

3 rd Semester	MCA03001	Software Engineering	L-T-P 3-0-0	3 CREDITS
--------------------------	----------	----------------------	----------------	--------------

Course Objectives

1. To discuss the software engineering discipline, its evolution, impact and emergence of software engineering and explain the development and use of different software life cycle models for real-life industrial applications.
2. To discuss different aspects of software project management, risk management and configuration management and explain various requirement elicitation, analysis and specification techniques.
3. To discuss various software design methodologies, the impact of cohesion and coupling measures on the goodness of the software design.
4. To discuss the importance of practicing different coding standards, guidelines and different testing strategies along with software reliability metrics and software quality management techniques and standards.

Course Outcome

After reading this subject, students will be able to:

1. Choose a proper life cycle model for different real-life industrial projects, prepare the SRS document, design the software using function-oriented approach (DFDs) and object-oriented approach (UML diagrams), code it, and test the developed software using different software testing strategies.
2. Understand the concepts of computer aided software engineering (CASE) and use different CASE tools in the development, maintenance and reuse of software systems.

Detailed Syllabus

Module I:Software development life cycle and Project Management: Software development life cycle (SDLC) models such as Waterfall model, Iterative waterfall model, Prototyping model, Evolutionary model, Spiral model, V model, RAD, Agile models etc., software project management, project planning, metrics for project size estimation such as LOC and FP, project estimation, COCOMO, Halstead's software science, Staffing level estimation, project scheduling, staffing, Organization and team structure, risk management, configuration management. [11hours]

Module II:Requirements analysis and specification: Requirements gathering and analysis, software requirements specification, formal systems specification. [3 hours]

Module III:Software Design: Outcome of a design process, cohesion and coupling, layered arrangement of modules, approaches to software design, function-oriented software design: overview of SA/SD methodology, structured analysis, DFDs, Data Dictionary, structured design, detailed design, object-oriented software design: UML diagrams such as use case diagram, class diagram, object diagram, sequence diagram, communication diagram, state chart diagram, activity diagram, etc., unified process, OOD goodness criteria. [11 hours]

Module IV:Coding and Testing: Coding standards and guidelines, code review, software documentation, unit testing, black-box testing, white-box testing, debugging, integration testing, system testing, performance testing, regression testing. [8 hours]

Module 5:Software reliability and Quality management: Software reliability, Statistical testing, software quality, software quality management system,ISO 9000, SEI CMM, PSP, Six sigma, CASE Tools, Software maintenance, Software reuse. [7 hours]

Books

1. R. Mall, Fundamentals of Software Engineering, 5th Edition, PHI Learning, 2018.
2. R. S. Pressman, Software Engineering: A Practitioner's Approach, McGraw Hill Publications, 2015.
3. I. Sommerville, Software Engineering, Pearson Education, 2015.
4. K. K. Aggarwal and Yogesh Singh, "Software Engineering", New Age International Publishing, 2007.
5. Pankaj Jalote, "An Integrated Approach to Software Engineering", Narosa Publication, 2019.
6. A. Behferooz and F. J. Hudson, Software Engineering Fundamentals, Oxford University Press, 2014.
7. James Peter, W. Pedrycz, "Software Engineering: An Engineering Approach", John Wiley & Sons, 2000.