

## NIRMA UNIVERSITY

<b>Institute:</b>	Institute of International Study
<b>Name of Programme:</b>	Bachelor of Science (Computer Science and Engineering) [2+2 Dual Degree]
<b>Faculty</b>	Faculty of Technology & Engineering
<b>Course Code:</b>	2CS509
<b>Course Title:</b>	Principles of Software Development
<b>Course Type:</b>	Core
<b>Year of Introduction:</b>	2023-24

L	T	Practical Component				C
		LPW	PW	W	S	
3	-	2	-	-	-	4

### Course Learning Outcomes (CLO):

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

1. explain various phases of the software development lifecycle (BL2)
2. analyse and document the requirement specifications for a software project (BL4)
3. develop the process model using standard tools and methodologies (BL3)
4. build a quality software project through effective team-building, planning, scheduling and risk assessment (BL6)

### Syllabus:

**Total Teaching hours: 45**

Unit	Syllabus	Teaching hours
Unit-I	<b>Introduction:</b> 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	05
Unit-II	<b>Software Process Models:</b> Generic Process Model (Defining Framework Activity, Identifying Task Set), Process Assessment and 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	07
Unit-III	<b>Project Management Concepts:</b> Management activities, Project Planning, Project Scheduling, Risk analysis and Management, Reactive vs. Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement Risk Mitigation, Monitoring and Management.	06
Unit-IV	<b>Software requirement engineering:</b> Software Requirements, Requirement Engineering, Extraction and Specification, Feasibility Study, Requirements Modelling, Object Oriented Analysis.	07
Unit-V	<b>Design Concepts:</b> Object oriented design, Architectural Design, Component level Design, User Interface Design, Distributed Systems Architecture, Real Time Software Design, User Interface Design, Pattern Based Design <b>Coding:</b> Top-down and bottom-up, structured programming, information hiding, programming style, and internal documentation.	06
Unit-VI	<b>Software Process and Metrics:</b> Metrics in the Process and Project Domains: Process metrics, project metrics, Software Measurement Metrics for Software Quality	06



Unit-VII **Software Testing:** Unit testing, integration testing, black box and white box testing, regression testing, performance testing, object-oriented testing. 08

**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

Self-Study: The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents

- Suggested Readings/References:
1. Roger Pressman, Software Engineering a Practitioner's Approach, McGraw Hill Publication
  2. Ian Sommerville, Software Engineering, Addison – Wesley
  3. Ivar Jacobson, Object Oriented Software Engineering A use case Approach, Pearson
  4. Rajib Mall, Fundamentals of Software Engineering, Prentice Hall of India

Suggested List of Experiments:	S. No.	Title	Hours
	0	Identify and form a group of 4 students' students within the same batch. Selecting a Project Title and Definition on which they will apply SE processes and work in a group collaboratively for entire semester during the lab sessions.	02
	1	Identify Project scope, Objectives, Problem Statement formulation and requirement identification for project.	02
	2	Define functional & non-functional requirements for same. Prepare a SRS document for the project.	04
	3	Define modules of the project and design the project plan (Gantt Chart) for the same and identify deliverables with time line.	02
	4	Design Use Case Diagrams and Use Case Specifications for your system.	02
	5	Construct Activity Diagram for your system.	02
	6	Design Class Diagram & CRC index cards for your System.	02
	7	Construct Sequence Diagram and Collaboration Diagram for project.	02
	8	Construct State Diagram for you project.	02
	9	Implement formal specification using Z notation.	02
	10	Implement at-least four functional modules of your project. Design test cases for your project and perform testing. Prepare test strategy document	08