

BÀI GIẢNG

NHẬP MÔN TƯỞNG TÁC NGƯỜI MÁY

GV: Đỗ Bảo Sơn

SĐT: 0163 233 6863

Mail: sondb@utt.edu.vn

Page: /sondo.utt

GIỚI THIỆU MÔN HỌC

1. Tên học phần: **NHẬP MÔN TƯƠNG TÁC NGƯỜI MÁY**

Mã học phần: DC3HT17

2. Số tín chỉ: 2

3. Đối tượng: SV ĐH chính quy năm 3

4. Phân bổ thời gian:

Tổng số: **30** tiết

- Lý thuyết & bài tập: 28 tiết;

- Kiểm tra: 2 tiết.

5. Mục tiêu học phần:

- Môn học cung cấp cho sinh viên các kiến thức cơ bản của lĩnh vực tương tác người máy: các nhân tố con người trong ứng dụng tương tác, các phương thức phát triển, truy cập giao diện tương tác, các vấn đề trong thiết kế và các kiểu tương tác.

- Sinh viên có khả năng vận dụng các mô hình, phương thức của lĩnh vực thiết kế, cài đặt, và đánh giá giao diện cho các ứng dụng tương tác HC.

GIỚI THIỆU MÔN HỌC

6. Mô tả vắn tắt nội dung học phần:

- Giới thiệu về lĩnh vực HCI
- Các mô hình, dạng thức tương tác
- Thiết kế tương tác người máy
- Đặc tả người dùng và phân tích nhiệm vụ
- Thiết kế giao diện người dùng
- Một số phương pháp đánh giá hệ thống
- Một số mô hình nhận thức

GIỚI THIỆU MÔN HỌC

7. Tài liệu học tập:

-Sách, giáo trình chính:

1. Lương Mạnh Bá (2005), *Tương tác người máy*, NXB Khoa học và kỹ thuật.

- Sách tham khảo:

1. Alan Dix, Janet E. Finlay, Gregory D. Abowd, Russell Beale, *Human-Computer Interaction*, Harlow, England: Prentice Hall, 2004
2. Jenifer Tidwell. *Designing Interfaces*, 2005
3. Jef Raskin. *Human Interface*, 2000
4. Caretta Software Ltd., *GUI Design Studio User Manual*, Version 2.3, March 2007.

GIỚI THIỆU MÔN HỌC

8. Nhiệm vụ của sinh viên:

- Nghiên cứu trước giáo trình, tài liệu học tập để thảo luận;
- Tham gia đầy đủ các giờ lên lớp và thảo luận;
- Hoàn thành các bài tập được giao đúng thời gian quy định;
- Tham gia kiểm tra giữa kỳ và thi kết thúc học phần.

9. Tiêu chuẩn đánh giá sinh viên:

- Điểm chuyên cần: 10% (≥ 5)
- Điểm kiểm tra giữa kỳ: 20%
- Điểm kiểm tra cuối kỳ: 70%

CHƯƠNG 1

GIỚI THIỆU TƯƠNG TÁC NGƯỜI MÁY

1.1. Định nghĩa

□ Có nhiều định nghĩa khác nhau:

■ Định nghĩa 1

□ *Tập các quá trình, đối thoại và các hành động, qua đó con người sử dụng và tương tác với máy tính (Backer & Buxton, 1987).*

■ Định nghĩa 2

□ *Là một lĩnh vực liên quan đến thiết kế, đánh giá và cài đặt Hệ thống máy tính tương tác cho con người sử dụng và nghiên cứu các hiện tượng chính xảy ra trên đó (ACMSIGCHI 1992).*

1.2. Vai trò của HCI

- Tạo ra các Hệ thống an toàn và *sử dụng được* (Usability) như các Hệ thống chức năng.

Usability: Là khái niệm trong HCI có thể hiểu là làm cho HT *dễ học* và *dễ dùng*.

1.3. Lịch sử phát triển

- Giai đoạn đầu của máy tính, UI không được xem trọng
- Đầu những năm 70: Hình thành khái niệm giao diện người máy (MMI - Man Machine Interface) sau đó thay đổi thành khái niệm UI
- Cuối những năm 70, đầu những năm 80: Xuất hiện khái niệm *''thân thiện người sử dụng''* dành cho các sản phẩm có UI tốt

1.3. Lịch sử phát triển

- ❑ Những năm 80: Xuất hiện khái niệm tương tác người máy (HCI - Human Computer Interaction). Xuất hiện trong MS Windows với GUI.
- ❑ Những năm 90 và những năm 2000: Có các nghiên cứu thực tại ảo, nhận dạng tiếng nói, nhận dạng chữ viết tay và ứng dụng chúng vào việc thiết kế vào/ra của HCI

1.4. Các lĩnh vực liên quan

- HCI nghiên cứu 3 phần:
 - Hình thức: Các hình thức giao tiếp giữa người và máy
 - Chức năng: Các chức năng mới trong giao tiếp người máy
 - Cài đặt: Cài đặt các giao diện

