JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY ANANTAPUR
(Established by Govt. of A.P., ACT No.30 of 2008)
ANANTHAPURAMU – 515 002 (A.P) INDIA

MASTER OF COMPUTER APPLICATIONS

Course Code	SOFTWARE ENGINEERING	L	T	P	C
21P00102		4	0	0	4
	Semester	I			

Course Objectives:

- To learn the basic concepts of software engineering and life cycle models
- To explore the issues in software requirements specification and enable to write SRSdocuments for software development problems
- To elucidate the basic concepts of software design and enable to carry out proceduraland objectoriented design of software development problems
- To understand the basic concepts of black box and white box software testing andenable to design test cases for unit, integration, and system testing
- To reveal the basic concepts in software project management

Course Outcomes (CO): Student will be able to

- Ability to apply software engineering principles and techniques.
- Ability to develop, maintain and evaluate large-scale software systems.
- To produce efficient, reliable, robust and cost-effective software solutions.
- Ability to work as an effective member or leader of software engineering teams.
- Ability to understand and meet ethical standards and legal responsibilities.

UNIT – I Lecture Hrs:

Basic concepts: abstraction versus decomposition, evolution of software engineering techniques, Software development life cycle (SDLC) models: Iterative waterfall model, Prototype model, Evolutionary model, Spiral model, RAD model, Agile models, software project management: project planning, project estimation, COCOMO, Halstead's Software Science, project scheduling, staffing, Organization and team structure, risk management, configuration management.

UNIT – II Lecture Hrs:

The nature of software, The Unique nature of Webapps, Software Myths, Requirements gathering and analysis, software requirements specification, Traceability, Characteristics of a Good SRS Document, IEEE 830 guidelines, representing complex requirements using decision tables and decision trees, overview of formal system development techniques. Axiomatic specification, algebraic specification.

UNIT - III Lecture Hrs:

Good Software Design, Cohesion and coupling, Control Hierarchy: Layering, Control Abstraction, Depth and width, Fan-out, Fan-in, Software design approaches, object oriented vs. function oriented design. Overview of SA/SD methodology, structured analysis, Data flow diagram, Extending DFD technique to real life systems, Basic Object oriented concepts, UML Diagrams, Structured design, Detailed design, Design review, Characteristics of a good user interface, User Guidance and Online Help, Mode-based Vs. Mode-less Interface, Types of user interfaces, Component-based GUI development, User interface design methodology: GUI design methodology.

UNIT – IV Lecture Hrs:

Coding standards and guidelines, code review, software documentation, Testing, Black Box Testing, White Box Testing, debugging, integration testing, Program Analysis Tools, system testing, performance testing, regression testing, Testing Object Oriented Programs.

UNIT – V

Software reliability, Statistical testing, Software quality and management, ISO 9000, SEI capability maturity model (CMM), Personal software process (PSP),Six sigma, Software quality metrics, CASE and its scope, CASE environment, CASE support in software life cycle, Characteristics of software maintenance, Software reverse engineering, Software maintenance processes model, Estimation maintenance cost. Basic issues in any reuse program, Reuse approach, Reuse at organization level.

Text Books:

1. RajibMall, "Fundamentals of Software Engineering", 5th Edition, PHI, 2018.
2.Pressman R, "Software Engineering- Practioner Approach", McGraw Hill.