



UML VÀ CÔNG CỤ PTTK HỆ THỐNG

Giảng viên: Nguyễn Tu Trung
BM HTTT, Khoa CNTT, Trường ĐH Thủy Lợi

Hà Nội, 2018

Nội dung

- ❖ Lịch sử ra đời của UML
- ❖ UML – Ngôn ngữ mô hình hoá hướng đối tượng
- ❖ Các khái niệm cơ bản trong UML
- ❖ Các biểu đồ UML
- ❖ Phương pháp luận hướng đối tượng
- ❖ Bộ công cụ StartUML

Lịch sử ra đời của UML

- ❖ Các ngôn ngữ hướng đối tượng ra đời khá sớm: Simula-67 (năm 1967), Smalltalk (đầu những năm 1980), C++, CLOS (giữa những năm 1980)...
- ❖ Những năm cuối 80, đầu 1990, các phương pháp luận và ngôn ngữ mô hình hóa hướng đối tượng mới ra đời: Booch của Grady Booch, OMT của James Rumbaugh, OOSE của Ivar Jacobson, hay OOA and OOD của Coad và Yordon
- ❖ Mỗi phương pháp luận và ngôn ngữ trên đều có hệ thống ký hiệu riêng, phương pháp xử lý riêng và công cụ hỗ trợ riêng
=> Cần thống nhất thành 1 chuẩn
- ❖ James Rumbaugh, Grady Booch và Ivar Jacobson đã cùng cố gắng xây dựng được một Ngôn Ngữ Mô Hình Hoá Thống Nhất và đặt tên là UML (Unified Modeling Language)
- ❖ UML đầu tiên được đưa ra năm 1997 và sau đó được chuẩn hoá để trở thành phiên bản 1.0, hiện nay là bản 2.

UML – Ngôn ngữ mô hình hoá hướng đối tượng

❖ UML

- ❖ Là ngôn ngữ mô hình hoá được xây dựng để đặc tả, phát triển và viết tài liệu cho các hệ phần mềm hướng đối tượng
- ❖ Bao gồm một tập các khái niệm, các ký hiệu, các biểu đồ và hướng dẫn sử dụng

❖ Mục đích của ngôn ngữ UML

- ❖ Mô hình hoá các hệ thống bằng cách sử dụng các khái niệm hướng đối tượng
- ❖ Thiết lập sự liên hệ từ nhận thức của con người đến các sự kiện cần mô hình hoá
- ❖ Giải quyết vấn đề về mức độ thừa kế trong các hệ thống phức tạp với nhiều ràng buộc khác nhau
- ❖ Tạo một ngôn ngữ mô hình hoá có thể sử dụng được bởi người và máy

UML – Ngôn ngữ mô hình hoá hướng đối tượng

- ❖ UML hỗ trợ phân rã hệ hướng đối tượng dựa trên cấu trúc tĩnh và hành vi động của hệ thống
 - ❖ Các cấu trúc tĩnh (static structure) xác định các kiểu đối tượng quan trọng của hệ thống và mối quan hệ giữa các đối tượng đó nhằm đến cài đặt sau này
 - ❖ Các hành vi động (dynamic behavior) xác định các hành động của các đối tượng theo thời gian và tương tác giữa các đối tượng

Các khái niệm cơ bản trong UML

- ❖ Khái niệm mô hình
- ❖ Các hướng nhìn (View) trong UML
- ❖ Các phần tử mô hình và các quan hệ

Khái niệm mô hình

- ❖ Mô hình
 - ❖ Là một biểu diễn của sự vật hay một tập các sự vật trong một lĩnh vực áp dụng nào đó theo một cách nhất định
 - ❖ Nhằm nắm bắt các khía cạnh quan trọng của sự vật, bỏ qua các khía cạnh không quan trọng và biểu diễn theo một tập ký hiệu và quy tắc nào đó
- ❖ Mục đích sử dụng mô hình khi xây dựng hệ thống
 - ❖ Nắm bắt chính xác yêu cầu và tri thức miền mà hệ thống cần phát triển
 - ❖ Thể hiện tư duy về thiết kế hệ thống
 - ❖ Trợ giúp ra quyết định thiết kế dựa trên việc phân tích yêu cầu
 - ❖ Tổ chức, tìm kiếm, lọc, kiểm tra và sửa đổi thông tin về các hệ thống lớn
 - ❖ Làm chủ được các hệ thống phức tạp

