

Chương 11

■ Thiết kế giao diện người dùng

Slide được xây dựng theo cuốn

Software Engineering: A Practitioner's Approach, 7/e
của **Roger S. Pressman**

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

Chỉ dành cho mục đích giáo dục phi lợi nhuận.

Sinh viên đại học có thể sao chép slide này nếu dùng chung với cuốn *Software Engineering: A Practitioner's Approach, 7/e*. Mọi hình vi tái bản hoặc sử dụng đều bị cấm khi chưa có sự cho phép của tác giả.

Tất cả các thông tin bản quyền phải xuất hiện trong các slide nếu xuất hiện trên trang web dành cho sinh viên.

Thiết kế giao diện

Dễ học?

Dễ sử dụng?

Dễ hiểu?



Thiết kế giao diện

Lỗi thiết kế thông thường

**thiếu nhất quán
quá nhiều ghi nhớ
không có hướng dẫn / giúp đỡ
không nhạy cảm với ngữ cảnh
đáp ứng kém
Phức tạp / không thân thiện**



Quy tắc vàng

- Đặt người dùng trong sự kiểm soát
- Giảm tải bộ nhớ cho người dùng
- Làm cho giao diện nhất quán



Đặt người dùng trong sự kiểm soát

- Xác định phương thức tương tác theo một cách mà không ép người dùng tới những hành động không cần thiết hoặc không mong muốn.
- Cung cấp sự tương tác linh hoạt.
- Cho phép tương tác người dùng được ngắt và hoàn tác .
- Hợp lý hóa tương tác như trình độ kỹ năng cao và cho phép tùy chỉnh tương tác.
- Ẩn kỹ thuật bên trong với người sử dụng bình thường.
- Thiết kế cho tương tác trực tiếp với các đối tượng xuất hiện trên màn hình.

Giảm tải bộ nhớ cho người dùng

- Giảm nhu cầu về bộ nhớ ngắn hạn.
- Thiết lập các mặc định có ý nghĩa.
- Xác định các phím tắt trực quan.
- Các thiết kế trực quan của giao diện phải được dựa trên một phép ẩn dụ thế giới thực.
- Tiết lộ thông tin theo kiểu lũy tiến.



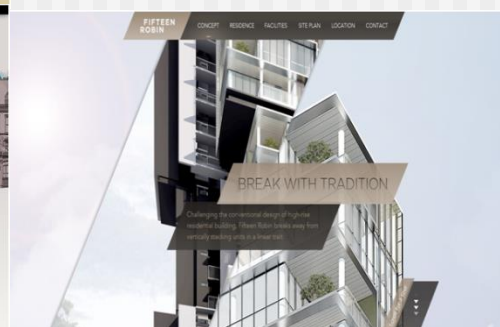
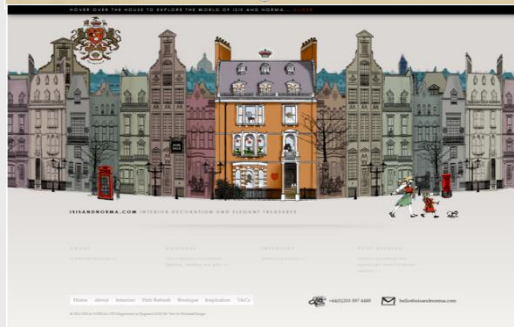
Làm cho giao diện nhất quán



Cho phép người sử dụng đưa các tác vụ hiện hành vào một ngữ cảnh có ý nghĩa.

Duy trì tính nhất quán giữa một họ các ứng dụng.

