MANIPAL UNIVERSITY JAIPUR

School of Computing and Information Technology

Department of Computer Science & Engineering
Course Hand-out

Software Engineering| CS 3201 | 4 Credits | 3 1 0 4

Session: Jan 2024- May 2024 | Faculty: Dr Manmohan Sharma, Dr Ankit Srivastava, Dr Praneet Saurabh, Ms. Bali Devi, Ms Neha, Dr Susheela Vishnoi, Dr Prashant Vats, Dr Gireesh Kumar, Dr Sakshi Shringi, Dr Roheet Bhatnagar | VI Sem

A. **Introduction:** The aim of this course is to enable a clear understanding and knowledge of the foundations, techniques, tools and processes in the area of software engineering and its practices followed in software industry. The course will prepare students to apply engineering methods and processes on software projects.

B. **Course Outcomes:** At the end of the course, students will be able to

[CS 3201.1] Describe basic concept related to software engineering Methods, tool, Process Model and use them in software project.

[CS 3201.2] Apply different Estimation Techniques based on project Metrics, Measures and indicators.

[CS 3201.3] Design SRS, detailed and architecture design based on user requirements.

[CS 3201.4] Select appropriate testing methods and apply testing techniques for software projects.

[CS 3201.5] Show entrepreneurship skills by understanding quality of software projects based on software quality assurance techniques.

C. **PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES**

[PO.1]. **Engineering knowledge**: : Apply the knowledge of basic science and fundamental computing in solving complex engineering problems

[PO.2]. **Problem analysis**: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

[PO.3]. **Design/development of Computing solutions**: Design solutions for complex IT engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the Information oriented public health and safety, and the cultural, societal, and environmental considerations

[PO.4]. **Conduct investigations of complex problems**: Use IT domain research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

[PO.5]. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

[PO.6]. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice

[PO.7]. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

[PO.8]. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practices

[PO.9]. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse IT teams, and in multidisciplinary settings.

[PO.10]. **Communication**: Communicate effectively on complex computing engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions

[PO.11]. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments

[PO.12]. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

[PSO.1] Will be able to design, develop and implement efficient software for a given real life problem.

[PSO.2] Will be able to apply knowledge of AI, Machine Learning and Data Mining in analysing big data for extracting useful information from it and for performing predictive analysis.

[PSO.3] Will be able to design, manage and secure wired/ wireless computer networks for transfer and sharing of information.

D. Assessment Plan:

Criteria	Description	Maximum Marks
Internal Assessment (Summative)	Midterm Exam	30
	Quizzes (Two)	10
	Assignments/Presentations	10
	Online certificate course	10
End Term Exam (Summative)	End Term Exam (Closed Book)	40
	Total	100
Attendance (Formative)	A minimum of 75% Attendance is required to be maintained by a student to be qualified for taking up the End Semester examination. The allowance of 25% includes all types of leaves including medical leaves.	
Homework/ Home Assignment/ Activity Assignment (Formative)	There are situations where a student may have to work in home, especially before a flipped classroom. Although these works are not graded with marks. However, a student is expected to participate and perform these assignments with full zeal since the activity/ flipped classroom participation by a student will be assessed and marks will be awarded.	

E. SYLLABUS

Introduction: The Evolving Role of Software, The changing nature of software, Legacy software, Software Myths, Software Engineering: A Layered Technology, A Process Framework, umbrella activities, measures, metrics, indicators, The Capability Maturity Model Integration (CMMI), Specialized Process Models, The Unified Process; Agile development: Agile Process Models Software Engineering Practice, Communication Practice, Planning Practices, Modeling Practices, Construction Practice, Deployment Computer–Based Systems, The System Engineering Hierarchy, Business Process Engineering: An Overview; Product Engineering: An Overview, Data Modeling Concepts, Object Oriented Analysis, Flow-Oriented Modeling, Taxonomy of Quality Attributes, Perspectives of Quality, Quality System, Software Quality Assurance, Capability Maturity Model Observation on Estimation, The Project Planning Process, Software Scope and Feasibility, Human Resources, Empirical Estimation Model

F. TEXT BOOKS

T1. Pressman R, Software Engg. Practioner Approach (MGH), 2006
T2. Jalote Pankaj, An integrated approach to Software Engineering (Narosa)
T3. Rajib Mall, Introduction to Software Engineering, TMH

G. REFERENCE BOOKS

R1. Sommerville, Software Engineering 10e, Pearson

Lecture Plan:

S.NO	Topics as per the university and syllabus	Topics	Session Outcome	Mode of Delivery	Corresponding CO	Mode of Assessing the Outcome
		Introduction to software engineering	To explain teachers expectations and understand student expectations	**Lecture**	NA	NA
1.	Introduction	Software, Emergence of software engineering, evolving role of software, The changing nature of software.	Define role and nature of software	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
2.		Legacy software, Software Myths, Software crisis.	Analyze types of software and myths associated with them.	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
3.		Introduction to software engineering: Layered Approach, Generic approach, Process Framework.	Recall software engineering layered approach and process framework	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
4.		The process, software products, Software Characteristics, Applications.	Recall software products, Software Characteristics, Applications.	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
5.	Models		**Appraise s**oftware development life cycle, Assessment Model-CMMI	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
6.		Process Models- Conventional Process Model (Traditional Waterfall Model, prototype, RAD)	**Choose between** process Models	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
7.		Evolutionary Process Model (Incremental Model, Spiral models, Component based process model, unified process model), Comparison of various models	**Compare** Evolutionary Process Model	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term

