ST. JOSEPH'S
COLLEGE OF ENGINEERING
AND TECHNOLOGY,
- PALAI -
AUTONOMOUS

Choondacherry P.O., Pala, Kottayam - 686579
Kerala, India

MINOR
in
SOFTWARE ENGINEERING

Offered by CS, CA & CC Departments

Eligible Branches : CE, EC, EE, ME

2024 SCHEME

Minor (Software Engineering)

Sl. No:	Semester	Course Code	Course Title (Course Name)	Credit Structure			SS	Total Marks		Credits	Hrs./ Week
				L	T	P		CIA	ESE		
1	3	24SJMNCST309	Software Engineering /MOOC[#]	3	1	0	5	40	60	4	4/5
2	4	24SJMNCST409	Object Oriented Software Engineering /MOOC[#]	3	1	0	5	40	60	4	4/5
3	5	24SJMNCST509	Software Testing /MOOC	3	1	0	5	40	60	4	4
4	6	24SJMNCST609	Software Project Management /MOOC	3	0	0	4	40	60	3	3
Total										**15**	**15/ 17**

** Students from the CE, ECE, EEE, and ME branches are eligible for the Minor Programme.*

** Students must register for theory courses listed in the 3rd and 4th semesters of the Minor curriculum.*

[#]Students who fail a theory course listed in the Minor curriculum are permitted to register for an alternate MOOC course specified in the Minor curriculum.

SEMESTER 3

SOFTWARE ENGINEERING

Course Code	24SJMNCST309	**CIE Marks**	40
Teaching Hours/Week (L: T:P: R)	3:1:0:0	**ESE Marks**	60
Credits	4	**Exam Hours**	2 Hrs. 30 Min.
Prerequisites (if any)	Nil	**Course Type**	Theory

Course Objectives:

- Understand Software Engineering lifecycle, model and basics of project management.
- Learn and evaluate various software measurement techniques.
- Apply manual and automated software testing & debugging techniques.

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	**Software Engineering Fundamentals** Introduction, SDLC life cycle, SDLC Models, Project Management **Agile Development** Agility and Agile Process model, Extreme Programming, SCRUM development, other process models of Agile Development and Tools	**12**
2	**Software Requirement Analysis and Design** Understanding the Requirement, Requirement Modeling, Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, Requirement Engineering, Feasibility Studies, Structural design, behavior design, functional design, UML Diagrams, Design Pattern	**12**
3	**Software Architectural Design** Data centered architecture, data flow architecture, call and return architecture, object-oriented architecture and layered architecture, Wireframing, Software analysis for legacy systems.	**12**
4	**Software Measurement Techniques** Size and Cost Estimation: Function point analysis, LOC estimation, COCOMO. Software metrics, risk estimation, effort estimation	**12**

Course Assessment Method (CIE: 40 marks, ESE: 60 marks)

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
• 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks **(8x3 =24 marks)**	• Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 sub divisions. **(4x9 = 36 marks)**	**60**

Course Outcomes (COs)

At the end of the course students should be able to:

	Course Outcome	Bloom's Knowledge Level (KL)
CO1	Illustrate the key components of the Software Engineering field	**K3**
CO2	Analyze and apply best SDLC models as per business need	**K3**
CO3	Design appropriate system requirement specifications	**K3**
CO4	Design and implement appropriate measurement Quality	**K3**

Note*: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create*

CO-PO Mapping Table (Mapping of Course Outcomes to Program Outcomes)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	✓	✓	✓								✓
CO2	✓	✓	✓			✓					✓
CO3	✓	✓	✓			✓					✓
CO4	✓	✓	✓							✓	✓

Note: *1: Slight (Low), 2: Moderate (Medium), 3: Substantial (High), -: No Correlation*

Text Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Software Engineering	Sommrville	Pearson Education	1998
2	Software Engineering – A Practitioner's Approach	Roger S Pressman	McGraw-Hill	2023

Reference Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Pankaj Jalote,	Software Engineering	A Precise Approach Wiley	2010
2	Software Engineering Fundamentals	Ali Behhforoz & Frederick Hudson	Oxford	1996
3	Rajib Mall	Fundamentals of software Engineering	Prentice Hall of India	2014

