ITA5005	Object Oriented Software Engineering	L	T	P	J	C
		3	0	0	0	3
Pre-requisite	**Nil**	**Syllabus version**				
		v. 1.1				

Course Objectives:

1. To learn various SDLC models and requirement gathering techniques.
2. To focus on understanding the user and their task, mapping to object oriented modeling.
3. To focus on techniques needed to develop a complete and consistency product.

Expected Course Outcomes:

1. Analyse various SDLC models and select appropriate model as per project nature and complexity.
2. Produce accurate and complete software product.
3. Develop a specialised knowledge, skills and judgement for complex software development.
4. Produce appropriate documentation accurately and to a professional standard.
5. Reinforce the requirement changes by achieving interoperability and integrity at each stages of the software development process.
6. Develop the products using object oriented techniques.

Student Learning Outcomes (SLO):	**2, 6**

Module:1	**Software and Software Engineering**	**6 hours**

The nature of software-Types of software- Characteristic of software-Stakeholders in software engineering – SDLC Process Models- Waterfall, RAD, Agile Software Development. – RUP

Module:2	**Review of object orientation**	**6 hours**

Introduction to object orientation- Classes and objects- inheritance- types of inheritance- Aggregation-Instance variables - Methods, operations and polymorphism -Organizing classes into inheritance hierarchies

Module:3	**Developing requirements**	**6 hours**

Domain analysis - Functional Requirement and Non-Functional requirements – Requirements gathering – object-based requirements analysis - Use cases: describing how the user will use the system - techniques for gathering requirements- Managing changing requirements, class-based requirements design

Module:4	**Modeling with classes**	**7 hours**

Introduction to UML - Essentials of UML class diagrams – Use case diagram- Activity diagram- Class diagram with Associations and multiplicity - Generalization - More advanced features of class diagrams

Module:5	**Focusing on users and their tasks**	**6 hours**

User-centered design - Characteristics of users - The basics of user interface design -Usability principles - Evaluating user interfaces- Modeling interactions and behavior: Interaction diagrams - State diagrams - Activity diagrams - Implementing classes based on interaction and state diagrams

- Difficulties and risks in modeling interactions and behavior.		
Module:6	**Architecting and designing software**	**6 hours**
The process of design - Principles leading to good design - Design Principles- Techniques for making good design decisions - Model Driven Development		
Module:7	**Basing software development on reusable technology**	**6 hours**
Reuse: building on the work and experience of others -Incorporating reusability and reuse into software engineering-Frameworks: reusable subsystems,the client-server architecture -Technology needed to build client-server systems -The Object Client-Server Framework (OCSF)		
Module:8	**Contemporary issues**	**2 hours**
Expert Talk		
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Timothy C Lethbridge, Object-Oriented Software Engineering Practical Software Development using UML and Java, 2010, 3rd Edition, McGraw-Hill Higher Education.
Reference Books	
1.	Ivar Jacobson, Object-Oriented Software Engineering: A Use Case Driven Approach, 2004, 1stEdition, Addison Wesley Longman Publishing

Recommended by Board of Studies	12-08-2017		
Approved by Academic Council	No. 46th	Date	24.08.2017