Các hướng nhìn (View) trong UML

- ❖ Một hướng nhìn trong UML là một tập con các biểu đồ UML được xây dựng để biểu diễn một khía cạnh nào đó của hệ thống
- ❖ Có thể có những biểu đồ UML có mặt trong cả hai hướng nhìn

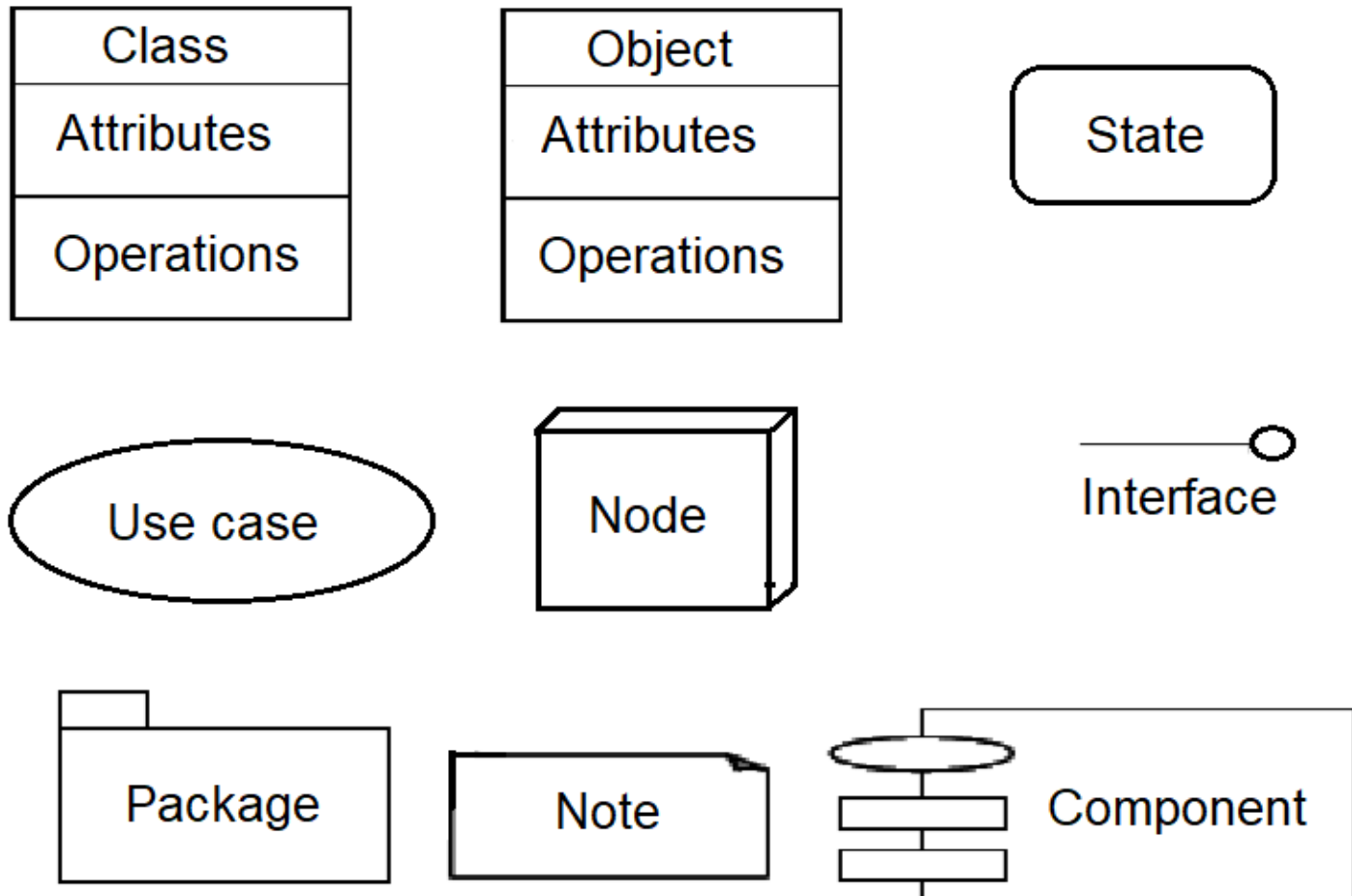
Khía cạnh	Hướng nhìn	Biểu đồ	Các khái niệm chính
Khía cạnh cấu trúc hệ thống	Hướng nhìn tĩnh (static view)	view) Biểu đồ lớp	Lớp, liên hệ, kế thừa, phụ thuộc, giao diện
	Hướng nhìn use case (Use case view)	Biểu đồ use case	Use case, tác nhân, liên hệ, extend, include ...
	Hướng nhìn cài đặt (implementation view)	Biểu đồ thành phần	Thành phần, giao diện, quan hệ phụ thuộc ...
	Hướng nhìn triển khai (deployment view)	Biểu đồ triển khai	Node, thành phần, quan hệ phụ thuộc, vị trí (location)