1.4. Các lĩnh vực liên quan

- ❑ HCI sử dụng tri thức của nhiều ngành khác nhau
 - Khoa học máy tính
 - Trí tuệ nhân tạo
 - Nhân loại học
 - Ngôn ngữ học
 - Triết học
 - Nghệ thuật
 - Điện, điện tử

1.4. Các lĩnh vực liên quan

- Xã hội học
- Thiết kế đồ họa, công nghiệp, âm thanh, điện ảnh,...
- Tâm lý học
 - Ứng dụng lý thuyết tiến trình nhận thức và phân tích theo kinh nghiệm ứng xử của người sử dụng
- Kỹ nghệ
- Sinh học
- Công thái học
 - Cải tiến thiết kế máy móc để con người dễ sử dụng

1.4. Tại sao phải nghiên cứu HCI

- ❑ UI là nơi giao tiếp giữa người dùng và máy tính. Không thể xâm nhập vào máy tính nếu không có UI.
- ❑ Trong lập trình:
 - Một phần lớn mã liên quan đến giao diện
 - Nếu thiết kế giao diện sai=> phải làm lại
 - Nếu không sửa được=> người dùng phải sử dụng giao diện không tốt
 - Thiết kế giao diện tốt => giảm thời gian lập trình

1.4. Tại sao phải nghiên cứu HCI

□ Về mặt kinh tế:

■ Tăng năng suất lao động

20 người dùng

x 230 ngày

x 100 màn hình giao tiếp 1 ngày

x 10 giây mỗi màn hình giao tiếp

= 1278 giờ(32 tuần)

1.4. Tại sao phải nghiên cứu HCI

- Tăng năng suất lao động
 - 5 người điều hành
 - x 500 lần chọn bảng một ngày
 - x 3 giây một lần chọn
 - x 230 ngày một năm

= 480 giờ (12 tuần)

1.4. Tại sao phải nghiên cứu HCI

- Giảm chi phí đào tạo

20 nhân viên

x 2 ứng dụng mỗi năm

x 2.5 ngày mỗi ứng dụng

= 100 ngày (20 tuần)

1.4. Tại sao phải nghiên cứu HCI

- Giảm những lỗi người dùng
500 người dùng
x 20 lỗi một năm
x 15 phút cho một lỗi

= mất 2500 giờ (63 tuần)

1.4. Tại sao phải nghiên cứu HCI

- Người sử dụng hài lòng
- Tạo ra những sản phẩm có chất lượng cao hơn. Người dùng tiết kiệm thời gian khi sử dụng giao diện nên có thể tập trung vào công việc chính

Ví dụ: tìm kiếm dữ liệu, định dạng văn bản....

1.4. Tại sao phải nghiên cứu HCI

- ❑ Giảm lỗi nguy hiểm đến tính mạng con người
 - Hệ thống liệu pháp bức xạ chữa bệnh ung thư Therac-25 đã
 - gây chết người do có UI tồi
 - Hệ thống rada Aegis trên tàu chiến USS Vincennes đã bắn
 - nhầm máy bay dân sự của Iran do có UI thiết kế tồi
 - Lá phiếu bầu cử tổng thống Mỹ 2000 gây tranh cãi do có thiết kế tồi (*Presidential Voting Ballot*, www.taberbrains.com)
 - Một chiếc máy bay đâm vào vách núi ở Colombia năm 1996 giết chết tất cả mọi người trên máy bay. Lý do người lái gõ phím “R” thay vì tên đầy đủ của sân bay. Hệ thống dẫn đường lấy ra trong hệ thống sân bay đầu tiên bắt đầu bằng chữ “R” → sai sân bay sân bay → đâm vào núi

1.4. Tại sao phải nghiên cứu HCI

- ❑ Tăng khả năng bán được của sản phẩm
 - DOS không thể so sánh được với các hệ điều hành khác cùng thời
 - Windows và Explorer đem lại cho Microsoft lợi nhuận cực lớn
 - Windows được sao chép lại từ giao diện của Macintosh !!!
 - Giao diện của Macintosh được sao chép lại từ Bravo – phát triển tại Xerox PARC !!!
- ❑ Giao diện đẹp dễ nhận được hợp đồng
- ❑ Giao diện tồi có thể bị loại ngay từ đầu cho dù chương trình tốt đến mấy

