- Apply and design code optimization techniques for any input code with error recovery
- Design a compiler by incorporating the various phases of the compiler for any new source language

Category of Course	Continuous Assessment	Mid – Semester Assessment	End Semester
Theory Integrated with Practical	**15(T) + 25 (P)**	**20**	**40**

CO - PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓									✓
CO2	✓	✓	✓	✓	✓	✓					✓	✓
CO3	✓	✓	✓	✓	✓	✓	✓				✓	✓
CO4	✓	✓	✓	✓	✓	✓					✓	✓
CO5	✓	✓	✓	✓	✓	✓	✓				✓	✓
CO6	✓	✓	✓	✓	✓	✓	✓		✓		✓	✓

CS6110 OBJECT ORIENTED ANALYSIS AND DESIGN

Prerequisites for the course: None

OBJECTIVES:

- To capture the requirements specifications of an intended software system
- To design software with static and dynamic UML diagrams
- To map the design properly to code
- To improve the software design with design patterns
- To test the software against its requirements specifications

OBJECT ORIENTED ANALYSIS AND DESIGN	**L**	**T**	**P**	**EL**	**CREDITS**
	3	**0**	**4**	**3**	**6**

MODULE I :	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Introduction to OOAD with OO Basics - Unified Process – UML diagrams

SUGGESTED ACTIVITIES :

- EL - Identifying a suitable case study to work on for a complete end-end implementation
- EL – Document the Software Requirement Specifications(SRS) for the identified case study
- Practical – Getting familiar with the case tool

SUGGESTED EVALUATION METHODS:

- Assignment problems
- Quizzes

MODULE II :	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Use Cases –Case study – the Next Gen Point of Sale(POS) system, Inception Use case Modelling

SUGGESTED ACTIVITIES :

- EL – Identify use cases for the chosen case study and develop the Use Case model.
- Practical – Presenting the SRS for the chosen case study and obtaining approval

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE III :	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Use case modeling - Relating Use cases – include, extend and generalization - Class Diagram— Elaboration – Domain Model – Finding conceptual classes and description classes – Associations – Attributes

SUGGESTED ACTIVITIES :

1. EL - Identify the conceptual classes to develop a DomainModel and Class Diagram.
2. Practical – Presenting the use case model (for the chosen case study) along with use case diagrams.

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE IV :	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Domain Modeling using class diagrams - Domain model refinement – Finding conceptual class Hierarchies – Aggregation and Composition

SUGGESTED ACTIVITIES :

- EL – Expand the domain model by identifying the hierarchies, association, aggregation and composition
- Practical – Present the refined use case model and the basic domain model

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE V :	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Dynamic Diagrams - UML interaction diagrams - System sequence diagram – Collaboration diagram - Communication diagram

SUGGESTED ACTIVITIES :

- EL – Develop sequence diagrams for the scenarios identified in the use case model
- Practical – Presenting the complete domain model(after refinement) and class diagrams for the chosen case study

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE VI:	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

State machine diagram and Modelling – State Diagram - Activity diagram

SUGGESTED ACTIVITIES :

- EL - Develop state and activity diagrams for the chosen case study

- Practical – Presenting the dynamic model with sequence diagrams

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE VII:	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Implementation Diagram - UML package diagram - Component and Deployment Diagrams

SUGGESTED ACTIVITIES :

- EL –Finalize the environment and initiate implementation
- Practical – Presenting the complete dynamic model with state and activity diagrams and refined sequence diagrams

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE VIII:	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Designing objects with responsibilities – Creator – Information expert – Low Coupling – High Cohesion – Controller. Design Patterns – Creational – Factory method – Structural – Bridge – Adapter – Behavioural– Strategy – Observer

SUGGESTED ACTIVITIES :

- EL– Continue with the implementation
- Practical – Demonstrate partial implementation

SUGGESTED EVALUATION METHODS:

- Practical demonstration
- Quizzes

MODULE IX:	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Applying Gang of Four design patterns – Mapping design to code

SUGGESTED ACTIVITIES :

- EL – Identifying suitable design patterns to improve the design and documenting the rationale behind their selection. Proceed with the refined implementation by applying them,
- Practical **–** Demonstrate complete implementation without the design patterns

SUGGESTED EVALUATION METHODS:

- Practical demonstration
- Quizzes

MODULE X:	**L**	**T**	**P**	**EL**
	3	**0**	**4**	**3**

Object Oriented Methodologies – Software Quality Assurance – Impact of object orientation on Testing – Develop Test Cases and Test Plans

SUGGESTED ACTIVITIES :

- EL – Developing a Test plan with all test cases
- Practical – Present the modified design with appropriate design patterns. Demonstrate the implementation after incorporating the implementation of suitable design patterns

SUGGESTED EVALUATION METHODS:

- Presentations
- Quizzes

MODULE XI	L	T	P	EL
	2	0	4	0
Revisiting and consolidating all salient points and key insights based on the team projects				
Suggested Activities: • Practical – Demonstrating the test plan and the various test cases				
Suggested Evaluation: • Presentations				

OUTCOMES:

Upon completion of the course, the students will be able to:

- Identify and map basic software system requirements in UML
- Express software design with UML diagrams
- Design and implement software systems using OO methodology
- Improve software design using design patterns
- Test the software system developed against the intended requirements

TEXT BOOK:

1. Craig Larman, "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development", 3rd. Edition, Pearson Education, 2005.

REFERENCES:

1. Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language", Third Edition, Addison Wesley, 2003.
2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Patterns: Elements of Reusable Object-Oriented Software", Pearson, 2015.

EVALUATION METHOD TO BE USED:

Category of Course	Continuous Assessment	Mid – Semester Assessment	End Semester
Theory Integrated with Practical	**15(T) + 25 (P)**	**20**	**40**

CO - PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓	✓					✓	✓		
CO2	✓	✓	✓	✓	✓				✓	✓		
CO3	✓	✓	✓	✓	✓	✓			✓	✓		
CO4	✓	✓	✓	✓	✓				✓			✓
CO5	✓	✓	✓	✓	✓							✓

CS 6111 COMPUTER NETWORKS

CS 6111 COMPUTER NETWORKS	L	T	P	EL	CREDITS
	3	0	4	3	6
OBJECTIVES • To understand the division of network functionality into layers					