

NIRMA UNIVERSITY
SCHOOL OF TECHNOLOGY, INSTITUTE OF TECHNOLOGY
Course Syllabus
Master of Computer Application (2-Years Programme) Semester-II

L	T	P	C
3	0	2	4

Course Code	6CS152
Course Name	Software Engineering

Course Outcomes:

At the end of the course, students will be able to –

1. explain various phases of software development lifecycle
2. analyze and document the requirement specifications for a software project
3. develop the process model using standard tools and methodologies
4. implement a quality software project through effective team-building, planning, scheduling and risk assessment

Syllabus:

**Teaching
Hours: 45**

Unit I

Introduction: introduction to software engineering, defining software, changing nature of software, attributes of a good software, software product, software development life cycle, software processes, software engineering practices, software myths

5

Unit II

Software process models: generic process model (defining framework activity, identifying task set), process assessment & improvement, waterfall process model, incremental process model, spiral process model, prototyping software process model, evolutionary process model, component based process model, introduction to basic concepts of agile software development

6

Unit III

Project management concepts: management activities, project planning, project scheduling, risk analysis and management, reactive vs. Proactive risk strategies, software metrics and measurement, software risks, risk identification, risk projection, risk refinement risk mitigation, monitoring and management.

6

Unit IV

Software requirement engineering: software requirements, requirement engineering, extraction and specification, feasibility study, requirements modelling, object oriented analysis.

6

Unit V

Design concepts: object oriented design, architectural design, component level design, user interface design, distributed systems architecture, real time software design, user interface design, pattern based design

6



Coding: top-down and bottom-up, structured programming, information hiding, programming style, and internal documentation.

Unit VI

5

Software process and metrics: metrics in the process and project domains: process metrics, project metrics, software measurement metrics for software quality

Unit VII

7

Software testing: unit testing, integration testing, black box and white box testing, regression testing, performance testing, object-oriented testing.

Verification and validation of software: software inspections and audit, automated analysis, critical systems validation software quality assurance, quality standards, quality planning and control, various quality models.

Software configuration management: software configuration items- SCM repository, SCM process, version control, change management

Unit VIII

4

Trends in software engineering: software release management, overview of reverse engineering and re-engineering, design patterns, case tools, DevOps tools, service oriented software engineering, aspect oriented software engineering., formal methods.

Self-Study:

Aspect Oriented Software Development, Unified Process Model, Computer Aided Software Engineering & its Tools, Software Maintenance & Reengineering

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 8 experiments to be incorporated that will be considered for evaluation.

Suggested Readings[^]:

1. Ian Sommerville, Software Engineering, Addison – Wesley
2. Roger Pressman, Software Engineering A Practitioners Approach, McGraw Hill Publication
3. Rajib Mall, Fundamentals of Software Engineering, Prentice Hall of India
4. Ivar Jacobson, Object Oriented Software Engineering A use case Approach, Pearson

L=Lecture, T=Tutorial, P=Practical, C=Credit

[^]this is not an exhaustive list