Các hướng nhìn (View) trong UML

Khía cạnh	Hướng nhìn	Biểu đồ	Các khái niệm chính
Khía cạnh động	Hướng nhìn máy trạng thái (state machine view)	Biểu đồ trạng thái	Trạng thái, sự kiện, chuyển tiếp, hành động
	Hướng nhìn hoạt động (activity view)	Biểu đồ động	Trạng thái, sự kiện, chuyển tiếp, kết hợp, đồng bộ...
	Hướng nhìn tương tác (interaction view)	Biểu đồ tuần tự	Tương tác, đối tượng, thông điệp, kích hoạt ...
		Biểu đồ cộng tác	Cộng tác, vai trò cộng tác, thông điệp...
Khía cạnh quản lý mô hình	Hướng nhìn quản lý mô hình	Biểu đồ lớp	Gói, hệ thống con, mô hình

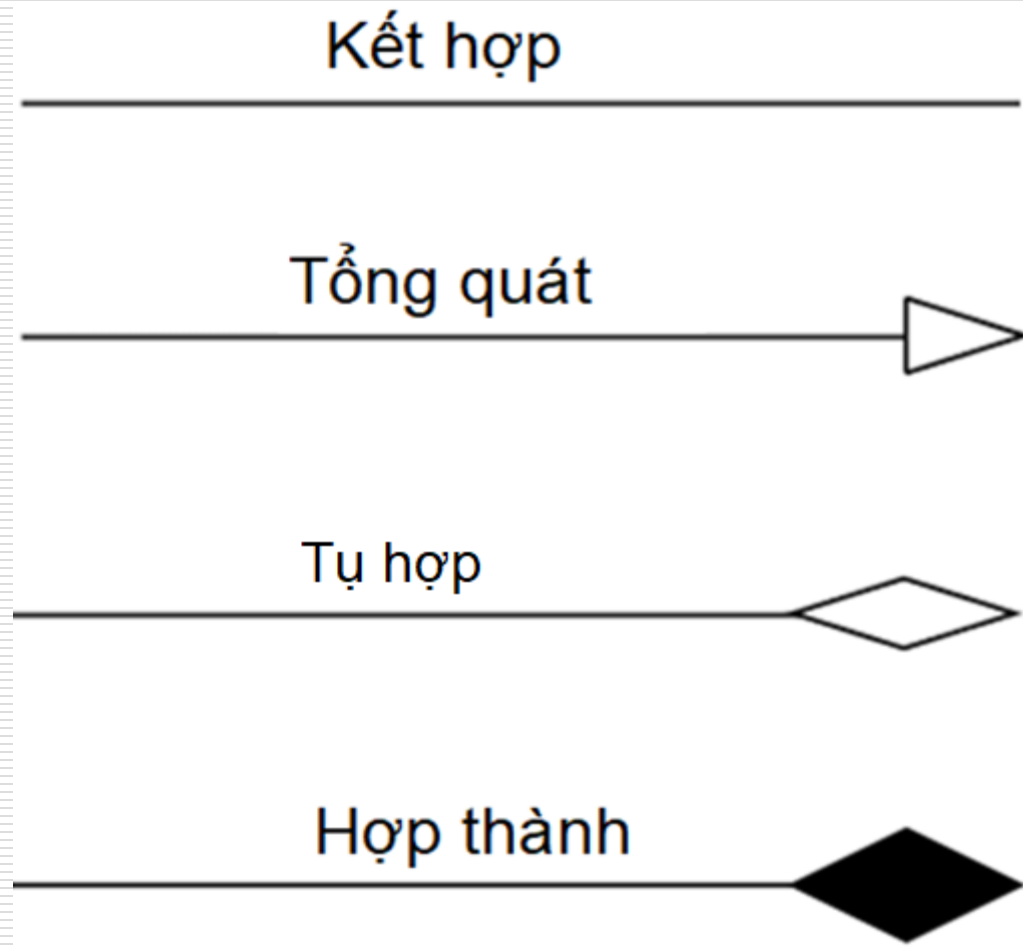
Các phần tử mô hình và các quan hệ

- ❖ Một số ký hiệu để mô hình hướng đối tượng thường gặp trong UML



Các phần tử mô hình và các quan hệ

- ❖ Đi kèm với các phần tử mô hình này là các quan hệ
- ❖ Các quan hệ này có thể xuất hiện trong bất cứ mô hình nào của UML dưới các dạng khác nhau (như quan hệ giữa các use case, quan hệ trong biểu đồ lớp...)



Các biểu đồ UML

- ❖ Thành phần mô hình chính trong UML là các biểu đồ
 - ❖ Biểu đồ use case
 - ❖ Biểu đồ hoạt động
 - ❖ Biểu đồ lớp
 - ❖ Các biểu đồ tương tác
 - ❖ Biểu đồ tuần tự
 - ❖ Biểu đồ cộng tác
 - ❖ Biểu đồ trạng thái
 - ❖ Biểu đồ thành phần
 - ❖ Biểu đồ triển khai
- ❖ Dựa trên tính chất của các biểu đồ, UML chia các biểu đồ thành hai lớp mô hình: Biểu đồ cấu trúc (Structure Diagram) và Biểu đồ hành vi (Behavior Diagram)

