28-12-2025 09:49:15 PM

SOFTWARE ENGINEERING **Course Code : 315323**

Programme Name/s	**: Computer Technology/ Computer Engineering/ Computer Science & Engineering/ Computer Hardware & Maintenance/ Computer Science & Information Technology/ Computer Science**
Programme Code	**: CM/ CO/ CW/ HA/ IH/ SE**
Semester	**: Fifth**
Course Title	**: SOFTWARE ENGINEERING**
Course Code	**: 315323**

I. RATIONALE

Software Engineering is the foundation for professional processes to be followed for designing, developing, testing and maintaining software involving principles, different techniques, and practices for software development. This course enable students to develop requisite abilities to follow systemic and disciplined approach to software development that aims to create high quality, reliable and maintainable software.

II. INDUSTRY / EMPLOYER EXPECTED OUTCOME

Apply software engineering principles to develop software product.

III. COURSE LEVEL LEARNING OUTCOMES (COS)

Students will be able to achieve & demonstrate the following COs on completion of course based learning

- CO1 - Select suitable software development process model
- CO2 - Prepare software requirement specification.
- CO3 - Construct different Software design models
- CO4 - Apply different planning and cost estimation techniques for a software product
- CO5 - Apply project management techniques in software development.
- CO6 - Use quality assurance principles in software development

IV. TEACHING-LEARNING & ASSESSMENT SCHEME

Course Code	Course Title	Abbr	Course Category/s	Learning Scheme					Credits	Assessment Scheme												
				Actual Contact Hrs./Week			SLH	NLH		Paper Duration	Theory				Based on LL & TL				Based on SL		Total Marks
															Practical							
				CL	TL	LL					FA-TH	SA-TH	Total		FA-PR		SA-PR		SLA		
											Max	Max	Max	Min	Max	Min	Max	Min	Max	Min	
315323	SOFTWARE ENGINEERING	STE	DSC	4	-	4	1	9	3	3	30	70	100	40	25	10	25@	10	25	10	175

SOFTWARE ENGINEERING **Course Code : 315323**

Total IKS Hrs for Sem. : 0 Hrs
Abbreviations: CL- ClassRoom Learning , TL- Tutorial Learning, LL-Laboratory Learning, SLH-Self Learning Hours, NLH-Notional Learning Hours, FA - Formative Assessment, SA -Summative assessment, IKS - Indian Knowledge System, SLA - Self Learning Assessment
Legends: @ Internal Assessment, # External Assessment, *# On Line Examination , @$ Internal Online Examination
Note :

1. FA-TH represents average of two class tests of 30 marks each conducted during the semester.
2. If candidate is not securing minimum passing marks in FA-PR of any course then the candidate shall be declared as "Detained" in that semester.
3. If candidate is not securing minimum passing marks in SLA of any course then the candidate shall be declared as fail and will have to repeat and resubmit SLA work.
4. Notional Learning hours for the semester are (CL+LL+TL+SL)hrs.* 10 Weeks
5. 1 credit is equivalent to 30 Notional hrs.
6. * Self learning hours shall not be reflected in the Time Table.
7. * Self learning includes micro project / assignment / other activities.

V. THEORY LEARNING OUTCOMES AND ALIGNED COURSE CONTENT

Sr.No	Theory Learning Outcomes (TLO's)aligned to CO's.	Learning content mapped with Theory Learning Outcomes (TLO's) and CO's.	Suggested Learning Pedagogies.
1	TLO 1.1 Select the attributes that match with standards for the given software application. TLO 1.2 Suggest the relevant software solution for the given problem. TLO 1.3 Select the relevant software process model for the given problem. TLO 1.4 Suggest the relevant activities in Agile Development Process.	**Unit - I Software Development Process** 1.1 Software characteristics, Types of software. 1.2 The Process: Software Engineering: A Layered approach -Process, Methods and Tools 1.3 Software development framework. 1.4 Software Process Model: Waterfall Model 1.5 Incremental Process Model: RAD Model 1.6 Evolutionary Process Models: Prototyping model, Spiral model 1.7 Agile Process Model: Extreme Programming, Adaptive Software Development (ASD), Scrum, Dynamic System Development Method (DSDM), CRYSTAL. Agile Unified Process (AUP)	Presentations Lecture Using Chalk-Board
2	TLO 2.1 Apply principles of software engineering for the given problem. TLO 2.2 Select the relevant requirement engineering steps for the given problem. TLO 2.3 Construct the Requirement Engineering model for the given problem. TLO 2.4 Prepare SRS for the given problem.	**Unit - II Software Requirement Engineering** 2.1 Software Engineering core principles. 2.2 Software Practices: Communication, Planning, Modelling, Construction, Software deployment (Statement and meaning of each principles for each practice). 2.3 Requirement Engineering: Requirement Gathering and Analysis, Types: Functional, Product, organizational, External Requirements, Eliciting Requirements, Developing Use-cases, Building requirement models, Negotiation, Validation. 2.4 Software Requirement Specification: Need, Format, and its Characteristics.	Lecture Using Chalk-Board Presentations Case Study