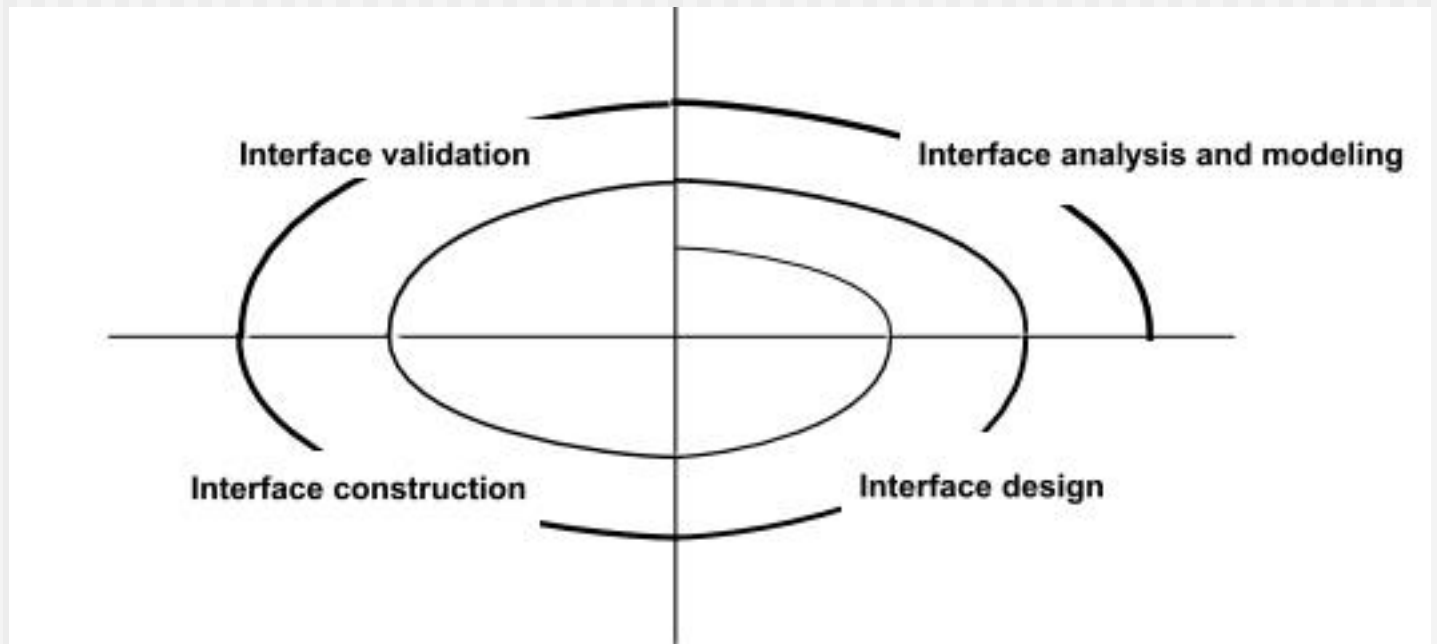
Nếu mô hình tương tác cũ đã tạo ra những kỳ vọng của người dùng, không làm thay đổi trừ khi có một lý do thuyết phục để làm như vậy.



Mô hình thiết kế giao diện người dùng

- **Mô hình người dùng** — một hồ sơ của tất cả người dùng cuối của hệ thống
- **Mô hình thiết kế** — một nhận thức thiết kế của các mô hình sử dụng
- **Mô hình về tinh thần (nhận thức hệ thống)** — hình ảnh nhận thức của người dùng về những gì giao diện là
- **Mô hình triển khai** — giao diện "nhìn và cảm nhận" cùng với thông tin hỗ trợ mô tả cú pháp và ngữ nghĩa giao diện

Quá trình thiết kế giao diện người dùng



Phân tích giao diện

- Phân tích giao diện nghĩa là hiểu được:
 - (1) những người (người dùng cuối), người sẽ tương tác với các hệ thống thông qua giao diện;
 - (2) các tác vụ mà người dùng phải thực hiện để làm công việc của họ,
 - (3) các nội dung được trình bày như là một phần của giao diện
 - (4) môi trường trong đó những công việc này sẽ được tiến hành.

