

Chương 13

■ Thiết kế ứng dụng web

Slide đi cùng với sách

Software Engineering: A Practitioner's Approach, 7/e

Của Roger S. Pressman

Slides copyright © 1996, 2001, 2005, 2009 by Roger S. Pressman

Chỉ sử dụng cho giảng dạy phi lợi nhuận

Chỉ có thể được sao chép cho sinh viên tại trường đại học khi sử dụng kết hợp với *Software Engineering: A Practitioner's Approach, 7/e*. Cấm sao chép khi không được sự cho phép của tác giả

Toàn bộ thông tin bản quyền phải được ghi đầy đủ trên các trang web cung cấp tài liệu cho sinh viên.

Thiết kế ứng dụng web (WebApp)

“Có hai phương pháp thiết kế cơ bản: ý tưởng nghệ thuật để thể hiện bản thân và ý tưởng kỹ thuật để giải quyết vấn đề cho khách hàng.”

Jakob Nielsen

■ *Khi nào cần nhấn mạnh thiết kế WebApp*

- Khi các nội dung và chức năng rất phức tạp
- Khi kích thước của WebApp bao gồm hàng tram đối tượng, hàm và lớp phân tích
- Khi sự thành công của WebApp sẽ tác động trực tiếp tới sự thành công của doanh nghiệp

Thiết kế và chất lượng WebApp

- **An ninh**

- Ngăn ngừa tấn công từ bên ngoài
- Loại trừ truy cập trái phép
- Đảm bảo sự riêng tư của người dùng/khách hàng

- **Khả dụng**

- Ước lượng được tỉ lệ thời gian WebApp khả dụng

- **Khả năng mở rộng**

- WebApp có thể xử lý sự thay đổi đáng kể trong khối lượng người dùng/giao dịch

- **Thời gian ra thị trường**

Chất lượng với người dùng

- **Thời gian**

- Trang Web thay đổi bao nhiêu sau khi nâng cấp
- Các phần thay đổi có được highlight?

- **Cấu trúc**

- Các bộ phận của website kết nối với nhau như thế nào?
- Mọi liên kết trong/ngoài trang web có hoạt động?
- Tất cả các hình ảnh có được hiển thị?
- Có phần nào của website không được kết nối?

- **Nội dung**

- Nội dung các trang quan trọng có khớp với thông tin tại đó?
- Các từ khóa có tồn tại liên tục với các trang thay đổi nhanh?
- Các trang quan trọng có duy trì được chất lượng nội dung?
- Có thể tự động tạo các trang HTML?

Chất lượng với người dùng

- ***Độ chính xác, nhất quán***
 - Bản sao của ngày hôm qua có giống hôm nay?
 - Dữ liệu được trình bày chính xác, đầy đủ?
- ***Thời gian đáp ứng và độ trễ***
 - Server web có đáp ứng yêu cầu trình duyệt trong thời gian nhất định?
 - Trong thương mại điện tử, làm sao đảm bảo thời gian đáp ứng sau khi gửi một lệnh
 - Có phần nào của web quá chậm, khiến người dùng không tiếp tục sử dụng?
- ***Hiệu năng***
 - Kết nối có đủ nhanh?
 - Hiệu suất thay đổi như thế nào theo thời gian trong ngày?
 - Hiệu suất có đáp ứng được các ứng dụng thương mại điện tử