Biểu đồ cấu trúc (Structure Diagram)

- ❖ Nhóm các biểu đồ này biểu diễn các cấu trúc tĩnh của hệ phần mềm cần được mô hình hoá
- ❖ Các biểu đồ trong mô hình tĩnh tập trung biểu diễn khía cạnh tĩnh liên quan đến cấu trúc cơ bản cũng như các phần tử chính của hệ thống
- ❖ Các biểu đồ trong mô hình tĩnh bao gồm
 - ❖ Biểu đồ gói
 - ❖ Biểu đồ lớp và đối tượng
 - ❖ Biểu đồ thành phần
 - ❖ Biểu đồ triển khai

Biểu đồ hành vi (Behavior Diagram)

- ❖ Nhóm biểu đồ này nhằm nắm bắt các hoạt động và hành vi của hệ thống, cũng như tương tác giữa các phần tử bên trong và bên ngoài hệ thống
- ❖ Các biểu đồ trong mô hình động bao gồm
 - ❖ Biểu đồ use case
 - ❖ Biểu đồ hoạt động
 - ❖ Biểu đồ tuần tự
 - ❖ Biểu đồ cộng tác
 - ❖ Biểu đồ trạng thái

Phương pháp luận hướng đối tượng

- ❖ Tuân theo ba nguyên tắc:
 - ❖ Dựa vào ca sử dụng
 - ❖ Dựa vào hướng kiến trúc
 - ❖ Dựa vào lặp và gia tăng
- ❖ Tiến trình phát triển phần mềm hợp nhất

Dựa vào ca sử dụng

- ❖ Ca sử dụng là công cụ mô hình ban đầu để xác định hành vi của hệ thống
- ❖ Ca sử dụng mô tả cách người dùng tương tác với hệ thống để tiến hành một hoạt động nào đó như tạo hóa đơn, đặt hàng hay tìm kiếm thông tin
- ❖ Ca sử dụng được dùng để xác định và chuyển tải các yêu cầu hệ thống cho những người lập trình sau này