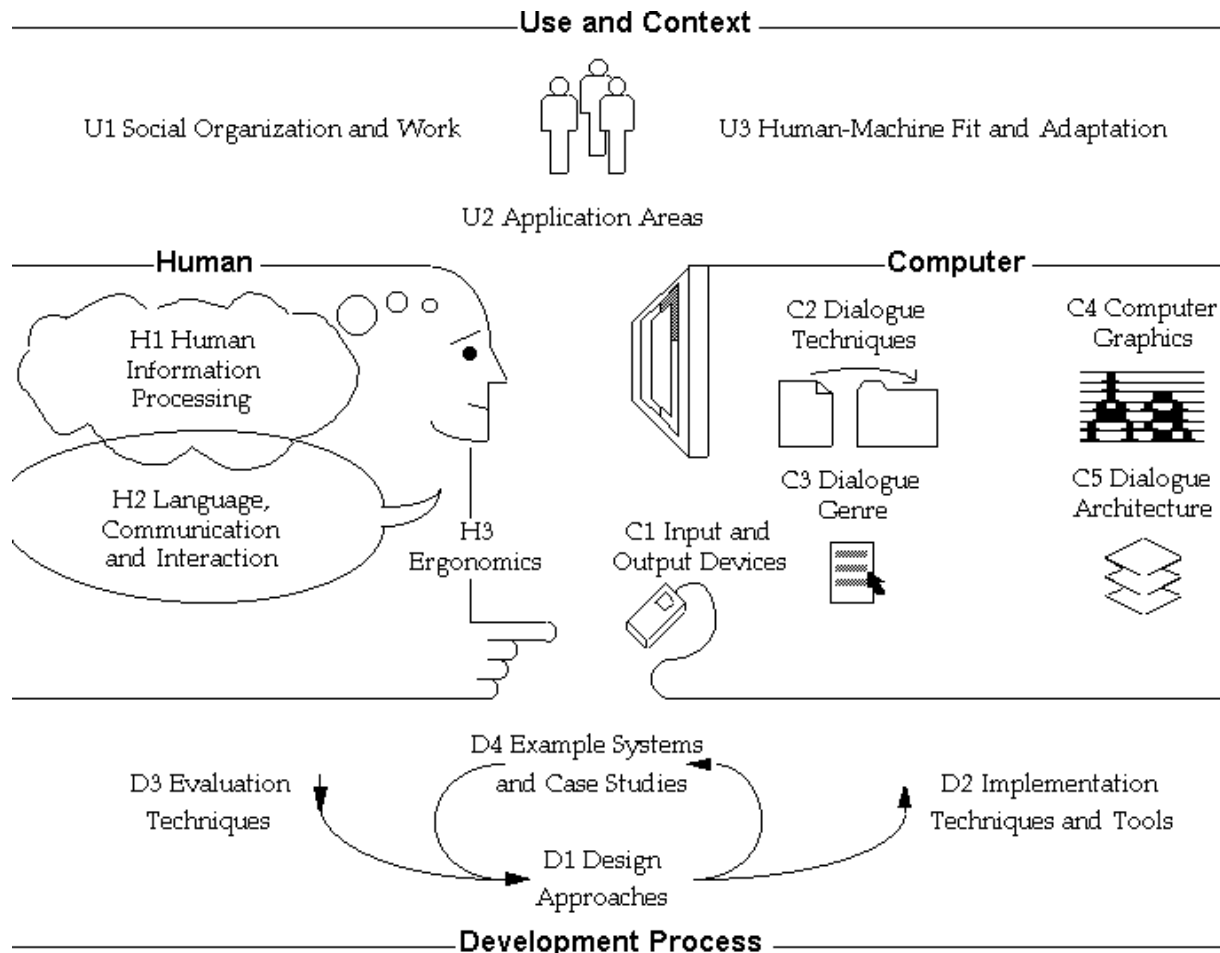
1.4. Tại sao phải nghiên cứu HCI

- ❑ Máy tính đã xuất hiện khắp mọi nơi: điều khiển máy bay, ô tô, dàn nghe nhạc
- ❑ Giao diện người – máy tính tốt => giao diện người –các thiết bị tốt

1.5. Nội dung nghiên cứu của HCI

- Gồm 4 thành phần chủ yếu:
 - Môi trường
 - Con người
 - Máy tính
 - Quá trình phát triển

1.5. Nội dung nghiên cứu của HCI



1.5. Nội dung nghiên cứu của HCI

- ❑ Sự sử dụng ngữ cảnh ứng dụng (Use and Context)
 - Tổ chức xã hội loài người (Human Social Organization): Liên quan đến việc xem loài người như thực thể xã hội tương tác:
 - ❑ Mô hình hoạt động con người
 - ❑ Mô hình nhóm và tổ chức
 - ❑ Mô hình công việc và luồng công việc
 - ❑ Hệ thống kỹ thuật – xã hội

1.5. Nội dung nghiên cứu của HCI

- ❑ Sự sử dụng ngữ cảnh ứng dụng (Use and Context)
 - Các lĩnh vực ứng dụng (Application Areas): Tập trung nghiên cứu vào các lớp của lĩnh vực ứng dụng từ góc độ mà giao diện cần phải đáp ứng
 - ❑ Đặc trưng của các miền ứng dụng: ứng dụng riêng hay nhóm
 - ❑ Giao diện hướng tài liệu: soạn thảo văn bản, bảng tính,...
 - ❑ Giao diện hướng truyền tin: Email, hội nghị truyền hình, điện thoại,...
 - ❑ Môi trường thiết kế: IDE, CAD/CAM
 - ❑ Các hệ thống học trực tuyến, trợ giúp
 - ❑ Các hệ thống điều khiển: điều khiển quá trình, games
 - ❑ Các hệ thống nhúng: điều khiển máy sao chụp, thang máy,...