Mục tiêu thiết kế

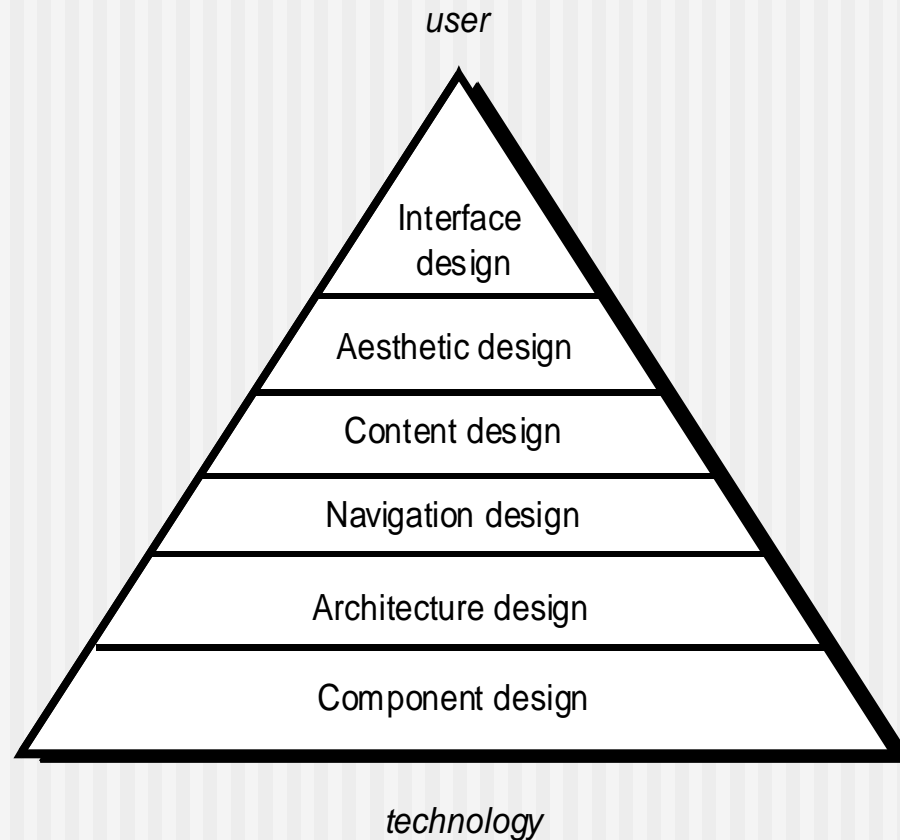
■ Nhất quán

- **Nội dung** cần được xây dựng nhất quán
- **Thiết kế đồ họa** nên có một cách nhìn nhất quán trên các phần của ứng dụng web
- **Thiết kế kiến trúc** cần đưa ra mẫu dẫn đến một cấu trúc siêu phương tiện phù hợp
- **Thiết kế giao diện** nên xác định các tương tác, chuyển hướng và hiển thị phù hợp
- **Cơ chế điều hướng** nên sử dụng nhất quán trên tất cả các yếu tố của ứng dụng

Mục tiêu thiết kế

- **Identity**
 - Establish an “identity” that is appropriate for the business purpose
- **Robustness**
 - The user expects robust content and functions that are relevant to the user’s needs
- **Navigability**
 - designed in a manner that is intuitive and predictable
- **Visual appeal**
 - the look and feel of content, interface layout, color coordination, the balance of text, graphics and other media, navigation mechanisms must appeal to end-users
- **Compatibility**
 - With all appropriate environments and configurations

Tháp thiết kế



Thiết kế giao diện

- *Giao diện cần*
 - Cung cấp dấu hiệu truy cập
 - Thông báo cho người sử dụng vị trí của họ trong nội dung phân cấp
- *Giao diện luôn giúp người dùng hiểu tùy chọn hiện tại của họ*
 - Những chức năng nào khả dụng
 - Những liên kết nào còn sử dụng được
 - Những nội dung nào có liên quan
- *Giao diện cần tạo điều kiện chuyển hướng*
 - Cung cấp một “bản đồ” dễ hiểu để người dùng có thể chuyển hướng trong ứng dụng

Giao diện WebApp hiệu quả

- Bruce Tognozzi [TOG01] đề nghị
 - **Giao diện hiệu quả cần trực quan**, người sử dụng có thể nhanh chóng xem các lựa chọn, nắm bắt xem làm thế nào để đạt mục tiêu và làm việc
 - **Người sử dụng không liên quan đến hoạt động bên trong của hệ thống**. Công việc thực hiện xuyên suốt và thường xuyên được lưu, với các tùy chọn đầy đủ để có thể hoàn tác.
 - **Thực hiện công việc một cách tối đa**, trong khi nhận thông tin ít nhất từ người dùng

