among Team Members**: Explain who did what. • **Development Phases or Timeline**: Provide a summary of key milestones and their timelines.

7. Conclusion & Future Work *(1- 2 pages)*	Summarize the project outcome and reflect on future improvements. Include: • **Summary of Achievements**: What was completed successfully. • **Challenges Faced**: Key difficulties or blockers during the project. • **Suggestions for Improvement or Extension**: What could be done better or added if more time/resources were available.
8. Individual Contribution, Teamwork & Lessons Learned *(1- 2 pages)*	In this section, each student should reflect on their **personal involvement** and **team experience** during the project. Please include the following: • **Your specific role** in the project: What tasks or responsibilities did you handle? • **Challenges you faced**: Describe any difficulties you encountered during the project. • **How you overcame those challenges**: Explain the steps you took to solve the problems. • **Key lessons learned**: What did you learn from working on this project? (about teamwork, tools, time management, etc.) • **Reflection on the project**: What aspects of the project went well? What could have been improved? **Note:** • Use a table where each row represents a group member's reflection. • Each student in the group must complete one row, explaining their own role and experience in the project.
9. References *(1- 2 pages)*	List all the external resources you referred to in your project. Include: • Books, websites, tools, APIs, and libraries. • Follow the **IEEE referencing style** strictly.
10. Appendix *(optional) – Max 2 pages*	Include any supplementary materials. You may attach: • GitHub or repository link to your source code • Additional diagrams • Meeting minutes, sprint logs, or planning documents

Marking Rubrics

Project Proposal Presentation and Report - Marking Rubric

Total: 100 Marks (Presentation: 70 — Report: 30)

Criteria (Total Marks)	Excellent	Good	Satisfactory	Needs Improvement	Poor
Problem Analysis (8 marks)	Clear, concise explanation of problem, purpose, goals, and background (6–8 marks)	Clear explanation with minor omissions (5–6 marks)	Covers basic context, lacks clarity in purpose (3–5 marks)	Lacks detail and clarity (2–3 marks)	Not understandable or irrelevant (0–2 marks)
System Overview Diagram (8 marks)	Well-structured, easy-to-understand diagram showing components and relationships (6–8 marks)	Mostly clear diagram with minor issues (5–6 marks)	Diagram exists but lacks clarity or completeness (3–5 marks)	Poorly presented diagram (2–3 marks)	No meaningful diagram (0–2 marks)
Functional Requirements (10 marks)	All major functions well defined, relevant, and clearly explained (8–10 marks)	Most functions defined with minor gaps (6–8 marks)	Basic functions mentioned but not clearly elaborated (4–6 marks)	Few unclear or irrelevant functions (2–4 marks)	Requirements missing or incorrect (0–2 marks)
Non-Functional Requirements (7 marks)	Quality attributes like usability, performance, and security clearly presented (6–7 marks)	Mostly relevant quality aspects included (4–6 marks)	Some quality attributes covered (3–4 marks)	Limited or unclear attributes (1–3 marks)	Missing or incorrect content (0–1 marks)

Stakeholder Identification (7 marks)	All relevant stakeholders identified, and roles explained (6–7 marks)	Most stakeholders identified (4–6 marks)	Basic stakeholder list with limited role explanation (3–4 marks)	Vague or incomplete stakeholder info (1–3 marks)	No stakeholder information (0–1 marks)
Assigned Functions (6 Major + Minor) (10 marks)	6 major functions mapped to users and tasks + minor functions explained (8–10 marks)	Functions assigned with minor gaps (6–8 marks)	Most functions assigned but roles not clearly defined (4–6 marks)	Few functions assigned, unclear ownership (2–4 marks)	No functional mapping (0–2 marks)
System Limitations / Constraints (7 marks)	Realistic, clearly explained limitations and constraints (6–7 marks)	Reasonable limitations included (4–6 marks)	Basic limitations mentioned but not discussed (3–4 marks)	Vague or incomplete constraints (1–3 marks)	No mention of limitations (0–1 marks)
Team Coordination & Delivery (7 marks)	Smooth transitions, equal participation, well-managed time (6–7 marks)	Minor issues in time or transitions (4–6 marks)	Unequal participation or unbalanced timing (3–4 marks)	Poor coordination, evident confusion (1–3 marks)	No coordination or disorganized (0–1 marks)
Communication Skills (6 marks)	Clear, confident speaking; professional language and effective visuals (5–6 marks)	Mostly clear with minor errors (4–5 marks)	Adequate presentation; some language/clarity issues (2–4 marks)	Unclear or unconfident delivery (1–2 marks)	Difficult to understand (0–1 marks)
Report					
Report Structure & Format (10 marks)	Well-structured, neat formatting, logical flow of content (8–10 marks)	Mostly well organized (6–8 marks)	Basic structure; minor formatting issues (4–6 marks)	Unclear structure or inconsistent format (2–4 marks)	Disorganized, hard to follow (0–2 marks)

