



Chapter 1 : Introduction To Software Engineering **1-1 to 1-14**

Syllabus : Software Engineering-process framework, Capability Maturity Model (CMM), Advanced Trends in Software Engineering.

1.1	Nature of Software	1-1
1.1.1	Absence of Fundamental Theory	1-1
1.1.2	Ease of Change	1-1
1.1.3	Rapid Evolution of Technologies	1-2
1.1.4	Low Manufacturing Cost	1-2
1.2	Software Definition	1-2
1.3	Software Engineering : A Layered Technology	1-2
1.3.1	Quality Focus	1-3
1.3.2	Process	1-3
1.3.3	Methods	1-3
1.3.4	Tools	1-4
1.4	The Characteristics of Software	1-4
1.5	The Software Engineering Process Framework	1-4
1.5.1	Umbrella Activities	1-5
1.6	Capability Maturity Model (CMM)	1-6
1.7	Advanced Trends in Software Engineering	1-10

Chapter 2 : Process Models **2-1 to 2-16**

Syllabus : Prescriptive Process Models : The Waterfall Model, Incremental Process Models, Evolutionary Process Models, RAD & Spiral.

2.1	A Generic Process Model (or Generic Process Framework)	2-1
2.1.1	Communication	2-1
2.1.2	Planning	2-1
2.1.3	Modelling	2-1
2.1.4	Construction	2-1
2.1.5	Deployment	2-2
2.2	Prescriptive Process Models	2-3
2.2.1	The Waterfall Model	2-3
2.2.1(A)	V-Model (Software Development)	2-6
2.2.2	Incremental Process Models	2-6
2.2.2(A)	The Incremental Model	2-6
2.2.2(B)	The RAD Model	2-8



2.3	Evolutionary Process Models.....	2-10
2.3.1	The Prototyping Paradigm	2-10
2.3.2	The Spiral Model.....	2-12
2.3.3	The Concurrent Development Model.....	2-13
2.3.4	Differentiation between Prescriptive and Evolutionary Process Models.....	2-15

Chapter 3 : Agile Process Models
3-1 to 3-20

Syllabus : Agile process : Extreme Programming (XP), Scrum, Kanban Model.
--

3.1	Agile Process Model	3-1
3.1.1	Comparison between the Agile and Evolutionary Process Models	3-2
3.2	Agile Software Development.....	3-2
3.2.1	Agile methods	3-3
3.2.2	Agile Manifesto	3-3
3.2.3	Agility Principles.....	3-4
3.3	Extreme Programming Practices	3-5
3.3.1	XP Values.....	3-5
3.3.2	The XP Process.....	3-6
3.4	Scrum.....	3-6
3.4.1	Process Flow.....	3-7
3.4.2	Scrum Roles	3-8
3.4.3	Scrum Cycle Description	3-9
3.4.4	Product Backlog.....	3-10
3.4.5	Sprint Planning Meeting.....	3-11
3.4.6	Sprint Backlog.....	3-12
3.4.7	Sprint Execution.....	3-13
3.4.8	Daily Scrum Meeting.....	3-13
3.4.9	Maintaining Sprint Backlog and Burn-Down Chart.....	3-14
3.4.10	Sprint Review and Retrospective	3-15
3.5	Introduction to Agile Tools : Kanban	3-15
3.5.1	Kanban Boards	3-16
3.5.2	Kanban Cards	3-17
3.5.3	The Benefits of Kanban	3-17
3.5.4	Comparison between Kanban and Scrum.....	3-19

**Chapter 4 : Software Requirements Analysis and Modeling****4-1 to 4-26**

Syllabus : Requirement Engineering, Requirement Modeling, Data flow diagram, Scenario based model, Software Requirement Specification document format(IEEE)

4.1	Introduction.....	4-1
4.2	Requirement Engineering.....	4-1
4.2.1	Inception	4-2
4.2.2	Elicitation.....	4-2
4.2.3	Elaboration.....	4-3
4.2.4	Negotiation.....	4-3
4.2.5	Specification.....	4-3
4.2.6	Validation.....	4-4
4.2.7	Requirement Management.....	4-4
4.2.8	Initiating the Requirement Engineering Process.....	4-4
4.3	Requirement Modelling.....	4-6
4.4	Data Flow Diagram	4-7
4.4.1	Flow-Oriented Modelling.....	4-7
4.4.1(A)	Data Flow Model.....	4-7
4.4.1(B)	Control Flow Model	4-9
4.4.1(C)	Control Specifications.....	4-9
4.4.1(D)	Process Specifications (PSPEC).....	4-11
4.5	Scenario Based Modelling : UML Models	4-11
4.5.1	Diagramming in UML.....	4-12
4.5.2	Developing Use Cases Diagram	4-14
4.5.3	Developing Activity Diagram.....	4-15
4.5.4	Swim Lane Diagram.....	4-16
4.5.5	Class Diagram.....	4-17
4.6	Software Requirements Specification (SRS) Document Format(IEEE).....	4-20
4.6.1	Writing Software Requirements Specifications	4-21
4.6.2	What is a Software Requirements Specification?	4-22
4.6.3	What Kind of Information Should an SRS Include?	4-22
4.6.4	SRS Template.....	4-23
4.6.5	Characteristics of an SRS.....	4-23
4.6.6	Structured Specifications for an Insulin Pump Case Study	4-24
4.6.7	Tabular Specifications for an Insulin Pump Case Study	4-25

