

**UNIT I****Chapter 1 : Introduction to Software Engineering 1-1 to 1-10**

**Syllabus : Introduction :** What is software engineering ?  
Software Development Life Cycle, Requirements  
Analysis, Software Design, Coding, Testing,  
Maintenance etc.

1.1	Introduction.....	1-1
1.2	Software .....	1-1
1.2.1	Characteristics of Software .....	1-1
1.2.2	Classes of Software .....	1-2
1.2.3	Changing Nature of Software .....	2
1.3	What is Software Engineering ? .....	5
1.3.1	History of Software Engineering.....	6
1.4	Software Engineering - A Layered Technology.....	6
1.4.1	Quality Focus.....	6
1.4.2	Software Process .....	6
1.4.3	Software Engineering Methods .....	7
1.4.4	CASE Tools .....	7
1.5	Core Principles of SE Practice .....	7
1.6	Software Development Life Cycle (SDLC) .....	8
1.6.1	Requirements Analysis.....	9

**Chapter 2 : Software Requirements 2-1 to 2-21**

**Syllabus : Software Requirements :** Functional and Non-  
functional requirements, User Requirements, System  
Requirements, Interface Specification, Documentation of  
the software requirements.

2.1	Software Requirements.....	2-1
2.2	Functional and Non-Functional Requirements.....	2-1
2.2.1	Non-functional Requirements.....	2-2
2.3	User Requirements and System Requirements.....	2-3
2.4	Interface Specification .....	2-3
2.5	Documentation of Software Requirements.....	2-4
2.5.1	Need of SRS.....	2-4
2.5.2	Benefits of a SRS .....	2-4
2.5.3	Characteristics (Quality Criteria) of a Great SRS.....	2-5
2.5.4	IEEE Standard SRS Format.....	2-6
2.5.5	SRS Case Studies.....	2-7
2.5.5(A)	Online Library System .....	2-7

2.5.5(B)	Purchase Order System .....	2-11
2.5.5(C)	Hospital Management System.....	2-14
2.5.5(D)	Catering System .....	2-17

**Chapter 3 : Software Processes 3-1 to 3-2**

**Syllabus : Software Processes :** Process and Project,  
Component Software Processes.

3.1	Introduction .....	3-1
3.2	Process and Project.....	3-1
3.2.1	Software Process.....	3-1
3.2.2	Software Project.....	3-1
3.3	Components of Software Processes .....	3-1

**Chapter 4 : Software Development Process****Models****4-1 to 4-23**

**Syllabus : Software Development Process Models :** Waterfall  
Model, Prototyping, Iterative Development, Rational  
Unified Process, The RAD Model, Time boxing Model.

4.1	Software Development Process Models.....	4-1
4.1.1	Prescriptive Process Models .....	4-1
4.2	Waterfall Model (Oldest SDLC) .....	4-2
4.2.1	Preliminary Investigation.....	4-2
4.2.1(A)	Problem Identification (Problem Analysis).....	4-2
4.2.1(B)	Project Feasibility Study .....	4-3
4.2.2	System Analysis.....	4-4
4.2.3	Software Design.....	4-5
4.2.4	Coding.....	4-5
4.2.5	Testing .....	4-6
4.2.6	System Implementation .....	4-6
4.2.7	Maintenance .....	4-7
4.3	Iterative Development.....	4-8
4.3.1	Incremental Model .....	4-8
4.3.2	RAD Model.....	4-11
4.4	Evolutionary Process Models .....	4-12
4.4.1	Prototyping Model.....	4-13
4.4.2	The Spiral Model.....	4-15
4.4.2(A)	Win-Win Spiral Model .....	4-16
4.5	Rational Unified Process .....	4-17
4.5.1	RUP Phases .....	4-18
4.5.2	Six Best Practices .....	4-19
4.5.3	The "Nine Workflows" .....	4-20

