Program : **Diploma in Computer Engineering / Computer Hardware Engineering / Information Technology / Robotic Process Automation**	
Course Code :**6131C**	Course Title: **Software Testing**
Semester : **6 / 6 / 6 / 4**	Credits: **4**
Course Category: **Program Elective**	
Periods per week: **4(L:4 T:0 P:0)**	Periods per semester:**60**

Course Objectives:

- Understand dynamic and static testing techniques.
- Familiarize automation testing tools.
- Lay foundation for building industrial strength software.

Course Prerequisites:

Topic	Course code	Course name	Semester
Basic knowledge on software engineering concepts.		Project Management and Software Engineering	5

Course Outcomes:

On completion of the course, the student will be able to:

COn	Description	Duration (Hours)	Cognitive level
CO1	Identify Software Testing Concepts, its Life Cycle and Methodology.	13	Understanding
CO2	Make use of dynamic Testing techniques to design test cases.	15	Applying
CO3	Make use of Static testing techniques, Validation activities and Regression testing	17	Applying
CO4	Explain test automation – Compare commercial and Open Source testing tools.	13	Understanding
	Series Test	2	

CO – PO Mapping

Course Outcomes	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7
CO1	2						
CO2		3	3				
CO3		3					
CO4	2						

3-Strongly mapped, 2-Moderately mapped, 1-Weakly mapped

Course Outline

Module Outcomes	Description	Duration (Hours)	Cognitive Level
CO1	**Identify Software Testing Concepts, its Life Cycle and Methodology.**		
M1.01	Explain the concepts of Software testing, goals of software testing, Software testing Model	3	Understanding.
M1.02	Explain the software testing terminologies, States of bugs and bug classification	3	Understanding.
M1.03	Identify principles of software testing, Software Testing Life Cycle and its phases	3	Understanding.
M1.04	Explain Software testing methodology	4	Understanding
Contents: Introduction to software Testing -Software Testing Process, Goals of software testing, Model for software testing- States of a bug- Bug classification based on criticality- Bug classification based on SDLC- Testing principles- Software Testing Life Cycle (STLC) and different phases- Software testing methodology- Software testing strategy, Test strategy matrix, Validation activities, Testing tactics.			
CO2	**Make use of dynamic Testing techniques to design test cases.**		
M2.01	Black box testing techniques - Implement the test case design techniques offered by BVA-BVC and Robustnesstesting method.	3	Understanding
M2.02	Demonstrate test case design using Equivalence Class testing.	2	Understanding
M2.03	Explain state table based and decision table based testing.	3	Understanding

M2.04	Illustrate the white box testing techniques.	3	Understanding
M2.05	Develop test scenarios and test cases for applications using BVA and Equivalence class testing	4	Applying
	Series Test – I	1	

Contents:

Dynamic Testing techniques with test case design, Black box testing techniques, Boundary Value Analysis- Boundary value checking, Robustness testing method- Equivalence class testing, State table based testing, Decision table based testing, White box testing techniques- Need of white box testing- Logic coverage criteria- Basis path testing

CO3	**Make use of Static testing techniques, Validation activities and Regression testing**		
M3.01	Identify different static testing techniques – Inspection, Structured walkthrough, Technical reviews	3	Understanding
M3.02	Explain Unit validation testing.	2	Understanding
M3.03	Explain integration testing.	2	Understanding
M3.04	Explain function testing and system testing.	3	Understanding
M3.05	Summarize Acceptance testing techniques.	2	Understanding
M3.06	Explain regression testing techniques.	2	Understanding
M3.07	Develop test scenarios and test cases for applications	3	Applying

Contents:

Static testing, Inspections - Inspection team, Inspection process, Benefits of inspection process- Structured walkthroughs, Technical reviews, Validation Testing, Unit validation testing, Integration testing- Decomposition based integration- Function testing, System testing- Categories of system testing- Acceptance testing- Alpha testing , Beta testing- Regression Testing- Regression testability, Regression testing types, Objectives of regression testing- Regression testing techniques- Selective retest technique.

CO4	**Explain test automation – Compare commercial and Open Source testing tools.**		
M4.01	Illustrate the concept of automation testing.	2	Understanding
M4.02	Illustrate the use of open-source testing tool, Selenium for test automation.	3	Understanding
M4.03	Outline features of Commercial testing tools – WinRunner, SilkTest, LoadRunner, Jmeter, TestDirector.	2	Understanding
M4.04	Explain Object Oriented Testing	3	Understanding
M4.05	Explain testing of web based systems	3	Understanding

	Series Test – II	1	

Contents:

Test Automation- Automation and Testing tools- Need for automation- Categories of testing tools- Selection of testing tools- Guidelines for automated testing- Understand open source testing tool – Selenium- Basic browser navigation commands in Selenium – get, navigate, manage, quit, wait, submit, close, send keys- Overview of commercial testing tools – WinRunner, Silk Test, LoadRunner, J meter, Test Director- Object Oriented Testing – Issues in OOT, Strategy and tactics of Testing OOS, Verification and Validation of OOS, Testing OO classes, Inheritance testing, Integration testing- Understand testing of web based systems- Challenges in testing for web based system- Testing of web based systems

Text / Reference

T/R	Book Title/Author
T1	Naresh Chouhan, ***Software Testing Principles and Practices*** , Pearson, Second Edition
R1	Ian Sommerville, ***Software Engineering*****,** Pearson, Ninth Edition
R2	Roger S Pressman, ***Software Engineering a practitioner's approach*** – Mc Graw Hill , Seventh Edition

Online Resources

Sl.No	Website Link
1	https://www.guru99.com/selenium-tutorial.html
2	https://www.wisdomjobs.com/e-university/winrunner-tutorial-171.html
3	https://www.etestinghub.com/testdirector.php
4	https://www.tutorialspoint.com/jmeter/index.htm