Course Code	PCS21G02J	Course Name	SOFTWARE ENGINEERING	Course Category	G	Generic Elective Course	L	T	P	C
							3	0	2	4

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Science	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 lifecycle 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 managing a technology project*																		
CLR-3 :	*Examine basic methodologies for software design, development, testing, closure and implementation*																		
CLR-4 :	*Understand manage users expectations and the software development team*																		
CLR-5 :	*Acquire the latest industry knowledge, tools and comply to the latest global standards for project management*																		
Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:	Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)	Disciplinary Knowledge	Critical Thinking	Problem Solving	Analytical Reasoning	Research Skills	Team Work	Scientific Reasoning	Reflective Thinking	Self-Directed Learning	Multicultural Competence	Ethical Reasoning	Community Engagement	PSO 1	PSO 2	PSO 3
CLO-1 :	*Identify the process of life cycle model and process project*	3	80	70	L	H	-	H	L	-	-	-							
CLO-2 :	*Analyze and specify software requirements through a productive working Relationship with project stakeholders*	3	85	75	M	H	L	M	L	-	-	-					-	-	-
CLO-3 :	*Design the system based on Functional Oriented and Object Oriented Approach for Software Design.*	3	75	70	M	H	M	H	L	-	-	-					-	-	-
CLO-4 :	*Develop the correct and robust code for the software products*	3	85	80	M	H	M	H	L	-	-	-					-	-	-
CLO-5 :	*Perform by applying the test plan and various testing techniques*	3	85	75	H	H	M	H	L	-	-	-					-	-	-

Duration(Hour)		15	15	15	15	15
S-1	SLO-1	*Introduction to software Engineering*	*System Engineering*	*Introduction to Testing*	*Project Management Spectrum*	*Risk Management*
	SLO-2	*Characteristics of software*	*Components of System Engineering*	*Definition , Characteristics of Testing*	*Four P's*	*Reactive and Proactive Risk Strategies*
S2	SLO-1	*The Changing Nature of software*	*Requirements Engineering Tasks*	*Testing Strategies for Conventional Software*	*The People and the Product*	*Software Risks*
	SLO-2	*Legacy Software and Software myths*	*Process, Initiating and Eliciting requirements.*	*Unit testing and Integration testing*	*Role of People*	*Risk Identification and Risk Projection*
S3	SLO-1	*A Generic view of process Software Engineering*	*Building the Analysis Model*	*Validation Testing*	*The Process and the Project*	*Risk refinement*
	SLO-2	*A layered Technology*	*Analysis Modeling Approaches*	*Verification Vs Validation*	*Role of Process*	*Risk Mitigation*
S4 –5	SLO-1	*Laboratory1:Identifing Project Objective and Scope*	*Laboratory 4:Project Planning*	*Laboratory 7: Function Oriented Diagram*	*Laboratory 10:Test Case design for unit testing*	*Laboratory 13: Preparation of Timeline charts and Tracking the Scheduling*
	SLO-2					
S6	SLO-1	*A process framework*	*Data Modeling Concepts*	*System Testing*	*Metrics for Process and Projects-Estimation*	*Monitoring and Management*
	SLO-2	*Capability Maturity Model Integration*	*Example Diagram*	*Non-Functional testing*	*LOC, FP, Object Oriented.*	*Example*
S7	SLO-1	*Process Models*	*Scenario based Modeling*	*Debugging Process*	*Estimation*	*Quality Concepts*
	SLO-2	*Water fall , RAD model*	*USE-CASE Diagram*	*Testing Tactics*	*Estimation models*	*SQA Activities*
S8	SLO-1	*Iterative Process Models*	*Flow Oriented Modeling*	*White Box Testing, Basic-Path testing*	*The Project Planning Process*	*Software Reviews and FTR*
	SLO-2	*Incremental ,Prototype and Spiral*	*Data Flow Diagram*	*Cyclomatic complexity calculation*	*Resources*	*Statistical Quality Assurance*
S9-10	SLO-1	*Laboratory 2:Selection of Suitable*	*Laboratory 5:Performing Various*	*Laboratory 8:User's View Analysis*	*Laboratory 11:Test Case design for*	*Laboratory 14: Estimation of Effort and*

Duration(Hour)		15	15	15	15	15
	SLO-2	*software process Model of the suggested system*	*Requirement Analysis*		*Integration testing*	*Risk Identification*
S-11	SLO-1	*Prescriptive models*	*Design Engineering*	*Black Box Testing*	*Decomposition Techniques*	*The Software Configuration Management*
	SLO-2	*Phases of the model*	*Example*	*Equivalence Partitioning*	*calculations of Decomposition techniques*	*SCM Repository*
S-12	SLO-1	*Specialized Process Models*	*Software Design Concepts*	*BVA , Error Guessing*	*Empirical Estimation Models*	*Business Process Reengineering*
	SLO-2	*The Unified Process Model*	*Example Diagrams*	*Cause-Effect Graphing*	*COCOMO model*	*Reengineering Diagram and Example.*
S-13	SLO-1	*An agile view of Process*	*The Design Model*	*Testing for Specialized Environments*	*Project Scheduling Concepts*	*Reverse Engineering*
	SLO-2	*Case study on Best SDLC selection based on the Scenario*	*Examples for all designs*	*Preparation of Test case Plan and Report*	*Examples*	*Forward Engineering*
S14-15	SLO-1 SLO-2	*Laboratory 3:Problem Statement Preparation*	*Laboratory 6:Develop Software Requirement Specification Sheet (SRS)*	*Laboratory 9:Structure view diagram*	*Laboratory12:Perforing Testing and Debugging for a sample code*	*Laboratory 15: Software Quality Assurance Components.*

Learning Resources	1. Roger, S. Pressman (2004), Software Engineering: A Practitioner Approach, McGraw Hill International Edition, Sixth Edition, New Delhi 2. Waman, S Jawadekar (2004), Software Engineering: Principles and Practice, McGraw Hill Education Pvt. Limited, New Delhi.	3. RohitKhurana (2011), Software Engineering-Principles and Practices, Vikas Publishing House Pvt. Ltd., Second Edition, New Delhi. 4. Chairperson, Counting Practices Committee, Valerie Marthaler, EDS, Troy, Michigan, Function Point Counting Practices Manual Release 4.1.1, The International Function Point User Group, April 2000. 5. Carlo Ghezzi, Mehdi Jazayari, Dino Mandrioli (1991), Fundamentals of Software Engineering, Prentice Hall of India, New Delhi.

Learning Assessment											
Bloom's Level of Thinking		**Continous 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 Understand	20%	20%	15%	15%	15%	15%	15%	15%	15%	15%
Level 2	Apply Analyze	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
Level 3	Evaluate Create	10%	10%	15%	15%	15%	15%	15%	15%	15%	15%
	Total	100 %		100 %		100 %		100 %		100%	

CLA – 4 can be from any combination of these: Assignments, Seminars, Short 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. S. Karthik, Assistant Consultant, Tata Consultancy Services	Dr.S.Sasikala, Associate Professor and Head, Dept. of Computer Science, University of Madras	Dr.Kanchana Dr.Kalpana