

Course Code	USA20503J	Course Name	SOFTWARE ENGINEERING AND TESTING	Course Category	C	Professional Core Course	L	T	P	C
							4	0	2	5

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Applications	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to,	Learning	Program Learning Outcomes (PLO)
----------------------------------	--	----------	---------------------------------

CLR-1 :	Familiarize the software life cycle models and software development process	1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CLR-2 :	Understand the various techniques for requirements, planning and Testing																		
CLR-3 :	Examine the basic methodologies for software design, development, testing																		
CLR-4 :	Manage user expectations and software development team																		
CLR-5 :	Acquire the latest industry knowledge like agile for development																		
CLR-6 :	Usage of tools and comply the global standards for testing																		

Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)	Fundamental Knowledge	Application of Concepts	Link with Related Disciplines	Procedural Knowledge	Skills in Specialization	Ability to Utilize Knowledge	Skills in Modeling	Analyze, Interpret Data	Investigative Skills	Problem Solving Skills	Communication Skills	Analytical Skills	ICT Skills	Professional Behavior	Life Long Learning
CLO-1 :	Identify the process of project life cycle model and process	2	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-
CLO-2 :	Analyze and specify software requirements through a productive working Relationship Customers.	3	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-
CLO-3 :	Design the system based on Functional Oriented and Object Oriented Approach for Software Design.	3	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-
CLO-4 :	Develop the correct and robust code for the software products	3	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-
CLO-5 :	Perform by applying the test plan and various testing techniques	3	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-
CLO-6 :	Analyze the key issues of Software maintenance	3	85	80	L	H	H	H	H	-	-	M	M	L	-	H	-	-	-

Duration (hour)		18	18	18	18	18
S-1	SLO-1	The Evolving Role of Software	Computer-Based Systems	Principles of Testing	Integration testing	Performance Testing
	SLO-2	Software Engineering Definition	The System Engineering Hierarchy – System Modeling	Introduction-Testing Definition	Top down Integration testing	Factors of Governing
S-2	SLO-1	Software Characteristics	System Simulation	Phases of software	Bottom up Integration testing	Regression testing
	SLO-2	Software Applications and A Crisis	Comparison of various software Development	Error, Fault, Bug-Failure of the system –Comparison of the terms	Bi-Directional Integration	Types of regression testing
S-3	SLO-1	Software Myths	Business Process Engineering: An Overview	Types of testing-	System Integration	Software testing strategy
	SLO-2	Types Of Myths	Requirements Engineering process	Quality assurance	System Acceptance Testing	Best practice in regression testing
S-4	SLO-1	Software Engineering : Layered Technology	Software requirements specification	Quality Control	Functional testing	Methodology for Performance Testing



S 5-6	SLO-1	Lab :Problem Statement Preparation	Lab : Software Requirement Specification Document Preparation	Lab : Preparation of DFD of any Project	Lab : Test Case Design	Lab : Usage of Text
	SLO-2					
S-7	SLO-1	Software Process	Characteristics of Good Requirements	Testing verification and validation	Non Functional testing	Tools for Performance Testing
	SLO-2	Software Process Models	Types of Requirements	White Box Testing	Functional Vs Non Functional Testing	Challenges for Performance Testing
S-8	SLO-1	Linear Sequential Model	Requirements Elicitation	Techniques of White Box Testing	System Testing	Performing Initial Test, Understanding the Criteria
	SLO-2	Advantages And Disadvantages	Requirements Analysis and Negotiation	Black box testing	Design and Architectural Verification	Classifying Test Cases.
S-9	SLO-1	Prototyping Model	Requirement Documentation	Techniques of Black box testing	Deployment Testing	Resetting the Test Cases
	SLO-2	Advantages And Disadvantages	Requirement Specification and Analysis	Static Testing	Beta Testing	
S-10	SLO-1	Rapid Application Development Model	Requirement Review, Validation	DYNAMIC Testing	Certification, Standards	Concluding the Results of Regression Testing
	SLO-2		Software Requirement Specification and System Requirement Specifications			
S 11-12	SLO-1	Lab :Problem Statement Preparation	Lab: Drawing E-R Diagram for any project	Lab : Preparation of DFD of any Project	Lab :Testing – Calculator	Lab : Testing – Sorting
	SLO-2					
S-13	SLO-1	Evolutionary Process Models	Characteristics of Good SRS Document	Challenges in white box testing	Testing for Compliance	Configuration testing
	SLO-2	Incremental Model	Requirement Management	Black Box Testing	Scalability Testing	compatibility testing
S-14	SLO-1	Advantages and Disadvantages	Software Prototyping	Techniques of Black Box Testing	Reliability testing	Test plan with debugging
	SLO-2	Spiral Model, WIN WIN Model	Selecting the prototyping approach	Structural testing	Stress testing	Levels of testing
S-15	SLO-1	Concurrent Development Model	Specification Principles, Representation	Static testing	Acceptance Testing	Testing tools
	SLO-2	Component Based Development	Specification Review	Verification & Validation Techniques	Acceptance Criteria	Key Issues in Software maintenance
S-16	SLO-1	Comparison of Process models	Characteristics of Good E-R Diagrams	Cyclomatic complexity	Selecting Test Cases	Examples University Previous Question Papers Discussion
	SLO-2	Advantages and Disadvantages	SRS Document	Control flow graph	Executing Tests	
S 17-18	SLO-1	Lab : Software Requirement Specification Document Preparation	Lab: Drawing E-R Diagram for any project	Lab : Test Case Design	Lab : Testing – Mark sheet	Lab : Testing – Login Form
	SLO-2					

Learning Resources	1. Roger S. Pressman, (2001), "Software Engineering ", Fifth edition, McGraw-Hill Higher Education - A Division of The McGraw-Hill Companies.	3. William E. Perry (2006), "Effective Methods of Software Testing", 3rd Ed, Wiley India.
	2. Srinivasan Desikan and Gopalasamy Ramesh, "Software Testing for Principles and Practices", Pearson Education.	4. Renu Rajani, Pradeep Oak (2007), "Software Testing", TMH



Learning Assessment											
Level	Bloom's Level of Thinking	Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (10%)		CLA – 3 (20%)		CLA – 4 (10%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	20%	20%	15%	15%	15%	15%	15%	15%	15%	15%
	Understand										
Level 2	Apply	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
	Analyze										
Level 3	Evaluate	10%	10%	15%	15%	15%	15%	15%	15%	15%	15%
	Create										
	Total	100 %		100 %		100 %		100 %		100 %	

# CLA – 4 can be from any combination of these: Assignments, Seminars, Tech Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.,

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Mr.G.Muruganandam, Group Project Manager, HCL Technologies, Chennai	Dr.S.Gopinathan, Professor, University of Madras, Chennai	Mr .J. Venkata Subramanian, SRMIST
Mr.M. Hemachandar, Tech Lead, Wipro Limited, Chennai		Mrs. M. R. Sudha, SRMIST