**SOFTWARE ENGINEERING**

**MICRO SYLLABUS**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | 23  CS3403 | 23  IT3403 | **Year** | II | **Semester** | II |
| **Course Category** | PC  Professional Core | | **Branch**  **Branch** | CSE/IT  **CSE,IT** | **Course Type** | Theory |
| **Credits** | 3 | | **L – T – P** | 3-0-0 | **Pre requisites** | - |
| **Continuous Evaluation** | 30 | | **Semester End Evaluation** | 70 | **Total Marks** | 100 |

|  |  |  |
| --- | --- | --- |
| **Course Outcomes** | | |
| Upon successful completion of the course, the student will be able to: | | |
| **CO1** | Understand the fundamentals of Software Engineering and various process models | L2 |
| **CO2** | Apply project management and requirement analysis techniques for the software Projects. | L3 |
| **CO3** | Use various design elements along with testing to prepare software system. | L3 |
| **CO4** | Use of CASE to improve Software development and Software maintenance. | L3 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (3: Substantial,2: Moderate,1: Slight)** | | | | | | | | | | | | | | |
|  | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| **CO1** | 3 |  |  |  |  |  |  |  | 1 |  |  |  |  |  |
| **CO2** |  | 2 |  |  |  |  |  |  |  |  | 1 |  | 1 |  |
| **CO3** |  | 2 | 1 |  | 1 |  |  |  |  |  |  |  |  |  |
| **CO4** |  |  |  |  | 1 |  |  |  | 1 |  | 1 |  |  |  |