Dựa vào hướng kiến trúc

- ❖ Kiến trúc phần mềm cơ sở của hệ thống sẽ định hướng cách đặc tả, cách xây dựng và cách viết tài liệu hệ thống
- ❖ Mọi cách tiếp cận phân tích và thiết kế các hệ hướng đối tượng phải hỗ trợ ít nhất ba quan điểm theo kiến trúc của hệ thống:
 - ❖ Quan điểm chức năng:
 - ❖ Mô tả hành vi bên ngoài của hệ thống theo cách nhìn của người sử dụng
 - ❖ Ca sử dụng và biểu đồ ca sử dụng là cách tiếp cận ban đầu được sử dụng để minh họa quan điểm chức năng

Dựa vào hướng kiến trúc

❖ Quan điểm tĩnh

- ❖ Mô tả cấu trúc của hệ thống theo các lớp, thuộc tính, phương thức và quan hệ giữa các lớp
- ❖ Biểu đồ cấu trúc phác họa cách nhìn tĩnh của hệ hướng đối tượng đang tiến hóa và được biểu diễn bởi các biểu đồ cấu trúc trong UML

❖ Quan điểm động

- ❖ Mô tả hành vi bên trong của hệ thống theo các thông điệp được truyền giữa các đối tượng và sự thay đổi trạng thái bên trong một đối tượng
- ❖ Quan điểm động được biểu diễn bởi các biểu đồ hành vi trong UML

Dựa vào lặp và gia tăng

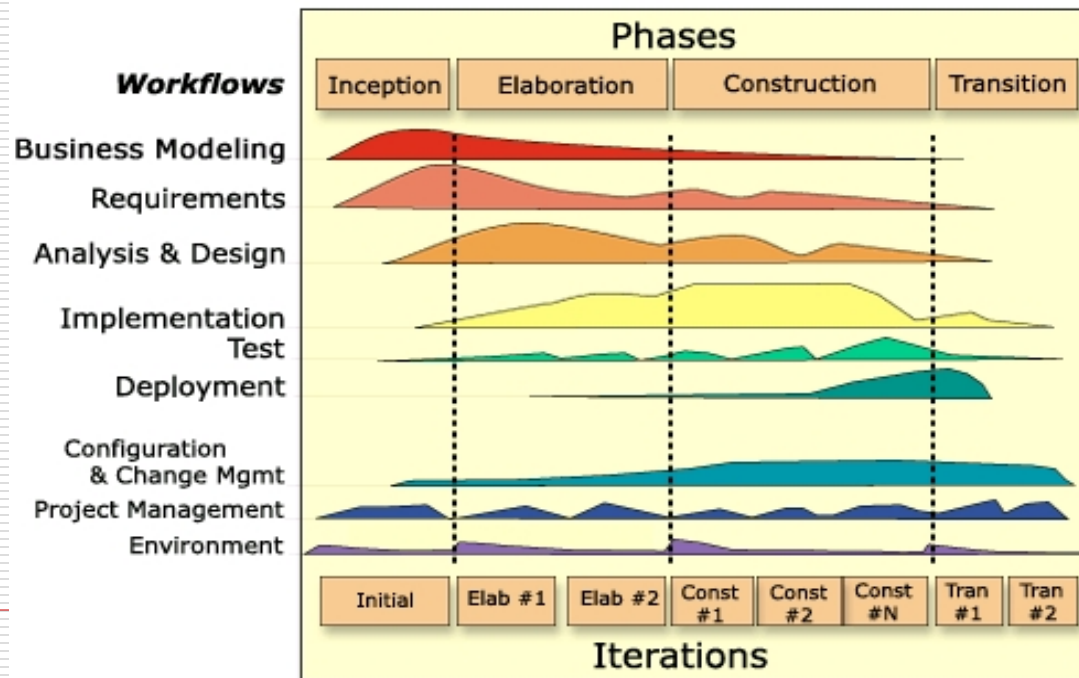
- ❖ Các cách tiếp cận phân tích và thiết kế hệ hướng đối tượng hiện nay nhấn mạnh phát triển lặp và tăng dần bằng cách tiến hành kiểm thử và mịn hóa liên tục suốt trong vòng đời của dự án
- ❖ Mỗi quá trình lặp trong phát triển sẽ làm cho hệ thống tiến gần hơn với yêu cầu thực sự của người sử dụng
- ❖ Các phương pháp luận được đưa ra đều là những phát triển dựa trên mô hình tiến trình hợp nhất (Unified Software Development Process) hay viết tắt là UP (Unified Process)
- ❖ Phổ biến nhất là Rational Unified Process (RUP)

