S.Y. M.Sc. Semester III

Theory Core Paper -8 (CSA 5302): Software Engineering and UML

[Credits-4]

Course Outcomes

At the end of this course, students will be able to

CO1 Understand various models used for Software development.
CO2 Learn and Understand about requirement engineering
CO3 Implement the object-oriented programming with the help of UML diagrams.
CO4 Learn about various agile process models.

Unit	Details	Lectures
I	**Introduction to Software development** 1.1 Overview of Software Development with SSAD 1.2 System Development Life Cycle, different types of users and their roles 1.3 Models for System Development: Waterfall Model, Spiral Model, Prototyping Model, RAD Model, Unified Process Model	[6]
II	**Requirement Engineering** 2.1 Types of Requirements: Functional and Non-functional 2.2 Four Phases of Requirement Engineering 2.3 Software requirement Specification (SRS): Structure and contents of SRS, IEEE standard format for SRS	[5]
III	**Use-case Driven Object-oriented Analysis** 3.1 Introduction to UML 3.2 Requirement Analysis - Use-case Diagram, Identify Actors, Identify Use cases, Develop use-case Model 3.3 Basic Structural Modeling: Class Diagram and Object diagram Advanced Structural Modeling: Associations and links, Aggregation, Composition and containment, Inheritance, Sub Types and IS-A hierarchy Package Diagram	[10]
IV	**Basic Behavioral Modeling** 4.1 Interaction Diagram 4.2 Sequence Diagram 4.3 Activity Diagram 4.4 Collaboration Diagram 4.5 State Chart Diagram 4.6 State Transition Diagram	[15]
V	**Architectural Modeling** 5.1 Component Diagram 5.2 Deployment Diagram	[6]
VI	**Current Trends in Software Engineering** 6.1 Introduction to Web Engineering 6.2 Agile Process	[6]

	6.3	Agile Process Models: Extreme Programming (XP), Adaptive Software Development (ASD), Dynamic Systems Development Method (DSDM) : Scrum, Crystal Feature Driven Development (FDD)	

Books-

1. Ali Bahrami, Object Oriented System Development - McGRAW-HILL International Edition, 2017.
2. UML in Nutshell, O'reilly Publication, 2015.
3. Roger Pressman, Software Engineering (6th edition), 2009.
4. Grady Booch, James Rumbaugh, Ivar Jacobson,The Unified Modeling Language user guide, 2005.
5. James Rumbaugh, Michael Blaha, Object Oriented Modeling and Design with UML 2004.
6. Tom Pender, UML 2 Bible, 2002.
7. Ivan Jacobson, Object-Oriented Software Engineering: A Use Case Driven Approach 1992

Web References

1. https://www.tutorialspoint.com/software_engineering
2. https://www.javatpoint.com/software-engineering-tutorial
3. https://www.edx.org/course/uml-class-diagrams-for-software-engineering
4. https://www.tutorialspoint.com/uml
5. https://www.smartdraw.com/uml-diagram