1.5. Nội dung nghiên cứu của HCI

- Sự sử dụng ngữ cảnh ứng dụng (Use and Context)
 - Sự phù hợp người – máy (Human Machine Fit): Thiết kế để đạt tới sự phù hợp giữa các đối tượng thiết kế và việc sử dụng chúng
 - Các kỹ thuật để đạt tới sự phù hợp
 - Lựa chọn và thích nghi hệ thống
 - Lựa chọn và thích nghi người sử dụng: Sự tương thích giữa đặc điểm người sử dụng và hệ thống, tính dễ sử dụng, dễ huấn luyện
 - Hướng dẫn người sử dụng: tài liệu hướng dẫn, tài liệu kỹ thuật và tài liệu quản lý lỗi

1.5. Nội dung nghiên cứu của HCI

□ Các đặc trưng con người (Human)

- Nghiên cứu các tính chất về quá trình xử lý thông tin của con người, cấu trúc hành động, bản chất giao tiếp và yêu cầu về vật lý, sinh lý học của con người
- Xử lý thông tin của con người (Human Information Processing): Đặc trưng của con người như bộ xử lý thông tin
 - Mô hình nhận thức
 - Mô hình khái niệm
 - Hiện tượng và bản chất nhớ, cảm nhận, vận động, học, giải quyết vấn đề, thu thập kỹ năng, ...

1.5. Nội dung nghiên cứu của HCI

□ Các đặc trưng con người (Human)

- Ngôn ngữ giao tiếp và tương tác (Language, Communication)
 - Các khía cạnh của ngôn ngữ: Cú pháp, ngữ nghĩa,...
 - Mô hình ngôn ngữ
 - Các ngôn ngữ chuyên dụng: Truy vấn, giao tiếp đồ họa, lệnh,...
- Nghiên cứu về lao động (ergonomics): Đặc điểm nhân trắc học, sinh lý học của con người và quan hệ giữa con người với môi trường, không gian làm việc
 - Thiết kế không gian làm việc
 - Bố trí màn hình, các điều khiển
 - Giới hạn của nhận thức, cảm biến

1.5. Nội dung nghiên cứu của HCI

- Hệ thống máy tính và kiến trúc giao diện (Computer)
 - Các thiết bị hỗ trợ vào ra (I/O Devices): Kỹ thuật xây dựng
 - các kỹ thuật hỗ trợ giao tiếp giữa người – máy
 - Các loại thiết bị vào/ra máy tính
 - Tính chất của thiết bị vào/ra: trọng lượng, băng thông,...
 - Thiết bị ảo

1.5. Nội dung nghiên cứu của HCI

- ❑ Hệ thống máy tính và kiến trúc giao diện (Computer)
 - Kỹ thuật đối thoại (Dialogue Techniques): Kiến trúc phần
 - mềm cơ sở và kỹ thuật để tương tác với người sử dụng
 - ❑ Các kỹ thuật đối thoại vào: Kỹ thuật sử dụng bàn phím (lệnh, menu), kỹ thuật sử dụng chuột (nhấp chuột), kỹ thuật sử dụng bút (nhận dạng ký tự, điệu bộ), kỹ thuật sử dụng tiếng nói,...
 - ❑ Các kỹ thuật đối thoại ra: Trượt màn hình, cửa sổ, hoạt hình,...
 - ❑ Các kỹ thuật đối thoại tương tác: Kỹ thuật chữ số, điền form, chọn lệnh trong menu, biểu tượng, ngôn ngữ tự nhiên. Các kỹ thuật dẫn đường. Quản lý lỗi

1.5. Nội dung nghiên cứu của HCI

- ❑ Hệ thống máy tính và kiến trúc giao diện (Computer)
 - Ẩn dụ giao tiếp (Interface Metaphor)
 - ❑ Ẩn dụ tương tác
 - ❑ Ẩn dụ nội dung
 - Thiết kế đồ họa (Graphic Design):
 - ❑ Hình học 2D, 3D, biến đổi hình học
 - ❑ Biểu diễn thuộc tính của đối tượng đồ họa
 - ❑ Mô hình hóa vật thể,...

1.5. Nội dung nghiên cứu của HCI

- ❑ Tiến trình phát triển (Development Process)
 - Các tiệm cận thiết kế (Design Approaches): Tiến trình thiết kế
 - ❑ Cơ sở thiết kế đồ họa: Ngôn ngữ thiết kế, kỹ thuật in, sử dụng màu, tổ chức không gian 2D và 3D, trình tự thời gian,...
 - ❑ Kỹ thuật phân tích nhiệm vụ, định vị nhiệm vụ
 - ❑ Kỹ thuật đặc tả thiết kế
 - ❑ Kỹ thuật phân tích thiết kế: Đối tượng, hành động
 - ❑ Thiết kế mẫu (Case Study)

1.5. Nội dung nghiên cứu của HCI

- ❑ Tiến trình phát triển (Development Process)
 - Kỹ thuật và công cụ cài đặt (Implementation Techniques and Tools): Thủ thuật và công cụ cài đặt phần mềm
 - ❑ Quan hệ giữa thiết kế, đánh giá và cài đặt
 - ❑ Các kỹ thuật làm bản mẫu: Sản xuất phim hoạt hình, phần mềm máy tính,...
 - ❑ Các công cụ phần mềm phát triển đối thoại: GUI Design Studio,...
 - ❑ Phương pháp hướng đối tượng
 - ❑ Trình diễn dữ liệu và các thuật toán

