

Branch Name:	IMCA
Program Code:	CS301
Course Name:	SOFTWARE ENGINEERING
Course Code:	1CS3010504T
Pre-requisite Course:	Systems & Object-Oriented Design Methodologies and Fundamentals of Structured Programming.

Course Objective (CO):

1. The educational objectives of the Software Engineering are to prepare students for employed in industry, government and private sector
2. Students are expected to demonstrate professional advancement through significant technical achievements and to expand leadership responsibility.
3. This course is intended to demonstrate the ability to work effectively as a team member and team leader in an ever-changing professional environment.
4. Get better Understand, Analyze and Model User's Requirements, Select Appropriate Process Model Apply it to All Stages of Software Development Life Cycle (SDLC)
5. Select and Apply Appropriate Metrics to Estimate Software Size, Effort, and Cost

Teaching and Examination Scheme:

Teaching Scheme (Hours per week)				Evaluation Scheme (Marks)				
Lecture	Tutorial	Practical	Credit	Theory		Practical		Total
				University Assessment	Continuous Assessment	University Assessment	Continuous Assessment	
4	-	-	4	60	40	-	-	100

Subject Contents

Sr. No	Topic	Total Hours	Weight (%)
1	Introduction to Software Engineering : Introduction to Software Engineering :Software , Definition of Software Engineering ,Software Development Approach , Evolving ,Role of Software , Software Characteristics , Different between Software Engineering and Computer Science , Different between Software Engineering and System Engineering, Software Application , Evolution of Software Engineering, Software Process, Layered Technology ,Principal of Software Engineering .	12	25
2	Process Models – Waterfall, Incremental, Evolutionary Process Model– Prototype and Spiral Model. Agile Process; Extreme Programming (XP), Brief Overview of Other Agile Process Models: Adaptive Software Development, Scrum Principles of Software Engineering: Introduction, Core Principles of Process and Practice, Principles Guiding Each Framework Activity.	12	25
3	Software Requirement Specification : Requirement Engineering Processes, Types of Requirement, SRS (System Requirement Specification), Software Eng. Benefits , Role of Management in Software Development ,Role of Metrics and Measurement	12	25

	System Design : Software Design Strategy Design Patterns, Becoming Master Designer, Implementing a Design, Evaluating a Design, Problem Partitioning, Abstraction, Strategy of Design, Function, Oriented vs. Object Oriented Approaches		
4	Coding : Programming Practices, Top Down Approach & Bottom Up Approach, Structure Programming, Information hiding , Programming Style. Testing : Testing Fundamentals, Tops Down And Bottom Up Approaches Test Cases and Test criteria, Psychology of Testing, Regression testing, Functional Testing, Structure Testing, Equivalence Class Partitioning, Boundary Value Analysis, Cause – Effect Graphing, Types of Testing under While/Glass Box Testing Strategy	12	25

Text Books:

1. Roger S. Pressman, “Software Engineering – A Practitioner’s Approach”, 7th Edition, McGraw Hill Publications
2. Sommerville, “Software Engineering”, 8th Edition, Pearson Education

References Books:

1. Waman S. Jawadekar, “Software Engineering – Principles and Practices”, TMGH Publication
2. Systems Analysis and Design by Kendall & Kendall, PHI Publication, 7th Edition
3. The Unified Modeling Language - User Guide by Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education Publication, 2009 Reprint
4. Fundamentals of Software Engineering by Rajib Mall, Prentice Hall India.
5. An integrated approach to Software Engineering by Pankaj Jalote, Springer
6. Practical Approach of Software Engineering by Munesh Trivedi, N.N.Jani, S.S.Sarangdevot, Avinash Dwivedi

E-Resources / Web Links:

1. https://www.tutorialspoint.com/software_engineering/index.htm
2. <https://www.slideshare.net/sivaayyakutti5/software-engineering-lecture-notes>
3. <https://www.slideshare.net/blpgirl/introduction-to-software-engineering-1508707>
4. <http://www.tutorialride.com/software-engineering/software-engineering-tutorial.htm>

Course Learning Outcomes (CLO): On completion of this course, the students will be able to:

CLO	Description	Bloom’s Taxonomy Level
CLO1	To apply knowledge of science and engineering in Software industry.	3 Applying,
CLO2	To design and conduct experiments, as well as to analyze and interpret data.	3 Applying,
CLO3	Understand common lifecycle processes including waterfall (linear), incremental approaches (such as Unified process), and agile approaches.	2 Understanding, 1 Remembering
CLO4	Learn Key modeling concepts that apply to both the traditional structured approach and the newer object- oriented approach.	4 Analyzing
CLO5	Identify and analyze the system requirements using various system analysis techniques.	2 Understanding,
CLO6	Design information system using structured and object-oriented techniques	6 Creating
CLO7	Recognize current and future trends of system analysis and design	6 Creating 2 Understanding,

Mapping of CLOs with Pos & PSOs

Course Learning Outcomes	Program Out comes (POs)												Program Specific Outcomes (PSOs)	
	PO1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO11	PO12	PSO1	PSO2
CLO1	H	H	M	L		L			M			M	M	L
CLO2		L			M		L	M			M		M	M
CLO3	M		M		H	M			M			M	M	M
CLO4	L	M		L		M	L		M	L		M	L	H
CLO5		M	M		L		M		H		L		L	H
CLO6	M		M		M	L	M	M		M	L		M	M
CLO7	M	L		L	M	M	H		M		M	L	H	M

H: High, M: Medium, L: Low