


**Nhập môn**  
**Phân tích thiết kế hướng đối tượng**

Giảng viên: Mai Thúy Nga



---

---

---

---

---

---

---

---

**Nội dung trình bày**

☆ Tổng quan về Phân tích và Thiết kế

- ◆ Nội dung môn học

OOAD using UML introduction 1 - 3

---

---

---

---

---


---

---

---

**Coding!!!**

- ◆ Nhiều LTV cho rằng phần mềm chủ yếu được xây dựng bằng cách gõ "code" từ bàn phím
  - Không dành đủ thời gian cho quá trình phân tích và thiết kế phần mềm
- ◆ Quan điểm này dẫn tới việc LTV tìm đến các ứng dụng hỗ trợ lập trình như Delphi, C++ Builder, JBuilder hay VB (RAD IDE).
- ◆ Với những "trận chiến coding" không bao giờ kết thúc, nhiều LTV phải "cày bừa" để hoàn thành chương trình vì
  - Họ không hiểu hoặc hiểu sai yêu cầu
  - Giao tiếp với các thành viên không tốt
  - Không tích hợp được với module của đồng nghiệp...
- ◆ Họ nhận ra rằng "Phân tích" và "Thiết kế" cần được coi trọng hơn, nhưng đã quá muộn



OOAD using UML introduction 1 - 4

---

---

---

---

---

---

---

---

**Phân tích và Thiết kế phần mềm (1)**

"Điều gì giúp LTV (kể cả QLDA) có thể tập trung vào việc thiết lập cách thức phân tích và thiết kế bài toán một cách hoàn thiện?"

OOAD using UML Introduction

1 - 5

---

---

---

---

---

---

---

---

**Phân tích và Thiết kế phần mềm (2)**

- ♦ Cần xây dựng một quy trình phát triển phần mềm hiệu quả
  - Quản lý yêu cầu hệ thống
  - Lập kế hoạch và kiểm soát công việc
  - Đảm bảo chất lượng sản phẩm...
- ♦ Thiết lập một cơ chế để nắm bắt yêu cầu, xây dựng thiết kế là điều rất quan trọng
  - Sử dụng các mẫu, các kỹ thuật để có thể đặc tả và minh họa quá trình phân tích và thiết kế
  - Sử dụng các chương trình hỗ trợ phân tích và thiết kế (tools)
- ♦ Cơ chế này phải như là một "ngôn ngữ thống nhất" giúp cho quá trình hợp tác hiệu quả giữa các thành viên và khách hàng trong nhóm phát triển phần mềm

OOAD using UML Introduction

1 - 6

---

---

---

---

---

---

---

---

**Mô hình hóa là gì? (Modeling)**

- ♦ Mô hình hóa là quá trình trừu tượng hóa các đối tượng (thực thể) trong thế giới thực
- ♦ Trừu tượng hóa đảm bảo đơn giản hóa những gì phức tạp trong thế giới thực:
  - Bỏ qua những thông tin không cần thiết
  - Chỉ thể hiện những thông tin cần thiết mang tính đặc thù của thực thể đó trong một ngữ cảnh cụ thể
- ♦ Khái niệm thế nào là cần thiết và không cần thiết phụ thuộc vào mục đích của mô hình đó, người quan tâm đến mô hình đó và một số nhân tố khác

OOAD using UML Introduction

1 - 7

---

---

---

---

---

---

---

---

## Tại sao cần mô hình hóa

- ♦ Khó khăn
  - Phần mềm ngày càng trở nên phức tạp
    - Windows XP > 40M LoC
  - Tìm hiểu mã nguồn luôn là một vấn đề khó đối với LTV
- ♦ Chúng ta cần một cách thể hiện các bài toán phức tạp một cách đơn giản hơn
  - Mô hình hóa là một phương tiện giúp chúng ta đơn giản hóa bài toán
- ♦ Mô hình chính là một phương tiện để giao tiếp giữa các thành viên tham gia phát triển dự án