1.5. Nội dung nghiên cứu của HCI

- Tiến trình phát triển (Development Process)
 - Các kỹ thuật đánh giá (Evaluation Techniques)
 - Các thước đo đánh giá (thời gian, lỗi, tính học được,...)
 - Kỹ thuật kiểm thử của tính sử dụng được

1.6. Các yếu tố cần quan tâm

- Khi thiết kế HCI cần quan tâm đến các yếu tố
 - Các yếu tố tổ chức: Huấn luyện, thiết kế công việc, chính sách, vai trò và tổ chức công việc
 - Các yếu tố môi trường (nhiều, nóng, chiếu sáng,...)
 - Yếu tố về sức khỏe và an toàn: stress, đau đầu, mỏi mắt, cơ – xương,...
 - Người sử dụng: Động cơ, thỏa mãn, thích thú, mức độ kinh nghiệm,...
 - Yếu tố thuận tiện: Tư thế ngồi, bố trí thiết bị,...

1.6. Các yếu tố cần quan tâm

- ❑ Khi thiết kế HCI cần quan tâm đến các yếu tố
 - Giao diện sử dụng: Thiết bị vào ra, cấu trúc đối thoại, sử dụng màu sắc, biểu tượng, lệnh, đồ họa, ngôn ngữ tự nhiên, 3D, đa phương tiện
 - Yếu tố nhiệm vụ: Dễ, phức tạp, mới, phân bổ công việc, tính lặp, kỹ năng,...
 - Các ràng buộc: Giá cả, thời gian, nguồn kinh phí, đội ngũ, thiết bị,...
 - Chức năng hệ thống: Phần cứng, phần mềm, ứng dụng
 - Yếu tố năng suất: Tăng đầu ra, tăng chất lượng, giảm giá thành, giảm lỗi, giảm yêu cầu lao động, giảm thời gian chế tạo

1.7. Ai tham gia xây dựng HCI

- ❑ Chuyên gia thiết kế đồ họa
- ❑ Người thiết kế tương tác/giao diện
- ❑ Người làm tài liệu kỹ thuật
- ❑ Nhân viên tiếp thị
- ❑ Kỹ sư kiểm thử
- ❑ Kỹ sư phần mềm

CHƯƠNG 2

TÍNH SỬ DỤNG CỦA HỆ THỐNG TƯƠNG TÁC

2.1. Định nghĩa tính sử dụng

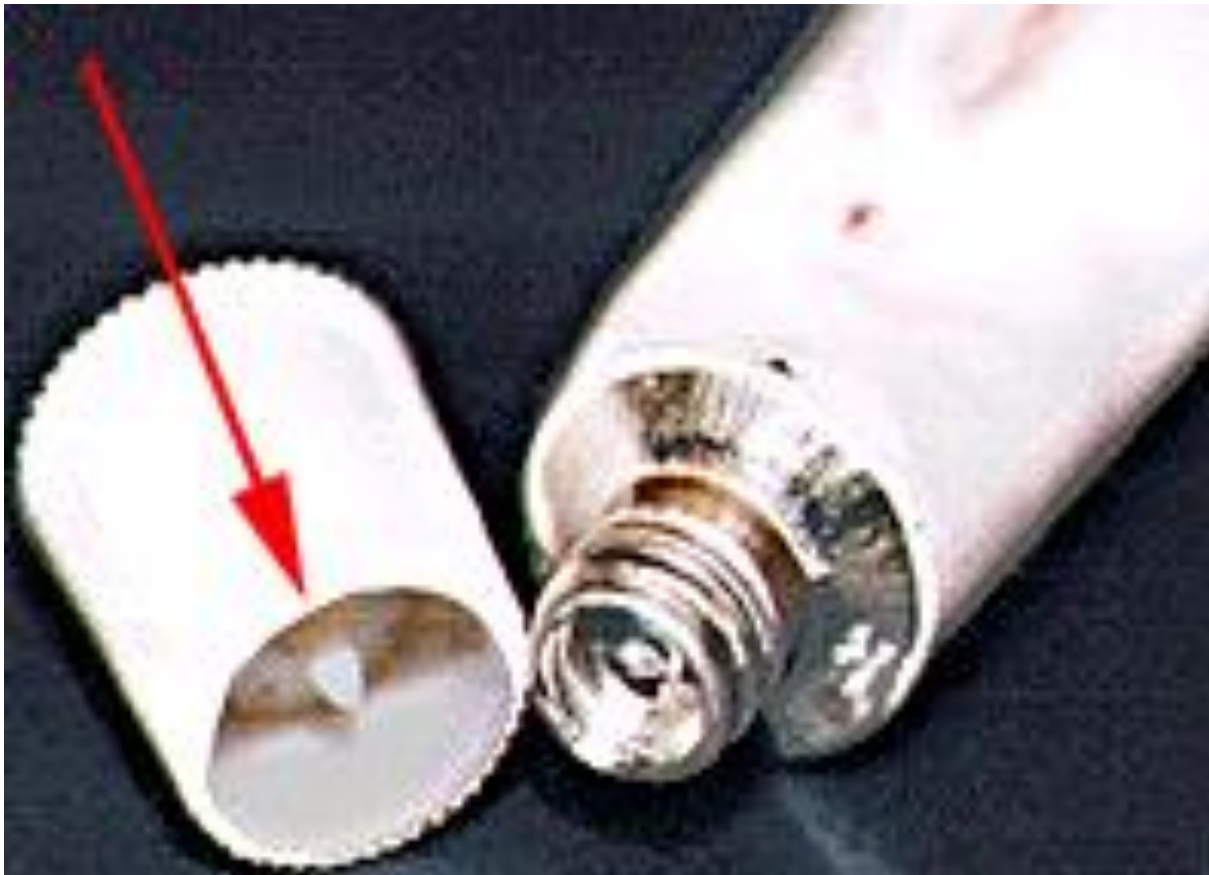
- ❑ Khả năng hệ thống được sử dụng bởi con người một cách dễ dàng và hiệu quả
- ❑ Ba tính chất của tính sử dụng
 - Hiệu quả: Tính chính xác và đầy đủ mà với nó người sử dụng đạt được mục tiêu xác định trước
 - Năng suất: Tài nguyên cần sử dụng để đạt được tính chính xác và đầy đủ mà với nó người sử dụng đạt được mục tiêu xác định trước
 - Thỏa mãn: Không bức dọc, lo lắng và có quan điểm tích cực với việc sử dụng sản phẩm