SOFTWARE ENGINEERING **Course Code : 315323**

Sr.No	Theory Learning Outcomes (TLO's)aligned to CO's.	Learning content mapped with Theory Learning Outcomes (TLO's) and CO's.	Suggested Learning Pedagogies.
3	TLO 3.1 Identify the elements of analysis model for the given software requirements. TLO 3.2 Apply the specified design concepts for software requirements modeling. TLO 3.3 Construct software design using standard design notation. TLO 3.4 State the purpose of software testing. TLO 3.5 Draw Use-Case ,Class Diagrams, Sequence Diagrams for software project. TLO 3.6 Explain basic types of software testing.	**Unit - III Software Modelling and Design** 3.1 Translating Requirement model into design model: Data Modelling. 3.2 Analysis Modelling: Elements of Analysis model. 3.3 Design modelling: Fundamental Design Concepts (Abstraction, Information hiding, Structure, Modularity, Concurrency, Verification, Aesthetics). 3.4 Design notations: Data Flow Diagram (DFD), Structured Flowcharts, Decision Tables. 3.5 UML Modelling :Use-Case ,Class Diagrams, Sequence Diagrams. 3.6 Testing – Meaning and purpose, testing methods - Black-box and White-box, Static and Dynamic testing, Level of testing, V-model.	Lecture Using Chalk-Board Presentations Demonstration
4	TLO 4.1 Explain the management spectrum for software project. TLO 4.2 Estimate size of software product. TLO 4.3 Estimate cost of software product using the empirical method. TLO 4.4 Compute size of the given software using COCOMO model. TLO 4.5 Apply RMMM strategy in Identified risks for any software development problem.	**Unit - IV Software Project Cost Estimation** 4.1 The Management Spectrum – 4P's. 4.2 Metrics for Size Estimation: Line of Code (LoC), Function Points (FP). 4.3 Project Cost Estimation Approaches: Overview of Heuristic, Analytical, and Empirical Estimation. 4.4 COCOMO (Constructive Cost Model), COCOMO II. 4.5 Risk Analysis and Management: Risk identification, Risk assessment, Risk management and monitoring, Risk Refinement and Mitigation, RMMM Plan.	Lecture Using Chalk-Board Presentations Case Study Flipped Classroom
5	TLO 5.1 Apply CPM/PERT scheduling technique for software project. TLO 5.2 Construct timeline chart/ Gantt chart to track progress of the given software project.	**Unit - V Software Project Management** 5.1 Project Scheduling: Basic principles, Work breakdown structure, Activity network 5.2 Project Tracking: Timeline charts, Earned Value Analysis, Gantt Charts. 5.3 Scheduling techniques: Critical Path Method(CPM), Program Evaluation Review Technique(PERT)	Lecture Using Chalk-Board Presentations Demonstration

SOFTWARE ENGINEERING **Course Code : 315323**

Sr.No	Theory Learning Outcomes (TLO's)aligned to CO's.	Learning content mapped with Theory Learning Outcomes (TLO's) and CO's.	Suggested Learning Pedagogies.
6	TLO 6.1 Differentiate between Software Quality Management and Software Quality Assurance. TLO 6.2 Apply the phases of Software Quality Assurance in software development project TLO 6.3 Apply software quality evaluation standards.	**Unit - VI Software Quality Assurance** 6.1 Software Quality Management vs. Software Quality Assurance. 6.2 Phases of Software Quality Assurance: Planning, activities, audit, and review. 6.3 Quality Evaluation standards: Six Sigma, CMMI: Levels, Process areas.	Lecture Using Chalk-Board Presentations Case Study

VI. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL / TUTORIAL EXPERIENCES.