8.	Agile development	An Agile view of process, Human Factors	Define agile view of process	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
9.		Agile Process Models (Extreme programming, Adaptive Software Development, Dynamic System Development method)	Apprise Agile process models and its adaptive software development using Extreme programming	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
10.		Agile process Models (Scrum, Crystal, Feature Driven Development, Agile Modeling)	Assess agile process Models (Scrum, Crystal, Feature Driven Development, Agile Modeling)	**Lecture**	[3201.1]	Mid Term, Quiz, Assignment & End Term
		Case study Based on Agile Model	Build systems based on Agile Model			
11.		Software development life cycle, Assessment Model-CMMI		**Lecture/Lab**	[3201.1]	Mid Term, Quiz, Assignment & End Term
		<mark>Test Driven Development</mark>				
12.	Requirements Analysis and specification	Requirement Engineering, Type of requirements, Software requirements specifications	Illustrate methods of Requirement Engineering, Type of requirements, Software requirements specifications	**Lecture/Lab**	[3201.3]	Mid Term, Quiz, Assignment & End Term
13.		Software Engineering Practice (elicitation, analysis, documentation, validation, Management)	Explain Software Engineering Practice (elicitation, analysis, documentation, validation, Management)	**Lecture/Lab**	[3201.3]	Mid Term, Quiz, Assignment & End Term
14.	Estimation Techniques	Size estimation- LOC Estimation	Evaluate the size of project and estimation- size of project using LOC Estimation	**Lecture/Lab**	[3201.2]	Mid Term, Quiz, Assignment & End Term

15.		Function Count Method	Apply Function Count method	**Lecture/Lab** [3201.2]		Mid Term, Quiz, Assignment & End Term
16.		Cost Estimation	Apply Cost Estimation techniques in software projects	**Lecture/Lab** [3201.2]		Mid Term, Quiz, Assignment & End Term
17.		Halstead Size Estimation	Evaluate software project using Halstead Size Estimation	**Lecture/Lab** [3201.2]		Mid Term, Quiz, Assignment & End Term
18.		Effort Estimation-COCOMO Model	Effort Estimation- using COCOMO Model	**Lecture/Lab** [3201.1]		Mid Term, Quiz, Assignment & End Term
21.		Risk Analysis and Risk Estimation	Determine Risk associated with software projects	**Lecture/Lab** [3201.1]		Mid Term, Quiz, Assignment & End Term
22.		Analysis Modeling: Data Modeling, Functional modeling and information flow, Data flow diagrams	Apply Analysis Modeling: Data Modeling, Functional modeling and information flow, Data flow diagrams	**Lecture/Lab** [3201.3]		Mid Term, Quiz, Assignment & End Term
23.	Software designing	Behavioral Modeling; The mechanics of structured analysis: Creating entity/ relationship diagram	Apply Behavioral Modeling; The mechanics of structured analysis: Creating entity/ relationship diagram	**Lecture/Lab** [3201.3]		Mid Term, Quiz, Assignment & End Term
24.		Data flow model, control flow model, the control and process specification; The data dictionary	Design Data flow model, control flow model, the control and process specification; The data dictionary	**Lecture/Lab** [3201.3]		Mid Term, Quiz, Assignment & End Term
25.		Unified modeling language – an introduction, Discussion on Class diagrams, Object diagrams, Sequence diagrams, use case diagram.	Apply Unified modeling language and develop UML Diagrams	**Lecture/Lab** [3201.3]		Mid Term, Quiz, Assignment & End Term
26.		Case Study Based on Software Design	Develop Case Study Based on Software Design	**Flipped Classroom** [3201.3]		Mid Term, Quiz, Assignment & End Term

27.	System Design	Design concepts and principles, the design process	Design concepts and principles and show design process	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
28.		Design and software quality, design principles	Relate design and software quality, design principles	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
		Design concepts: Abstraction, refinement, modularity	Recall concept of Abstraction, refinement and modularity	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
29.		software architecture, control hierarchy, structural partitioning, data structure, software procedure, information hiding	Develop software architecture, control hierarchy, structural partitioning, data structure, software procedure, information hiding	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
30.		Effective modular design: Functional independence, Cohesion, Coupling	Appraise effective modular design: Functional independence, Cohesion, Coupling	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
31.	Architecture Design	Software architecture: Data Design: Data modeling, data structures, databases and the data warehouse.	Develop Software architecture: Data Design: Data modeling, data structures, databases and the data warehouse.	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
32.		Analyzing alternative Architectural Designs ,architectural complexity;	Analyzing alternative Architectural Designs ,architectural complexity;	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
33.		Mapping requirements into a software architecture;	Translate requirements into a software architecture;	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
		Transform flow, Transaction flow	Recall Transform flow, Transaction flow	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term

