BCAC 0027: SOFTWARE ENGINEERING AND TESTING

Objective: The aim of the subject is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

Credits: 04 **L–T–P: 4–0–0**

Module No.	Content	Teaching Hours
I	**Introduction:** Introduction to Software Engineering, Software characteristics, Software Crisis, Software Engineering Process. **Software Development Life Cycle (SDLC) Models:** Waterfall, Incremental, Iterative Enhancement, Prototype, RAD and Spiral Models. **Software Requirements Engineering:** Types of Requirements, Requirement Elicitation Techniques Like Interviews, FAST & QFD, Use case Approach, Requirements Analysis Using DFD, Data Dictionaries & ER Diagrams, Requirements Documentation, and SRS. **Software Project Planning:** Size Estimation like Lines of Code & Function Count, Cost. **Estimation Models:** COCOMO (Basic, Intermediate) **Software Design:** Cohesion & Coupling, Classification of Cohesion & Coupling, Function Oriented Design, Object Oriented Design, Structure chart. **Coding:** Characteristics of Coding and Coding style.	26
II	**Software Metrics:** Software Measurements, Token Count, Halstead Software, Measures. **Software Reliability & Quality:** Introduction of Mc Call's & Boehm's Quality Model, Capability Maturity Models **Software Reliability Models**: Basic Execution Time Model. **Software Testing:** **Testing Fundamentals:** Test Case Design, Black Box Testing Strategies, White Box Testing, Unit Testing, Integration Testing, System Testing. **Introduction to Automation Testing and Testing Tools:** Automated Testing Process, Framework for Automation Testing, Introduction to Automation Testing Tool. **Software Maintenance:** Maintenance Process **Maintenance models**: Belady and Lehman Model, Boehm Model Regression Testing, Software Configuration Management; Implementation, Introduction to Reengineering and Reverse Engineering. **Software Risk Management:** Risk Identification and Risk Analysis	26

Text Book:

- P Jalote, (2006), "*Integrated Approach to Software Engineering"*, Narosa Book Distributors Pvt. Ltd, New Delhi, 3rdedition.

Reference Books:

- K. K. Aggarwal & Yogesh Singh, (2008), *"Software Engineering"*, New Age International, 3rdedition.
- R. S. Pressman, (2001), *"Software Engineering – A Practitioner's Approach"*, McGraw Hill Int., 5thedition.
- Stephen R. Schach, (1996), *"Classical & Object Oriented Software Engineering"*, IRWIN.
- James Peter, W. Pedrycz, *"Software Engineering: An Engineering Approach"*, John Wiley & Sons.
- I. Sommerville, (2004), *"Software Engineering"*, Addison Wesley, New Delhi, 7thedition.

Focus: This course focuses on Employability under CO2, CO3, CO4, CO5 and CO7.

Outcome: After the completion of the course, the student will be able to:
CO1: Understand the basic concepts of software engineering.
CO2: Apply software processes to solve real world problems.
CO3: Estimate the cost, effort and schedule of software using COCOMO Model.
CO4: Analyze the software design techniques (structure chart, SDM, sequence diagram).
CO5: Develop the test cases to validate the software.
CO6: Understand the basic models of software Quality and maintenance.
CO7: Automate the software testing using Selenium and TestNG.

Mapping of Course Outcomes (Cos) with Program Outcomes (Pos) and Program Specific Outcomes (PSOs):

Cos	Pos/PSOs
CO1	PO1,PO7/PSO1
CO2	PO2,PO3/PSO4
CO3	PO2,PO11/PSO3
CO4	PO3,PO10/PSO4
CO5	PO3,PO7/PSO1
CO6	PO5,PO12/PSO2
CO7	PO3,PO10/PSO4