

**SREENIVASA INSTITUTE of TECHNOLOGY and MANAGEMENT STUDIES**

<b>II MCA - I Semester</b>	<b>L</b>	<b>P</b>	<b>C</b>
	<b>4</b>	<b>0</b>	<b>4</b>

**16MCA213 SOFTWARE ENGINEERING**

**Course Objectives:**

- To Understand, Analyze and Model User's Requirements
- To Select Appropriate Process Model Apply it to All Stages of Software Development Life Cycle (SDLC)
- To Select and Apply Appropriate Design Methodology
- To Decide the Feasibility of Using and Applying Agile Development Process
- Assure Software Quality, Select and Apply Appropriate Testing Strategies
- To Select and Apply Appropriate Metrics to Estimate Software Size, Effort, and Cost
- Project, Mitigate, Monitor, and Manage Risk

**Syllabus:**

**UNIT I : Introduction to Software Engineering and A Generic view of Process**

The Evolving role of Software- Changing nature of Software - Legacy Software- Software myths.

A layered technology- A Process Framework- CMMI- Process assessment - Personal and team Process Models.

**UNIT II : Process Models**

The waterfall model- Incremental process models- Evolutionary process models- Specialized Process Models- Agile process - Agile process Model: Extreme programming.

**UNIT III : Software Requirements and System Models**

Functional and non-functional requirements- User requirements- System requirements- Interface specification- The software requirements document-Feasibility studies- Requirements elicitation and analysis- Requirements validation- Requirements management.

Context Models- Behavioral models- Data models- structured methods.

#### **UNIT IV : Design Engineering& Architecture, Testing Strategies**

Design process and Design quality- Design concepts- the design model - Creating an architectural design: software architecture- Data design- Architectural styles and patterns- Architectural Design.

A strategic approach to software testing- Test strategies for conventional Software - Validation testing-System testing- The art of debugging.

#### **UNIT V : Testing Tactics, Software Measurement and Estimation**

Software testing fundamentals- White-Box testing- Basis path testing- Control structure Testing- Black box testing.

Size oriented metrics- Function oriented metrics- Metrics for software quality- Empirical Estimation Models: - Quality Management: Software quality assurance- Formal Technical Reviews.

#### **Course Outcome:**

- Gain an insight into the concepts of software engineering
- Be able to cultivate the art of building good software
- Would be better equipped in implementing software using test case designing and testing techniques.

#### **TEXT BOOKS:**

1. Software Engineering, A practitioner's Approach, 6/e ,2005, Roger S Pressman, Tata McGraw-Hill International Edition .
2. Software Engineering, 7/e ,2004,Ian Sommerville, Pearson Education, India.(Units:2,3)

#### **REFERENCE BOOKS:**

1. Fundamentals of Software Engineering , 2/e, 2005, Rajib Mall , Prentice Hall Inc, India.
2. Software Engineering: A Precise Approach , 1/e, 2010, Pankaj Jalote , Wiley, India.
3. Software Engineering: A Primer , 1/e,2008, Waman S Jawadekar , Tata McGraw Hill , India.
4. Software Engineering - Principles and Practices ,1/e, Deepak Jain , Oxford University Press.
5. Software Engineering – A Supporting Processes, 1/e, 2005, Richard H. Thayer and Merlin Dorfman, Wiley.