|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SOFTWARE ENGINEERING AND PROJECT MANAGEMENT**  [As per Choice Based Credit System (CBCS) scheme]  **SEMESTER – V** | | | | | | |
| **Subject Code** | : |  |  | **Credits** | : | 03 |
| **Hours / Week** | : | 03 Hours |  | **Total Hours** | : | 39 Hours |
| **L–T–P–S** | : | 3–0–0–0 | | | | |
| **Course Learning Objectives:**  1. **Outline** software engineering principles, activities and different process models involved in building large software programs. Identify ethical and professional issues and explain why they are of concern to Software Engineers.  2. **Explain** the role of Agile development and Extreme Programming (XP).  3**.Devise** the process of requirement gathering, requirement classification, requirement specification and requirements validation.  4. **Recognize** the importance Project Management with its methodologies.  5. **Identify** software quality parameters and quantify software using measurements and metrics. | | | | | | |
| **Teaching-Learning Process (General Instructions)**  These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.   1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes. 2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying. 3. Show ***Video/animation*** films to explain functioning of various concepts. 4. Encourage ***Collaborative*** (Group Learning) Learning in the class. 5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class. 6. Adopt ***Problem Based Learning***, which fosters students’ Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it. 7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding. | | | | | | |
| **UNIT – I** | | | | | | **08 Hours** |
| **Introduction to Software Engineering**: The nature of Software, The unique nature of WebApps, Software Engineering, The software Process, The software Engineering practice, The software myths, How it all starts  **Textbook** **1:** **Chapter** **1:** **1.1** **to** **1.7**  **Process** **Models**: A generic process model, Process assessment and improvement, Prescriptive process models, Waterfall model, Incremental process models, Evolutionary process models, Concurrent models, Specialized process models.  **Textbook** **1:** **Chapter** **2:** **2.1** **to** **2.4** | | | | | | |
|  | | | | | | |
| **UNIT – II** | | | | | | **08 Hours** |
| **Agile** **Development**: What is Agility?, Agility and the cost of change. What is an agile Process?, Extreme Programming (XP), Other Agile Process Models, A tool set for Agile process  **Principles that guide practice**: Software Engineering Knowledge, Core principles, Principles that guide each framework activity  **Textbook** **1:** **Chapter** **3:** **3.1** **to** **3.6,** **Chapter** **4:** **4.1** **to** **4.3** | | | | | | |
|  | | | | | | |
| **UNIT – III** | | | | | | **08 Hours** |
| **Understanding** **Requirements**: Requirements Engineering, Establishing the ground work,  Eliciting Requirements, Developing use cases, Building the requirements model, Negotiating Requirements, Validating Requirements  **Requirements** **Modeling** **Scenarios,** **Information** **and** **Analysis** **classes:** Requirement Analysis, Scenario based modeling, UML models that supplement the Use Case, Data modeling Concepts class Based Modeling.  **Textbook** **1:** **Chapter** **5:** **5.1** **to** **5.7, Chapter** **6:** **6.1** **to** **6.5**  Case study: Software Requirement specification | | | | | | |
|  | | | | | | |
| **UNIT – IV** | | | | | | **08 Hours** |
| **Introduction** **to** **Project** **Management:**  Introduction, Project and Importance of Project Management, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, some ways of categorizing Software Projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, Management and Management Control  Case study- W5HHH Principles, CMMI Levels.  **Textbook** **2:** **Chapter** **1:** **1.1** **to** **1.14** | | | | | | |
|  | | | | | | |
| **UNIT – V** | | | | | | **07 Hours** |
| **Software** **Quality:**  Introduction, the place of software quality in project planning, Importance of software quality, Defining software quality, quality models, ISO 9126, product and process metrics, product versus process quality management, Quality Management systems, process capability models, techniques to enhance software quality, testing-types, Software reliability, quality plans.  Case study: Writing Test cases on real time projects.  **Textbook** **2:** **Chapter** **13:** **(13.1** **to** **13.11)** | | | | | | |
|  | | | | | | |

|  |  |  |
| --- | --- | --- |
| **Course Outcome** | **Description** | **Bloom’s Taxonomy Level** |
| At the end of the course the student will be able to: | | |
| 1 | **Understand** the activities involved in software engineering and analyze the role of various process models | L1 & L2 |
| 2 | **Explain** various software testing methods and to understand the importance of agile methodology and DevOps | L2 |
| 3 | **Describe** the basics of object-oriented concepts and build a suitable class model using modelling techniques | L2 |
| 4 | **Illustrate** the role of project planning and quality management in software development | L2 & L3 |
| 5 | **Interpret** the importance of activity planning and different planning models. | L2 & L3 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table: Mapping Levels of COs to POs / PSOs** | | | | | | | | | | | | | | |
| **COs** | **Program Outcomes (POs)** | | | | | | | | | | | | **PSOs** | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **1** | **2** |
| CO1 | 2 |  |  |  |  |  |  |  |  |  | 2 | 2 |  | 3 |
| CO2 | 2 |  |  |  |  |  |  |  |  |  | 2 | 2 |  | 3 |
| CO3 | 3 |  |  | 2 | 3 |  |  |  |  |  | 2 | 2 |  | 3 |
| CO4 | 3 | 2 |  | 3 | 3 |  |  |  |  |  | 3 | 3 |  | 3 |
| CO5 | 3 | 2 |  | 3 | 2 |  |  |  |  |  | 2 | 3 |  | 3 |

**3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)**

**TEXT BOOKS:**

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
2. Bob Hughes, Mike Cotterell, Rajib Mall: Software Project Management, 6th Edition, McGraw Hill Education, 2018.

**REFERENCE BOOKS:**

* + - 1. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India.
      2. Software Engineering, by Ian Sommerville 10th edition, Pearson global edition
      3. Software Engineering, an Engineering approach-JamesF. Peters, Witold Percy, John Wiley

**E-Resources:**

1. <https://onlinecourses.nptel.ac.in/noc20_cs68/preview>
2. [https://www.youtube.com/watch?v=WxkP5KR\_Emk&list=PLrjkTql3jnm9b5nr-](https://www.youtube.com/watch?v=WxkP5KR_Emk&list=PLrjkTql3jnm9b5nr-ggx7Pt1G4UAHeFlJ) [ggx7Pt1G4UAHeFlJ](https://www.youtube.com/watch?v=WxkP5KR_Emk&list=PLrjkTql3jnm9b5nr-ggx7Pt1G4UAHeFlJ)
3. <http://elearning.vtu.ac.in/econtent/CSE.php>
4. <http://elearning.vtu.ac.in/econtent/courses/video/CSE/15CS42.html>

<https://nptel.ac.in/courses/128/106/128106012/> (DevOps)

**Activity Based Learning (Suggested Activities in Class)**

1. Demonstration of solution to a problem using different process models like V model.
2. Real world problem solving and its cost estimation using COCOMO models.
3. Project management activity- Project scheduling using Tools for near real time projects.
4. Testing activity-Writing Test case & Test plan for the real time projects.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*