GOPALAN COLLEGE OF ENGINEERING AND MANAGEMENT

Department of Computer Science and Engineering

Academic Year: **2016-17** Semester**: EVEN**

COURSE PLAN

Semester: **VIII** Subject Code& Name: **10CS842 &System Testing**

Name of Subject Teacher: **APARNA N**

Name of Subject Expert (Reviewer): **N S SARADHA DEVI**

For the Period: From:
Details of Book to be referred:

Text Books	T1. Paul C Morgan: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publication,2008. T2. Mauro Pezze, Michal Young: Software Testing and Analysis- Process, Principles and Techniques, Wiley India, 2009.
Reference Books	R1. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008. R2. Srinivasan Desikan, Gopalaswamy Ramesh: Software Testing Principles and Practices,2nd Edition, Pearson Education, 2007. R3. Brian Marrick: The Craft of Software Testing, Pearson Education, 1995.

Lecture NO	Topic Planned	Practical Applications & Brief objectives	Book referred with Pg No.	Planned Date	Executed Date	Deviation Reasons thereof	How Made Good / Reciprocate arrangement	Remarks by HOD
1.	**Introduction to the subject**	Introduction to all the units in brief.		13-02-17				
2.	**UNIT-1-A Perspective on Testing Examples:** Basic definitions, Test cases	**Objective:** Understand the basics of testing, test cases and examples.	T1: 3-5	14-02-17				
3.	Insights from a Venn diagram		T1:6-7	15-02-17				

4.	Identifying test cases	**Application:** This Unit helps to understand different test scenarios and can be easily implement in real time. **Outcome:** Compare and pick out the right type of software testing process for any given real world problem.	T1:7-9	16-02-17				
5.	Error and fault taxonomies, Levels of testing.		T1:10-13	20-02-17				
6.	Examples: Generalized pseudocode, The triangle problem		T1:15-21	21-02-17				
7.	Examples: The NextDate function		T1:22-25	22-02-17				
8.	Examples: The commission problem		T1:26	22-02-17				
9.	Examples: The SATM (Simple Automatic Teller Machine) problem.		T1:27-30	23-02-17				
10.	Examples: The currency converter, Saturn windshield wiper		T1:30-31	27-02-17				
11.	Revision/Test		T1: 3-31	28-02-17				
12.	**UNIT 2- Boundary Value Testing, Equivalence Class Testing, Decision Table Based Testing:** Boundary value analysis	**Objective:** Understand different Testing Methods such as Boundary Value Testing, Equivalence Class Testing, Decision Table Based Testing **Application:** Helps to understand the situation where	T1:75-78	1-03-17				
13.	Robustness testing, Worst-case testing, Special value testing		T1:78-80	1-03-17				
14.	Examples, Random Testing		T1:80-86	2-03-17				
15.	Equivalence classes: Weak Normal Equivalence Class Testing, Strong Normal Equivalence Class Testing,		T1:89-91	6-03-17				
16.	Equivalence classes: Weak Robust Equivalence Class Testing , strong Robust Equivalence Class Testing, Equivalence Class Test Cases for		T1:91-92	7-03-17				

	the Triangle Problem	the different testing can be applied. **Outcome:** Carry out the software testing process in efficient way.						
17.	Equivalence classes: Next Date function		T1:93-98	8-03-17				
18.	Equivalence classes: Commission problem,		T1:98-100	8-03-17				
19.	Decision Table Testing: Decision table, Technique.		T1:103-108	13-03-17				
20.	Decision Table Testing: Test cases for the triangle problem, Next Date function, and the commission problem		T1:108-109	14-03-17				
21.	Decision Table Testing: Test cases for Next Date function, First try, second try, Third try.		T1:109-114	15-03-17				
22.	Revision/Test		T1:75-114	15-03-17				
23.	**UNIT 3- Path Testing, Data Flow Testing:** DD paths, Test coverage metrics	**Objective:** Understand different Testing Methods such as BasisPath Testing, Definition-Use Testing, Slice-based Testing **Application:** Helps to understand the situation where the different testing can be applied. **Outcome:** Automate the testing process by using several testing tools.	T1:131-139	16-03-17				
24.	Basis path testing: McCabe's Basis Path Testing		T1:139-142	20-03-17				
25.	Basis path testing: observation, Essential complexity		T1:142-147	21-03-17				
26.	Definition-Use testing: Introduction		T1:151-158	22-03-17				
27.	Definition-Use testing: Commission problem.		T1:158-161	22-03-17				
28.	Slice-based testing: Observation		T1:161-164	23-03-17				
29.	Slice-based testing: Example.		T1:164-167	27-03-17				