Tiến trình phát triển phần mềm hợp nhất

- ❖ Nguyên lý lặp và tăng dần trong mô hình UP cho phép chia nhỏ các pha trong vòng đời dự án thành các giai đoạn nhỏ
- ❖ Mỗi giai đoạn có thể được lặp đi lặp lại và gia tăng theo thời gian và có thể theo các phiên bản (version) của sản phẩm trong dự án
- ❖ => có thể thêm và cải tiến các chức năng so với phiên bản trước của sản phẩm trong cùng một giai đoạn và việc cho phép thêm và cải tiến chức năng này => sự phát triển tăng dần của các giai đoạn theo thời gian
- ❖ UP là quá trình phát triển hệ thống theo hai chiều được mô tả bởi tập các pha (phase) và các công đoạn (workflow)
- ❖ Các pha bao gồm: khởi đầu, triển khai, xây dựng và chuyển giao

Tiến trình phát triển phần mềm hợp nhất

- ❖ Các công đoạn bao gồm 7 công đoạn kỹ thuật và 3 công đoạn hỗ trợ:
 - ❖ Các công đoạn kỹ thuật: mô hình nghiệp vụ, xác định yêu cầu, phân tích yêu cầu, thiết kế, cài đặt, kiểm thử, triển khai
 - ❖ Các công đoạn hỗ trợ: quản lý dự án, quản lý cấu hình và thay đổi và môi trường



Pha khởi đầu (inception)

- ❖ Cần trả lời các câu hỏi sau:
 - ❖ Chúng ta có khả năng về mặt kỹ thuật để xây dựng hệ thống hay không? (Tính khả thi về mặt kỹ thuật)
 - ❖ Nếu xây dựng được, nó có đem lại giá trị về mặt kinh tế hay không? (Tính khả thi về mặt kinh tế)
 - ❖ Nếu xây dựng được, tổ chức hay doanh nghiệp có sử dụng nó hay không? (Tính khả thi về mặt tổ chức)
- ❖ Các công việc cơ bản liên quan: mô hình nghiệp vụ, xác định yêu cầu và phân tích yêu cầu
- ❖ Những sản phẩm chính của pha này bao gồm:
 - ❖ Tài liệu trong đó cần đề cập phạm vi của dự án, những yêu cầu và ràng buộc, kế hoạch dự án ban đầu và mô tả tính chấp nhận được của những rủi ro liên quan đến dự án
 - ❖ Sự thừa nhận về môi trường cần thiết để phát triển hệ thống

Pha triển khai (Elaboration)

- ❖ Các luồng phân tích và thiết kế cơ bản tập trung suốt trong pha này
- ❖ Tiếp tục bằng phát triển tài liệu bao gồm kết thúc các ca nghiệp vụ, hiệu chỉnh lại đánh giá rủi ro, hoàn thiện kế hoạch dự án với đầy đủ chi tiết để các bên liên quan dự án có thể thỏa thuận xây dựng hệ thống thực sự cuối cùng
- ❖ Giải quyết vấn đề thu thập yêu cầu, bằng cách xây dựng các mô hình của bài toán theo các biểu đồ hành vi và cấu trúc của UML và đồng thời chi tiết hóa cách mô hình bài toán tương thích với kiến trúc hệ thống thế nào
- ❖ Sản phẩm chính của pha này bao gồm:
 - ❖ Các biểu đồ cấu trúc và hành vi trong UML
 - ❖ Phiên bản nền thực thi được của hệ thống

