THAI NGUYEN UNIVERSITY
UNIVERSITY OF INFORMATION AND COMMUNICATION TECHNOLOGY

SOCIALIST REPUBLIC OF VIET NAM
Independence - Freedom - Happiness

COURSE SYLLABUS

(Training level: Undergraduate)

Course Title:

Vietnamese Course Title: Phát triển phần mềm hướng đối tượng với UML

English Course Title: Develop Object-Oriented Software with UML

Course Code: UML333

Major: Information Technology, Software Engineering

Version: 2017

1. General Information

- Number of credits: 3 (Theory: 2 ; Practice: 1)

- Types of Knowledge:

General Education		**Basic core courses**		**Major core courses**		**Concentration courses**		**Others**
Required ☐	Optional ☐			Information Technology Software Engineering				
		Required ☐	Optional ☐	Required ☒	Optional ☐	Required ☐	Optional ☐	Alternative subject of Graduation Thesis ☐

- Pre-requisite: Database
- Co-requisite: Data structure and algorithms, Object-oriented programming, Dot Net technology.
- Practice Room: Software: Rational Rose, StarUML, Astah professional, SoftwareIdiaModeler
- Departments in Charge: Faculty Information Technology

2. Time Allocated

Total: 60 Periods	Theory: 28 Periods
	Discussion/ Group Presentation: 0
	Assignment/ Essay/ Practice: 28 Periods
	Tests: 04 + Theory*: 2* Number of Tests*: 2* Periods*:2* + Practice*:2* Number of Tests*: 1* Periods*:2*
Self-Study: 90 Periods Other Activities: 0	

3. Lecturer's Information

No.	**Lecturer name**	**Phone number**	**Email**	**Note**
1	MSc. Nguyen Van Viet	0912660246	nvviet@ictu.edu.vn	Leader
2	Ph.D Nguyen Van Nui	0964.719.929	nvnui@ictu.edu.vn	Member
3	MSc. Nguyen Hong Tan	0943252165	nhtan@ictu.edu.vn	Member

No.	Lecturer name	Phone number	Email	Note
4	MSc. To Huu Nguyen	0989146142	thnguyen@ictu.edu.vn	Member
5	MSc. Bui Anh Tu	0914400246	batu@ictu.edu.vn	Member

4. Objectives

- Knowledge**:** General knowledge of software engineering, software design and construction. Core knowledge of analysis, design, and construction of object-oriented software. Introduction to software, processes, requirements specification, planning a software project, designing and building software. Introduction to object-oriented software development tools. Details and examples of system requirements, use cases, class diagrams, interaction diagrams, function diagrams, state diagrams, activity diagrams, interfaces, and programs in object-oriented software development.

- Skills: Skills in using aided tools in designing, building software in general and object-oriented software in particular. Skills in designing and building software for a real system.

- Attitude: Attend class fully; Good prepare for class time as requested by the teacher.

- Position of the course: The "Develop Object-Oriented Software with UML" course is an important course of majors: software, information technology, and information systems; Support for course approaching to programming. Problem analysis and design is a phase in the process of software development.

- The course contributes to the PLOs L7, L9

5. Description of content and output standards:

- Knowledge Standards: *(1) Remember ⇨ (2) Understand ⇨ (3) Apply ⇨ (4) Analyze ⇨ (5) Create.*

- Attitude Standards: *(1) Copy ⇨ (2) Self-manipulation ⇨ (3) Masterfully repeating to the norm ⇨ (4) Combining multiple activities ⇨ (5) Completely proactive.*

Notation CLOs	Contents	Level		PLOs
		Knowledge	Skills	
C1	Understand of basic knowledge about UML, software development models and object-oriented principles	2		L7
C2	Apply rational rose software, starUML… to build, analyze, and design class, object, and diagram models.	3	2	L9
C3	Understand the overview of the process of object-oriented analysis and design process	2	2	L7
C4	Understand of knowledge related to analysis, architecture design, use cases, interaction diagrams, class diagrams	2	2	L7
C5	Apply knowledge of detailed design the software components to real problems	3	2	L7
C6	Understand the mechanisms of system building and designing	2	2	L9
C7	Understand knowledge related to execution and distributed architecture	2	2	L7
C8	Understand the features of tools to specify use cases, subsystems, and classes	2	2	L9
C9	Understand the relevant new technology trends	2		L7