Coverage of Requirements (10 marks)	Complete documentation of functional, non-functional, stakeholders, and limitations (8–10 marks)	Mostly complete with some minor omissions (6–8 marks)	Some components missing or unclear (4–6 marks)	Few components covered or poorly explained (2–4 marks)	Very limited or missing sections (0–2 marks)
Written Communication (10 marks)	Professional language, clear visuals, no grammar or spelling issues (8–10 marks)	Minor grammar or style issues (6–8 marks)	Moderate grammar or clarity issues (4–6 marks)	Hard to read, many errors (2–4 marks)	Poorly written or missing visuals (0–2 marks)

Progress Presentation - Marking Rubric

Total: 100 Marks

Criteria (Total Marks)	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor (1)
Working Prototype (UI + Backend) (25 marks)	Functional UI with partial backend demonstrating expected behavior (20–25 marks)	Most features shown; minor missing parts (15–20 marks)	Basic functionality shown; lacks polish (10–15 marks)	Prototype incomplete or unstable (5–10 marks)	No working prototype or does not meet minimum expectations (0–5 marks)
Agile Sprint Summaries (15 marks)	Clearly explains completed sprints, tasks assigned, and backlog tracking (12–15 marks)	Sprint summaries mostly complete with small gaps (9–12 marks)	Basic sprint updates with little detail (6–9 marks)	Vague sprint info or inconsistent tracking (3–6 marks)	No usable summary of sprints (0–3 marks)
Final Use Case Diagram (10 marks)	Accurate and updated diagram with all actors and functions (8–	Mostly correct diagram with minor issues (6–8	Some elements missing or misplaced (4–6 marks)	Confusing or inaccurate diagram (2–4 marks)	No diagram or irrelevant (0–2 marks)

	10 marks)	marks)			
Class Diagram (10 marks)	Well-structured showing relationships, methods, and attributes (8–10 marks)	Mostly complete; minor issues (6–8 marks)	Some basic structure present (4–6 marks)	Limited or unclear diagram (2–4 marks)	No diagram or mostly incorrect (0–2 marks)
Activity Diagrams (per major function) (10 marks)	Clear diagrams showing detailed logical flows (8–10 marks)	Mostly correct diagrams with small errors (6–8 marks)	Basic flow represented (4–6 marks)	Incorrect or incomplete flow (2–4 marks)	Missing or irrelevant diagrams (0–2 marks)
Ethical Considerations (10 marks)	Addresses issues like privacy, accessibility, and fairness (8–10 marks)	Covers most important ethical aspects (6–8 marks)	Basic mention of ethics (4–6 marks)	Unclear or vague treatment of ethics (2–4 marks)	No ethical concerns discussed (0–2 marks)
Team Communication & Demo Delivery (10 marks)	Smooth delivery with good transitions and confident explanations (8–10 marks)	Minor delivery issues, good team involvement (6–8 marks)	Acceptable coordination (4–6 marks)	Disorganized delivery (2–4 marks)	Uncoordinated or unclear demo (0–2 marks)
Effective Communication (10 marks)	Each member explains their system component clearly and effectively (8–10 marks)	Most members communicate well (6–8 marks)	Mixed levels of clarity (4–6 marks)	Many unclear or nervous presenters (2–4 marks)	Very weak individual communication (0–2 marks)

Phase 3 - Final Presentation and Report - Marking Rubric

Total: 100 Marks (Presentation: 80 — Report: 20)