34.		Transform mapping: Refining the architectural design.	Apply transform mapping: Refining the architectural design.	**Lecture/Lab** [3201.3]	Mid Term, Quiz, Assignment & End Term
35.		Software Testing Techniques, software testing fundamentals (Software re-engineering, reverse engineering, restructuring, forward engineering).	Analyze Software Testing Techniques, software testing fundamentals	**Lecture/Lab** [3201.4]	Mid Term, Quiz, Assignment & End Term
36.		Functional Testing (black Box): Boundary value analysis	Experiment with functional Testing(black Box): Boundary value analysis	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
37.		Equivalence class Testing,	Apply Equivalence class Testing,	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
38.		Decision table based testing	Make use of Decision table based testing	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
39.		Structural Testing (White box) : Path testing	Experiment with Structural Testing (White box) : Path testing	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
40.	Software testing	Cyclomatic Complexity	Evaluate Cyclomatic Complexity	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
41.		Graph Matrices	Examine Graph Matrices for testing	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
42.		Data Flow Testing	Make use of Data Flow Testing	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
43.		Unit testing, Integration Testing, System Testing, Validation Testing, Testing Tools (Static Testing tools, Dynamic Testing tools, characteristics of Modern tools).	Develop test cases using different methods	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
44.		Debugging: Debugging techniques, Debugging Approaches, Debugging Tools.	Choose Debugging tools and techniques	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term
45.		Software re-engineering, reverse engineering, restructuring, forward engineering.	Apply concepts of Software re-engineering, reverse engineering, restructuring, forward engineering.	**Lecture/Lab** [3201.4]	Quiz, Assignment & End Term

46.		Case Study based on Software Testing	Take part in Case Study based on Software Testing	**Flipped Classroom**	[3201.4]	Quiz, Assignment & End Term
47.		Quality concepts, Software quality assurance , SQA activities, Software reviews	Apply Quality concepts, Software quality assurance , SQA activities, Software reviews	**Lecture**	[3201.5]	Quiz, Assignment & End Term
48.		Formal technical reviews: The review meeting, review reporting and record keeping, review guidelines, Formal approaches to SQA.	Recall Formal technical reviews and Formal approaches to SQA.	**Lecture**	[3201.5]	Quiz, Assignment & End Term
49.	Software Quality Assurance, Software Maintenance	Statistical software quality assurance; software reliability: Measures of reliability and availability, The ISO 9000 Quality standards: The ISO approach to quality assurance systems, The ISO 9001 standard	Explain Statistical software quality assurance and ISO 9000 Quality standards	**Lecture**	[3201.5]	Quiz, Assignment & End Term
50.		Characteristics of software maintenance, Software maintenance processes model	Identify characteristics of software maintenance, Software maintenance processes model	**Lecture**	[3201.5]	Quiz, Assignment & End Term
51.		Case Study based on Software Quality Assurance and Software Maintenance	Take part in Case Study based on Software Quality Assurance and Software Maintenance	**Flipped Classroom**	[3201.5]	Quiz, Assignment & End Term
	Total	51 lectures				

H. Course Articulation Matrix: (Mapping of COs with POs and PSOs)

CO	STATEMENT	CORRELATION WITH PROGRAM OUTCOMES												CORRELATION WITH PROGRAM SPECIFIC OUTCOMES		
		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
3201.1	[3201.1]. Describe basic concept related to software engineering Methods, tools, Process Model and use them in software project.	1								2	1	1	3	3		
3201.2	[3201.2]. Apply different Estimation Techniques based on project Metrics, Measures and indicators										1	3		3		
3201.3	[3201.3]. Design software and architecture at different detail levels based on requirements for software projects			3	3	3				3	1	3	3	3		
3201.4	[3201.4]. Demonstrate the Testing methods and their procedures to implement in any project										1	1		3		
3201.5	[3201.5]. Assess quality of software projects based on software quality assurance techniques.										1	2		3		

1- Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

I. Course Outcome Attainment Level Matrix:

CO	STATEMENT	ATTAINMENT OF PROGRAM OUTCOMES THRESHOLD VALUE: 40%												ATTAINMENT OF PROGRAM SPECIFIC OUTCOMES		
		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
3201.1	Describe basic concept related to software engineering Methods, tool, Process Model and use them in software project.															
3201.2	Apply different Estimation Techniques based on project Metrics, Measures and indicators															
3201.3	Design software and architecture at different detail levels based on															

	requirements for software projects																	
3201.4	Demonstrate the Testing methods and their procedures to implement in any project																	
3201.5	Improve entrepreneurship skills by Assess quality of software projects based on software quality assurance techniques.																	

0-No Attainment; 1- Low Attainment; 2- Moderate Attainment; 3- Substantial Attainment