University of Colombo School of Computing

Computer Science Degree Program

Course Detail Document

Course Outline									
Course Code :	**SCS1303**								
Course Name :	Introduction to Software Engineering								
(Bold and underline the appropriate)	**Year :**	**1**	2	3	4		**Semester :**	**1**	2
Number of Credits	2L + [1P]								
Core/Optional	Core								
Evaluation Criteria	**Assignments :**	40%*weight [if final exam mark >25-40, weight =01.-1.0, otherwise weight =0]							
	Final Exam :	60 %							
Requisites for following the Course	None								

Method of Delivery		**Per Week**	**Total**
	Lectures	2 Hours	30 Hours
	Tutorials/Case Studies	2 Hours in a week Not more than 30 Hours in the semester	
	Lab Work		
	Group Work		

Course Definition	
Description	This introductory course in Software Engineering aims to equip first-year students with fundamental skills to identify requirements, document system design, and maintain developed software systems. Presuming a basic understanding of computers and programming, this course delves into the systematic approach essential for managing the increasing demand for software projects. Software engineering has emerged to address the complexities and challenges of developing software across various domains. This course provides a comprehensive understanding of the software engineering process, concepts, and methodologies necessary for the systematic development and management of software projects.

Course Aim/Goal and the Intended Learning Outcomes	Course Aims: • To provide a broad understanding of the software engineering process, concepts, and the systematic development and management of software projects. • To explain the software engineering principles and techniques used in developing quality software products. • To apply software engineering principles and techniques appropriately to develop moderately complex software systems. Learning Outcomes: By the end of this course, students should be able to: • LO1: Explain software engineering principles and techniques in developing quality software products, including the software development lifecycle, software quality, and software testing. • LO2: Apply software engineering principles and techniques appropriately in developing a moderately complex software system. • LO3: Use software engineering tools and technologies to support the development process. • LO4: Apply ethical and professional principles to software engineering practice.
Assessment Plan	Online Quizzes, individual and group case studies for assignments given during the tutorial time periods.
References/Reading Materials	Sommerville Ian, Software Engineering, 10th Edition. Pearson. 2016 Software Engineering: A practitioner's approach by Roger S. Pressman, 9th edition, McGraw-Hill International edition, 2020.

Course Content
Topic 1: Software and Software Engineering 1.1. Professional software development 1.2. Software engineering ethics 1.3. Case studies **Duration**: **2 weeks**
Topic 2: Software Processes

2.1. Software process models 2.2. Process activities 2.3. Coping with change 2.4. Process improvement **Duration: 2 weeks**
Topic 3: Requirements Engineering 4.1. Functional and non-functional requirements 4.2. Requirements engineering processes 4.3. Requirements elicitation 4.4. Requirements specification 4.5. Requirements validation 4.6. Requirements change **Duration: 2 weeks**
Topic 4: Agile software development 3.1. Agile methods 3.2. Agile development techniques 3.3. Agile project management 3.4. Scaling agile methods **Duration: 1 weeks**
Topic 5: System Modeling 4.7. Context models 4.8. Interaction models 4.9. Structural models 4.10. Behavioral models 4.11. Model-driven architecture **Duration: 2 weeks**
Topic 6: Architectural Design 4.12. Architectural design decisions 4.13. Architectural views 4.14. Architectural patterns 4.15. Application architectures **Duration: 2 weeks**
Topic 7: Design and implementation 4.16. Object-oriented design using the UML 4.17. Design patterns 4.18. Implementation issues 4.19. Open-source development **Duration: 1 week**
Topic 8: Software Testing 4.20. Development testing 4.21. Test-driven development 4.22. Release testing 4.23. User testing **Duration: 2 weeks**
Topic 9: Software Evolution 4.24. Evolution processes 4.25. Legacy systems

4.26. Software maintenance **Duration: 1 week**

Course Administration Details			
Lecturer in-charge	Prof. K. P. Hewagamage		
Telephone	0112581245 Ext. 6503	**Email**	kph@ucsc.cmb.ac.lk