

# 10B17CI572: Software Engineering Lab

**Course Credit: 1**

**Semester: V**

## Objective:

The Software Engineering Lab has been developed by keeping in mind the following objectives:

- To impart state-of-the-art knowledge on Software Engineering and UML in an interactive manner.
- Present case studies to demonstrate practical applications of different concepts
- Provide a scope to students where they can solve small, real life problems

## Learning Outcomes:

1. Case Study based on Software life cycle.
2. Develop, implement, and demonstrate the learning through a project that meet stated specifications.
3. You will learn User Interface Design.
4. Understand Software Cost Estimation and web engineering.

## List of Experiments

S NO	Topics
1	Identifying the Requirements from Problem Statements Requirements   Characteristics of Requirements   Categorization of Requirements   Functional Requirements   Identifying Functional Requirements
2	E-R Modeling from the Problem Statements Entity Relationship Model   Entity Set and Relationship Set   Attributes of Entity   Keys   Weak Entity   Entity Generalization and Specialization   Mapping Cardinalities   ER Diagram   Graphical Notations for ER Diagram   Importance of ER modeling
3	Identifying Domain Classes from the Problem Statements Domain Class   Traditional Techniques for Identification of Classes   Grammatical Approach Using Nouns   Advantages   Disadvantages   Using Generalization   Using Subclasses   Steps to Identify Domain Classes from Problem Statement   Advanced Concepts
4	Modeling UML Use Case Diagrams and Capturing Use Case Scenarios Use case diagrams   Actor   Use Case   Subject   Graphical Representation   Association between Actors and Use Cases   Use Case Relationships   Include

	Relationship   Extend Relationship   Generalization Relationship   Identifying Actors   Identifying Use cases   Guidelines for drawing Use Case diagrams
5	Modeling UML Class Diagrams and Sequence diagrams Structural and Behavioral aspects   Class diagram   Elements in class diagram   Class   Relationships   Sequence diagram   Elements in sequence diagram   Object   Life-line bar   Messages
6	Modeling Data Flow Diagrams Data Flow Diagram   Graphical notations for Data Flow Diagram   Explanation of Symbols used in DFD   Context diagram and leveling DFD
7	Statechart and Activity Modeling Statechart Diagrams   Building Blocks of a Statechart Diagram   State   Transition   Action   Guidelines for drawing Statechart Diagrams   Activity Diagrams   Components of an Activity Diagram   Activity   Flow   Decision   Merge   Fork   Join   Note   Partition   A Simple Example   Guidelines for drawing an Activity Diagram
8	Estimation of Project Metrics Project Estimation Techniques   COCOMO   Basic COCOMO Model   Intermediate COCOMO Model   Complete COCOMO Model   Advantages of COCOMO   Drawbacks of COCOMO   Halstead's Complexity Metrics
9	Estimation of Test Coverage Metrics and Structural Complexity Control Flow Graph   Terminologies   McCabe's Cyclomatic Complexity   Computing Cyclomatic Complexity   Optimum Value of Cyclomatic Complexity   Merits   Demerits

## References

1. R.S. Pressman, "Software Engineering: A Practitioner's Approach", 7Edition, McGraw Hill, 2010
2. Sommerville, "Introduction to Software Engineering", 8Edition, Addison-Wesley, 2007
3. Ghezzi, Jazayeri and Mandrioli, "Fundamentals of Software Engineering", 2Edition, Prentice-Hall, 2003
4. Peters and Pedrycz, "Software Engineering: An Engineering Approach, John Wiley, 2004
5. Len Bass, "Software Architecture in Practice", 2Edn. Addison Wesley, 2003
6. Allamaraju, "Professional Java Server Programming", Apress, 2004

**Evaluation Scheme:**

1. Mid Term Exam (Viva and Written Exam)	20
2. End term Exam (Viva and Written Exam)	30
3. Lab Records	5
4. Regular Assessment (Quality and quantity of experiment performed, Learning laboratory skills, Attendance etc.)	30
5. Project	15

---

<b>Total</b>	<b>100</b>
--------------	------------