Software Testing and Quality Metrics

Course code: 18IS7DCSTQ **Credits: 04**
L: P: T: S: 4: 0: 0: 0 **CIE Marks: 50**
Exam Hours: 03 **SEE Marks: 100**
Total Hours: 50

Course Objectives:

1. To provide foundations in the area of Software Testing Methodologies.
2. To analyze and apply the different testing methods for various types of software applications.
3. To design qualitative test cases within project constraints.
4. To analyze and apply the basic types of testing.
5. To bring awareness about software quality metrics and its significance

Course Outcomes: At the end of the course, student will be able to:

CO1	Bring an awareness to generate test cases based on the given specifications.
CO2	Compare different testing techniques.
CO3	Design test cases with techniques such as Equivalence Class, Boundary Value Analysis etc.
CO4	Understand the appropriate technique for the design of control flow graph.
CO5	Understand and design test cases for Integration Testing and OO Testing.
CO6	Comprehend the significance of quality and metrics in software projects.

Mapping of Course outcomes to Program outcomes:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	-	-	-	-	-	-	-	-	-	-	3
CO2	3	-	-	-	-	-	-	-	-		-	3
CO3	3	2	2	-	-	-	-	-	-	-	-	3
CO4	3	2	2	-	-	-	-	-	-	-	-	2
CO5	3	2	2	-	-	-	-	-	-	-	-	1
CO6	3	2	3	-	-	-	-	-	-	-	-	3

Module	Course Content	Hours	CO's
1	**Introduction to software testing:** Basics of Software Testing: Basic definitions, Software Quality, Requirements, Behavior and Correctness, Correctness versus Reliability, Testing and Debugging, Test cases, Insights from a Venn diagram, Identifying test cases, Test-generation Strategies, Test Life Cycle, Levels of testing, Testing and Verification, Static Testing. **Problem Statements:** Generalized pseudo code, the triangle problem, the NextDate function, the commission problem, the SATM (Simple Automatic Teller Machine) problem, the currency converter, Saturn wind shield wiper	**10**	**CO1**
2	**Functional Testing:** Boundary value analysis, Robustness testing, Worst-case testing, Robust Worst testing for triangle problem, Nextdate problem and commission problem, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, NextDate function and commission problem, Guidelines and observations.	**10**	**CO2, CO3**
3	**Structural Testing:** Overview, Statement testing, Branch testing, Condition testing. **Path testing:** DD paths, Test coverage metrics, Basis path testing, guidelines and observations. **Data–Flow testing:** Definition-Use testing, Slice based testing. Guidelines and observations, Cyclomatic complexity and Examples.	**10**	**CO2, CO4**
4	**Model Based Testing:** FSM & EFSM **Integration Testing:** Introduction, Top Down, Bottom Up and Sandwich approaches. Introduction to Neighborhood, Pairwise and Combinatorial Testing, Object Oriented Testing, System, Acceptance and Regression Testing, MM Path Testing and Test adequacy	**10**	**CO2, CO5**
5	Views on quality, Cost of quality, Quality models, Ishikawa's Seven Basic Tools, Product Quality Metrics, The Defect Density Metric, Lines of Code, Function Points, Example: Function Point, Halstead's Software Science.	**10**	**CO5, CO6**

SELF STUDY COMPONENT:

1. Discrete Mathematics for Testers: Set Theory, Functions, Relations, Propositional Logic, Graph Theory for Testers

TEXT BOOKS:

1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2008.
2. Stephen H. Kan, "Metrics and Models in Software Quality Engineering, 2nd Edition, Pearson, 2003
3. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009.
4. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.

REFERENCE BOOKS:

1. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2008.
3. Pressman, R.S., Software Engineering: A Practitioner & Approach, 7th (Alternate) Edition, McGraw Hill International Edition, 2010
4. Kshirasagar Naik and Priyadarshi Tripathy (Eds); Software Testing and Quality Assurance: Theory and Practice ", John Wiley, 2008
5. Gordon G Schulmeyer, "Handbook of Software Quality Assurance", Third Edition, Artech House Publishers 2007.