Uka Tarsadia University

B. Tech.
Semester V

SOFTWARE TESTING
CE5020

EFFECTIVE FROM June-2023

Syllabus version: 1.00

Subject Code	Subject Title	Teaching Scheme			
		Hours		Credits	
		Theory	Practical	Theory	Practical
CE5020	Software Testing	3	2	3	1

Subject Code	Subject Title	Theory Examination Marks		Practical Examination Marks	Total Marks
		Internal	External	CIE	
CE5020	Software Testing	40	60	50	150

Objectives of the course:
- To unfold the principles of software testing.
- To expose students with various tools and techniques of software testing.

Course outcomes:

Upon completion of the course, the student shall be able to,

CO1: Understand the process and significance of software testing and Apply software verification and validation.
CO2: Understand and apply black box software testing techniques.
CO3: Understand and apply white box software testing techniques.
CO4: Understand the concept of optimized testing.
CO5: Understand and implement object oriented testing strategies.
CO6: Understand and implement automated testing.

Sr. No.	Topics	Hours
	Unit – I	
1	**Introduction to Software Testing:** Introduction, The testing process, Selection of good test cases, Measurement of testing, Incremental testing approach, Basic terminology related to software testing, Testing life cycle, Principles of testing, Limitations of testing, Available testing tools, Techniques, and Metrics.	5

2	**Software Verification and Validation:** Introduction, Differences between Verification and Validation, Differences between Quality analysis and Quality control, Evolving nature of area, V&V limitations, Categorizing V&V techniques, Role of V&V in SDLC, Proof of correctness, Simulation and Prototyping, Requirements tracing, Software V&V Planning (SVVP), Software Technical Reviews (STRs), Independent V&V Contractor (IV&V), Positive and negative effects of software V&V on projects, Standard for software test documentation (IEEE829).	
Unit – II		
3	**Black-Box Testing Techniques:** Introduction to Black-Box testing, Boundary Value Analysis (BVA), Equivalence class testing, Decision table based Testing, Cause-effect graphing technique, Comparison on Black-box testing techniques, Kiviat charts.	6
Unit – III		
4	**White-Box Testing Techniques:** Introduction to White-box testing, Static versus dynamic White-box testing, Dynamic White-box testing techniques, Mutation testing versus error seeding – differences in tabular form, Comparison of Black-box and White-box testing in tabular form, Introduction to Gray-box testing.	7
Unit – IV		
5	**Reducing the Number of Test Cases and Levels of Testing:** Prioritization guidelines, Priority category scheme, Risk analysis, Regression testing, Prioritization of test cases for regression testing, Unit, Integration, System, and Acceptance testing Relationship, Integration testing.	5
Unit – V		
6	**Object-Oriented Testing:** Basic unit for testing, Inheritance, and testing, Basic concepts of state machines, Testing object–oriented systems, Heuristics for class testing, Levels of object-oriented testing, Unit testing a class, integration testing of classes, System testing, Regression and Acceptance testing managing the test process, Design for testability.	7
Unit – VI		
7	**Automated Testing:** Introduction to automated testing, Consideration during automated testing, Types of testing tools – static v/s dynamic,	6

	Problems with manual testing, Benefits of automated testing, Disadvantages of automated testing, Criteria for selection of test tools, Characteristics of modern testing tools.	

Sr. No.	Software Testing (Practicals)	Hours
1	Case study on function and non-functional testing	4
2	Case study on testing of e-learning management system	4
3	Case study on testing of e-commerce web application	4
4	To test a static web application	6
5	To test dynamic web application	6
6	To prepare a test cases for a mobile application	4
7	To develop a simple system for automated software testing	9
8	To perform automated software testing on web application	8

Text book:
1. Rajiv Chopra, "Software testing – A Self-Teaching Introduction", Mercury Learning and Information, ISBN 978-1-683921-66-0

Reference books:
1. Naresh Chauhan, "Software Testing – Principles and Practice", Oxford University press, ISBN 978-0-1980618-47
2. Stephen Vance, "Quality Code- Software Testing Principles, Practice, and Patterns", Addison-Wesley, ISBN 0-321-83298-1

Course objectives and Course outcomes mapping:
- To unfold the principles of software testing: CO1, CO2, CO3, and CO5.
- To expose students with various tools and techniques of software testing: CO4, CO5, and CO6.

Course units and Course outcomes mapping:

Unit No.	Unit Name	Course Outcomes					
		CO1	CO2	CO3	CO4	CO5	CO6
1	Introduction to Software Testing and Software Verification and Validation	✓					
2	Black-Box Testing Techniques		✓				
3	White-Box Testing Techniques			✓			
4	Reducing the Number of Test Cases and Levels of Testing				✓		
5	Object-Oriented Testing					✓	
6	Automated Testing						✓

Programme outcomes:

PO 1: Engineering knowledge: An ability to apply knowledge of mathematics, science, and engineering.

PO 2: Problem analysis: An ability to identify, formulates, and solves engineering problems.

PO 3: Design/development of solutions: An ability to design a system, component, or process to meet desired needs within realistic constraints.

PO 4: Conduct investigations of complex problems: An ability to use the techniques, skills, and modern engineering tools necessary for solving engineering problems.

PO 5: Modern tool usage: The broad education and understanding of new engineering techniques necessary to solve engineering problems.

PO 6: The engineer and society: Achieve professional success with an understanding and appreciation of ethical behavior, social responsibility, and diversity, both as individuals and in team environments.

PO 7: Environment and sustainability: Articulate a comprehensive world view that integrates diverse approaches to sustainability.

PO 8: Ethics: Identify and demonstrate knowledge of ethical values in non-classroom activities, such as service learning, internships, and field work.

PO 9: Individual and team work: An ability to function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give/receive clear instructions.

PO 11: Project management and finance: An ability to demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: recognition of the need for, and an ability to engage in life-long learning.

Programme outcomes and Course outcomes mapping:

Programme Outcomes	Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
PO1						
PO2		✓	✓		✓	✓
PO3		✓	✓		✓	✓
PO4						
PO5						✓
PO6						
PO7						
PO8						
PO9	✓					
PO10						
PO11						
PO12						