L	T	P	C
3	-	-	3

20MC109 SOFTWARE ENGINEERING

Course Description and Objectives:

This course focuses on the concepts of software life cycle, role of process models and methods to prepare software requirement specification document. In addition to that, it also imparts knowledge of design, development and testing of software. The objective of this course is to enable the student to develop efficient, cost effective, feasible software as per client requirements.

Course Outcomes:

The student will be able to:

- ➢ Prepare a Software Requirement Specification (SRS) document for any software project.
- ➢ Identify the importance of system analysis and design in solving complex problems.
- ➢ Distinguish between object-oriented approach and traditional approach in system analysis and design.
- ➢ Analyze various metrics to measure software product size and complexity.

Skills:

- Design a process for developing/completing different kinds of projects on time with expected quality.
- Analyze software requirements and find out various ways to gather them and specifying them.
- Analyze and model a software product.
- Design an effective, user-friendly interface for a given software product.
- Fix bugs in a software product through software testing.

Activities:

- Analyze the feasibility from customer requirements and design the process for developing the product.
- Draft the system and software requirements to prepare the SRS document.
- Prepare test cases for testing the software product to ensure that it is defect-free.
- Make a list of possible risks and prepare a mitigation plan.

Syllabus

UNIT – 1   **9 Hours**

SOFTWARE ENGINEERING FRAMEWORK: The Nature of Software, Software Engineering, The Software Process, Software Engineering Practice.

PROCESS MODELS: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models.

UNIT – 2   **9 Hours**

REQUIREMENTS MODELING: Requirements Engineering, Eliciting Requirements, Developing Use Cases, Building the requirements model, Data modeling concepts, Class-based modeling.

UNIT – 3   **9 Hours**

DESIGN CONCEPTS: Design process, Design concepts, Design model, Golden rules, User interface analysis and design, Interface analysis, Interface design steps, Design evaluation.

UNIT – 4   **9 Hours**

PRODUCT METRICS AND SOFTWARE TESTING: Metrics- A framework for product metrics, Metrics for the requirements model, Metrics for the design model, Metrics for source code, Metrics for testing, Metrics for maintenance; Testing- Software testing fundamentals, Internal and external views of testing, White-box testing, Basis path testing, Control sequence testing, Black-box testing.

UNIT – 5   **9 Hours**

RISK MANAGEMENT: Reactive vs. Proactive risk strategies, Software risks, Risk identification, Risk projection, Risk refinement, Risk Mitigation Monitoring and Management, The RMMM Plan, The SQA plan.

Text Book:

Roger S. Pressman, "Software Engineering A Practitioner's Approach", 7th Edition, Mc Graw Hill, 2012.

Reference Books:

1. Ian Sommerville, "Software Engineering", 7th Edition, Pearson Education India, 2004.
2. Pankaj Jalote, "Software Engineering A Precise Approach", 1st Edition, John Wiley and Sons, 2010.