OOAD using UML Introduction

1 - 8

---

---

---

---

---

---

---

---

## Nội dung trình bày

- ♦ Tổng quan về Phân tích và Thiết kế
- ☆ Nội dung môn học

OOAD using UML Introduction

1 - 9

---

---

---

---

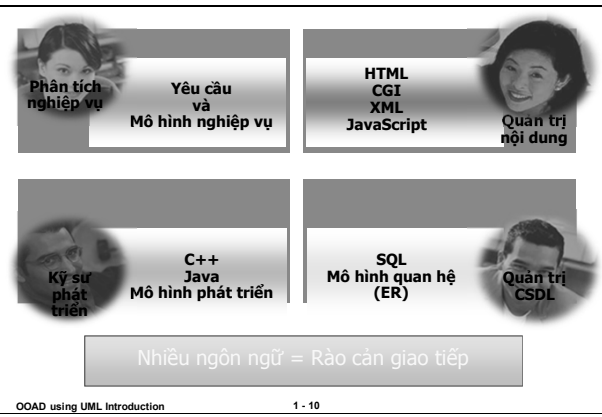
---

---

---

---

## Ngôn ngữ trong quá trình phát triển



OOAD using UML Introduction

1 - 10

---

---

---

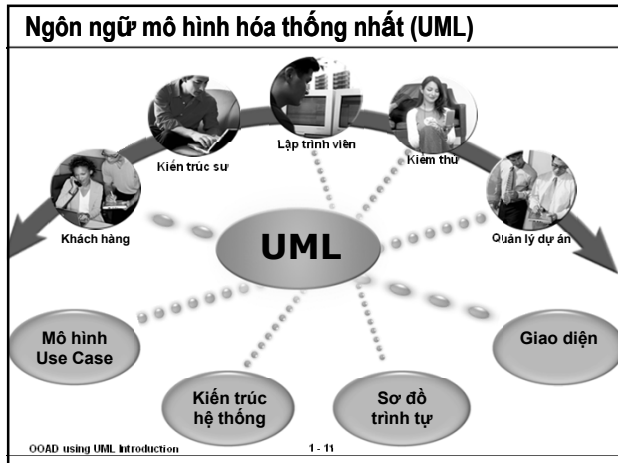
---

---

---

---

---




---

---

---

---

---

---

---

---

**UML (1)**

- ♦ UML được xây dựng bởi 3 chuyên gia phân tích hướng đối tượng
  - Grady Booch: Booch91
  - Ivar Jacobson: Object-Oriented Software Engineering (OOSE)
  - Jim Rumbaugh: Object Modeling Technique (OMT)
- ♦ Thiết lập một phương thức thống nhất để xây dựng và “vẽ” ra các yêu cầu & thiết kế theo hướng đối tượng trong quá trình phân tích và thiết kế phần mềm
- ♦ UML có thể được áp dụng trong nhiều quy trình phát triển phần mềm khác nhau
- ♦ UML có thể áp dụng được với phần lớn các ngôn ngữ OOP như Java, .NET...

OOAD using UML Introduction 1 - 12

---

---

---

---

---

---

---

---

**UML (2)**

- ♦ Ba chuyên gia hợp nhất các kỹ thuật của họ vào năm 1994
- ♦ UML được công nhận là chuẩn chung vào năm 1997

OOAD using UML Introduction 1 - 13

---

---

---

---

---

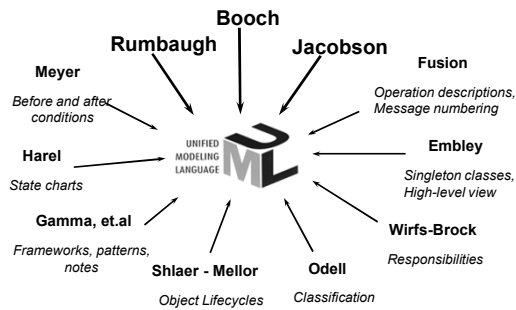
---

---

---

## UML (3)

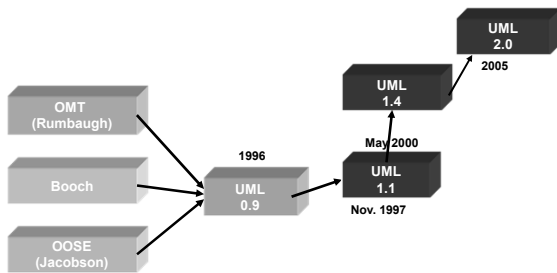
- ♦ UML cũng sử dụng một số kỹ thuật của các chuyên gia khác