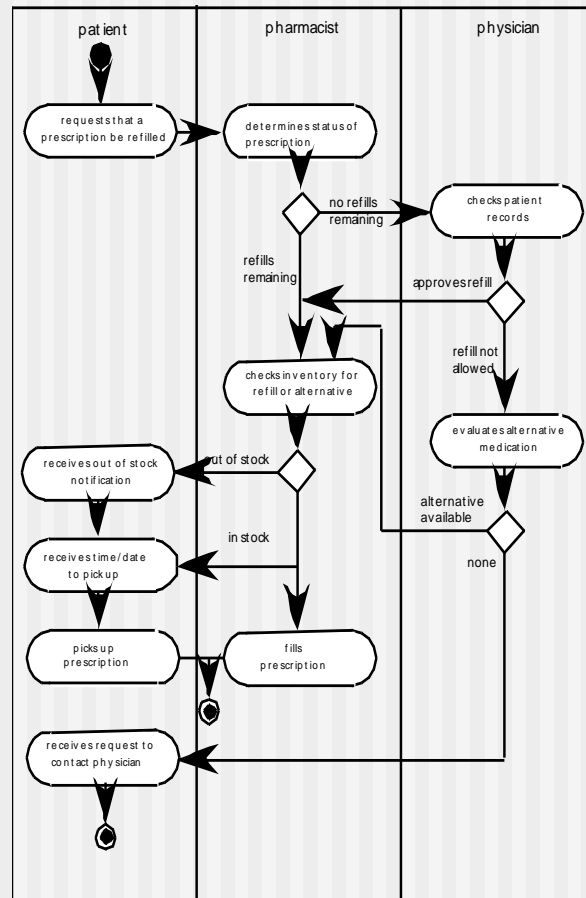
Phân tích người dùng

- Liệu người dùng là các chuyên gia đã qua đào tạo, kỹ thuật viên, văn thư hay là công nhân sản xuất?
- Mức độ giáo dục trung bình của người dùng là gì?
- Liệu người dùng có thể học từ những tài liệu viết hay họ bày tỏ mong muốn một chương trình luyện tập kiểu lớp học?
- Người dùng là những chuyên gia đánh máy hay họ sợ bàn phím?
- Độ tuổi của cộng đồng người dùng là bao nhiêu?
- Liệu người dùng sẽ được đại diện chủ yếu bởi một giới tính?
- Người dùng sẽ trả cho công việc họ thực hiện như thế nào?
- Liệu người dùng chỉ làm việc trong giờ công sở hay họ sẽ làm việc đến khi công việc hoàn thành?
- Liệu phần mềm sẽ trở thành một phần quan trọng trong công việc của người dùng hay nó sẽ chỉ thỉnh thoảng được sử dụng?
- Ngôn ngữ chính giữa các người dùng là gì?
- Sẽ có những hệ quả nào nếu người dùng gây ra lỗi khi sử dụng hệ thống?
- Liệu người dùng có phải là chuyên gia trong lĩnh vực liên quan được xử lý bởi hệ thống?
- Liệu người dùng có muốn biết về công nghệ phía sau giao diện?

Phân tích tác vụ và mô hình hóa

- Trả lời những câu hỏi sau...
 - Công việc gì mà người dùng sẽ thực hiện trong những trường hợp cụ thể?
 - Tác vụ hay tác vụ con nào sẽ được thực hiện khi người dùng làm việc?
 - Những vấn đề miền đối tượng cụ thể nào mà người dùng sẽ thao tác khi công việc được thực hiện?
 - Chuỗi các nhiệm vụ-quy trình là gì?
 - Hệ thống cấp bậc của tác vụ là gì?
- Use-cases xác định tương tác cơ bản.
- Xây dựng tác vụ điều chỉnh các nhiệm vụ tương tác.
- Xây dựng đối tượng xác định đối tượng giao diện (lớp)
- Phân tích quy trình làm việc xác định cách một quá trình làm việc được hoàn thành khi một số người (và vai trò) đều tham gia