6. Reading List

- Main Syllabus:

[1] Department of Software Engineering, Faculty of Information Technology, Thai Nguyen University of Information and Communication Technology (2017), Develop Object-Oriented Software with UML

- References:

[2] Associate Professor, PhD. Dang Van Duc (2002), Analyze Object Oriented Design Using UML, Education publisher.

[3] Doan Van Ban (2005), Analysis of object-oriented design using UML, Education publisher.

[4] Huynh Van Duc (2006), UML Introductory Syllabus, Social Labor publisher.

[5] Rational (2003), Object-Oriented Analysis and Design with UML. SYBEX Inc.

7. Score Assessment

- Score Scale: 10.
- Components Assessment:

Evaluation time	**Components Assessment**	**Learning Outcomes**	**Factor**	**Score**	**Weight**
During the duration of the course	Attendance: (b_0)		$b_0=1$	d=($b_0+b_1+b_2+b_3$)/4	30%
According to the teaching plan in section 9	Test No.1: (b_1)	$b_1=1$	$b_1=1$		
	Test No.2: (b_2)	$b_2=1$	$b_2=1$		
	Test No.*3*: (b_3)	$b_3=1$	$b_3=1$		
The end of the term.	Final exam	C1, C2, C3, C4, C5, C6, C7, C8, C9		e	70%
Final Score: (f)				f = d ×30% + e ×70%	

- Final exam: *Written*

8. Regulations for students

8.1. Student's Duties

- Read materials and prepare for each lesson before attending class
- Complete assigned assignments.
- Prepare the practical content of the course.

8.2. Regulations on Exams and Academic Studies

- Students must attend classes, ensuring at least 80% of classroom sessions.
- Complete the assigned tasks for the course.
- Participate in the full number of regular tests.

9. Teaching Plan

No.	**Period**	**Contents**	**Teaching Methodology**	**CLOs**	**References**

1	3	**Chapter 1: High quality software development**
		1.1. Incremental development
		1.2. Requirements managment
		1.3. Component-Oriented Architecture
		1.4. Visual modeling
		1.5. Quality verify
		1.6 Change managment
			Teaching;
			Discussion;
			QA
				C1	[1].Chapter 1
					[2].Chapter 1

2	3	**Chapter 2: Object-Oriented Modeling**
		2.1. Object-Oriented Schema
		2.2. Schema representation by UML modeling language
			Teaching;
			Discussion;
			QA
				C1	[1].Chapter 2
					[2].Chapter 2
					[3].Chapter 1
					[4].Chapter 1

3	3	**Chapter 3: Overview of Software**	Teaching;	C1, C2	[1].Chapter 2

No.	Period	Contents	Teaching Methodology	CLOs	References
		requirements 3.1. Introduction 3.2. Basic concepts 3.3. Use-case model 3.4. Dictionary terms	Discussion; QA		[2].Chapter 2 [3].Chapter 1 [4].Chapter 1

4	3	**Chapter 4: Overview of analysis and design**
		4.1. Basic concepts
		4.2. Workflow and design analysis
			Teaching;
			Discussion;
			QA
				C3	[1].Chapter 3
					[2].Chapter 3
					[3].Chapter 2
					[4].Chapter 2

5	1	Test No.1	Testing	C1, C3	[1].Chapter 3 [2].Chapter 3 [3].Chapter 2 [4].Chapter 2
6	5	Practice 1: Instructions for installing and exploiting the features of StarUML or Rational Rose tools	Discussion; QA	C2	[1].Chapter 3 [2].Chapter 3 [3].Chapter 2 [4].Chapter 2