Chapter 5 : Software Estimation Metrics **5-1 to 5-14**

Syllabus : Software Metrics, Software Project Estimation : LOC, FP, COCOMO II	
5.1 Software Metrics.....	5-1
5.1.1 Process Metrics	5-1
5.1.2 Project Metrics.....	5-2
5.2 Software Project Estimation.....	5-2
5.3 Observations on Estimation	5-2
5.3.1 Software Sizing	5-3
5.3.2 Problem-Based Estimation	5-4
5.3.3 An Example of LOC-Based Estimation.....	5-4
5.3.4 An Example of FP-Based Estimation.....	5-5
5.3.5 Process-Based Estimation	5-6
5.3.6 An Example of Process-Based Estimation.....	5-7
5.3.7 Estimation with Use-Cases.....	5-8
5.3.8 An Example of Use-Case Based Estimation.....	5-8
5.3.9 Reconciling Estimates	5-9
5.3.10 Software Scope and Feasibility.....	5-9
5.3.10(A)Obtaining Information Necessary for Scope.....	5-10
5.3.10(B)Feasibility	5-10
5.3.10(C) A Scoping Example	5-11
5.4 Empirical Estimation Models.....	5-11
5.4.1 The Structure of Estimation Models.....	5-11
5.4.2 The COCOMO II Model.....	5-12
5.4.3 The Software Equation.....	5-13

Chapter 6 : Project Scheduling and Tracking **6-1 to 6-8**

Syllabus : Project Scheduling, Tracking Software Project Estimation	
6.1 Project Scheduling.....	6-1
6.1.1 Defining a Task Set for the Software Project	6-2
6.1.2 Scheduling	6-3
6.1.2(A) Time-line Charts	6-3
6.2 Tracking Software Project Estimation.....	6-3
6.3 Schedule Tracking Tools	6-4
6.3.1 Microsoft Project	6-4



6.3.2	Daily Activity Reporting and Tracking (DART).....	6-6
6.4	Earned Value Analysis.....	6-7

Chapter 7 : Software Design	7-1 to 7-14
------------------------------------	--------------------

Syllabus : Design Principles, Design Concepts, Effective Modular Design, Cohesion and Coupling, Architectural design

7.1	Design Principles	7-1
7.2	Concept of Design.....	7-1
7.2.1	Abstraction.....	7-2
7.2.2	Architecture	7-2
7.2.3	Patterns.....	7-2
7.2.4	Modularity	7-2
7.2.5	Information Hiding	7-3
7.3	Effective Modular Design.....	7-3
7.4	Cohesion and Coupling.....	7-5
7.5	Architectural Design	7-8
7.6	Architectural Design Decisions	7-10
7.7	Architectural Views.....	7-11

Chapter 8 : Software Testing	8-1 to 8-12
-------------------------------------	--------------------

Syllabus : Unit testing, Integration testing, Validation testing, System testing

8.1	Introduction to Software Testing	8-1
8.1.1	Verification and Validation	8-2
8.1.1(A)	Difference between Verification and Validation.....	8-2
8.1.2	Organizing for Software Testing	8-3
8.2	Unit testing	8-4
8.3	Integration testing	8-5
8.4	Validation Testing.....	8-7
8.4.1	Validation Test Criteria.....	8-7
8.4.2	Configuration Review	8-8
8.4.3	Acceptance Testing.....	8-8
8.4.4	Alpha and Beta Testing.....	8-8
8.4.4(A)	Difference between Alpha Testing and Beta Testing.....	8-9
8.5	System Testing	8-10
8.5.1	Recovery Testing	8-10
8.5.2	Security Testing.....	8-11



8.5.3	Stress Testing.....	8-11
8.5.4	Performance Testing.....	8-12
8.5.5	Verification and Validation	8-12

Chapter 9 : Testing Techniques**9-1 to 9-12**

Syllabus : White-box testing: Basis path, Control structure testing, black-box testing: Graph based, Equivalence, Boundary Value

