Subject Description Form

Subject Code	COMP3211
Subject Title	Software Engineering
Credit Value	3
Level	3
Pre-requisite / Co-requisite / Exclusion	**Pre-requisite**: COMP2011/COMP2013
Objectives	The objectives of this subject are to: 1. familiarise students with the concepts, theories, and techniques concerning various aspects of software development; and 2. provide students with opportunities to apply the theories and techniques to the development of software systems.
Intended Learning Outcomes	Upon completion of the subject, students will be able to: *Professional/academic knowledge and skills* (a) appreciate the importance of software engineering; (b) become familiar with software engineering concepts, theories, and techniques; (c) apply software engineering theories and techniques at various stages of the software development lifecycle; *Attributes for all-roundedness* (d) work with others as a team to develop serious software systems; and (e) communicate both verbally and in writing about important aspects of software development.

Subject Synopsis/ Indicative Syllabus	**Topic** **1. Introduction to Software Engineering** Importance of software engineering, software engineering diversity, software engineering ethics. **2. Software Processes** Software process models, software process activities, coping with changes, process improvement. **3. Agile Software Development** Agile methods, agile development techniques, agile project management, scaling agile methods. **4. Requirements Engineering** functional and non-functional requirements, requirements engineering processes, requirements elicitation, specification, validation, and change. **5. System Modeling** Context models, interaction models, structural models, behavioural models, model-driven architecture. **6. Architectural Design** Architectural design decisions, architectural views, architectural patterns, application architectures. **7. Object-Oriented Methodology** Object-oriented analysis, object-oriented design, object-oriented programming. **8. Software Testing** Development testing, test-driven development, release testing, user testing. **9. Software Evolution** Evolution processes, legacy systems, software maintenance, software configuration management. **10. Advanced Topics in Software Engineering**
Teaching/ Learning Methodology	Lectures focus on introduction and explanation of key concepts and techniques. Tutorial/lab sessions provide students with opportunities to apply the theories and techniques in selected software engineering scenarios. Assignments, in-class exercises/quizzes, and the examination will be used to assess the students' understanding of the learned knowledge. The project requires the students to work in groups and apply the theories and techniques to solve problems in the development of serious software systems.

Assessment Methods in Alignment with Intended Learning Outcomes

Specific assessment methods/tasks	% weighting	Intended subject learning outcomes to be assessed				
		a	b	c	d	e
Continuous Assessment	**60%**					
1. Assignments		✓	✓			
2. In-Class Exercises/Quizzes		✓	✓	✓		
3. Project				✓	✓	✓
Examination	**40%**	✓	✓			
Total	100%					

Explanation of the appropriateness of the assessment methods in assessing the intended learning outcomes:

Assignments, in-class exercises/quizzes and the examination will be used to assess students' understanding of the concepts, theories, and techniques in software engineering.

The project will provide students with opportunities to apply the theories and techniques to the development of serious software systems. Students can develop their analytical and problem-solving skills and practise teamwork in the project. The students can also improve their presentation and communication skills through the project presentation.

Student Study Effort Expected

Class contact:	
▪ Lecture	39 Hrs.
Other student study effort:	
▪ Assignments, Project and Self-study	66 Hrs.
Total student study effort	105 Hrs.

Reading List and References

Textbook:

1. Sommerville, I., *Software Engineering*, 10th Edition, Pearson, 2015.

Reference Books:

1. Pressman, R., *Software Engineering: A Practitioner's Approach*, 8th Edition, McGraw-Hill, 2014.

2. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, Kelli A. Houston, *Object Oriented Analysis & Design with Applications*, 3rd Edition, Addison-Wesley, 2007.

	3. Jacobson, I., Booch, G. and Rumbaugh, J., *The Unified Software Development Process*, Addison-Wesley, 1999. 4. Pierre Bourque, Richard E. Fairley, *Guide to the Software Engineering Body of Knowledge*, IEEE Computer Society, 3rd Edition, 2014.