UNIVERSITI TEKNOLOGI MARA

CSC565: SOFTWARE ENGINEERING

Course Name (English)	SOFTWARE ENGINEERING `APPROVED`
Course Code	CSC565

MQF Credit	3

Course Description	The subject introduces the theories and practices of Software Engineering which includes requirement analysis, design, coding, testing, validation, verification and maintenance. Students will collaboratively define requirements, design and implement a mini project using CASE tools. Upon successful completion of the module, students will have demonstrated knowledge of the strengths and weaknesses of the software development lifecycle and the ability to design and develop software solutions for a given problem. Apply and reflect upon project management techniques used to solve a given problem.

Transferable Skills	Solution Provider

Teaching Methodologies	Lectures, Lab Work, Discussion

CLO	CLO1	Demonstrate an engineering practices in software development.
	CLO2	Comply appropriate process models and methodologies in software development.
	CLO3	Use appropriate software engineering techniques in a small-scale software project.
	CLO4	Formulate appropriate software development documents for requirement and design.

Pre-Requisite Courses	No course recommendations

Topics
1. Introduction to Software Engineering 1.1) What is Software? 1.2) Systems level considerations and its challenges 1.3) Software Crisis 1.4) The Cost of (software) quality
2. Software Processes 2.1) Introduction to Software Process 2.2) Generic Activities 2.3) Software Process Model (Waterfall, Interactive/Incremental-Spriral/RUP, Agile) 2.4) Programming in the large vs. Individual programming
3. Project Planning and Management 3.1) Team participation 3.2) Scheduling and Tracking 3.3) Project risks
4. Requirements Engineering 4.1) Introduction to Requirements Engineering 4.2) Requirements Specification 4.3) Requirements Validation 4.4) Requirements Modelling

5. Software Design and Construction
5.1) System design principles
5.2) Design Paradigm - OOAD
5.3) Relationships between requirements and designs
5.4) Software architecture concepts and standard architectures
5.5) Design Pattern
5.6) Design modeling
5.7) Coding practices
5.8) Coding standards
5.9) Development context: "green field" vs. existing code base

6. Software Verification and Validation
6.1) Verification and validation concepts
6.2) Inspections, reviews, audits
6.3) Testing fundamentals

7. Software Evolution
7.1) Software development in the context of large, pre-existing code bases
7.2) Software evolution
7.3) Characteristics of maintainable software

8. Tools and Environments
8.1) Requirements analysis and design modeling tools
8.2) Software configuration management and version control

Assessment Breakdown	%
Continuous Assessment	70.00%
Final Assessment	30.00%

Details of Continuous Assessment	Assessment Type	Assessment Description	% of Total Mark	CLO
	Group Project	n/a	10%	CLO1
	Group Project	N/A	10%	CLO2
	Group Project	N/A	30%	CLO4
	Test	n/a	10%	CLO3
	Test	n/a	10%	CLO3

Reading List	Recommended Text	• Ian Sommerville 2015, *Software Engineering*, 10 Ed., Pearson [ISBN: 978-013394303] • Seidl, M., Scholz, M., Huemer, C., Kappel, G. 2015, *UML @ Classroom An Introduction to Object-Oriented Modeling*, Springer [ISBN: 978-3319127]
	Reference Book Resources	• Mark Richards and Neal Ford 2020, *Fundamentals of Software Architecture: An Engineering Approach*, 1st Edition Ed., O'Reilly Media Inc Sebastopol, Canada [ISBN: 1492043451] • Suraiya Hussain 2020, *Software Engineering*, I.K International New Delhi, India [ISBN: B08CB3RNF8] • Titus Winters, Tom Manshreck, Hyrum Wright 2020, *Software Engineering at Google: Lessons Learned from Programming Over Time*, O'Reilly Media Inc Sebastopol, Canada [ISBN: 1492082791] • Eric J. Braude and Michael E. Bernstein 2016, *Software Engineering: Modern Approaches*, 2nd Edition Ed., Waveland Press Long Grove, Illinois [ISBN: 978-147863230]
Article/Paper List	This Course does not have any article/paper resources	
Other References	This Course does not have any other resources	