United College of Engineering and Research
Department of Computer Science and Engineering
Lecture Plan

Name of Course	Software Engineering
AKTU Course Code	KCS-601
Branch	IT
Semester	6
Section	H
Total Number of Students	
Name of Faculty	Abhishek Kesharwani
Number of Lecture Proposed	50

S.No	Unit No	Topic	CO	No of Lectures Required	No of Student present	Actual Date of Completion	TLM	Signature of Incharge
1	1	Introduction to Software Engineering	1	1				
2		Software Components		1				
3		Software Characteristics		1				
4		Software Crisis		1				
5		Software Engineering Processes		1				
6		Similarity and Differences from Conventional Engineering Processes		1				
7		Software Quality Attributes		1				
8		Software Development Life Cycle (SDLC) Models		1				
9		Water Fall Model		1				
10		Prototype Model		1				
11		Spiral Model		1				
12		Evolutionary Development Models		1				
13		Iterative Enhancement Models		1				
No of Lectures Required to complete Unit 1				13	No of Lectures Taken:			
14	2	Software Requirement Specifications (SRS		1				
15		Requirement Engineering Process, Elicitation, Analysis, Documentation, Review and Management of User Needs,		1				

		Feasibility Study							
16		**Data Flow Diagrams**		**3**					
17		**Entity Relationship Diagrams**		**2**					
18		**Decision Tables**		**1**					
19		**SRS Document**		**1**					
20		**IEEE Standards for SRS**		**1**					
21		**Software Quality Assurance (SQA)**		**1**					
22		**Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model**		**1**					
No of Lectures Required to complete Unit 2				**12**	**No of Lectures Taken:**				
23	3	**Software Design: Basic Concept of Software Design, Architectural Design, Low Level Design:**		**1**					
24		**Modularization, Design Structure Charts, Pseudo Codes**		**1**					
25		**Flow Charts, Coupling and Cohesion Measures,**		**1**					
26		**Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design**		**2**					
27		**Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science**		**2**					
28		**Function Point (FP) Based Measures**		**1**					
29		**Cyclomatic Complexity Measures: Control Flow Graphs**		**2**					
No of Lectures Required to complete Unit 3				**10**	**No of Lectures Taken:**				
30	4	**Software Testing: Testing Objectives, Unit Testing**		**1**					
31		**Acceptance Testing, Regression Testing,**		**1**					

32		**Integration Testing, Testing for Functionality and Testing for Performance**		**1**				
33		**Top Down and Bottom Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing),**		**1**				
34		**Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products.**		**1**				
35		**Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection,**		**1**				
36		**Compliance with Design and Coding Standards.**		**1**				
No of Lectures Required to complete Unit 4				**7**	**No of Lectures Taken:**			
37		**Software Maintenance and Software Project Management: Software as an Evolutionary Entity.**		**1**				
38		**Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance,**		**1**				
39		**Software Re- Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process**		**1**				
40		**Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration**		**2**				

41		**Constructive Cost Models (COCOMO),**		**1**				
42		**Resource Allocation Models,**		**1**				
43		**Software Risk Analysis and Management.**		**1**				
No of Lectures Required to complete Unit 5				**8**	**No of Lectures Taken:**			

Teaching and Learning Methods	
TLM1	**Chalk and Talk**
TLM2	**PPT**
TLM3	**Tutorial**
TLM4	**Online**

Text Books & References	
1	**RS Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.**
2	**Pankaj Jalote, Software Engineering, Wiley**
3	**Rajib Mall, Fundamentals of Software Engineering, PHI Publication.**
4	**KK Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.**
5	**Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication**
6	**Ian Sommerville, Software Engineering, Addison Wesley.**
7	**Kassem Saleh, “Software Engineering”, Cengage Learning.**
8	**P fleeger, Software Engineering, Macmillan Publication**

Course Outcomes (COs)	
At the end of this course students will demonstrate the ability to:	
CO1	**Explain various software characteristics and analyze different software Development Models.**
CO2	**Demonstrate the contents of a SRS and apply basic software quality assurance practices to ensure that design, development meet or exceed applicable standards.**
CO3	**Compare and contrast various methods for software design**
CO4	**Formulate testing strategy for software systems, employ techniques such as unit testing, Test driven development and functional testing.**
CO5	**Manage software development process independently as well as in teams and make use of Various software management tools for development, maintenance and analysis.**

Faculty Instructor
Course Coordinator
Lecture Plan Incharge
Programme Coordinator
Head of Department