Pha xây dựng (Construction)

- ❖ Tập trung vào lập trình nên công việc chủ yếu liên quan đến luồng cài đặt
- ❖ Các luồng xác định yêu cầu, phân tích và thiết kế cũng liên quan đến pha này
- ❖ Bổ sung những yêu cầu còn thiếu sót và cuối cùng hoàn thành những mô hình phân tích và thiết kế
- ❖ Luồng quản lý cấu hình và thay đổi sẽ trở thành cực kỳ quan trọng trong pha xây dựng này
- ❖ Sản phẩm chính của pha này là phiên bản cài đặt beta của hệ thống với kiểm thử chấp nhận

Pha chuyển giao (Transition)

- ❖ Đến pha này, các luồng mô hình nghiệp vụ, xác định yêu cầu và phân tích ắt đã được hoàn thiện trong các vòng lặp trước
- ❖ Pha này tập trung chủ yếu vào các luồng kiểm thử và triển khai
- ❖ Phụ thuộc vào kết quả kiểm thử, có thể cần phải thiết kế lại hay cài đặt lại nhưng phải là tối thiểu tại thời điểm này
- ❖ Sản phẩm chính của pha này là hệ thống thông tin thực thi thực sự
- ❖ Các sản phẩm khác bao gồm tài liệu hướng dẫn sử dụng, bản kế hoạch để hỗ trợ người dùng và nâng cấp hệ thống trong tương lai

Các công đoạn

- ❖ Mô hình hóa nghiệp vụ: bao gồm tất cả kỹ thuật trực quan hóa mô hình nghiệp vụ
- ❖ Xác định yêu cầu: Định nghĩa hệ thống cần làm gì
- ❖ Phân tích & Thiết kế: Chỉ ra làm thế nào các ca sử dụng (usecases) của hệ thống được cài đặt
- ❖ Thực thi: Coding
- ❖ Kiểm thử: Testing
- ❖ Cài đặt: Cung cấp sản phẩm phần mềm đến người sử dụng
- ❖ Quản lý thay đổi & Cấu hình: Điều khiển và theo dõi sự thay đổi
- ❖ Quản lý dự án: Bảo đảm tất cả các công việc được lập lịch, cung cấp và hoàn chỉnh phù hợp với lịch, yêu cầu và kinh phí dự án
- ❖ Môi trường: Định nghĩa và quản lý môi trường trên đó hệ thống được phát triển