OOAD using UML Introduction

1 - 14

## Lịch sử phát triển của UML



OOAD using UML Introduction

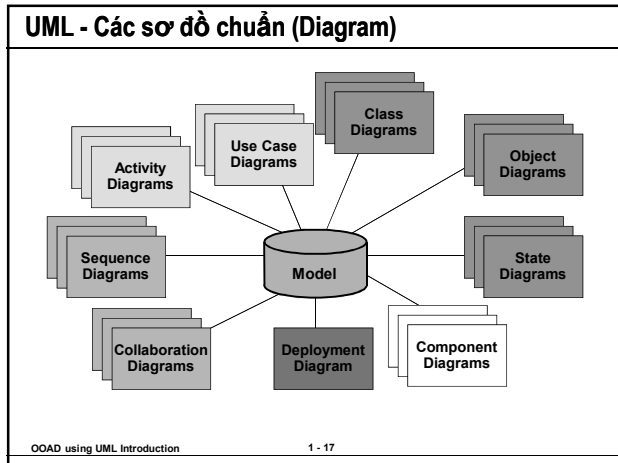
1 - 15

## Một số khái niệm

- ♦ Hệ thống (System): là một hệ thống thực tế cần phải mô tả hoặc xây dựng
- ♦ Mô hình (Model): là kết quả của quá trình trừu tượng hóa của hệ thống thực
- ♦ Khung nhìn (View): phản ánh một khía cạnh nào đó của mô hình
  - Mỗi một Model có thể có nhiều View khác nhau tùy thuộc vào mục đích của View đó
- ♦ Ký hiệu (Notation): là một ký hiệu đồ họa hoặc văn bản cho phép mô tả khung nhìn

OOAD using UML Introduction

1 - 16




---

---

---

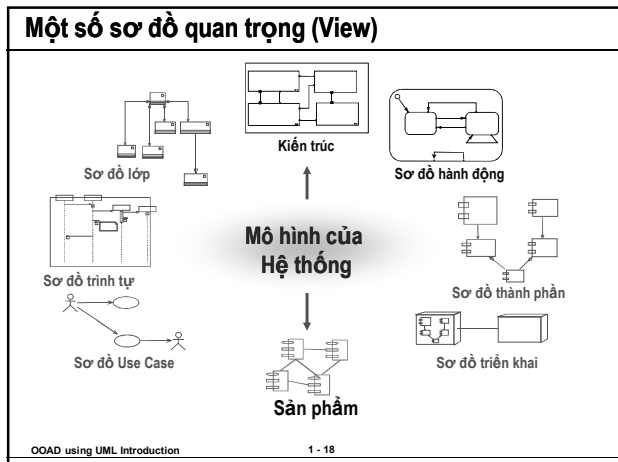
---

---

---

---

---




---

---

---

---

---

---

---

---

## Các thuật ngữ, từ viết tắt

- ♦ Object-Oriented Analysis and Design (OOAD): Phân tích thiết kế hướng đối tượng
- ♦ Unified Modeling Language (UML): Ngôn ngữ mô hình hoá
- ♦ Một số khái niệm hướng đối tượng
  - Abstraction: Sự trừu tượng hoá
  - Encapsulation: Sự đóng gói
  - Inheritance: Sự thừa kế
  - Polymorphism: Sự đa hình
  - Hierarchy: Sự phân cấp
  - Modularity: Sự mô đun hoá
- ♦ QLDA, LTV, KH

OOAD using UML Introduction 1 - 19

---

---

---

---

---

---

---

---

## Tổng quan về môn học

- ♦ Phân tích các chức năng của bài toán theo hướng use-case, phát triển phần mềm theo hướng tiếp cận mới
- ♦ Sử dụng UML để biểu diễn các mô hình phân tích và thiết kế
- ♦ Áp dụng các khái niệm hướng đối tượng: trừu tượng hoá, đóng gói, phân cấp, module hoá, đa hình để phát triển mô hình thiết kế
- ♦ Song song với việc học lý thuyết, sinh viên sẽ chia nhóm thực hành một đề tài thực tế

