MCPC1006 SOFTWARE ENGINEERING (3-0-0)

Objectives:

- To provide the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
- To provide an idea of using various process models in the software industry according to given circumstances.
- To gain the knowledge of how Analysis, Design, Implementation, Testing and Maintenance processes are conducted in a software project.

Course Outcomes(CO):

CO1: Students will be able to decompose the given project in various phases of a lifecycle.
CO2: Students will be able to choose appropriate process model depending on the user requirements.
CO3: Students will be able perform various life cycle activities like Analysis, Design, Implementation, Testing and Maintenance.
CO4: Students will be able to know various processes used in all the phases of the product.
CO5: Students can apply the knowledge, techniques and skills in the development of a software product.

Module-I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software. A Generic view of process: Software engineering- A layered technology, a process framework, Process patterns, process assessment, personal and team process models. Process models: The waterfall model, Incremental process models, Evolutionary process models, spiral, specialized process models, The Unified process.

Module-II

Requirement analysis: problems in information elicitation, methods of eliciting user requirements, functional and non-functional requirements, tools for requirement analysis, document flow charts, decision tables, data flow diagrams, data dictionaries, tools for analyzing real time systems, Use case diagrams, system sequence diagrams, CRC card, software requirement specification.

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management, software requirement specification.

Module-III

Software design: The design process, Function-oriented design, Data base oriented design, Object oriented design, Data base design. Coding: Code documentation, data declaration, statement construction, guidelines for input/output, efficiency with regard to code, memory and input/output.

Module-IV

Testing: Unit testing, black box and white box testing, test cases, integration testing, top-down and bottom-up testing, validation testing, alpha and beta testing, system testing. Maintenance: software reliability, availability, and maintainability, Reliability models. Risk management: software risks, Risk identification, Risk projection, Risk refinement, Quality Management: Quality concepts, Software quality assurance, Software reliability, The ISO 9000 quality standards.

Books

1. Software Engineering: A Practitioners Approach by Roger Pressman, 6th Edition, McGraw-Hill
2. Software Engineering by Ian Sommerville, Addison-Wesley
3. Fundamentals of Software Engineering by Rajiv Mall, PHI