Nguyên lý thiết kế giao diện I

- **Tiên đoán** — WebApp nên thiết kế để dự kiến được hành động tiếp theo của người dùng
- **Truyền thông** — Giao diện cần truyền thông các trạng thái của các hoạt động của người dùng
- **Nhất quán** — Sử dụng công cụ điều hướng, menu, biểu tượng
- **Kiểm soát quyền tự trị** — Giao diện tạo điều kiện cho người dùng di chuyển trong ứng dụng, theo các quy ước điều hướng được quy định trong ứng dụng
- **Hiệu quả** — Thiết kế của WebApp và giao diện nên được tối ưu hóa hiệu quả làm việc của người dùng, chứ không nên tối ưu hóa theo công việc của kỹ sư hay môi trường thực thi

Nguyên lý thiết kế giao diện II

- **Tập trung** — Giao diện và nội dung nên tập trung vào các tác vụ người dùng đang sử dụng
- **Luật Fitt** — “Thời gian đạt được mục tiêu là hàm của khoảng cách và kích thước mục tiêu”
- **Đối tượng giao diện người dùng** — Một thư viện lớn của các đối tượng giao diện tái sử dụng, phát triển cho WebApp
- **Giảm độ trễ** — Ứng dụng thực hiện đa nhiệm tuy nhiên vẫn cho phép người dùng thực hiện công việc liên tục
- **Thời gian học** — Giao diện nên thiết kế giảm thiểu thời gian học, và sau khi học để giảm thiểu thời gian học lại khi được truy cập lại.

Nguyên lý thiết kế giao diện III

- **Duy trì kết quả công việc** — Công việc của người dùng phải được lưu lại tự động, tránh mất mát nếu có lỗi xảy ra
- **Dễ đọc** — Mọi thông tin được trình bày phải dễ đọc với mọi người
- **Theo dõi trạng thái** — Khi thích hợp, các trạng thái tương tác của người dùng nên được theo dõi và ghi lại, để người dùng có thể đăng xuất và sau đó quay lại trạng thái làm việc cuối của họ.
- **Điều hướng** — Một giao diện tốt cung cấp các điều hướng rõ ràng, bố trí hợp lý