OOAD using UML Introduction

1 - 20

---

---

---

---

---

---

---

---

## Nội dung môn học

<b>Giới thiệu về PTTK hướng đối tượng với UML</b> 1	<b>Khái niệm về Hướng đối tượng trong UML</b> 2	<b>Đặc tả Yêu cầu với mô hình Use Case I</b> 3
<b>Đặc tả Yêu cầu với mô hình Use Case II</b> 4	<b>Tổng quan về Phân tích và Thiết kế</b> 5	<b>Phân tích Use Case I</b> 6
<b>Phân tích Use Case II</b> 7	<b>Mô hình hóa Thiết kế</b> 8	<b>Ôn tập</b> 9

OOAD using UML Introduction

1 - 21

---

---

---

---

---

---

---

---

## Nội dung thực hành

<b>Tìm hiểu mục đích của bài tập lớn</b> 1	<b>Thực hành UML với Rational Rose</b> 2	<b>Phân tích yêu cầu bài tập lớn</b> 3
<b>Xây dựng Đặc tả Yêu cầu I</b> 4	<b>Xây dựng Đặc tả Yêu cầu II</b> 5	<b>Xây dựng các sơ đồ phân tích Use Case I</b> 6
<b>Xây dựng các sơ đồ nhân tích Use Case I</b> 7	<b>Xây dựng sơ đồ Kiến trúc và Triển khai</b> 8	<b>Báo cáo bài tập lớn</b> 9

OOAD using UML Introduction

1 - 22

---

---

---

---

---

---

---

---

## Yêu cầu phần mềm

- ♦ Microsoft Word
- ♦ Microsoft PowerPoint
- ♦ Rational Rose 2002/2003 hoặc chương trình khác để thiết kế UML trong bài tập lớn

OOAD using UML Introduction

1 - 23

---

---

---

---

---

---

---

---

## Tài liệu tham khảo

- ♦ DEV485 – Object Oriented Analysis and Design with UML
  - IBM
- ♦ The Unified Modeling Language User Guide
  - Grady Booch, Ivar Jacobson, James Rumbaugh
- ♦ UML Distilled
  - Martin Fowler
- ♦ Writing Effective Use Cases
  - Alistair Cockburn
- ♦ Design Patterns
  - Nhà xuất bản Minh Khai
- ♦ Elements of Reusable Object Oriented Software
  - GoF
- ♦ Rational Web Site
  - <http://www.rational.com>

OOAD using UML Introduction

1 - 24

---

---

---

---

---

---

---

---

## Đánh giá môn học

- ♦ Bài tập lớn
  - Mỗi nhóm khoảng 4-6 sinh viên cho 1 dự án trong thời gian khóa học
  - Tài liệu (Tuần 1-9)
    - Tài liệu yêu cầu, thiết kế (theo mẫu cung cấp)
    - Báo cáo công việc thực hiện hàng tuần
  - Bảo vệ dự án (sử dụng slide) vào Tuần 9
- ♦ Thi cuối môn
  - Thi viết trong 2 tiết
  - Được sử dụng tài liệu
- ♦ Điểm môn học
  - Điểm quá trình (30%): Là điểm trung bình các bài trong tuần
  - Điểm thi viết cuối kì: 30%
  - Điểm dự án: Bảo vệ (20%), tài liệu (20%)
  - Nếu một trong các điểm thành phần <5 thì điểm cuối cùng là 4 điểm

OOAD using UML Introduction

1 - 25

---

---

---

---

---

---

---

---



### Thực tập

- ♦ Làm bài tập lớn theo phân công
- ♦ Tuân thủ kế hoạch và theo quy trình
- ♦ Hợp tác nhóm
- ♦ Tập trung vào tài liệu và cách thức giải quyết vấn đề
- ♦ Đặt câu hỏi nếu có cho các thành viên khác hoặc giảng viên

OOAD using UML Introduction

1 - 26

---

---

---

---

---

---

---

---

### Giới thiệu

- ♦ Kinh nghiệm của các bạn
  - Kinh nghiệm lập trình
  - Kinh nghiệm phát triển phần mềm
- ♦ Các bạn mong muốn gì từ khóa học này?
- ♦ Q & A?

OOAD using UML Introduction

1 - 27

---

---

---

---

---

---

---

---