PESIT Bangalore South Campus

10CS842-SOFTWARE TESTING

FACULTY: Mrs. Bidisha Goswami & Ms.Sudeepa

Total Hours Specified: 52

Objective: This course aims to introduce about fundamental of software testing. The syllabus discusses about different approaches of software techniques. Students have solve a few case study to understand what type of software testing should be addressed under a given situation.

Chapter Title/ Reference Literature	Topic to be covered	% of Portions covered	
		Reference Chapter	Cumulative
UNIT 1 **6 Hours** **A Perspective on Testing, Examples**	Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudocode, The triangle problem, The NextDate function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper	11.5%	11.5%
UNIT 2 **7 Hours** **Boundary Value Testing, Equivalence Class Testing, Decision Table Based Testing:**	Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, 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 the commission problem, Guidelines and observations.	13.5%	25%
UNIT 3 **7 Hours** **Path Testing, Data Flow Testing:**	DD paths, Test coverage metrics, Basis path testing, guidelines and observations. Definition-Use testing, Slice-based testing, Guidelines and observations.	13.5%	38.5%
UNIT 4 **6 Hours** **Levels of Testing, Integration Testing**	Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing. A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.	11.5%	50%

UNIT 5 **7 Hours** **System Testing, Interaction Testing:**	Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example. Context of 118 interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing,.	13.5%	63.5%
UNIT 6 **7 Hours** **Process Framework:**	Validation and verification, Degrees of freedom, Varieties of software. Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback. The quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis, Testing, Improving the process, Organizational factors.	13.5%	77%
UNIT 7 **6 Hours** **Fault-Based Testing, Test Execution:**	Overview, Assumptions in faultbased testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. Test Execution: Overview, from test case specifications to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay.	11.5%	88.5%
UNIT 8 **6 Hours** **Planning and Monitoring the Process, Documenting Analysis and Test:**	Quality and process, Test and analysis strategies and plans, Risk planning, Monitoring the process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.	11.5%	100%

Literature:

Book Type	**Titlle & Author**	**Publication Information** **Edition**
Text Book	1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, Auerbach Publications, 2008. (Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 1314, 15) 2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009.	3rd

	(Listed topics only from Chapters 2, 3, 4, 16, 17, 20, 24)	
Reference Book	1. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008. 2. Srinivasan Desikan, Gopalaswamy Ramesh: Software Testing Principles and Practices, 2nd Edition, Pearson Education, 2007. 3. Brian Marrick: The Craft of Software Testing, Pearson Education, 1995.	2nd

Question Bank

- What is Error, Fault, Failure.
- What are the quality attribute of a software.
- Explain CFG.
- Explain different type of classifier.
- What is Path testing? Discuss the DD path for triangle program and write a table for the types of DD paths with a graph.
- Explain Mc Cabe's basis path method with an example
- Explain Slice Based Testing

- With a Neat diagram explain validation and verification activities against actual user requirements.

- Explain the following principles:-
- i)Sensitivity
- ii)Redundancy
- iii)Visibility
- iv)Restriction
- v)Partition

- Enlist the Dependability properties of a software product and explain the relationship among the Dependability Properties.

- With reference to test execution, explain the concept of scaffolding and test oracles

- Explain Clean Room Process model in detail.

- Write short notes on:- i)Quality goals
    - ii)Test and Analysis Strategies and plan
- How can we Monitor a Process and Improve a Process within a project or across multiple projects?

- What do you mean by Risk Management? Explain the types and the control strategies.

- What is Integration Testing? Explain the strategies used in Integration testing.

- With neat diagram, explain the traditional view of testing levels of waterfall life cycle and
- Rapid prototyping life cycle.