**Chapter 5 : Agile Software Development 5-1 to 5-11**

**Syllabus : Agile software development :** Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.

5.1	Introduction.....	5-1
5.1.1	The 12 Principles of Agile.....	5-1
5.2	Agile Methodology.....	5-1
5.2.1	Feature Driven Development .....	5-2
5.2.2	Dynamic Systems Development Method .....	5-2
5.2.3	Scrum .....	5-2
5.3	Plan-Driven and Agile Development .....	5-3
5.4	Extreme Programming .....	5-4
5.4.1	Test Driven Development (TDD also known as Test First Development).....	5-8
5.4.2	Incremental Test Development from Scenario.....	5-9
5.4.3	User Involvement in Test Development and Validation .....	5-9
5.4.4	The use of Automated Test Harnesses.....	5-9
5.4.5	Comparison of all XP Tests.....	5-9
5.5	Agile Project Management .....	5-10
5.6	Scaling Agile Methods.....	5-10

**UNIT II****Chapter 6 : Socio-Technical System 6-1 to 6-6**

**Syllabus : Socio-technical system :** Essential characteristics of socio technical systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy System

6.1	Socio-Technical System.....	6-1
6.1.1	Essential Characteristics of Socio-technical Systems..	6-1
6.1.2	Emergent System Properties .....	6-2
6.2	Systems Engineering .....	6-3
6.2.1	System Engineering Process .....	6-3
6.2.2	Components of System .....	6-4
6.2.3	Computer Based Systems.....	6-5
6.3	Legacy System .....	6-6
6.3.1	Quality of Legacy Software .....	6-6

**Chapter 7 : Critical System 7-1 to 7-4**

**Syllabus : Critical System :** Types of critical system A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems.

7.1	Critical System.....	7-1
7.1.1	Types of Critical System.....	7-1

7.2	System Dependability .....	7-1
7.3	A Simple Safety Critical System .....	7-2
7.4	Availability and Reliability .....	7-2
7.5	Safety of the System.....	7-3
7.6	Security of the System.....	7-4

**Chapter 8 : Requirements Engineering Processes****8-1 to 8-14**

**Syllabus : Requirements Engineering Processes :** Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management.

8.1	Introduction .....	8-1
8.2	Requirements Engineering .....	8-1
8.3	Feasibility Study.....	8-2
8.3.1	Technical Feasibility .....	8-2
8.3.2	Operational Feasibility .....	8-3
8.3.3	Economical Feasibility .....	8-3
8.3.4	Organizational Feasibility.....	8-3
8.3.5	Cultural Feasibility.....	8-3
8.4	Requirements Engineering Tasks.....	8-3
8.5	Issues of Requirement Gathering .....	8-6
8.5.1	Problems of Requirements Gathering .....	8-7
8.6	Initiating the Requirements Engineering Process.....	8-7
8.6.1	Identifying the Stakeholders .....	8-7
8.6.2	Recognizing Multiple Viewpoints .....	8-8
8.6.3	Working Towards Collaboration.....	8-8
8.6.4	Asking the First Questions .....	8-9
8.7	Requirement Elicitation.....	8-9
8.7.1	Collaborative Requirements Gathering.....	8-9
8.7.2	Quality Function Deployment (QFD).....	8-10
8.7.3	User Scenarios .....	8-11
8.7.4	Elicitation Work Products .....	8-12
8.8	Requirements Analysis Model .....	8-12
8.8.1	Elements of the Analysis Model.....	8-12
8.8.2	Analysis Patterns .....	8-13
8.9	Negotiating Requirements .....	8-13
8.10	Requirements Validation.....	8-14

**Chapter 9 : System Models****9-1 to 9-25**

**Syllabus : System Models :** Models and its types, Context models, Behavioral models, Data models, Object models, Structural methods

9.1	Models and its Types.....	9-1
9.1.1	Analysis Modeling.....	9-1
9.1.2	Design Modeling Principles .....	9-2



9.1.3	Analysis Modeling Approaches .....	9-3
9.2	Context Model .....	9-3
9.3	Behavioral Model.....	9-3
9.3.1	Data Flow Diagram.....	9-4
9.3.2	State Diagram.....	9-9
9.4	Data Models .....	9-13
9.4.1	Data Dictionary .....	9-16
9.4.2	Entity Relationship Diagram .....	9-16
9.5	Object Model .....	9-23
9.6	Structured Methods.....	9-24

### UNIT III

#### Chapter 10 : Architectural Design      10-1 to 10-10

**Syllabus : Architectural Design :** Architectural design Concept, Architectural design decisions, System organisation, Decomposition styles, Control styles, Reference architectures.