2.1. Định nghĩa tính sử dụng

- ❑ 10 vi phạm ảnh hưởng đến tính sử dụng
 1. Menu và biểu tượng nhập nhằng
 2. Ngôn ngữ chỉ cho phép đi theo một hướng trong hệ thống
 3. Hạn chế đầu vào và thao tác trực tiếp
 4. Hạn chế lựa chọn và điểm nổi bật
 5. Trình tự các bước không rõ ràng
 6. Nhiều bước quản lý giao diện hơn là thực hiện nhiệm vụ
 7. Liên kết phức tạp với các ứng dụng khác và giữa các ứng dụng
 8. Phản hồi và khẳng định không phù hợp
 9. Hệ thống kém đề phòng và kém thông minh
 10. Các thông điệp lỗi, trợ giúp, tài liệu không phù hợp

2.1. Định nghĩa tính sử dụng

- ❑ Lấy kem đánh răng bằng cách nào



2.1. Định nghĩa tính sử dụng

□ Hộp không có tay cầm



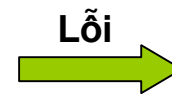
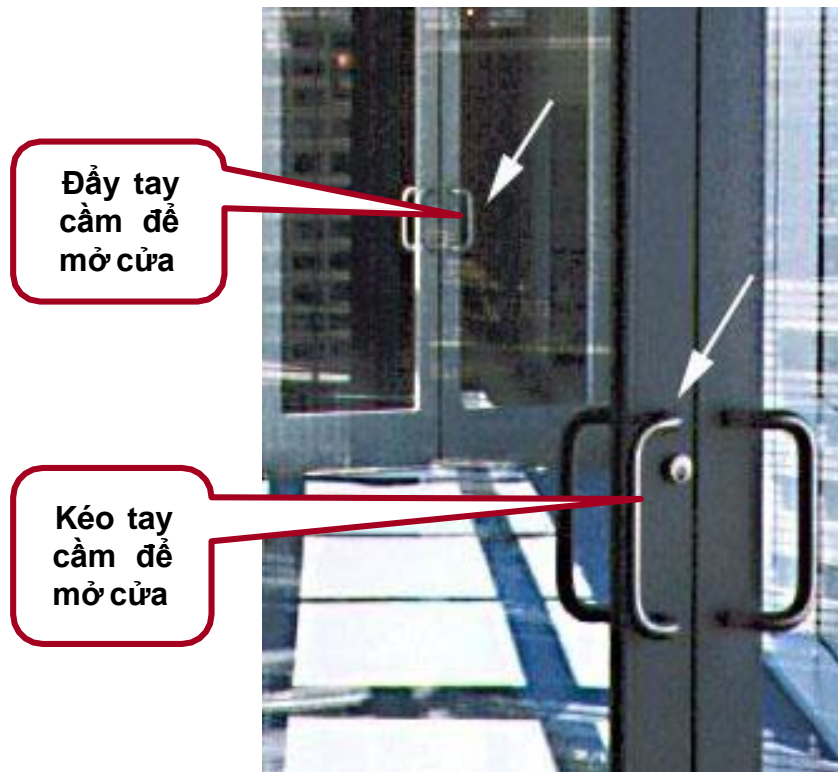
Để cầm vào nắp để nhấc hộp
lên và bị đổ bột