Video Links (NPTEL, SWAYAM…)	
Module No.	**Link ID**
1	Link: https://onlinecourses.nptel.ac.in/noc20_cs68/preview

SEMESTER S4

OBJECT ORIENTED SOFTWARE ENGINEERING

Course Code	**24SJMNCST409**	**CIE Marks**	40
Teaching Hours/Week (L:T:P: R)	3:1:0:0	**ESE Marks**	60
Credits	4	**Exam Hours**	2 Hrs. 30 Min.
Prerequisites (if any)	None	**Course Type**	Theory

Course Objectives:

1. To understand the fundamental concepts of object-oriented modeling, analysis, and design using UML for effective system development.
2. To enable students to apply object-oriented principles and methodologies in identifying classes, relationships, and constructing detailed design models for real-world applications.

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	Object Oriented Concepts and Modeling: Object Orientation, Importance of Modeling; Object Oriented Modeling, Object oriented analysis, Identifying the elements of an object model, Introduction to UML, Conceptual Model of UML, Architecture	**6**
2	Object Oriented methodologies, Basic and Advanced Structural Modeling, Classes Relationship, Class diagram, Advanced Relationship, Interface, Packages, Object Diagram, Basic Behavioral Modeling, Use cases, Use Case Diagram, Interaction Diagram, Activity Diagram, State chart Diagram. Analysis: Software Requirement: Functional and Non-functional Requirements, Known and Unknown Requirements. Characteristics of a Good Requirement, Software Requirements Specification Document: Nature of the SRS Document, Organization of the SRS Documents, Requirements Change Management, Overview of Analysis, Analysis Object Models and Dynamic Models, Entity, Boundary, and Control Objects, Structured Analysis versus Object-Oriented Analysis,	**12**
3	Identification of Classes: Entity Classes, Interface Classes, Control Classes, Identification of Relationships: Association, Aggregation, Multiplicity, Composition, Dependency, Generalization, Modeling Relationships.	**8**
4	Object Oriented Design: Interaction Diagrams, Refinement of Use Case Description, Construction of Detailed Class diagram, Development of Detailed Design and Creation of Software Design Document, Generating Test Cases from Use Cases, Object-Oriented Design Principles for Improving Software Quality, Commonly Used Testing Terminology, and Deriving Test Cases from Use Cases. Frameworks and design patterns	**10**

Course Assessment Method (CIE: 40 marks, ESE: 60 marks)

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
• 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks **(8x3 =24 marks)**	• Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 subdivisions. **(4x9 = 36 marks)**	**60**

Course Outcomes (COs)

At the end of the course students should be able to:

Course Outcome		Bloom's Knowledge Level (KL)
CO1	Explain the fundamental concepts of Object Orientation and the importance of modeling in software development.	**K2**
CO2	Apply object-oriented methodologies to construct various UML diagrams such as Class, Use Case, and Activity diagrams.	**K3**
CO3	Apply techniques to identify entity, interface, and control classes along with their relationships such as association, aggregation, and generalization.	**K3**
CO4	Apply object-oriented design principles to refine use cases, interaction diagrams, and detailed class diagrams for system design.	**K3**
CO5	Apply design patterns and testing concepts to improve software quality and reliability in object-oriented systems.	**K3**

Note*: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create*

CO-PO Mapping Table (Mapping of Course Outcomes to Program Outcomes)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	✓	✓	✓							✓	
CO2	✓	✓	✓	✓	✓				✓	✓	✓
CO3	✓	✓	✓	✓	✓				✓		
CO4	✓	✓	✓	✓	✓				✓	✓	
CO5	✓	✓	✓	✓	✓				✓		✓

Note: *1: Slight (Low), 2: Moderate (Medium), 3: Substantial (High), -: No Correlation*

Text Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Object-Oriented Modeling and Design with UML	Michael Blaha, James Rumbaugh	Pearson Education	2nd Edition, 2005
2	Object-Oriented Analysis and Design with Applications	Grady Booch, Robert A. Maksimchuk, Michael W. Engel, Bobbi J. Young	Pearson Education	3rd Edition, 2007
3	Design Patterns: Elements of Reusable Object-Oriented Software	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides	Addison-Wesley Professional	1st Edition, 1995