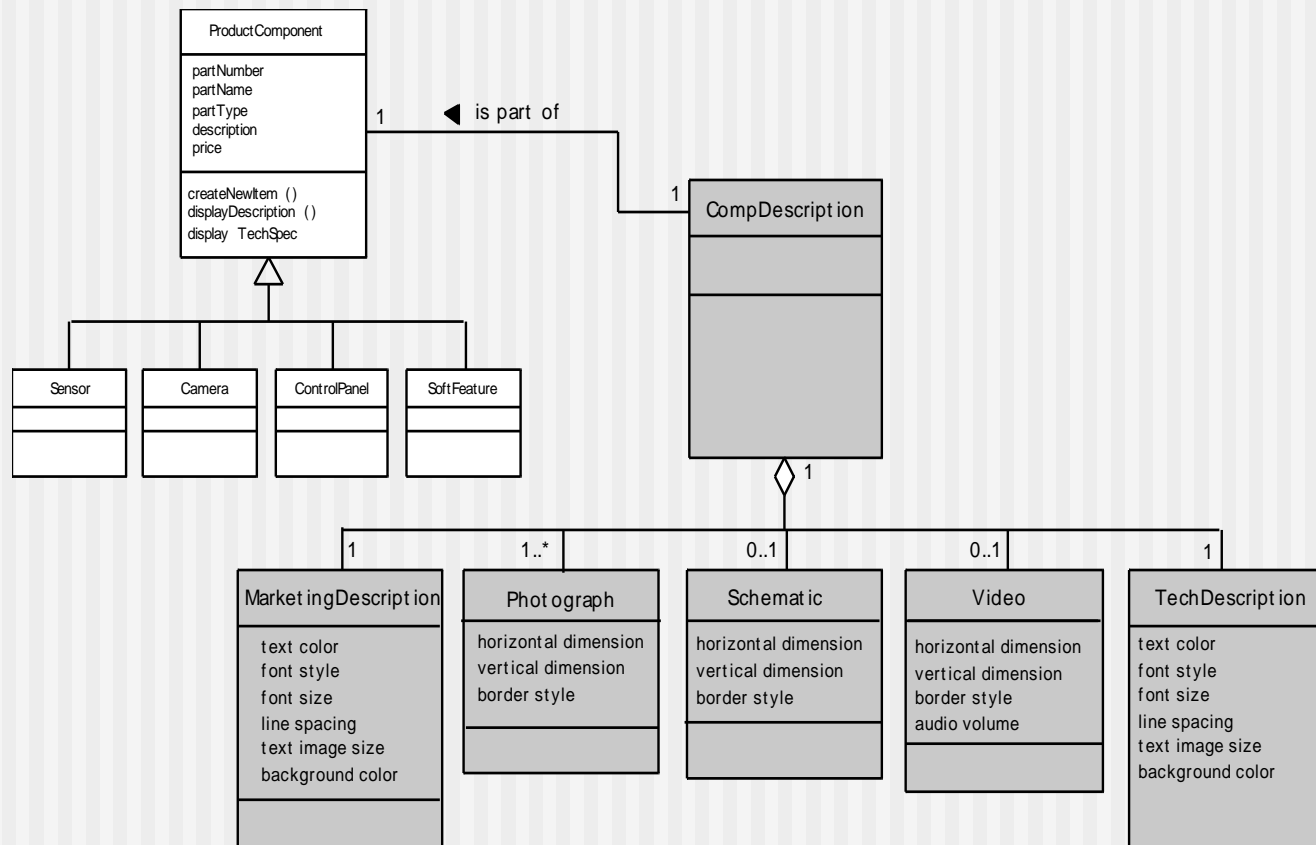
Tính thẩm mỹ

- Không ngại các khoảng trắng
- Nhấn mạnh nội dung
- Tổ chức các thành phần từ góc trên bên trái xuống góc dưới bên phải
- Nhóm các điều hướng, nội dung và các chức năng di chuyển trong trang
- Không mở rộng khung trang với thanh cuộn
- Xem xét độ phân giải và kích thước cửa sổ khi thiết kế

Nội dung

- Phát triển một thể hiện cho đối tượng nội dung
 - Với WebApp, đối tượng nội dung liên kết chặt chẽ với đối tượng dữ liệu hơn phần mềm thông thường
- Biểu diễn các cơ chế cần thiết để tạo các mối quan hệ giữa các đối tượng
 - Tương tự mối quan hệ giữa lớp phân tích và thiết kế thành phần trong chapter 11
- Một đối tượng nội dung gồm thông tin dành riêng cho nội dung và các thuộc tính hiển thị cụ thể được quy định như một phần của thiết kế

