

# National Institute of Technology, Jamshedpur

## Department of Computer Applications

Spring Semester Session 2020 – 2021

### Course Handout

Date: 11-01-2021

<b>Batch</b>	:	<b>MCA (4<sup>th</sup> Semester)</b>
<b>Course No.</b>	:	<b>CA3401</b>
<b>Course Title</b>	:	<b>Software Engineering</b>
<b>Instructor In-Charge</b>	:	<b>Prof. D. A. Khan</b>

### Course Description

**Introduction:** Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes  
**Software Development Life Cycle (SDLC) Models:** Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models  
**Software Requirement Specifications (SRS):** Requirement Engineering Process, Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS.  
**Software Quality Assurance (SQA):** Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.  
**Software Design:** Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures.  
**Design Strategies:** Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design.  
**Software Measurement and Metrics:** Various Size Oriented Measures: Halstead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.  
**Software Testing:** Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.  
**Software Maintenance and Software Project Management:** Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.

### Scope

- To provide an in-depth knowledge of the most important aspects of Software Engineering
- To study about software development process, various SDLC models, SRS, SQA, Design strategies, Software measurement and metrics, Software testing, Software maintenance and software project management.
- To develop excellent understanding about Software development process for the development of quality software.

### Objectives

- At the end of this course, the student will be able to understand software engineering, the processes required to design and develop a quality software.
- At the end of this course, the student will be able to understand the importance of software engg.

### Text Book

T1. R.S. Pressman, "Software Engineering – A Practitioner's Approach", Third Edition. McGraw Hill International Edition, ISBN: 0073655783

T2. Pankaj Jalote, A Concise Introduction to Software Engineering, Springer, ISBN: 978-1-84800-301-9.

### Reference Books

- R1. Carlo Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.
- R2. Ian Sommerville, Software Engineering, Addison Wesley.

## Course Plan

Lec. No.	Learning Objectives	Topics to be covered	Refer to chapter see (text book)
1-4	Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes	Introduction of Software Engg.	Chapter 1 (T1)
5-7	SDLC Models: Water Fall Model, Iterative Model, Prototype Model, Spiral Model, V Model, Big Bang Model, Agile Model, RAD Model.	SDLC Models	Chapter 2 (T1) and Notes
8-12	Requirement Engineering Process, Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS.	Software Requirement Specifications	Chapter 3 (T2) Chapter 10 (T1)
13-16	Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.	Software Quality Assurance	Chapter 8 (T1) Chapter 24 (R2)
17-20	Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures.	Software Design	Chapter 13, 14 (T1)
21-23	Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design	Design Strategies	Chapter 6 (T2)
24-27	Various Size Oriented Measures: Halstead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.	Software Measurement and Metrics	Chapter 4 (T1)
28-35	Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance with Design and Coding Standards.	Software Testing	Chapter 17, 18, 19 (T1) Chapter 7, 8 (T2) Chapter 6 (R1)
36-42	Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.	Software Maintenance and Software Project Management	Chapter 3, 5, 6, 7 (T1) Chapter 22, 23 (R2)

### Evaluation Scheme (EC)

EC No.	Evaluation Component	Duration	Weightage	Date & Time	Nature of Component
1.	Mid Term Examination	02 Hours	30%	Academic Calendar	Closed Book
2.	End Term Examination	03 Hours	50%	Academic Calendar	Closed Book
3.	Internal Assessment	--	20%	TBA	(Class Test, Attendance, Assignments/Reports/Projects/Seminars)
Class Test/Reports/Projects/Seminars - 10 Marks, Assignment- 05 Marks, Attendance & Punctuality in class- 05 Marks					

Chamber consultation hour: Thursday, 5.00 Pm - 6.00 Pm, Chamber

Notices: All notices regarding the course will be displayed only on the Department of Computer Applications notice board.

Instructor In-Charge (CA3401)