Gayatri Vidya Parishad
College of Engineering for Women
Approved by AICTE, New Delhi and Affiliated to JNTU, Kakinada, Andhra Pradesh, India.

LECTURE SCHEDULE

Branch & Section: B. Tech (IT), III Year- II Semester **Regulation: R13**
Subject: Software Testing (RT32121) **Academic year: 2017-18**
Faculty Name: Kuppili N Satya Chitra

COURSE OBJECTIVES:

At the end of the course, the students will be able to:

1. To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
2. To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
3. To learn how to planning a test project, design test cases and data, conduct testing operations, manage software problems and defects, generate a testing report.
4. To expose the advanced software testing topics, such as object-oriented software testing methods, and component-based software testing issues, challenges, and solutions.
5. To gain software testing experience by applying software testing knowledge and methods to practice-oriented software testing projects.
6. To understand software test automation problems and solutions.
7. To learn how to write software testing documents, and communicate with engineers in various forms.
8. To gain the techniques and skills on how to use modern software testing tools to support software testing projects.

COURSE OUTCOMES:

After completing the course, the student must demonstrate the knowledge and ability to:

1. Have an ability to apply software testing knowledge and engineering methods.
2. Have an ability to design and conduct a software test process for a software testing project.
3. Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.
4. Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.
5. Have an ability to use various communication methods and skills to communicate with their teammates to conduct their practice-oriented software testing projects.
6. Have basic understanding and knowledge of contemporary issues in software testing, such as component based software testing problems
7. Have an ability to use software testing methods and modern software testing tools for their testing projects.

Unit No.	Topic No.	Name of the Concept	No. of Classes Required
Unit – 1 :			
Unit - 1	1	Introduction	1
	2	Evolution, Myths and facts, goals, psychology	1
	3	Definition, model for testing	1
	4	Effective vs. Exhaustive, terminology	1
	5	STLC, relations between SDLC and STLC	2
	6	ST Methodology**,** tutorial/discussion	1
Total number of hours			7
Unit – 2 :			
Unit – 2	1	Verification and validation activities	1
	2	Verification, ver. Of requirements, levels of designs	1
	3	Verifying the code, validation	1
	4	Black box testing techniques, Boundary value analysis	1
	5	Equivalence class testing, state table based testing	1
	6	Decision table based testing, cause-effect graphs, error guessing	1
	7	Tutorial/slip test	1
Total number of hours			7
Unit – 3 :			
Unit – 3	1	White box testing, transparency, need	1
	2	Logic coverage criteria, basis path testing, examples	1
	3	Statement, branch, decision coverage, examples	2
	4	Decision, condition, decision/condition	1
	5	Cyclomatic complexity, example	1
	6	BP testing, DFG, CFG, V(G), examples	2
	7	Graph Matrix, control/connection matrix	1
	8	k-path set, d,u,k,c,p, dft, example of terminology usage on a C program	1
	9	Du, dk, anomalies, Static DFT, examples, disadvantages	1
Total number of hours			11

Unit – 4 :			
Unit – 4	1	Validation, definition, types, one-to-one	1
	2	Unit testing, drivers and stubs	1
	3	Examples	1
	4	Integration testing, definition, examples	1
	5	Incremental, non-incremental	1
	6	Top-down, bottom-up, examples	1
	7	Call graph, adjacency matrix	1
	8	Examples	1
	9	Pair-wise integration, neighbourhood integration	1
	10	Path based integration, examples	1
	11	MM path graph, examples	1
	12	Functional testing, types	2
	13	System Testing, elaboration	1
	14	Recovery testing, examples	1
	15	Security Testing, performance	1
	16	Acceptance testing, regression testing	1
	17	Definitions, types, methods	1
	18	Prioritization	1
	19	Tutorial	1
		Total number of hours	20

Unit – 5 :			
Unit – 5	1	Test Suite	1
	2	Prioritization types, methods and examples	1
	3	Slice & dice, examples	1
	4	Risk management	1
	5	Requirements	1
	6	Software quality metrics	1
	7	Product metrics	1
	8	Process metrics, models	1
	9	SQA Models, ISO-9126	1
	10	CMM, SQTM, Six Sigma	3
	11	Debugging Processes, techniques	1
	12	Test Management Tools – JIRA	1
		Total number of hours	14

Unit – 6 :			
Unit – 6	1	Testing tools, goals	1
	2	Motivation, Testing tools types	1
	3	Working of the tools	1
	4	WinRunner, LoadRunner	1
	5	Object oriented testing, Modelling, UML	1
	6	OO testing features, types, methods, Web App Testing	1
	7	Mobile App Testing, Tutorial	1
	8	Final Discussion	1
		Total number of hours	8

OVERALL NUMBER OF CLASSES REQUIRED: 67

TEXTBOOKS:

1. **Software Testing – Principles and Practices, Naresh Chouhan, Oxford.**
2. **Foundations of Software Testing, Aditya P Mathur, 2nd Edition, Pearson.**
3. **Software Testing, Yogesh Singh, Cambridge.**

References:

1. **Software Testing techniques, Boris Beizer, 2nd edition, International Thompson Computer Press.**
2. **Software Testing, Principles, Techniques and Tools, MG. Limaye, Tata McGraw-Hill.**
3. **Effective Methods for Software Testing, William E. Perry, 3rd edition, John Wiley.**

SIGNATURE OF FACULTY