Thiết kế đối tượng nội dung

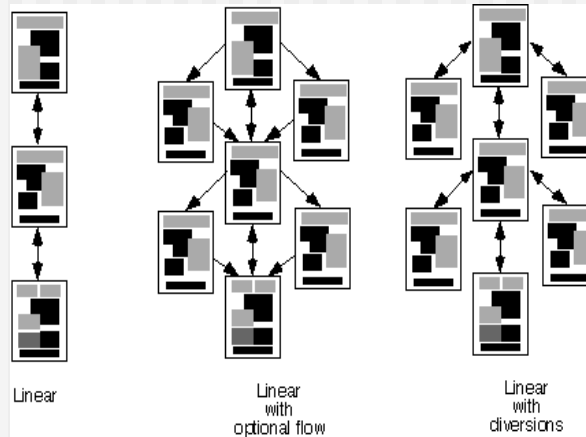


Thiết kế kiến trúc

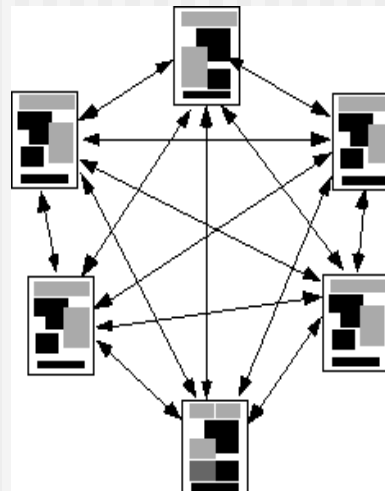
- *Kiến trúc nội dung* tập trung vào cách các đối tượng nội dung được tổ chức có cấu trúc để chuyển hướng/hiển thị
 - Khái niệm kiến trúc thông tin cũng được sử dụng cho các cấu trúc giúp tổ chức, đánh chỉ mục, chuyển hướng, ... tốt hơn
- *Kiến trúc WebApp* địa chỉ hóa theo cách các ứng dụng được xây dựng để quản lý tương tác người dùng, xử lý công việc nội bộ, chuyển hướng hiệu quả và hiển thị nội dung
- Thiết kế kiến trúc được tiến hành song song với thiết kế giao diện, thẩm mỹ và nội dung

