

SIR PADAMPAT SINGHANIA UNIVERSITY, UDAIPUR
SCHOOL OF ENGINEERING
Software Engineering

Code: CS 207

Credits: 4

L	T	P	C
3	0	1	4

Faculty: Sandeep Chaurasia

Learning Objective: Software engineering is a systematic and disciplined approach to developing software. It applies both computer science and engineering principles and practices to the creation, operation, and maintenance of software systems. Software Engineering requires expertise in data management, design and algorithm paradigms, programming languages, and human-computer interfaces. It also demands an understanding of and appreciation for systematic design processes, non-functional system properties, and large integrated systems. The program is designed to produce graduates who are capable of developing high-quality software systems. It emphasizes early development activities including requirements analysis, specification, which help to reveal errors early in the development process, when they are cheaper and easier to fix systematic, predictable processes that help to ensure that a software product satisfies its requirements.

Lesson Plan:

Topics	Lecture No.
Introduction:	
<i>The Evolving role of software</i>	1
<i>Changing nature of software, legacy software</i>	2
<i>Software Myths, Software Crisis</i>	3
Software Process:	
<i>SE – a layered technology, Process framework,</i>	4
<i>CMMI</i>	5
<i>Process pattern and assessment, Process technology, Product and process</i>	6
Process Models	
<i>Waterfall, Incremental, RAD</i>	7
<i>Prototype, Spiral, Concurrent Development</i>	8
<i>Component based, Formal methods models</i>	9
Requirement Engineering:	
<i>Tasks, Process, Eliciting requirement</i>	10
<i>Developing use cases</i>	11
<i>Building the analysis model</i>	12
<i>Negotiating and Validating requirements</i>	13
Analysis Model:	
<i>Requirement analysis, Modeling approaches</i>	14
<i>Data modeling concepts:</i>	15
<i>O-O analysis, Scenario based</i>	16
<i>Flow oriented, Class based modeling</i>	17

<i>Creating a behavioral model</i>	18
Design and Architecture:	
<i>Design process and quality</i>	19
<i>Design concepts & model</i>	20 – 21
<i>Software architecture, Data design, Architectural design</i>	22
<i>Mapping data flow into a software architecture</i>	23
Testing Strategy and Tactics:	
Approaches	24
Test strategies	25
Validation and System testing, Debugging	26
Black Box and White Box Testing - Basis path, Control structure testing	27 – 29
Project Management:	
Management spectrum, Metrics in the process and project domains	30
Software measurement	31
Metrics for software quality	32
Software project estimation, Decomposition techniques	33
Empirical estimation model	34
Project scheduling	35
Risk Management, RMMM plan	36
Quality Management, SQA	37
FTR, Statistical SQA	38
ISO 9000 quality standards	39
Software Configuration Management	40

Evaluation Criteria

Quizzes	10
Assignments	15
Midterm Examination	25
End-term Examination	50

Total 100

Suggested Reading (Books):

- 1 R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
- 2 Ian Sommerville, Software Engineering, Addison Wesley.
- 3 Carlo Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.
- 4 Pankaj Jalote, Software Engineering, Narosa Publication
- 5 Rajib Mall, Fundamentals of Software Engineering, PHI Publication.