



**R 20 Regulations**

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

**Computer Science & Engineering**

| Course Code  | Software Engineering   |  | L  | T | P            | C |
|--|--|--|----|---|--------------|---|
| 20A05403T  | (Common to CSE, IT, CSE( DS), CSE (IoT))                               |  | 3  | 0 | 0            | 3 |
| Pre-requisite  | Semester   |  | IV |   |              |   |
| Course Objectives:   |  |  |    |   |              |   |
| <ul style="list-style-type: none"><li>To learn the basic concepts of software engineering and life cycle models</li><li>To explore the issues in software requirements specification and enable to write SRS documents for software development problems</li><li>To elucidate the basic concepts of software design and enable to carry out procedural and object oriented design of software development problems</li><li>To understand the basic concepts of black box and white box software testing and enable to design test cases for unit, integration, and system testing</li><li>To reveal the basic concepts in software project management</li></ul>    |  |  |    |   |              |   |
| Course Outcomes (CO):  |  |  |    |   |              |   |
| After completion of the course, students will be able to <ul style="list-style-type: none"><li>Obtain basic software life cycle activity skills.</li><li>Design software requirements specifications for given problems.</li><li>Implement structure, object oriented analysis and design for given problems.</li><li>Design test cases for given problems.</li><li>Apply quality management concepts at the application level.</li></ul>  |  |  |    |   |              |   |
| UNIT - I   | Basic concepts in software engineering and software project management |  |    |   | Lecture 8Hrs |   |
| 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  | Requirements analysis and specification                                |  |    |   | Lecture 8Hrs |   |
| 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   | Software Design  |  |    |   | Lecture 9Hrs |   |
| 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  | Coding and Testing   |  |    |   | Lecture 9Hrs |   |
| 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 quality, reliability, and other issues                        |  |    |   | Lecture 9Hrs |   |
| 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.  |  |  |    |   |              |   |
| Textbooks:   |  |  |    |   |              |   |



*R 20 Regulations*

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

**Computer Science & Engineering**

- |   |
|---|
| <ol style="list-style-type: none"><li>1. Rajib Mall, “Fundamentals of Software Engineering”, 5th Edition, PHI, 2018.</li><li>2. Pressman R, “Software Engineering- Practioner Approach”, McGraw Hill.</li></ol> |
|---|

|                  |
|------------------|
| Reference Books: |
|------------------|

- |  |
|--|
| <ol style="list-style-type: none"><li>1. Somerville, “Software Engineering”, Pearson 2.</li><li>2. Richard Fairley, “Software Engineering Concepts”, Tata McGraw Hill.</li><li>3. JalotePankaj, “An integrated approach to Software Engineering”, Narosa</li></ol> |
|--|

|                            |
|----------------------------|
| Online Learning Resources: |
|----------------------------|

|  |
|--|
| <a href="https://nptel.ac.in/courses/106/105/106105182/">https://nptel.ac.in/courses/106/105/106105182/</a><br><a href="http://peterindia.net/SoftwareDevelopment.html">http://peterindia.net/SoftwareDevelopment.html</a> |
|--|