Sẽ nâng hộp lên bằng hai tay

2.1. Định nghĩa tính sử dụng

❑ Cửa thiết kế khó sử dụng



Lỗi

1. Tay cầm dùng để kéo, không phải để đẩy
2. Hai cửa làm việc không giống nhau

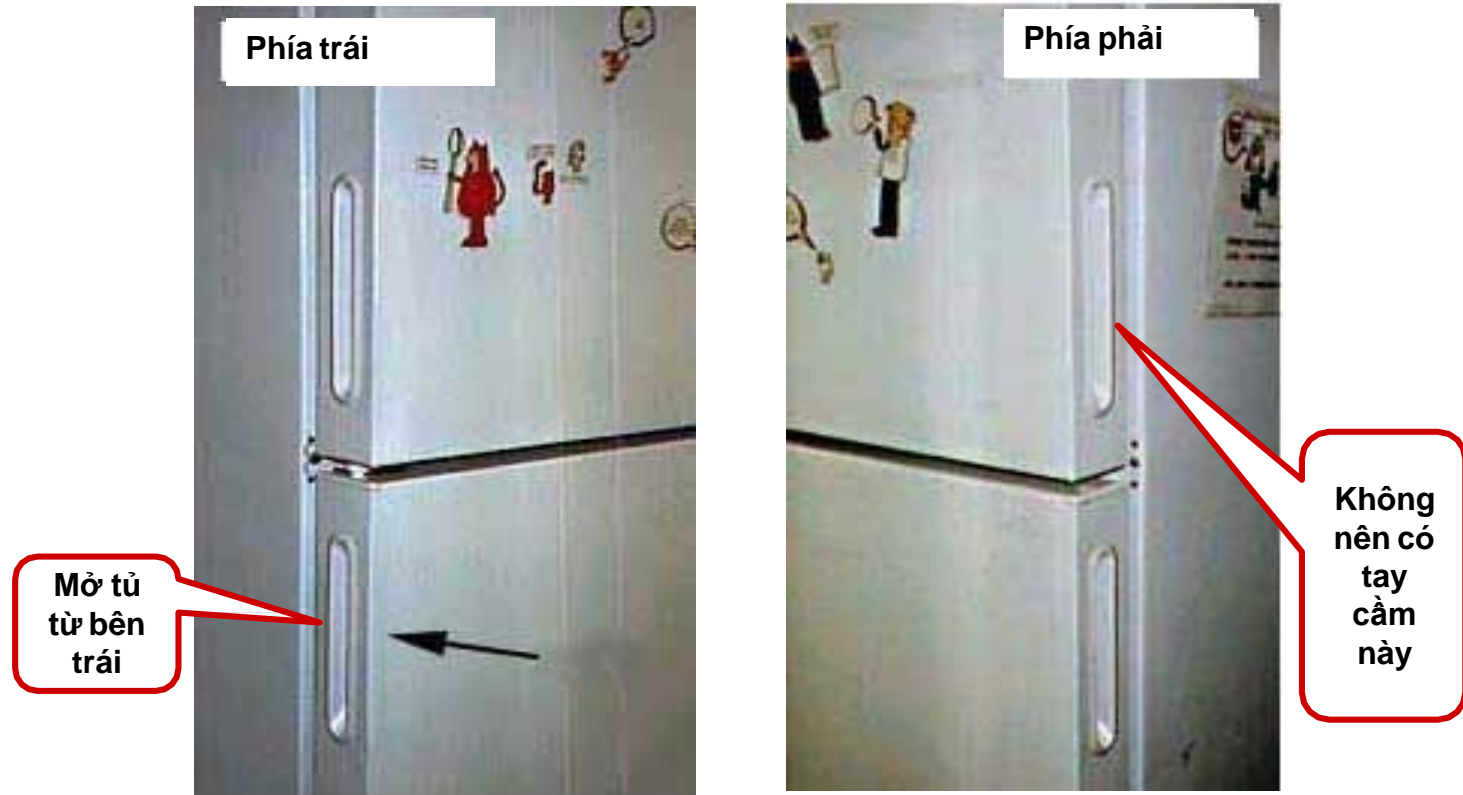


Giải pháp

1. Tay cầm dùng để đẩy phẳng dễ đẩy và khó kéo
2. Các cửa phải làm việc giống nhau

2.1. Định nghĩa tính sử dụng

- ❑ Làm thế nào để mở tủ lạnh



2.1. Định nghĩa tính sử dụng

- Chai nhìn giống nhau nhưng chứa các chất khác nhau



Dầu gội
đầu

Dầu dưỡng
tóc



Insulin R

Insulin N

2.1. Định nghĩa tính sử dụng

❑ Đèn màu vàng trên đường



Đèn giao thông đang có màu xanh



Không phân biệt được đèn giao thông khi màu vàng và hệ đèn đường

➡ Tai nạn ?