|  |  |  |
| --- | --- | --- |
| **Syllabus** | | |
| **Unit No.** | **CONTENTS** | **Mapped**  **CO** |
| **I** | **Introduction:** Evolution—From an Art Form to an Engineering Discipline- Evolution Pattern for Engineering Disciplines, A Solution to the Software Crisis, Software Development Projects-Types of Software Development Projects, Exploratory Style of Software Development- Principles Deployed by Software Engineering, Emergence of Software Engineering-Early Computer Programming, High-level Language Programming, Control Flow-based Design, Data Structure-oriented Design, Data Flow-oriented Design, Object-oriented Design. Notable Changes in Software Development Practices.  **Software Life Cycle Models:** Few Basic concepts, Waterfall Model and its Extensions-Classical Waterfall Model, Iterative Waterfall Model, V-Model, Prototyping Model, Incremental Development Model, Evolutionary Model, Rapid Application Development (RAD)- Working of RAD, Applicability of RAD Model, Comparison of RAD with Other Models, Agile Development Models- Essential Idea behind Agile Models, Agile versus Other Models, Extreme Programming Model, Scrum Model, Lean Software development, Spiral Model- Phases of the Spiral Model. | **CO1** |
| **I**  **I** | **Software Project Management:** Software Project Management Complexities, Responsibilities of a Software Project Manager- Job Responsibilities for Managing Software Projects, Skills Necessary for Managing Software Projects, Project Planning-Sliding Window Planning, The SPMP Document of Project Planning, Metrics for Project Size Estimation.-Lines of Code (LOC), Function Point (FP) Metric, Project Estimation Techniques- Empirical Estimation Techniques, Heuristic Techniques, Analytical Estimation Techniques, Empirical Estimation techniques- Expert Judgement, Delphi cost estimation, COCOMO—A Heuristic Estimation Technique-Basic COCOMO Model, Risk Management- Risk Management Approaches, Risk Identification, Risk Assessment, Risk Mitigation.  **Requirements Analysis and Specification:** Requirements Gathering and Analysis- Requirements Gathering, Requirements Analysis, Software Requirements Specification (SRS)-Users of SRS Document, Characteristics of a Good SRS Document, Attributes of Bad SRS Documents, Important Categories of Customer Requirements-Functional Requirements, how to identify the functional requirements, how to document functional requirements, traceability, Organisation of the SRS Document. | **CO1,**  **CO2** |
| **III** | **Software Design:** Overview of the Design Process- Outcome of the Design Process, classification of Design Activities, How to Characterise a Good Software Design- Understandability of a Design: A Major Concern, Cohesion and Coupling- Classification of Cohesiveness, Classification of Coupling, Layered Arrangement of Module, Approaches to software design- Function oriented design, Object oriented design.  **Function-Oriented Software Design:** Overview of SA/SD Methodology, Structured Analysis- Data Flow Diagrams, Developing the DFD Model of a System- Context Diagram, Level 1 DFD, Structured Design- Transformation of a DFD Model into Structure Chart, Detailed Design, Design Review.  **User Interface Design:** Characteristics of a Good User Interface, Basic Concepts- User Guidance and On-line Help, Mode-based versus modeless interface, Graphical User Interface (GUI) versus Text-based User Interface, Types of User Interfaces-Command Language-based Interface, Menu-based Interface, Direct Manipulation Interfaces, Golden Rules. | **CO1,**  **CO3** |
| **IV** | **Coding and Testing:** Coding- Coding Standards and Guidelines, Code Review- Code Walkthrough, Code Inspection, Clean Room Technique, Software documentation- Internal Documentation , External Documentation, Testing- Basic Concepts and Terminologies, Testing Activities, Why Design Test Cases?, Testing in the Large versus Testing in the Small, Unit Testing, Black-box Testing- Equivalence Class Partitioning, Boundary Value Analysis, White-Box Testing- Basic Concepts, Statement Coverage, Branch Coverage, Condition Coverage, Condition and Decision Coverage, Multiple Condition Coverage ,Path Coverage, McCabe’s Cyclomatic Complexity Metric , Debugging- Debugging Approaches, Integration Testing, Phased versus Incremental Integration Testing, Testing Object-Oriented programs- What is a Suitable Unit for Testing Object-Oriented Programs?, System Testing- Smoke Testing, Performance Testing  **Software Reliability and Quality Management:** Software Reliability- Hardware versus Software Reliability, Reliability Metrics of Software Products, Statistical Testing- Steps in Statistical Testing, Software Quality - Software Quality Models, Software Quality Management System- Evolution of Quality Systems, Product Metrics versus Process Metrics, ISO 9000-What is ISO 9000 Certification? ISO 9000 for Software Industry, SEI Capability maturity model- Comparison between ISO 9000 certification and SEI/CMM. | **CO1,**  **CO3** |
| **V** | **Computer-Aided Software Engineering (Case):** Case and its Scope, Case Environment- Benefits of CASE, CASE Support in Software Life Cycle- Prototyping Support, Structured Analysis and Design, Code generation, test case generator, Other Characteristics of Case Tools- Hardware and Environmental Requirements, Documentation Support, Project Management, External Interface, Reverse Engineering Support, Data dictionary interface, Towards Second Generation CASE Tool, Architecture of a Case Environment.  **Software Maintenance:** Characteristics of Software Maintenance- Characteristics of Software Evolution, Special Problems Associated with Software Maintenance, Software Reverse Engineering, Software maintenance process models and Estimation of maintenance cost. | **CO1,**  **CO4** |
| **Learning Resources** | | |
| **Text Books** | | |
| 1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI. 2. Software Engineering A practitioner’s Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition. | | |
| **Reference Books** | | |
| 1. Software Engineering, Ian Sommerville, 10th Edition, Pearson. 2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press. | | |
| **E-Resources & other digital material** | | |
| 1. <https://nptel.ac.in/courses/106/105/106105182/>  2.<https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview>  3.<https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview> | | |

**Course Coordinators Module Coordinators**

1. Dr A Haritha
2. Dr K Kavitha Dr. K Koteswara Rao, Dr A Haritha
3. Dr K Koteswara Rao

4. Dr. A Ramana Lakshmi

5. Ms A Divya

**Prof. & Head, Dept. of IT Prof. & Head, Dept. of CSE**

**(Dr. B V.Subbarao).**  **(Dr. A. JayaLakshmi)**