Thiết kế nội dung

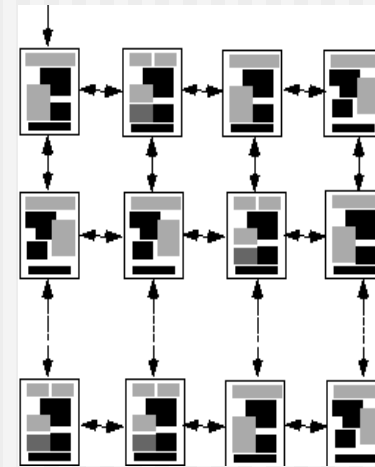
ear
cure



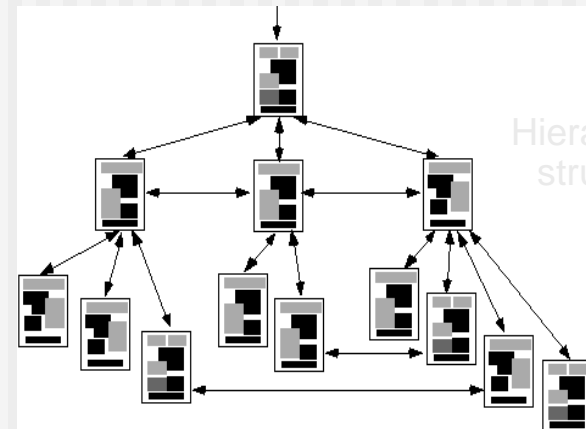
work
cure



Grid
structure



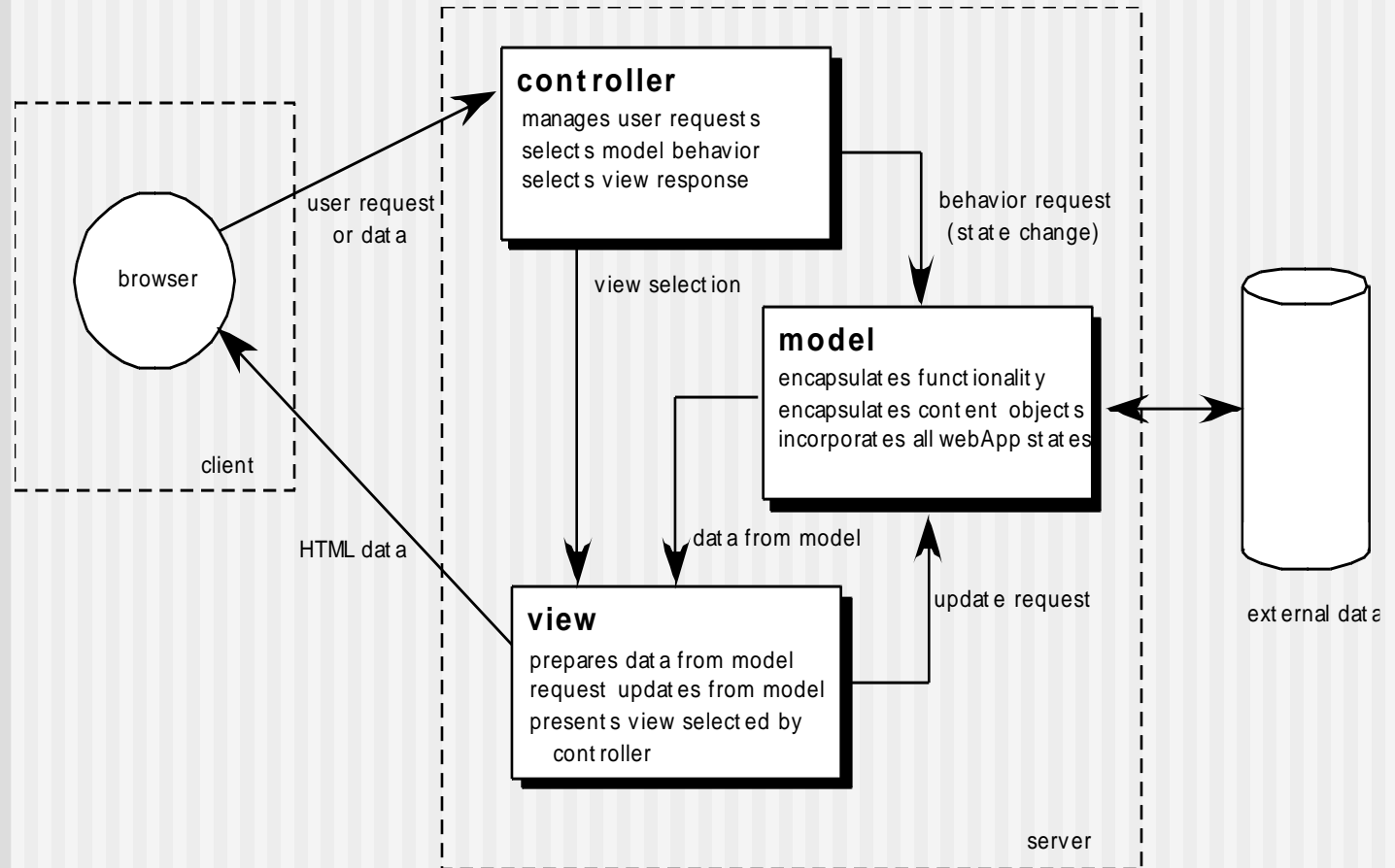
Hierarchical
structure



Kiến trúc MVC

- **Model** chứa tất cả các nội dung cụ thể của ứng dụng và xử lý logic, bao gồm cả:
 - Mọi đối tượng nội dung
 - Truy cập dữ liệu bên ngoài
 - Mọi chức năng xử lý trong ứng dụng
- **View** chứa tất cả các giao diện chức năng cụ thể và cho phép:
 - Trình bày mọi nội dung và chức năng logic
 - Truy cập dữ liệu bên ngoài
 - Các chức năng được người dùng yêu cầu
- **Controller** cung cấp truy cập vào Model và View, điều khiển dòng dữ liệu