Reference Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	*The Unified Modeling Language User Guide*	Grady Booch; James Rumbaugh; Ivar Jacobson	Addison-Wesley Professional	2nd Edition, **2005** (Amazon)
2	*Learning UML 2.0: A Pragmatic Introduction to UML*	Russ Miles; Kim Hamilton	O'Reilly Media	1st Edition, **2006** (ACM Digital Library)

Video Links (NPTEL, SWAYAM…)	
No.	**Link ID**
1	https://nptel.ac.in/courses/106105153

SEMESTER 5

SOFTWARE TESTING

Course Code	24SJMNCST509	**CIE Marks**	40
Teaching Hours/Week (L: T:P: R)	3:1:0:0	**ESE Marks**	60
Credits	4	**Exam Hours**	2 Hrs 30 Min.
Prerequisites (if any)	None	**Course Type**	Theory

Course Objectives:

1. To Cultivate proficiency in software testing methodologies and techniques.
2. To Foster expertise in software testing tools and technologies.

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	**Introduction to Software Testing** Introduction to Software Testing - Concepts, importance of testing, Testing Terminologies - Verification, validation, fault, error, bug, test cases, and coverage criteria; **Types of testing** Types of Testing - Unit, Integration, System, Acceptance, Performance (stress, usability, regression), and Security Testing;	**12**
2	**Unit Testing -** Unit Testing- Static and Dynamic Unit Testing, **Mutation Testing -** Mutation Testing- Mutation operators, mutants, mutation score, and modern mutation testing tools **JUnit Framework** - Automation of unit testing, frameworks for testing in real-world projects; Case Study - Mutation testing using JUnit	**12**
3	Testing Methods - Black-Box, White-Box, and Grey- Box Testing; **Advanced White Box Testing & Security Testing: -** Graph Coverage Criteria - Node, edge, and path coverage; prime path and round-trip coverage; Graph Coverage for Code - Control flow graphs (CFGs) for complex structures (e.g., loops, exceptions); Data Flow Criteria - du paths, du pairs, subsumption relationships;	**12**
4	**Black Box Testing -** Input space partitioning, domain testing, functional testing (equivalence class partitioning, boundary value analysis, decision tables, random testing); **Grey Box Testing** - Introduction, advantages, and methodologies (matrix testing, regression testing, orthogonal array testing);	**12**

Course Assessment Method (CIE: 40 marks, ESE: 60 marks)

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
• 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks **(8x3 =24 marks)**	• Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 subdivisions. **(4x9 = 36 marks)**	**60**

Course Outcomes (COs)

At the end of the course students should be able to:

Course Outcomes		Bloom's Knowledge Level (KL)
CO1	Basic concepts of software testing and the various testing methods	**K1**
CO2	Demonstrate the ability to apply software testing techniques unit testing and mutation testing using Junit.	**K2**
CO3	Explain and apply graph coverage criteria in terms of control flow and data flow graphs to improve code quality.	**K3**
CO4	Demonstrate the importance of black-box and grey box approaches	**K3**

Note: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create

CO-PO Mapping Table (Mapping of Course Outcomes to Program Outcomes)

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

Note: 1: Slight (Low), 2: Moderate (Medium), 3: Substantial (High), -: No Correlation

Text Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Introduction to Software Testing.	Paul Ammann, Jeff Offutt	Cambridge University Press	2/e, 2016
2	Software Testing and Quality Assurance : Theory and Practice	Kshirasagar Naik, Priyadarshi Tripathy	Wiley	1/e, 2008

Reference Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Software Testing	Ron Patten	Pearson	2/e, 2005
2	Software Testing: A Craftsman's Approach	Paul C. Jorgensen	CRC Press	4/e, 2017
3	Foundations of Software Testing	Dorothy Graham, Rex Black, Erik van Veenendaal	Cengage	4/e, 2021
4	The Art of Software Testing	Glenford J. Myers, Tom Badgett, Corey Sandler	Wiley	3/e, 2011