10.1	Software Design .....	10-1
10.1.1	Architectural Design Concept.....	10-1
10.1.2	Advantages.....	10-1
10.1.3	System Properties focused in Architectural Design Process.....	10-2
10.2	Architectural Design Decisions.....	10-2
10.2.1	Software Architecture Reuse.....	10-3
10.2.2	Software Architectural Models.....	10-3
10.3	System Organization .....	10-3
10.4	Modular Decomposition Styles.....	10-4
10.5	Control Styles .....	10-5
10.6	Reference Architectures .....	10-7
10.6.1	OSI Reference Model.....	10-7
10.7	Application Architecture.....	10-8
10.7.1	Data Processing Systems .....	10-9
10.7.2	Transaction Processing Systems.....	10-9
10.7.3	Event Processing Systems.....	10-9
10.7.4	Language Processing Systems.....	10-10

#### Chapter 11 : User-Interface Design      11-1 to 11-7

**Syllabus : User Interface Design :** Need of UI design, Design Issues, UI design process, User Analysis, User Interface prototyping, Interface Evaluation.

11.1	User Interface Design.....	11-1
11.1.1	Need of UI Design .....	11-1
11.2	Interface Design Issues.....	11-1

11.3	UI Design Process .....	11-2
11.4	User Analysis.....	11-3
11.5	UI Design Principles .....	11-4
11.6	UI Prototyping .....	11-4
11.7	Interface Evaluation .....	11-5
11.7.1	Need of Evaluation .....	11-5
11.7.2	Usability Attributes .....	11-5
11.7.3	Steps involved in the UI Evaluation .....	11-6
11.7.4	Evaluation Criteria.....	11-6

#### Chapter 12 : Project Management      12-1 to 12-12

**Syllabus : Project Management :** Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management.

12.1	Software Project Management .....	12-1
12.1.1	Need of software project management.....	12-1
12.2	Management Activities.....	12-1
12.3	Project Planning.....	12-2
12.4	Project Scheduling.....	12-3
12.5	Risk Management.....	12-4
12.5.1	Risk Management Process (A framework for dealing with Risks) .....	12-5
12.5.2	Risk Analysis (Risk Assessment) .....	12-7
12.5.3	Risk Planning.....	12-7
12.5.4	Risk Monitoring and Control (Risk Management).....	12-8
12.5.5	Risk Projection (Estimation).....	12-9
12.5.6	Risk Refinement.....	12-10
12.5.7	RMMM Plan .....	12-10
12.5.8	Boehm's Top 10 Risks and Counter Measures .....	12-10

#### Chapter 13 : Quality Management      13-1 to 13-7

**Syllabus :** Process and Product Quality, Quality assurance and standards, Quality Planning, Quality Control, Software Measurement and Metrics.

13.1	Process and Product Quality .....	13-1
13.1.1	Definition of Quality.....	13-1
13.1.2	Importance of Software Quality .....	13-1
13.1.3	Quality Challenges.....	13-2
13.2	Quality Assurance and Quality Control.....	13-2
13.2.1	Need of Quality Assurance (QA) .....	13-3
13.3	Quality Standards .....	13-3
13.3.1	ISO 9000 Quality Standard.....	13-3
13.3.2	ISO 9001.....	13-3

