Course Name (English)	SOFTWARE ENGINEERING: THEORIES AND PRINCIPLES `APPROVED`
Course Code	CSC577

MQF Credit	3

Course Description	This course introduces the theories and practices of Software Engineering, which includes software processes, requirement analysis, design, programming practices, verification and validation and software evolution for a large system. It also constructs a solid foundation for understanding and application of principles, techniques, technologies and tools in the development of a good software system. To help students understand these concepts, students will work in a team which lead a project flow through the entire software lifecycle.

Transferable Skills	Solution Provider

Teaching Methodologies	Lectures, Lab Work, Tutorial, Discussion

CLO	CLO1 Develop application based on theories and principles in software engineering.
	CLO2 Formulate software development document for requirement and design.
	CLO3 Demonstrate social skills in software engineering project.
	CLO4 Adhere to appropriate managerial skill in software engineering project.

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 Management
3.1) Team participation
3.2) Scheduling and Tracking
3.3) Project risks

4. Tools and Environments
4.1) Requirements analysis and design modeling tools
4.2) Software configuration management and version control

5. Requirements Engineering
5.1) Introduction to Requirements Engineering
5.2) Requirements Specification
5.3) Requirements Validation
5.4) Requirements Modelling

6. Software Design
6.1) System design principles
6.2) Design Paradigm - OOAD
6.3) Relationships between requirements and designs
6.4) Software architecture concepts and standard architectures
6.5) Design Pattern
6.6) Design modeling

7. Software Construction 7.1) Coding practices 7.2) Coding standards 7.3) Development context: "green field" vs. existing code base
8. Software Verification and Validation 8.1) Verification and validation concepts 8.2) Inspections, reviews, audits 8.3) Testing fundamentals
9. Software Evolution 9.1) Software development in the context of large, pre-existing code bases 9.2) Software evolution 9.3) Characteristics of maintainable software

Assessment Breakdown	%
Continuous Assessment	70.00%
Final Assessment	30.00%

Details of Continuous Assessment	Assessment Type	Assessment Description	% of Total Mark	CLO
	Group Project	Review 1 - Specification	5%	CLO3
	Group Project	Review 2 - Design	5%	CLO3
	Group Project	Review 1 - Project Management	5%	CLO4
	Group Project	Review 2 - Project management	5%	CLO4
	Group Project	Final Presentation	10%	CLO4
	Group Project	SRS Documentation	10%	CLO2
	Group Project	SDD Documentation	10%	CLO2
	Test	Test 1	10%	CLO1
	Test	Test 2	10%	CLO1

Reading List	Reference Book Resources	• **Ian Sommerville 2016, *Software Engineering, Global Edition*, 10 Ed., Pearson [ISBN: 1292096136]** • **R.S. Pressman, B.R. Maxim 2015, *Software Engineering: A Practitioner's Approach*, 8 Ed., McGraw-Hill Education [ISBN: 978-0-0-7-802]** • **A. Dennis, B.H. Wixom, D. Tegarden 2016, *System Analysis and Design: An Object Oriented Approach with UML*, 5 Ed., Wiley [ISBN: 78-1-118-8046]** • **J. Dick, E. Hull, K. Jackson 2017, *Requirements Engineering*, 4 Ed., Springer [ISBN: 978-3-319-610]** • **Raul Sidnei Wazlawick 2016, *Object-Oriented Analysis and Design for Information Systems : Modeling with UML, OCL, and IFML*, Morgan Kaufmann [ISBN: 9780124186736]**
Article/Paper List	This Course does not have any article/paper resources	
Other References	This Course does not have any other resources	