



VIETNAM NATIONAL UNIVERSITY OF HO CHI MINH CITY
UNIVERSITY OF NATURAL SCIENCES
FACULTY OF INFORMATION TECHNOLOGY

COURSE SYLLABUS

Course Code:	TH401
Title:	Object-oriented Software Development
Credits:	5
Workload:	Lecture hours: 4 periods * 15 weeks = 60 periods Laboratory hours: 2 periods * 15 weeks = 30 periods Preparative hours: 2 periods * 15 weeks = 30 periods
Prerequisites:	TH110 - Introduction to Software Engineering TH108 - Object-oriented Programming

Course Objectives:

The course is to provide students with the essential and advanced knowledge of Object-Oriented Software Engineering (such as SE procedures, SE methods, and CASE tools). To gain a general understanding, students will study different OO methods in which OMT (Object Modeling Techniques) is applied concretely and thoroughly to illustrate the analysis phase, design phase, and implementation phase.

Main Text: *Object-Oriented Modeling and Design*
James Rumbaugh et al., Prentice-Hall, 2001

References:

- *Conception orienté Objets et Application*
Grady Booch, Addison – Wesley, 1992.
- *Object-Oriented Analysis and Design*
James Martin, James J. Odell, Prentice-Hall, 1992.
- *Object-Oriented Modeling and Design*
James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen, Prentice-Hall, 2001.
- *Software Engineering – A Practitioner’s Approach*
Roger S. Pressman, McGraw-Hill, 2001.
- *Practical Software Engineering - A Case Study Approach*
Leszek A. Maciaszek, Bruce Lee Liong, Stephen Bills, Addison – Wesley, 2005.
- *Object-Oriented and Classical Software Engineering*
Stephen R. Schach, McGraw-Hill, 2002.

Course Outline:

Chapter 1 : Introduction (8 periods)

1. Basic concepts
 1. Characteristics and benefits of object-oriented design
 2. Object-oriented software engineering
2. Object-oriented methods
 1. Object-oriented analysis
 2. Object-oriented design
3. Object-oriented programming languages
4. Persistent objects and Database

Chapter 2 : Object-oriented Analysis (20 periods)

1. Class diagrams
 1. Basic concepts
 2. Simple diagrams
 3. Advanced diagrams
 4. Diagram construction methods
2. State diagrams
 1. Basic concepts
 2. Simple diagrams
 3. Advanced diagrams
 4. Diagram construction methods
3. Functional diagrams
 1. Basic concepts
 2. Diagrams
 3. Diagram construction methods
4. Relations of different diagrams
 1. Relation of Class and State Diagrams
 2. Relation of Functional to Class and State Diagrams
5. Analysis documents

Chapter 3 : Object-oriented design (20 periods)

1. System design
 1. Subsystem Design
 2. Shared resources
 3. Concurrency
2. Object design
 1. Combining the three diagrams
 2. Reusability
 3. Relationship optimization
 4. Non-functional requirements
3. Design documents

Chapter 4: From design to implementation (12 periods)

1. Implementation strategies

1. Implement using programming languages
2. Implement using relational and object-oriented database systems
3. Implement using other methods
2. Programming styles
 1. Object-oriented style
 2. Reusability
 3. Extendibility
 4. Robustness
3. Object-oriented programming languages
 1. Object-oriented language characteristics
 2. Class definitions and creating objects
 3. Implementing relationships
 4. Using inheritance
4. Non-object-oriented programming languages
 1. Non-object language characteristics
 2. Translating classes to data structures
 3. Allocating objects
 4. Implementing relationships
 5. Implementing inheritance

Grading

Final exam : 70%

Assignments: 30%