Practical / Tutorial / Laboratory Learning Outcome (LLO)	Sr No	Laboratory Experiment / Practical Titles / Tutorial Titles	Number of hrs.	Relevant COs
LLO 1.1 Use software tool to Write problem statement and identify scope of the project	1	*Write problem statement to define the project title with bounded scope of the project.	2	CO1
LLO 2.1 Use appropriate process model and activities related to project.	2	Select relevant process model to define activities and related tasks set	2	CO1
LLO 3.1 Apply the principles of requirement engineering.	3	*Gather application specific requirements for assimilate into RE (Requirement's engineering) model.	2	CO2
LLO 4.1 Create SRS document for the project.	4	*Prepare broad SRS (software requirement software) for the project.	2	CO2
LLO 5.1 Construct use case diagram for software models.	5	*Write use-cases and draw use-case diagram.	2	CO3
LLO 6.1 Design activity diagram for the project.	6	Draw the activity diagram to represent flow from one activity to another for software development.	2	CO3
LLO 7.1 Draw data flow diagram for the project. LLO 7.2 Create Decision tables and E-R diagram.	7	*Create DFDs (data flow diagram), Decision tables and E-R (entity-relationship) diagram.	2	CO3
LLO 8.1 Represent software project by class diagrams.	8	Draw class diagram and Sequence diagram, State Transition Diagram.	2	CO3
LLO 9.1 Prepare decision table for the project	9	* Create decision table for a project.	2	CO3
LLO 10.1 Design test cases by referring SRS document.	10	*Write test cases to validate requirements from SRS document.	2	CO3
LLO 11.1 Write test cases for Blackbox testing.	11	Prepare test cases for Black Box Testing.	2	CO3
LLO 12.1 Identify risk involved in the project LLO 12.2 Prepare RMMM Plan.	12	* Identify risks involved in the project and prepare RMMM (RMMM-Risk Management, Mitigation and Monitoring) plan.	2	CO4
LLO 13.1 Estimate size of project using function point matrix	13	* Calculate size of the project using Function point metric.	2	CO4

SOFTWARE ENGINEERING **Course Code : 315323**

Practical / Tutorial / Laboratory Learning Outcome (LLO)	Sr No	Laboratory Experiment / Practical Titles / Tutorial Titles	Number of hrs.	Relevant COs
LLO 14.1 Estimate size of project using COCOMO approach.	14	*Calculate cost of the project using COCOMO (Constructive Cost Model) / COCOMO II approach.	2	CO4
LLO 15.1 Prepare project schedule using CPM/PERT technique.	15	*Create software project scheduling charts using CPM (Critical Path Method) / PERT (Project Evaluation and Review Technique)	2	CO5
LLO 16.1 Monitor the progress of project using timeline/Gantt chart	16	Track progress of the project using Timeline charts/ Gantt charts.	2	CO5
LLO 17.1 Prepare SQA plan to ensure various quality processes.	17	Prepare SQA plan that facilitates various attributes of quality of process.	2	CO6
LLO 18.1 Prepare SQA plan to ensure quality product.	18	*Prepare SQA plan that facilitates various attributes of quality of product.	2	CO6

Note : Out of above suggestive LLOs -

- '*' Marked Practicals (LLOs) Are mandatory.
- Minimum 80% of above list of lab experiment are to be performed.
- Judicial mix of LLOs are to be performed to achieve desired outcomes.

VII. SUGGESTED MICRO PROJECT / ASSIGNMENT/ ACTIVITIES FOR SPECIFIC LEARNING / SKILLS DEVELOPMENT (SELF LEARNING)

Micro project

- Apply the principles of software engineering for Portfolio website for showcasing Skills and Work , Searchability and Online Presence , Demonstrating Growth and Progress , Career Advancement and Networking and Prepare complete technical document.
- Apply the principles of software engineering for Chatbot Application to create an intelligent chatbot to enhance customer support processes, providing efficient and personalized assistance and Develop technical document.
- Apply the principles of software engineering for Online Chat Application Project that enables users to exchange messages and communicate with each other in real-time. It allows individuals or groups to have conversations, share information, and collaborate instantly over the Internet. Online Chat Application is designed to provide a responsive and interactive experience, where messages are delivered and displayed immediately as they are sent and Prepare complete technical document.

Assignment

- Estimate Cost of software using any tool and risk involved in the library Management System
- Create DFDs, Activity Diagram, ER-Diagrams for Student Management System.
- Visit any medical shop and collect requirements from shop keeper and create SRS document

Other

- Complete the course basic of software engineering on Infosys Springboard or any MOOCs platforms.

SOFTWARE ENGINEERING **Course Code : 315323**

Note :

- Above is just a suggestive list of microprojects and assignments; faculty must prepare their own bank of microprojects, assignments, and activities in a similar way.
- The faculty must allocate judicial mix of tasks, considering the weaknesses and / strengths of the student in acquiring the desired skills.
- If a microproject is assigned, it is expected to be completed as a group activity.
- SLA marks shall be awarded as per the continuous assessment record.
- For courses with no SLA component the list of suggestive microprojects / assignments/ activities are optional, faculty may encourage students to perform these tasks for enhanced learning experiences.
- If the course does not have associated SLA component, above suggestive listings is applicable to Tutorials and maybe considered for FA-PR evaluations.