Mô hình MVC



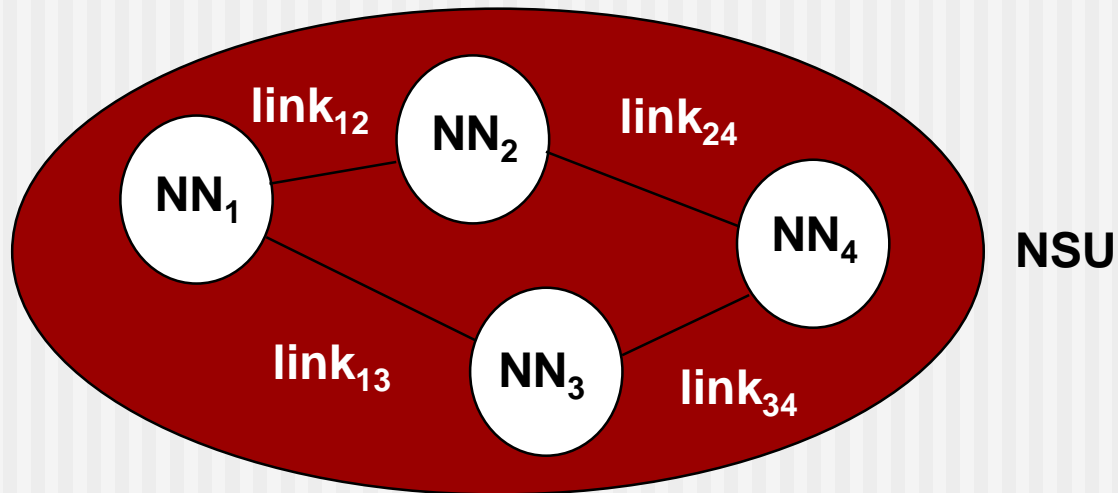
Thiết kế điều hướng

- Bắt đầu với việc xem xét cây phân cấp người dùng và use-case liên quan
 - Mỗi người dùng có thể sử dụng WebApp khác nhau và có yêu cầu điều hướng khác nhau
- Mỗi người dùng tương tác với WebApp sẽ xác định một *navigation semantic units* (NSUs)
 - NSU — “Tập thông tin, cấu trúc danh mục liên quan đến việc thực hiện một tập con của yêu cầu người dùng”

