

CSE11117	Software Engineering	L	T	P	C
Version 1.0	Contact Hours -45	3	0	0	3
Pre-requisites/Exposure	Software/Hardware evolution at basic level				
Co-requisites	--				

Course Objectives:

- 1. To help the student to acquire knowledge of software evolution process.
- 2. To enable students modelling software project with appropriate metric and precision at workplace.
- 3. To give the students a perspective to software design process variables by exposing them to software specification document; and also, to enrich their software testing ability.
- 4. To enable students, acquire testing and quality assessment of model required for their profession.

Course Outcomes:

On completion of this course, the students will be able to

CO1. Interpret the impact of software engineering.

CO2. Communicate with proper software model paradigm to pupils.

CO3. Enhancement of software metric engineering application in industry.

CO4. Effectively analyse testing and maintenance of software project.

CO5. Illustrate software metric analysis for an effective model.

Course Description:

There is a growing need for talented software developers across every industry. As technology advances, the ability to build quality software while considering design, development, security, and maintenance is sought after amongst all kinds of companies, from finance and banking to healthcare and national security.

Software Engineering applies the knowledge and theoretical understanding gained through computer science to building high-quality software products. As a maturing discipline, software is becoming more and more important in our everyday lives. Our software development and engineering professional program is Pace University's response to the tremendous growth of the software development industry.

Course Content:

Units	Lecture Hours
Unit-I: Software - Evolving role of it, a crisis on the Horizon and its Myths, Software process models: linear sequential model, prototyping model, RAD	9

Units	Lecture Hours
model, Evolutionary model, Formal methods model, Component based development, fourth generation techniques, Software development and requirement analysis using Agile, Scrum framework.	
Unit-II: Management spectrum, people, problem, process, project and few Critical approach, Software Process and project metrics: Measure, Metrics and Indicators, Process and Project Domain related metrics, Software Measurement, Reconciling of Different, Metrics Approaches, Software quality metrics, Validation management, Software project planning: Observations on estimation, Objectives of Project planning.	10
Unit-III: Resources: Software project estimation, Empirical models for estimation, Automated estimation tools, Risk management and Software risks: Identification, Risk projection, safety risks and hazards; RMMM plans, Risk management	8
Unit-IV: Project scheduling and tracking: Definition of task set and task network, Scheduling, earned value analysis, Tracking of Errors, Project planning, Software quality assurance: Concepts of Software Quality, Quality movement, Review of software quality assurance, Software reliability, Software quality metrics (MTTF, MTTR, MTBF ETC.)	9
Unit-V: Software configuration management: Object identification in software configuration, Configuring audit-SCM standards, Analysis concepts and principles: Requirement analysis, Software prototyping, Specification Review Analysis modeling, Data modeling, Functional modeling, Behavioral modeling, Software design, Software testing techniques: White box and black box testing, Software testing strategies - Unit testing, Integrating testing, System testing.	9

Text Books:

- 1. Software Engineering: A practitioner's approach, 8th Edition, Roger S. Pressman, McGraw Hill
- 2. An integrated approach to Software Engineering, Springer/Narosa Edition, Pankaj Jalote.

Reference Books:

- Fundamentals of Software Engineering, 4th Edition, Rajib Mall, Prentice Hall, India.

Examination Scheme:

Components	Mid term	Continuous Assessment	End Term
Weightage (%)	20	30	50

Relationship between Course Outcomes (COs) and Program Outcomes (POs):

1=weakly mapped

2= moderately mapped

3=strongly mapped

Course Code	Course Name	COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CSE11117	Software Engineering	CO11117.1	3	2	1	3	2	2	1	-	1	2	2	3	2	1	2
CSE11117	Software Engineering	CO11117.2	3	2	2	1	2	2	2	-	3	-	-	2	2	1	1
CSE11117	Software Engineering	CO11117.3	3	2	1	1	3	3	1	-	2	-	-	3	2	1	3
CSE11117	Software Engineering	CO11117.4	2	3	3	1	3	2	3	2	2	-	-	2	1	2	1
CSE11117	Software Engineering	CO11117.5	2	3	2	3	3	3	2	3	2	-	-	3	1	3	3
CSE11117	Software Engineering	CO11117	2.6	2.4	1.8	1.8	2.6	2.4	1.8	2.5	2.0	2.0	2.0	2.6	1.6	1.6	2.0