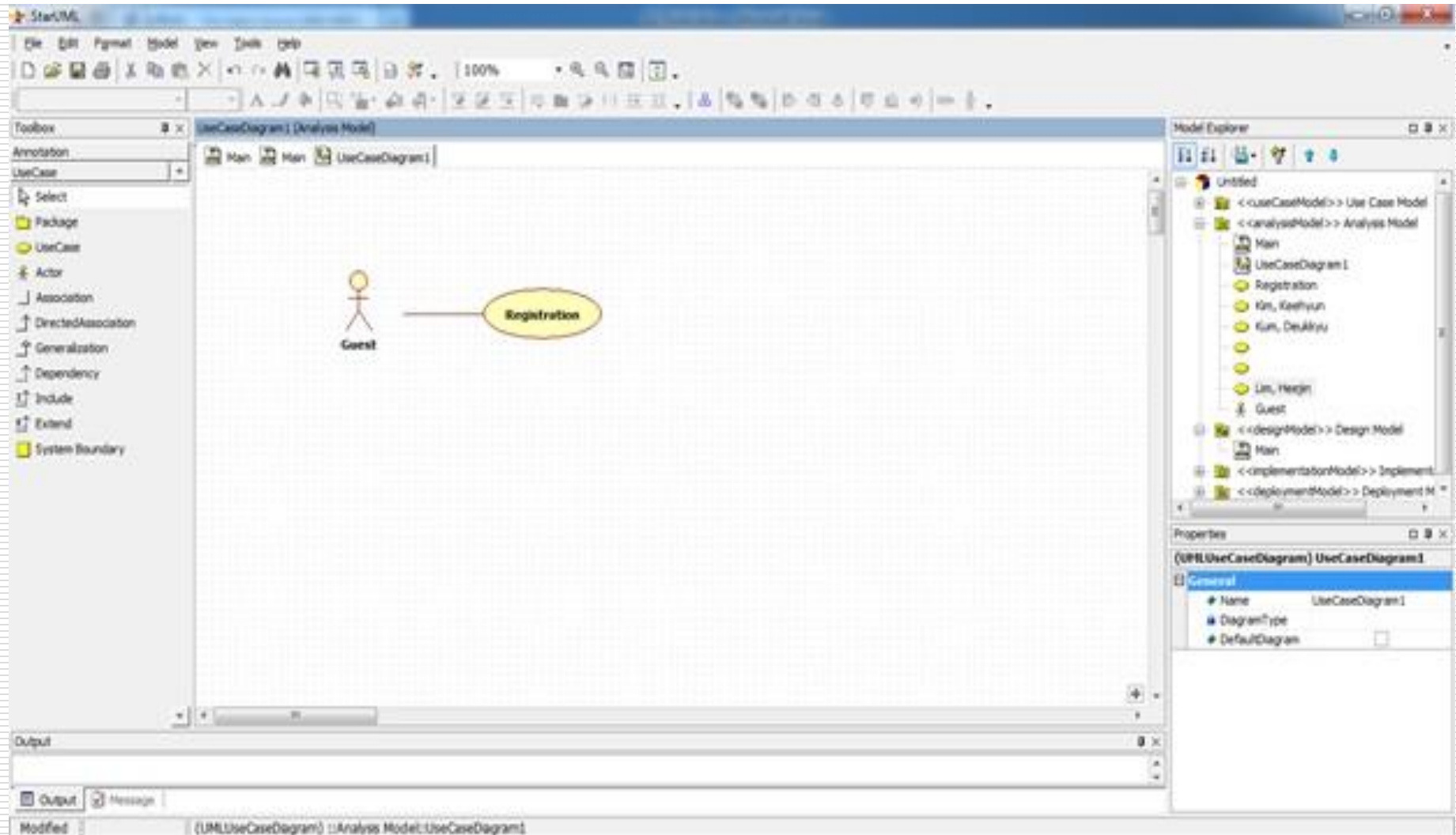
Bộ công cụ StartUML

- ❖ Giới thiệu bộ công cụ StartUML
- ❖ Màn hình chính của StartUML
- ❖ Các mô hình của StartUML

Giới thiệu bộ công cụ StartUML

- ❖ Có rất nhiều công cụ được sử dụng để vẽ các bản vẽ UML rất chuyên nghiệp như Rational Rose, Enterprise Architect, Microsoft Visio v.v.. và rất nhiều các công cụ phần mềm nguồn mở miễn phí có thể sử dụng tốt
- ❖ Các công cụ có cách sử dụng khá giống nhau và ký hiệu của các bản vẽ trên UML cũng đã thống nhất nên việc nắm bắt một công cụ khi chuyển sang làm việc với một công cụ khá không quá khó khăn
- ❖ Công cụ Start UML: phần mềm nguồn mở, miễn phí, có đầy đủ chức năng và có thể sử dụng tốt trên môi trường Windows
- ❖ Bạn có thể download bộ cài đặt của phần mềm Start UML tại <http://staruml.sourceforge.net/en/>
- ❖ Sau khi download và tiến hành các bước cài đặt chúng ta nhanh chóng có được công cụ này trên máy tính

Màn hình chính của StartUML



Các mô hình của StartUML

- ❖ StartUML có 5 model
 - ❖ Use Case Model: chứa các bản vẽ phân tích Use Case
 - ❖ Analysis Model: chứa các bản vẽ phân tích
 - ❖ Design Model: chứa các bản vẽ thiết kế
 - ❖ Implementation Model: chứa các bản vẽ cài đặt
- ❖ Để tạo các các bản vẽ
 - ❖ Chọn model mà bạn muốn sử dụng
 - ❖ Kích phải chuột
 - ❖ Chọn add diagram
 - ❖ Chọn bản vẽ cần xây dựng