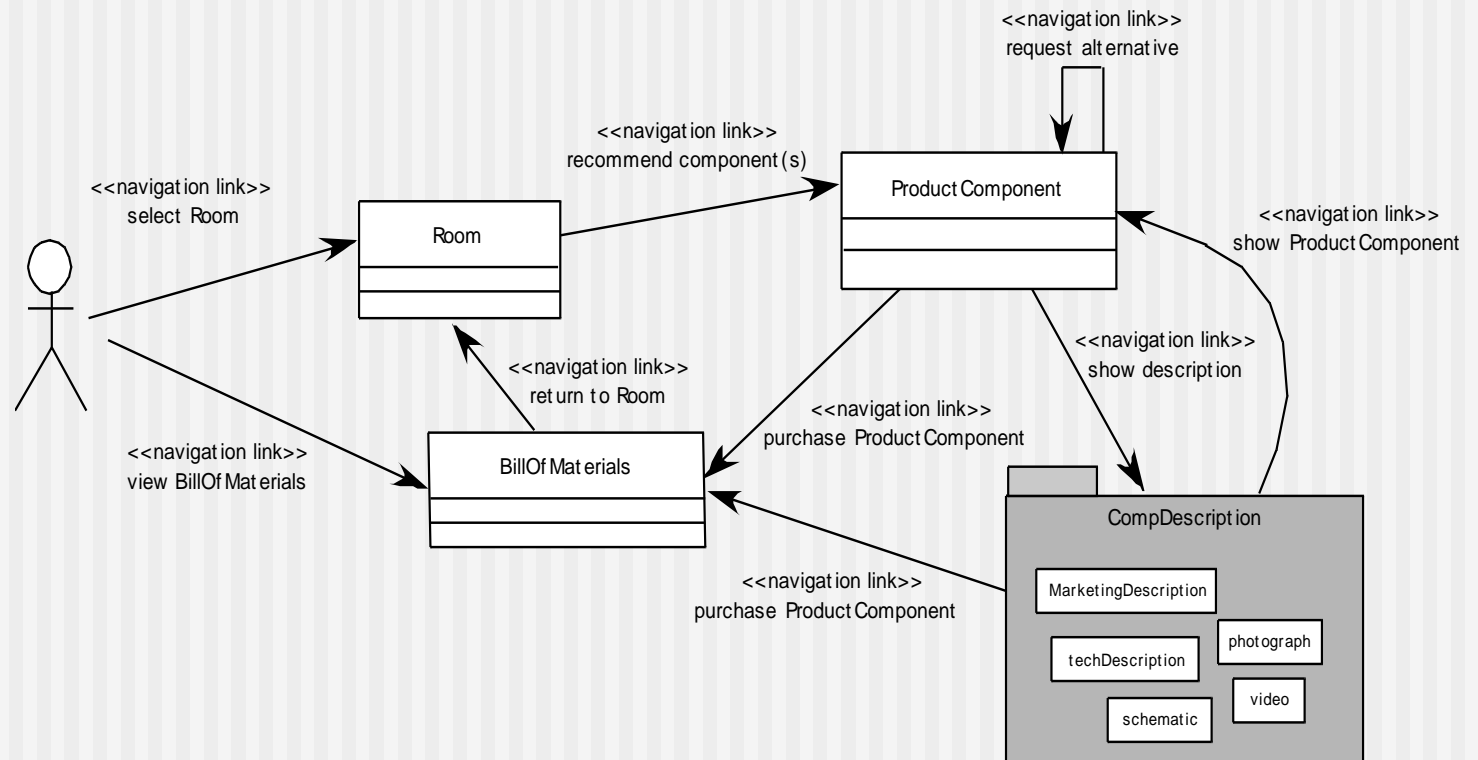
Navigation Semantic Units

- Navigation semantic unit

- Cách điều hướng — đại diện cho cách chuyển hướng tốt nhất để đạt được mục tiêu, gồm các nút điều hướng liên kết bởi các cạnh điều hướng



Tạo một NSU



Cú pháp điều hướng


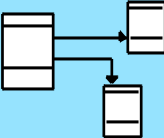
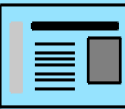
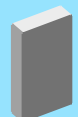
- *Liên kết điều hướng đơn* — liên kết dạng text, biểu tượng, nút, ...
- *Thanh điều hướng ngang* — danh sách các nội dung, chức năng có liên kết thích hợp (khoảng 4-7 loại)
- *Cột điều hướng dọc*
 - Liệt kê các nhóm nội dung, chức năng
 - Liệt kê tất cả các đối tượng, nội dung chính trong WebApp
- *Tabs* — một ẩn dụ tương tự như thanh/cột điều hướng, hiển thị trong một số tình huống
- *Site maps* — cung cấp một tab bao gồm tất cả các chuyển hướng đến các nội dung trong web

Thiết kế cấp thành phần

- Các thành phần của WebApp thực hiện chức năng:
 - Thực hiện xử lý địa phương, tạo nội dung, chuyển hướng
 - Cung cấp khả năng tính toán, xử lý dữ liệu
 - Cung cấp truy vấn cơ sở dữ liệu
 - Thiết lập giao diện dữ liệu với các hệ thống ngoài

OOHDM

■ *Object-Oriented Hypermedia Design Method (OOHDM)*

	 conceptual design	 navigational design	 abstract interface design	 implementation
work products	Classes, sub-systems, relationships, attributes	Nodes, links, access structures, navigational contexts, navigational transformations	Abstract interface objects, responses to external events, transformations	executable WebApp
design mechanisms	Classification, composition, aggregation, generalization, specialization	Mapping between conceptual and navigation objects	Mapping between navigation and perceptible objects	Resource provided by target environment
design concerns	Modeling semantics of the application domain	Takes into account user profile and task. Emphasis on cognitive aspects.	Modeling perceptible objects, implementing chosen metaphors. Describe interface for navigational objects	Correctness; Application performance; completeness

Lược đồ khái niệm