Sơ đồ Swimlane



Phân tích nội dung hiển thị

- Liệu các loại dữ liệu khác nhau có được gán cho vị trí địa lý nhất định trên màn hình (ví dụ, hình ảnh luôn luôn xuất hiện ở góc trên bên phải)?
- Liệu người dùng có thể tùy chỉnh vị trí màn hình cho nội dung?
- Liệu các nhận dạng phù hợp có được gán cho tất cả nội dung?
- Nếu một báo cáo lớn được trình bày, nó sẽ được phân chia như thế nào cho dễ hiểu?
- Liệu cơ chế có sẵn sàng để di chuyển trực tiếp tới thông tin tóm tắt cho lượng dữ liệu lớn?
- Liệu các đầu ra đồ họa có được căn chỉnh để vừa vặn với các giới hạn của thiết bị hiển thị được sử dụng?
- Màu sắc sẽ được sử dụng như thế nào để tăng tính dễ hiểu?
- Thông báo lỗi và cảnh báo sẽ được trình bày tới người dùng như thế nào?

Các bước thiết kế giao diện

- Sử dụng thông tin được phát triển trong quá trình phân tích giao diện, **xác định đối tượng giao diện và hành động (hoạt động)**.
- **Xác định các sự kiện (các hành động người dùng)** sẽ gây ra sự thay đổi trạng thái của giao diện người dùng. Mô hình hóa hành vi này.
- **Miêu tả mỗi trạng thái giao diện** như thể nó sẽ thực sự tìm đến người dùng cuối.
- **Xác định cách người dùng diễn giải các trạng thái của hệ thống** từ thông tin được cung cấp thông qua giao diện.

Các vấn đề thiết kế

- Thời gian trả lời
- Trợ giúp các tiện nghi
- Xử lý lỗi
- Gắn nhãn menu và câu lệnh
- Khả năng tiếp cận ứng dụng
- Quốc tế hóa



ĐĂNG KÝ NHÃN HIỆU QUỐC TẾ

Thiết kế giao diện WebApp

- *Tôi đang ở đâu?* Giao diện nên
 - cung cấp một dấu hiệu rằng các WebApp đó đã được tiếp cận
 - thông báo cho người sử dụng vị trí của họ trong hệ thống phân cấp nội dung.
- *Tôi có thể làm gì bây giờ?* Giao diện phải luôn luôn giúp người dùng hiểu các tùy chọn hiện tại của mình
 - Những chức năng nào đang có sẵn?
 - Những liên kết nào còn hoạt động?
 - Những nội dung nào có liên quan?
- *Tôi đã từng ở đâu, tôi sẽ đi đâu?* Giao diện phải tạo điều kiện cho việc điều hướng
 - Cung cấp một "bản đồ" (được thực hiện một cách dễ hiểu) về nơi mà người sử dụng đã đi qua và những con đường có thể được thực hiện để chuyển đến nơi khác trong phạm vi WebApp.

Giao diện WebApp hiệu quả

- Bruce Tognozzi [TOG01] gợi ý rằng...
 - Giao diện hiệu quả rất trực quan rõ ràng và rộng rãi, mang lại cho người dùng cảm giác kiểm soát. Người dùng nhanh chóng nhận ra sự rộng rãi của các tùy chọn, nắm bắt cách để đạt được mục tiêu và làm công việc của họ.
 - Giao diện hiệu quả khiến người dùng không phải quan tâm đến hoạt động bên trong hệ thống. Công việc được lưu trữ cẩn thận và liên tục, với đầy đủ tùy chọn cho người dùng được hoàn tác bất kì hoạt động nào tại thời điểm bất kì.
 - Ứng dụng và dịch vụ hiệu quả thực hiện công việc lượng công việc tối đa, trong khi yêu cầu lượng thông tin tối thiểu.