9.1	Testing Techniques	9-1
9.1.2	Types of Testing Techniques.....	9-1
9.2	White-Box Testing	9-1
9.2.1	Basis Path Testing.....	9-2
9.2.1(A)	Flow Graph Notation.....	9-2
9.2.1(B)	Independent Program Paths.....	9-3
9.2.2	Control Structure Testing.....	9-6
9.2.2(A)	Condition Testing	9-6
9.2.2(B)	Data Flow Testing.....	9-6
9.2.2(C)	Loop Testing	9-7
9.3	Black-Box Testing	9-8
9.3.1	Graph-Based Testing Method.....	9-9
9.3.2	Equivalence Partitioning.....	9-10
9.3.3	Boundary Value Analysis	9-10
9.3.4	Orthogonal Array Testing.....	9-11
9.3.5	Differentiation between White-box and Black-box Testing.....	9-12

Chapter 10 : Software Maintenance**10-1 to 10-6**

Syllabus : Types of Software Maintenance, Re-Engineering, Reverse Engineering
--

10.1	Software Maintenance.....	10-1
10.1.1	Types of Software Maintenance.....	10-2
10.1.1(A)	Corrective maintenance	10-2
10.1.1(B)	Adaptive maintenance	10-2
10.1.1(C)	Perfective maintenance	10-3
10.1.1(D)	Preventive maintenance.....	10-3
10.1.2	Maintenance Log.....	10-3
10.2	Software Re-engineering.....	10-4
10.3	Reverse Engineering	10-5
10.3.1	Abstraction Level.....	10-5



10.3.2	Completeness	10-5
10.3.3	Directionality.....	10-5

Chapter 11 : Risk Management**11-1 to 11-16****Syllabus : Risk Analysis & Management : Risk Mitigation, Monitoring and Management Plan (RMMM)**

11.1	Risk Analysis and Management	11-1
11.1.1	Software Risks	11-1
11.1.2	Reactive Versus Proactive Risk Strategies	11-2
11.2	Risk Identification	11-3
11.2.1	Assessing Overall Project Risk.....	11-4
11.2.2	Risk Components and Drivers	11-4
11.3	Risk Projection.....	11-6
11.3.1	Developing a Risk Table	11-6
11.3.2	Assessing Risk.....	11-7
11.3.3	Project Plan.....	11-8
11.4	Risk Refinement	11-8
11.5	Risk Mitigation, Risk Monitoring and Risk Management (RMMM)	11-9
11.5.1	The RMMM Plan.....	11-9
11.6	The RMMM Plan for Case Study Project.....	11-13
11.6.1	The general overview of RMMM Plan for WMITS.....	11-13
11.6.2	The Description of Risk for WMITS	11-14
11.6.3	Risk Mitigation, Monitoring and Management for WMITS.....	11-15

Chapter 12 : Quality Assurance**12-1 to 12-14****Syllabus : Quality Concepts, Software Quality assurance Metrics, Formal Technical Reviews, Software Reliability**

12.1	Quality Concept.....	12-1
12.1.1	McCall's Quality Factors	12-1
12.1.2	ISO 9126 Quality Factors	12-2
12.2	Software Quality Assurance (SQA).....	12-4
12.2.1	Software Quality Assurance Activities	12-5
12.2.2	SQA Relationships to Other Assurance Activities.....	12-6
12.3	Software Quality assurance Metrics.....	12-10
12.3.1	Measuring Quality	12-10
12.3.2	Defect Removal Efficiency	12-10

12.4	Formal Technical Reviews (FTR)	12-11
12.4.1	Review Meetings.....	12-11
12.4.2	Review Guidelines.....	12-12
12.5	Software Reliability	12-12
12.5.1	Measures of Reliability and Availability.....	12-13
12.5.2	Software Safety.....	12-13

Chapter 13 : SCM, Version and Change Control **13-1 to 13-12**

Syllabus : The Software Configuration Management (SCM), Version Control, Change Control
--

13.1	Software Configuration Management (SCM)	13-1
13.1.1	SCM Basics (Configuration Management System Elements).....	13-2
13.1.2	Baselines	13-2
13.1.3	Software Configuration Items	13-3
13.2	The SCM Repository	13-4
13.2.1	The Role of the Repository	13-4
13.2.2	General Features and Content	13-5
13.2.3	SCM Features.....	13-6
13.3	The SCM Process	13-7
13.3.1	Identification of Objects in the Software Configuration	13-8
13.3.2	Version Control	13-8
13.3.3	Change Control	13-9
13.3.4	Configuration Audit.....	13-11
13.3.5	Status Reporting	13-11

□□□