Criteria (Total Marks)	Excellent (5)	Good (4)	Satisfactory (3)	Needs Improvement (2)	Poor (1)
System Functionality Demo (25 marks)	Fully working system with all required	Most features functional; minor bugs or incomplete	Basic functionality demonstrat	System partially working or unstable (	Non-functional or very limited demonstratio

	features demonstrated (21–25 marks)	features (17–20 marks)	ed with notable issues (13–16 marks)	6–12 marks)	n (0–5 marks)
Team Communication & Presentation Flow (10 marks)	Smooth transitions, shared ownership, clear and confident delivery (8–10 marks)	Minor delivery issues; good team coordination (6–8 marks)	Acceptable presentation; uneven participation (4–6 marks)	Unclear transitions; limited teamwork (2–4 marks)	Disorganized presentation with poor communication (0–2 marks)
UI/UX Quality (10 marks)	Interface is attractive, user-friendly, and consistent (8–10 marks)	Minor UI issues but overall user-friendly (6–8 marks)	Acceptable interface with several usability flaws (4–6 marks)	Unappealing or difficult to use interface (2–4 marks)	No attention to design or usability (0–2 marks)
Application of Design Patterns (10 marks)	Two well-suited design patterns are correctly and effectively implemented with clear justification. Patterns are well-integrated into the system design. (8–10 marks)	Two relevant patterns applied with minor issues in implementation or justification. One pattern may be slightly less appropriate. (6–8 marks)	At least one pattern is applied correctly. Second pattern may be missing or poorly integrated. Partial justification provided. (4–6 marks)	Attempted use of design patterns, but with incorrect application, poor relevance, or lack of justification. (2–4 marks)	No meaningful application of design patterns. Patterns missing or completely incorrect. (0–2 marks)
Team Work / Individual Contribution (15 marks)	Each member clearly presents and explains their component (12–15 marks)	Most members explain their parts well (9–12 marks)	Some members unclear or limited input (6–9 marks)	Uneven participation (3–6 marks)	Minimal or no contribution from several members (0–3 marks)
Ethical & Real-World Considerations (10 marks)	Ethical issues and real-world impact clearly addressed (8–10 marks)	Most key ethical aspects mentioned (6–8 marks)	Basic ethics covered but lacks depth (4–6 marks)	Few or unclear ethical points (2–4 marks)	No ethical considerations mentioned (0–2 marks)
Report – 20 Marks					
Problem Analysis	Comprehensive and in-	Good analysis with mostly	Basic analysis	Incomplete or shallow	Little to no evidence of

(10 marks)	depth analysis of the topic. Clearly categorized functional and non-functional requirements with justification. Limitations are well-identified. All key stakeholders are correctly recognized. Use case diagram is accurate, complete, and effectively represents the system's behavior.(8–10 marks)	clear requirements . Minor gaps in justification or stakeholder identification. Use case diagram is mostly correct with minor issues. (6–8 marks)	provided. Functional and non-functional requirements are present but with limited clarity or depth. Stakeholders partially identified. Use case diagram is present but lacks detail or has errors. (4–6 marks)	analysis. Requirements are vague or not properly categorized. Stakeholder analysis is minimal. Use case diagram is unclear or incorrect. (2–4 marks)	analysis. Requirements and stakeholders are missing or completely inaccurate. No valid use case diagram provided. (0–2 marks)
Written Communication (Language, Grammar & Referencing) (10 marks)	Well-written with correct grammar, style, and references (8–10 marks)	Minor grammar or citation errors (6–8 marks)	Some unclear sections; inconsistent referencing (4–6 marks)	Poor grammar or structure (2–4 marks)	Hard to read and poorly structured (0–2 marks)

Submission Format

- All reports must be in PDF format.
- All source code should be submitted via GitHub with access granted.
- One group member will submit the documents on behalf of the team.
- Deadlines will be published on Course Web.

Important Notes

- Late submissions will incur penalties.
- All students must contribute equally.
- Plagiarism will result in zero marks and disciplinary action.
- Ethical risks must be identified early (e.g., data privacy, AI misuse).
- You may apply your prior knowledge from the OOP module to develop the web-based application.

———————————- End of the Assignment Specification———————————-