13.4 Quality Planning ..... 13-5  
 13.5 Software Metrics and Measurement..... 13-6

**UNIT IV**

**Chapter 14 : Verification and Validation 14-1 to 14-30**

**Syllabus : Verification and Validation :** Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods.  
**Software Testing :** System Testing, Component Testing, Test Case Design, Test Automation.  
**Software Measurement :** Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics  
**Software Cost Estimation :** Software Productivity, Estimation Techniques, Algorithmic Cost Modeling, Project Duration and Staffing

14.1 Verification and Validation..... 14-1  
 14.1.1 Difference between Software Verification and Validation..... 14-1  
 14.1.2 Goals of Verification and Validation ..... 14-2  
 14.2 Planning Verification and Validation..... 14-2  
 14.3 Software Inspections ..... 14-3  
 14.3.1 The Inspection Process..... 14-4  
 14.4 Automated Static Analysis..... 14-6  
 14.5 Verification and Formal Methods..... 14-7  
 14.6 Software Testing..... 14-8  
 14.6.1 System Testing..... 14-9  
 14.6.1(A) Types of System Testing..... 14-10  
 14.6.2 Component Testing..... 14-11  
 14.6.3 Test Case Design..... 14-12  
 14.6.4 Test Automation ..... 14-16  
 14.6.4(A) Automated Testing Process ..... 14-17  
 14.7 Software Measurement ..... 14-18  
 14.7.1 Size-Oriented Metrics ..... 14-19  
 14.7.2 Function-Oriented Metrics ..... 14-20  
 14.7.3 Extended Function Point Metrics..... 14-20  
 14.8 Software Cost Estimation ..... 14-21  
 14.8.1 Software Productivity..... 14-21  
 14.8.2 Estimation Techniques ..... 14-22  
 14.8.3 Algorithmic Cost Modeling..... 14-24  
 14.8.3(A) The COCOMO Model..... 14-25  
 14.8.3(B) Algorithmic cost models in project planning ..... 14-27  
 14.8.4 Project Duration and Staffing ..... 14-29

**UNIT V**

**Chapter 15 : Process Improvement 15-1 to 15-23**

**Syllabus : Process Improvement :** Process and product quality, Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework.  
**Service Oriented Software Engineering :** Services as reusable components, Service Engineering, Software Development with Services.  
**Software reuse :** The reuse landscape, Application frameworks, Software Product lines, COTS product reuse.  
**Distributed software engineering :** Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service.

15.1 Process Improvement..... 15-1  
 15.1.1 Process and Product Quality ..... 15-2  
 15.1.2 Process Classification..... 15-3  
 15.1.3 Process Measurement..... 15-4  
 15.1.4 Process Analysis and Modeling..... 15-4  
 15.1.5 Process Change ..... 15-5  
 15.1.6 CMMI Process Improvement Framework ..... 15-5  
 15.2 Service Oriented Software Engineering (SOSE) ..... 15-7  
 15.2.1 Services as Reusable Components..... 15-7  
 15.2.2 Service Engineering..... 15-8  
 15.2.3 Software Development with Services ..... 15-9  
 15.3 Software Reuse ..... 15-10  
 15.3.1 The Reuse Landscape..... 15-11  
 15.3.2 Application Frameworks ..... 15-12  
 15.3.3 Software Product Lines..... 15-13  
 15.3.4 COTS Product Reuse ..... 15-16  
 15.3.4(A) COTS-Solution Systems..... 15-16  
 15.3.4(B) COTS-Integrated Systems..... 15-17  
 15.4 Distributed Software Engineering ..... 15-18  
 15.4.1 Distributed System Issues ..... 15-19  
 15.4.2 Client- Server Computing ..... 15-21  
 15.4.3 Architectural Pattern for Distributed Systems ..... 15-22  
 15.4.4 Software as Service ..... 15-23  
 • **List of Practicals..... L-1 to L-12**  
 • **Appendix A : Solved University Question Paper of April 2018 ..... A-1 to A-3**  
 • **Appendix B : Solved University Question Papers of Nov. 2018 and April 2019 ..... B-1 to B-4**