Nguyên tắc thiết kế giao diện-I

- **Dự đoán**—Một WebApp nên được thiết kế sao cho nó dự đoán được động thái tiếp theo của việc sử dụng.
- **Truyền thông**—Giao diện nên truyền thông tình trạng của bất kỳ hoạt động khởi xướng bởi người sử dụng
- **Nhất quán**—việc sử dụng điều hướng, menu, biểu tượng, và thẩm mỹ (ví dụ: màu sắc, hình dạng, bố cục)
- **Tự điều khiển**—Giao diện điều khiển nên tạo điều kiện cho chuyển động sử dụng trong suốt WebApp, nhưng nó phải làm như vậy bằng cách thực hiện quy ước đã được thiết lập cho các ứng dụng.
- **Hiệu quả**—Thiết kế của WebApp và giao diện của nó nên tối ưu hóa hiệu quả công việc của người sử dụng, không phải của các kỹ sư thiết kế và xây dựng nó hoặc của các môi trường client-server thực hiện điều đó.

Nguyên tắc thiết kế giao diện-II

- **Tập trung**-Giao diện WebApp (và nội dung nó trình bày) nên tiếp tục tập trung vào các nhiệm vụ người dùng.
- **Luật Fitt** - "Thời gian để có được một mục tiêu là một hàm của khoảng cách đến và kích thước của các mục tiêu."
- **Đối tượng giao diện con người** - một thư viện tái sử dụng rộng lớn của các đối tượng giao diện con người đã được phát triển cho các ứng dụng web.
- **Giảm độ trễ**- WebApp nên sử dụng đa tác vụ bằng cách cho phép người sử dụng tiến hành với công việc nếu các hoạt động đã được hoàn thành.
- **Tính học được**- WebApp nên được thiết kế để giảm thiểu thời gian học tập, và một khi đã học, phải giảm thiểu sự học lại cần thiết khi WebApp được sử dụng lại.