Video Links (NPTEL, SWAYAM…)	
Module No.	**Link ID**
1	https://archive.nptel.ac.in/courses/106/101/106101163/
2	https://archive.nptel.ac.in/courses/106/101/106101163/
3	https://archive.nptel.ac.in/courses/106/101/106101163/
4	https://archive.nptel.ac.in/courses/106/101/106101163/
5	https://junit.org/

SEMESTER 6

SOFTWARE PROJECT MANAGEMENT

Course Code	24SJMNCST609	**CIE Marks**	40
Teaching Hours/Week (L: T:P: R)	3:0:0:0	**ESE Marks**	60
Credits	3	**Exam Hours**	2 Hrs. 30 Min.
Prerequisites (if any)	None	**Course Type**	Theory

Course Objectives:

1. To learn the techniques to effectively plan, manage, execute, and control projects within time and cost targets with a focus on Information Technology and Service Sector.
2. To learn agile project management techniques such as Scrum and DevOps.

SYLLABUS

Module No.	Syllabus Description	Contact Hours
1	**Project scheduling and feasibility study : -** Project Overview and Feasibility Studies - Identification, Market and Demand Analysis, Project Cost Estimate, Financial Appraisal; Project Scheduling - Project Introduction to PERT and CPM, Critical Path Calculation, Float Calculation and its importance.	12
2	**Resource Scheduling, Cost Control and Project management Features :-** Cost Control and Scheduling - Project Cost Control (PERT/Cost), Resource Scheduling & Resource Levelling; Project Management Features – Risk Analysis, Project Control, Project Audit and Project Termination.	12
3	**Agile Project Management :-** Agile Project Management - Introduction, Agile Principles, Agile methodologies, Relationship between Agile Scrum, Lean, DevOps.	12
4	**Scrum and DevOps in project management :-** Scrum - Various terminologies used in Scrum (Sprint, product backlog, sprint backlog, sprint review, retro perspective), various roles (Roles in Scrum), Best practices of Scrum, Case Study; DevOps - Overview and its Components,	12

Course Assessment Method (CIE: 40 marks, ESE: 60 marks)

Continuous Internal Evaluation Marks (CIE):

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written)	Total
5	15	10	10	40

End Semester Examination Marks (ESE)

In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions

Part A	Part B	Total
• 2 Questions from each module. • Total of 8 Questions, each carrying 3 marks **(8x3 =24 marks)**	• Each question carries 9 marks. • Two questions will be given from each module, out of which 1 question should be answered. • Each question can have a maximum of 3 subdivisions. **(4x9 = 36 marks)**	**60**

Course Outcomes (COs)

At the end of the course students should be able to:

Course Outcomes		Bloom's Knowledge Level (KL)
CO1	Understand how effectively plan, and schedule projects within time and cost targets	**K2**
CO2	Apply project estimation and evaluation techniques to real world problem	**K3**
CO3	Discuss different Agile Project Methodologies	**K2**
CO4	Apply various SCRUM practices in project management and demonstrate the techniques used in DevOps.	**K3**

Note*: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create*

CO-PO Mapping Table:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	✓	✓	✓							✓	✓
CO2	✓	✓	✓							✓	✓
CO3	✓	✓	✓							✓	✓
CO4	✓	✓	✓							✓	✓
CO5	✓	✓	✓							✓	✓

Text Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Succeeding with Agile: Software Development Using Scrum	Mike Cohn	Addison-Wesley	1/e, 2009

eference Books				
Sl. No	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
1	Agile Product Management with Scrum	Roman Pichler	Addison-Wesley	1/e, 2010
2	Agile Project Management with Scrum	Ken Schwaber	Microsoft Press	1/e, 2004

Video Links (NPTEL, SWAYAM…)	
No.	**Link ID**
1	https://archive.nptel.ac.in/noc/courses/noc19/SEM2/noc19-cs70/
2	https://www.youtube.com/watch?v=TPEgII1OilU
3	https://www.youtube.com/watch?v=7Bxdds2siU8