

Software Engineering					
Credit Hours:	3	Course Code:	INFT-2116	Prerequisites:	None
Course Learning Outcomes (CLOs):					
After the completion of this course the student should be able to:				Domain	BT Level*
1. Demonstrate Basic project management skills for solving real world problems.				C	2
2. Understand and be able to apply the principles of software engineering practice and process				C	3
3. Build Software using modern tools to solve real world problems.				C	3
4. Analyze Real world problems by using software engineering constructs				C	4
* BT= Bloom’s Taxonomy, C=Cognitive Domain, P=Psychomotor Domain, A=Affective Domain					
#	SDG	Relevant CLOs			
1	SDG 9: Industry, Innovation and Infrastructure	CLO 1: Demonstrate Basic project management skills for solving real world problems. CLO 3: Build Software using modern tools to solve real world problems.			
2	SDG 12: Responsible Consumption and Production	CLO 2: Understand and be able to apply the principles of software engineering practice and process			
3	SDG 16: Peace, Justice and Strong Institutions	CLO 1: Demonstrate Basic project management skills for solving real world problems.			
Course Contents:					
Nature of Software, Overview of Software Engineering, Professional software development, Software engineering practice, Software process structure, Software process models, Agile software development, Agile process models, Agile development techniques, Requirements engineering process, Functional and non-functional requirements, Context models, Interaction models, Structural models, Behavioral models, Model driven engineering, Architectural design, Design and implementation, UML diagrams, Design patterns, Software testing and quality assurance, Software evolution, Project management and project planning, Configuration management, Software process improvement, Software metrics and measurement, Software quality models (e.g., CMMI, ISO standards), Risk management and estimation techniques, Software maintenance and re-engineering, DevOps practices and Continuous Integration/Continuous Deployment (CI/CD), Usability and Human-Computer Interaction (HCI) fundamentals, Formal methods and model checking basics, Ethical issues in software engineering, Emerging practices: microservices architecture, cloud-native development.					
Teaching Methodology:					
Lectures, Written Assignments, Semester Project, Presentations.					
Course Assessment:					
Lectures, Written Assignments, Semester Project, Presentations.					
Reference Material:					
1. Software Engineering by Ian Sommerville, 10 th Edition, Year 2017, ISBN-13: 978-9332582699 2. Fundamentals of Software Architecture: A Modern Engineering Approach by Mark Richards and Neal Ford, 2nd Edition, Year 2025, ISBN-13: 978-1098175511 3. Software Engineering, A Practitioner’s Approach by Pressman R. S. and Maxim B. R., 9 th Edition, Year 2015, ISBN-13: 978-1260548006 4. Software Engineering at Google: Lessons Learned from Programming Over Time by Titus Winters, Tom Manshreck, Hyrum Wright, 1 st Edition, ISBN-13: 978492082798					