7	3	**Chapter 5: Architecture analysis**
		5.1. Overview of architecture analysis
		5.2. Basic concepts
		5.3. High level organization of subsystems
		5.4. Identify the analytical mechanisms
		5.5. Identify key abstractions
		5.6 Create use case realizations
			Teaching;
			Discussion;
			QA
				C3	[1].Chapter 3
					[2].Chapter 3
					[3].Chapter 2
					[4].Chapter 2

	5	Practice 2: Using software to build system use case diagram	Discussion; QA	C1, C2, C3	[1].Chapter 3 [2].Chapter 3 [3].Chapter 2 [4].Chapter 2

8	3	**Chapter 6: Use case analysis**
		6.1. Overview of use case analysis
		6.2. Adding to the use case description
		6.3 Finding classes from use case behavior
		6.4 Allocate use case behavior to classes
		6.5 Description of responsibilities
		6.6 Description of attributes and links
		6.7 Steps of use case analysis
		6.8 Unify the analytical classes
			Teaching;
			Discussion;
			QA
				C3, C4	[1].Chapter 3
					[2].Chapter 5
					[3].Chapter 5
					[4].Chapter 2

9	3	Practice 3: Building the interaction diagrams	Discussion; QA	C2, C3, C4	[1].Chapter 3 [2].Chapter 5 [3].Chapter 5 [4].Chapter 2
10	1	Test No.2	Testing	C2, C3, C4	[1].Chapter 3 [2].Chapter 5 [3].Chapter 5 [4].Chapter 2

11	3	**Chapter 7: Identify the design elements and mechanisms**
		7.1 Overview of identifying design elements
		7.2 Identify classes and subsystems
		7.3 Identify subsystem interfaces
		7.4 Identify chance of re-use
		7.5 Update the design models
			Teaching;
			Discussion;
			QA
				C5	[1].Chapter 3
					[2].Chapter 5
					[3].Chapter 5
					[4].Chapter 2

No.	Period	Contents	Teaching Methodology	CLOs	References
		7.6 Classify group using analytical mechanism 7.7 Documenting the architect mechanism			
12	5	Practice 4: Building the class diagrams	Discussion; QA	C5, C6, C7	[1].Chapter 3 [2].Chapter 5 [3].Chapter 5 [4].Chapter 2
13	3	**Chapter 8: Use case design** 8.1 Overview of use case design 8.2 Decribe the interaction between design objects 8.3 Simplify sequence diagrams using subsystems 8.4 Describe the behavior related to persistent storage 8.5 Smooth the description of event workflow 8.6 Unify the classes and subsystems	Teaching; Discussion; QA	C5, C6	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3
15	3	Chapter 9: Subsystem design 9.1 Overview of subsystem design 9.2 Distribution of subsystem behavior to its elements 9.3 Documenting the elements of subsystems 9.4 Describe the dependencies between subsystems	Teaching; Discussion; QA	C7, C8, C9	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3 [4].Chapter 4
15	5	Practice 5: Building the component diagrams and deployment diagrams	Discussion; QA	C5, C6, C7, C8	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3 [4].Chapter 4
16	3	Chapter 10: Class design 10.1 Overview of class design 10.2 Forming the design class 10.3 Identify the diagrams of design class 10.4 Identify the state diagrams 10.5 Harmonize the conflicts between use cases 10.6 Handling the non-functional requirements 0.7. Mapping the persistent storage class to the database	Teaching; Discussion; QA	C7, C8, C9	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3 [4].Chapter 4
17	3	Practice 6: Database building	Discussion; QA	C5, C6, C7, C8	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3 [4].Chapter 4
18	2	Test No.3	Testing	C2, C5, C6, C7, C8, C9	[1].Chapter 4 [2].Chapter 14 [3].Chapter 3 [4].Chapter 4

11. Competent Authority Approval: Thai Nguyen University of Information and Communication Technology

5th October, 2017

Vice Rector	Dean	Head of Department	Composer Team
Ph.D Do Dinh Cuong	**Ph.D Nguyen Hai Minh**	**MSc. Nguyen Hong Tan**	Nguyen Van Viet
			Nguyen Van Nui
			Nguyen Hong Tan
			To Huu Nguyen
			Bui Anh Tu