KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
Deemed to be University
BHUBANESWAR-751024

School of Computer Engineering
Autumn Semester 2024-25

Course Handout

1. **Course code** : CS 31001

2. **Course Title** : Software Engineering

3. **LTP Structure** :

L	T	P	Total	Credit
3	1	0	4	4

4. **Course Faculty** : **Sarita Mishra**

5. **Contact Address : F103 , Block C ,Campus 14**

6. **Course offered to the School :** Computer Engineering

7. **Course Objective:**

- To understand the Software Engineering Practice.
- To understand the Software Engineering Process Models.
- To understand the Design Engineering, Web applications.
- To gain knowledge of the software testing.
- To understand Software Project Management.

8. **Course Outcome:**

Upon completion of the course, the students will be able to:

CO #	Detail
CO1	Identify appropriate software process models for developing real life projects
CO2	Assess each module given the overall Software engineering practice
CO3	Enhance the software project management skills
CO4	Comprehend the systematic methodologies involved in SE
CO5	Work ethically in a team as well as independently on software projects and adapt to the ever changing dynamic real world situation
CO6	Design and develop a software product in accordance with SE principles

9. Course Contents

The course focuses on basic and essential topics in Software Engineering.

Unit #	Unit	Detailed Area
1	Introduction: Assessment:	Role of Software Engineer, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Quality Attributes. How Software Engineering Changes? Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Choosing a social relevant problem, Summary Team Report.
2	Requirement Engineering Process: Assessment:	Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Designing the architecture. Impact of Requirement Engineering in their problem, Decision Tables, SRS Document, IEEE Standards for SRS, Architectural design, component level design, user interface design, WebApp Design, Submission of SRS Document for Team Project.
3	Quality concepts, Review techniques, Software Quality Assurance (SQA): Assessment:	Verification and Validation, SQA Plans, Software Quality Frameworks. Framing SQA Plan, ISO 9000 Models, SEI-CMM Model and their relevance to project Management, Other emerging models like People CMM.
4	Testing: Assessment:	Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing, Software Testing Strategies, Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Testing conventional applications, object oriented applications, Web applications, Formal modeling and verification, Software configuration management, Product metrics. Team Analysis in Metrics Calculation.
5	Project Management Assessment:	Project Management Concepts, Process and Project Metrics, Estimation for Software projects, Project Scheduling, Risk Management, Maintenance and Re-engineering. Preparation of Risk mitigation plan.

10. Text Book:

1. R. S. Pressman, “Software Engineering: A Practitioners Approach”, Eighth Edition, McGraw Hill, 2010.

2. Rajib Mall, “Fundamentals of Software Engineering”, Fifth Edition, PHI Publication, 2009.

3. Pankaj Jalote, “Software Project Management in Practice”, Pearson Education, New Delhi, 2002.

11. Pre-requisites:

- Nil

12. Lesson Plan:

Class Days: *Pre-Mid Sem: 15th July to 6th Sep 2024*
Post Mid Sem: 23rd Sep to 7h Nov 2024

Module, Hrs	*Topics/Coverage*	*Lectures*
1. Introduction: (10 Hrs.)	Software and its characteristics, application. Software engineering. Role of a Software Engineer, software components, software crisis, software process, Emergence of Software Engineering, Similarity and differences from conventional engineering process, Quality Attributes	3
	Software process Models: Classical waterfall model, iterative waterfall model, prototyping model, evolutionary model, spiral model, RAD model. Essential Idea behind Agile Models, Agile models: Extreme programming and Scrum, Agile versus Other Models	6
Tutorial I	*Discussion of module 1*	1
2. Requirement and Design phases (11 Hrs.)		
Requirement Engineering (3 Hrs.)	Requirements gathering and Analysis, Requirements elicitation for software analysis, Analysis principles, Specifications (SRS document), IEEE 830 guidelines, Decision tables and trees, software change management and software configuration management *Tutorial III Discussion of module 3.*	3
Structural Analysis & Design (7 Hrs.)	Information Modeling, ERD, Overview of the Design Process, How to Characterize a Good Software Design, Approaches to Software Design, Software Design: cohesion & coupling, neat arrangement, Function–Oriented software design: Structural Analysis, Structural Design (DFD and Structured Chart)	3
	Object Oriented Analysis & Design (Basic Object-Orientation Concepts, Use case diagram, Class diagram, State chart diagram), Command language,	4

	menu and iconic interfaces, Architectural Design, Component level Design, User Interface Design, WebApp Design, UML Design	
Tutorial II	*Discussion of module 2*	1
3. Software Project Management (10 Hrs.)	Software Project Management Concept, project planning, project processes, Metrics for project size estimation	2
	Project estimation techniques, Empirical estimation techniques, COCOMO models	4
	Scheduling (Work Breakdown Structure, Activity Networks, Critical Path Method (CPM), PERT Charts, Gantt Charts), Organization & team structure, Risk Management	3
Tutorial III	*Discussion of module 3.*	1
	MID SEMESTER EXAMINATION (17.9.2024-21.9.2024)	
4. Testing Strategies (9 Hrs)	Code Review, Verification and Validation u	2
	Testing Objectives, Unit Testing, Integration Testing, System Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing, Software Testing Strategies, Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Automated Testing	3
	Testing conventional applications, object oriented applications, Web applications, Formal modeling	3
Tutorial IV	*Discussion of module 4*	1
5. Software Reliability and Software Maintenance (6Hrs)	SQA Plan, ISO 9000 Models, SEI-CMM Model and their relevance to project Management, Software reliability, People CMM	2
	Characteristics of software maintenance, software reverse engineering, software re-engineering. Software Reuse	*3*
Emerging Topics (2Hrs)	Software as a Service (SaaS), AI-based Software Engineering	*2*
Tutorial V	*Discussion of module 5*	1
	Total	48

13. **Assessment Components:**

Sr #	Assessment Component	Time	Weightage/ Marks	Course Lecture No.		Mode
				From	To	
1	Mid-Semester Examination	1.5 Hrs	20	1	24	Closed Book
2	Activity based Teaching and Learning	Through out semester	30	1	48	Open Book, Closed Book
3	End-Semester Examination	2 Hrs 30 Mins	50	1	48	Closed Book

14. **Assessment plan for activity based learning:**

Considering the guidelines circulated and after discussing with the faculty members, following activity based teaching and learning is proposed and Component wise distributions of the activities are listed bel`ow.

Activities may include one/multiple Individual/Group Assignment(s), Class Test(s), Quizzes) etc.

Sl.No.	Activity	Date of Submission	Marks
1	Activity 1	10-08-2024	5
2	Activity 2	24-08-2024	5
3	Activity 3	06-09-2024	5
4	Activity 4	10-10-2024	5
5	Activity 5	20-10-2024	5
6	Activity 6	08-11-2024	5