VIII. LABORATORY EQUIPMENT / INSTRUMENTS / TOOLS / SOFTWARE REQUIRED

Sr.No	Equipment Name with Broad Specifications	Relevant LLO Number
1	Software Project Management Tools: Open source Software such as Jira.	1,2,3,4,10,11,17,18
2	Software Design tools : Projectriskmanager	12
3	Software Design tools :Open Project, Ganttproject 3.3	15,16
4	Software Design tools: Free Use Case Diagram Creator	5
5	Software Design tools: Draw.io, Decision Table Maker, Tiny tools	6,7,8,9,13,14
6	Hardware: Personal computer, processor i3 and above, RAM minimum 4 GB	All
7	Operating system: Windows 10 and above	All

IX. SUGGESTED WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE (Specification Table)

Sr.No	Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	I	Software Development Process	CO1	7	2	6	4	12
2	II	Software Requirement Engineering	CO2	9	4	6	4	14
3	III	Software Modelling and Design	CO3	9	4	4	8	16
4	IV	Software Project Cost Estimation	CO4	8	2	2	8	12
5	V	Software Project Management	CO5	4	2	2	4	8
6	VI	Software Quality Assurance	CO6	3	2	2	4	8
Grand Total				**40**	**16**	**22**	**32**	**70**

X. ASSESSMENT METHODOLOGIES/TOOLS

Formative assessment (Assessment for Learning)

- Each practical will be assesed considering 60% weightage to process, 40% weighatge to product.
- Continuous assessment based on process and product related performance indicators.
- A Continuous assessment -based term work.

Summative Assessment (Assessment of Learning)

- End Semester Examination, Lab Performance, Viva voce

SOFTWARE ENGINEERING **Course Code : 315323**

XI. SUGGESTED COS - POS MATRIX FORM

Course Outcomes (COs)	Programme Outcomes (POs)							Programme Specific Outcomes* (PSOs)		
	PO-1 Basic and Discipline Specific Knowledge	PO-2 Problem Analysis	PO-3 Design/ Development of Solutions	PO-4 Engineering Tools	PO-5 Engineering Practices for Society, Sustainability and Environment	PO-6 Project Management	PO-7 Life Long Learning	PSO-1	PSO-2	PSO-3
CO1	1	2	2	2	1	-	1			
CO2	2	3	3	2	1	-	1			
CO3	2	2	3	3	-	-	1			
CO4	2	2	2	3	-	2	2			
CO5	2	3	2	3	-	3	2			
CO6	-	2	2	3	1	2	2			

Legends :- High:03, Medium:02,Low:01, No Mapping: -
*PSOs are to be formulated at institute level

XII. SUGGESTED LEARNING MATERIALS / BOOKS

Sr.No	Author	Title	Publisher with ISBN Number
1	Roger S. Pressman & Bruce R. Maxim	Software Engineering: A practitioner's approach	McGraw Hill Higher Education, New Delhi, (Ninth Edition) ISBN 93-5532-504-5
2	Richard Fairly	Software Engineering Concepts	McGraw Hill Education New Delhi -2001, ISBN-13: 9780074631218
3	Deepak Jain	Software Engineering: Principles and practices	Oxford University Press, New Delhi ISBN 9780195694840
4	Srinivasan Desikan, Gopalaswamy Ramesh	Software Testing: Principles and Practices	PEARSON Publisher: Pearson India 2007, ISBN: 978-81-7758-121-8
5	Ron Patton	Software Testing	Sams Publishing; 2nd edition, 2005 ISBN: 0672327988

XIII . LEARNING WEBSITES & PORTALS

Sr.No	Link / Portal	Description
1	https://www.geeksforgeeks.org/software-engineering-introduction-to-software-engineering/	Software engineering tutorials from Geeksforgeeks
2	https://www.tutorialspoint.com/software_engineering/index.htm	Software Engineering Tutorials
3	https://www.sei.cmu.edu/	Software Engineering Institute
4	https://www.youtube.com/watch?v=WjwEh15M5Rw	Agile Methodology
5	https://app.diagrams.net/	Software Design -DFDs, Class Diagrams, Use Case Diagrams

28-12-2025 09:49:15 PM

SOFTWARE ENGINEERING **Course Code : 315323**

Sr.No	Link / Portal	Description

Note :

- Teachers are requested to check the creative common license status/financial implications of the suggested online educational resources before use by the students

MSBTE Approval Dt. 24/02/2025 **Semester - 5, K Scheme**