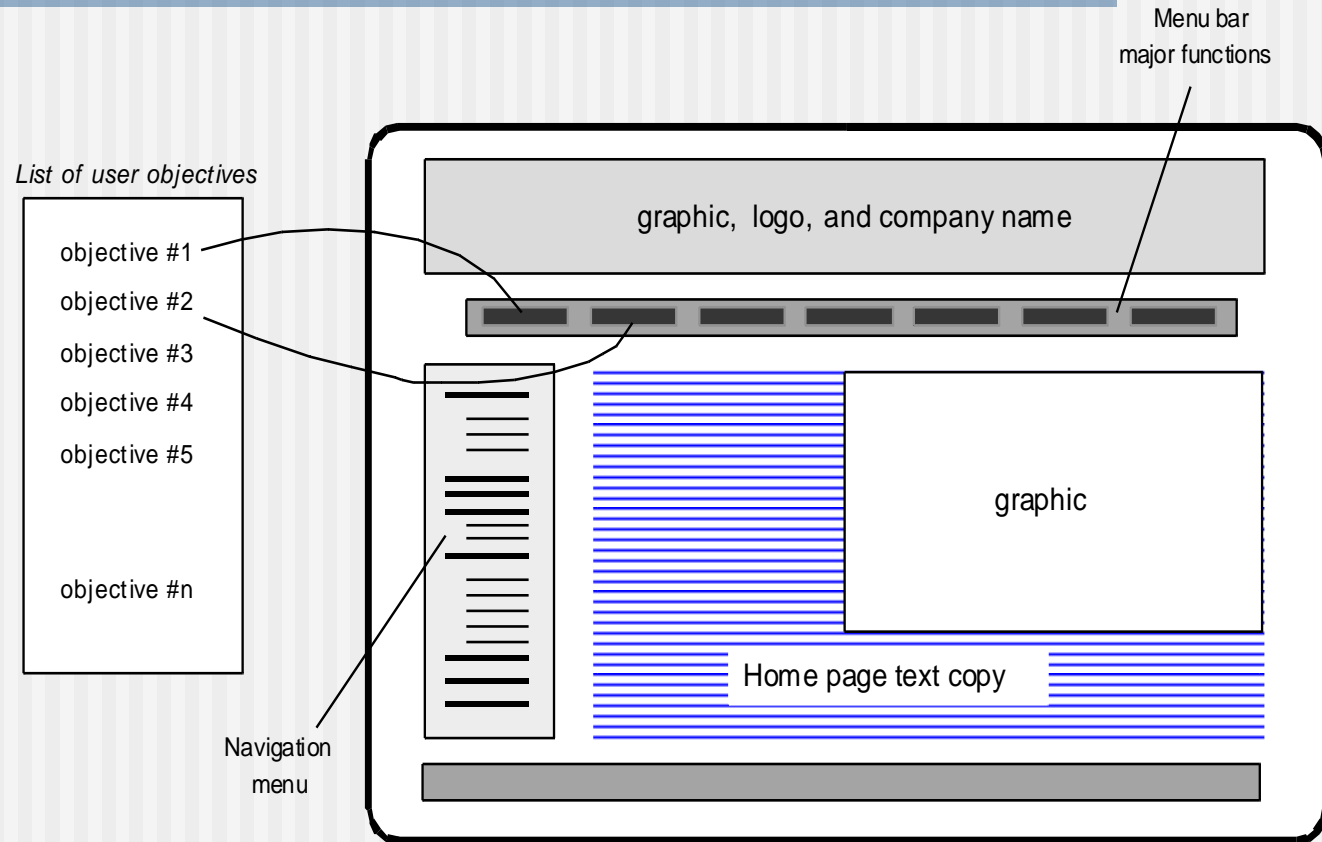
Nguyên tắc thiết kế giao diện-III

- **Duy trì tính toàn vẹn của sản phẩm công việc**—Một sản phẩm công việc (ví dụ: một biểu mẫu hoàn thành bởi người dùng, một danh sách chi tiết người dùng) cần được lưu trữ tự động để không bị mất khi có lỗi xảy ra.
- **Tính đọc được**—Tất cả thông tin trình bày qua giao diện cần phải đọc được bởi tất cả mọi người
- **Theo dõi trạng thái**—Lúc thích hợp, trạng thái của tương tác người dùng nên được theo dõi và lưu trữ để một người dùng có thể đăng xuất và sau đó trở lại đúng giao diện mà người đó đã rời đi.
- **Hiển thị điều hướng**—A giao diện WebApp được thiết kế tốt cung cấp "sự ảo tưởng rằng người sử dụng ở cùng một vị trí, với các công việc mang lại cho họ."

Quy trình thiết kế giao diện-I

- Xem lại thông tin chứa trong các mô hình phân tích và tinh chỉnh khi cần thiết.
- Xây dựng một phác thảo sơ bộ giao diện bố cục WebApp.
- Map các mục tiêu sử dụng vào các hoạt động giao diện cụ thể.
- Xác định một tập hợp các nhiệm vụ người dùng có liên quan đến từng hành động.
- Lên kịch bản hình ảnh màn hình cho mỗi hành động giao diện.
- Tinh chỉnh giao diện bố cục và kịch bản sử dụng đầu vào từ thiết kế thẩm mỹ.

Xác định mục tiêu người dùng



Quy trình thiết kế giao diện-II

- Xác định đối tượng giao diện người dùng được yêu cầu để thực hiện các giao diện. Xây dựng biểu diễn thủ tục của sự tương tác của người dùng với giao diện.
- Xây dựng một biểu diễn hành vi của giao diện.
- Mô tả bố cục giao diện cho mỗi trạng thái.
- Tinh chỉnh và xem xét các mô hình thiết kế giao diện.

Thiết kế thẩm mỹ

- Không ngại khoảng trắng
- Nhấn mạnh nội dung.
- Tổ chức các yếu tố định dạng từ góc trái trên đến phải dưới.
- Nhóm điều hướng, nội dung và chức năng địa lý trong trang.
- Không mở rộng thành phần bất động của bạn với thanh cuộn.
- Xem xét kích thước cửa sổ trình duyệt và độ phân giải khi thiết kế.

Chu kỳ đánh giá thiết kế