30.	Revision/Test		T1: 131-167	28-03-17				
31.	**UNIT 4- Levels of Testing, Integration Testing:** Traditional view of testing levels	**Objective:** Understand Testing Models and different Integration Testing **Application:** Helps to Implement the Testing conditions in real scenarios. **Outcome:** Establish a quality environment as specified in standards for developing quality software	T1:181-182	30-03-17				
32.	Alternative life-cycle models.		T1:183-185	3-04-17				
33.	The SATM system		T1:186-195	4-04-17				
34.	Separating integration and system testing.		T1:196-199	5-04-17				
35.	A closer look at the SATM system: Decomposition Based Integration,Top-Down Integration		T1:203-207	5-04-17				
36.	A closer look at the SATM system: Bottom-up Integration, Sandwich Integration, Call graph-based		T1:207-209	6-04-17				
37.	Path-based integrations.		T1:209-212	10-04-17				
38.	Revision/Test		T1: 181-212	11-04-17				
39.	**UNIT 7- Fault Based Testing, Test Execution:** Overview, Assumptions in fault based testing.	**Objective:** Study Fault Based Testing and how it will migrate from test specification to test cases. **Application:** Complete knowledge on test life cycle, so that anyone can be expert by collecting the test specification	T2:313-315	12-04-17				
40.	Mutation analysis		T2:315-318	12-04-17				
41.	Fault-based adequacy criteria.		T2:319-321	13-04-17				
42.	Variations on mutation analysis.		T2:321-325	20-04-17				

43.	Test Execution: Overview, From test case specifications to test cases	and derive proper test cases. **Outcome:** Have basic understanding and knowledge of contemporary issu es in software testing, such as component-based software testing problems	T2:327-328	24-04-17				
44.	Scaffolding, Generic versus specific		T2:329-331	25-04-17				
45.	Test oracles, Self-checks as oracles.		T2:332-336	26-04-17				
46.	Capture and replay.		T2:337-338	26-04-17				
47.	**UNIT8- Planning and Monitoring the process, Documenting Analysis and Test:** Quality and process	**Objective:** This unit deals with Planning, quality analysis, strategy management and Analysis of Document. **Application:** Knowledge in planning and analysis helps to improve the quality of processes in real time and periodical enhancement. **Outcome:** Analyze and improve the quality procedures based on the past experience.	T2:75-377	27-04-17				
48.	Test and analysis strategies and plans		T2:377-381	2-05-17				
49.	Risk planning		T2:382-388	3-05-17				
50.	Monitoring the process, Improving the process		T2:389-393	3-05-17				
51.	The quality team		T2:394-398	4-05-17				
52.	Organizing documents, Test strategy document		T2:399-402	8-05-17				
53.	Analysis and test plan		T2:455-457	9-05-17				
54.	Test design specifications documents		T2:458-460	10-05-17				
55.	Test and analysis reports.		T2:460-461	10-05-17				
56.	Test and analysis reports.		T2:462-465	11-05-17				
57.	**UNIT 5- System testing, Interaction Testing:** Threads, Basic concepts for requirements	**Objective:** Understand threads,	T1:229-236	18-05-17				

	specification	thread testing strategies, guideline for system testing **Application:** Helps to understand the situation where these testing can be applied. **Outcome:** Have an ability to Select software test models, criteria, strategies, and methods.						
58.	Finding threads, Structural strategies , Functional Strategies for thread testing. SATM test threads, System testing Guidelines		T1:237-256	22-05-17				
59.	ASF (Atomic System Functions) testing example, Context of interaction, A taxonomy of interactions, Interaction, composition, Client/Server Testing		T1:257-281	23-05-17				
60.	**UNIT 6- Process Framework:** Validation and verification, Degrees of freedom, Varieties of software.	**Objective:** Study the Process framework, verification and validation and different factors affecting this. **Application:** Helps to understand and implement testing by understanding the Organizational factors. Thus improve the quality of Product. **Outcome:** Have an ability to validate and verify test cases and better planning for quality product.	T2:15-25	24-05-17				
61.	Basic principles: Sensitivity, Redundancy, Restriction, Partition, Visibility, Feedback.		T2:29-38	24-05-17				
62.	The quality process Planning and monitoring Quality goals Dependability properties, Analysis, Testing Improving the Process Organizational factors.		T2:39-51	25-05-17				

63.	Revision	Solving VTU Question Papers		29-05-17				
64.	Revision			30-05-17				
65.	Revision			1-06-17				
66.	Revision			2-06-17				

Prepared By:_____
(Faculty)
Date & Sign_____

Reviewed by:_____
(Sub. expert)
Date & Sign_____

Approved by: _____
(HOD)
Date & Sign _____

Approved by: _____
(Principal/ Acad. Co)
Date & Sign _____