Khắc
phục

Làm nổi
bật đèn
giao
thông

2.1. Định nghĩa tính sử dụng

- Mở cửa theo cách nào



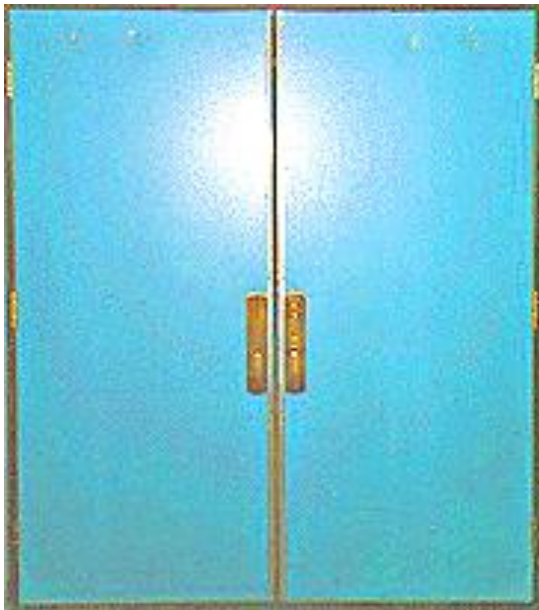
2.1. Định nghĩa tính sử dụng

- Đường giao nhau này sai gì?



2.1. Định nghĩa tính sử dụng

□ Cửa trong trường đại học



Cửa không có cửa sổ



Nên thiết kế có cửa sổ

2.1. Định nghĩa tính sử dụng

□ Bút chì



Bút chì có nắp bảo vệ ngòi khó dùng



Bút chì mà ngòi có thể thụt vào trong ống dễ dùng hơn

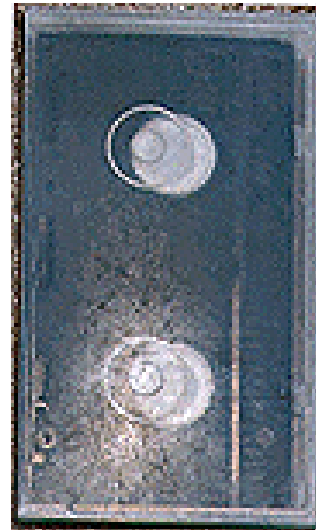
2.1. Định nghĩa tính sử dụng

□ Băng video



Nên thiết kế phẳng, không có phần trục này

Băng được bảo vệ bởi hộp nhựa



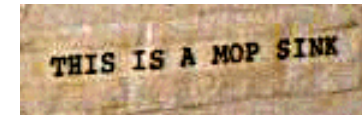
Băng video được đặt không đúng



Băng video được đặt đúng

2.1. Định nghĩa tính sử dụng

□ Bồn rửa còn sót lại

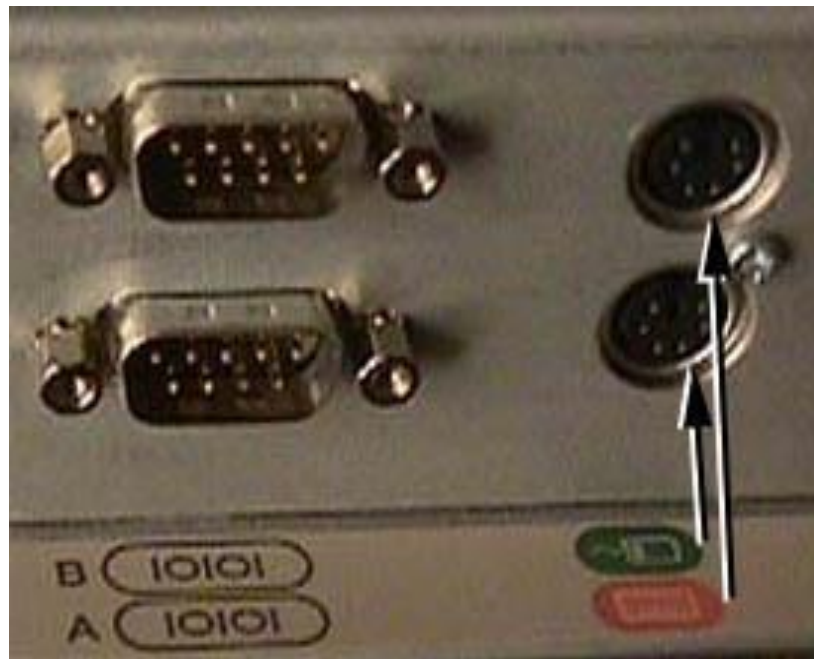


Biển chỉ dẫn
không rõ ràng

Bồn rửa sót lại
giống chỗ đi
tiểu

2.1. Định nghĩa tính sử dụng

❑ Cắm chuột vào đâu?



Cổng chuột và bàn phím khá giống nhau

Nhãn màu xanh ứng
với cổng nào ?

Đề xuất



Nhãn đi
cùng
với đối
tượng



Dùng
màu
tương
ứng
với đối
tượng

2.1. Định nghĩa tính sử dụng

□ Cái điều khiển ti vi



Nhằm chữ V
với mũi tên
xuống

Đề xuất

Thay chữ “V”
bằng chữ “vol”

2.1. Định nghĩa tính sử dụng

□ Vòi nước ở Zion National Park

Vòi nước ở trạng thái khóa



Nhằm là cần bơm



Vòi nước ở trạng thái mở

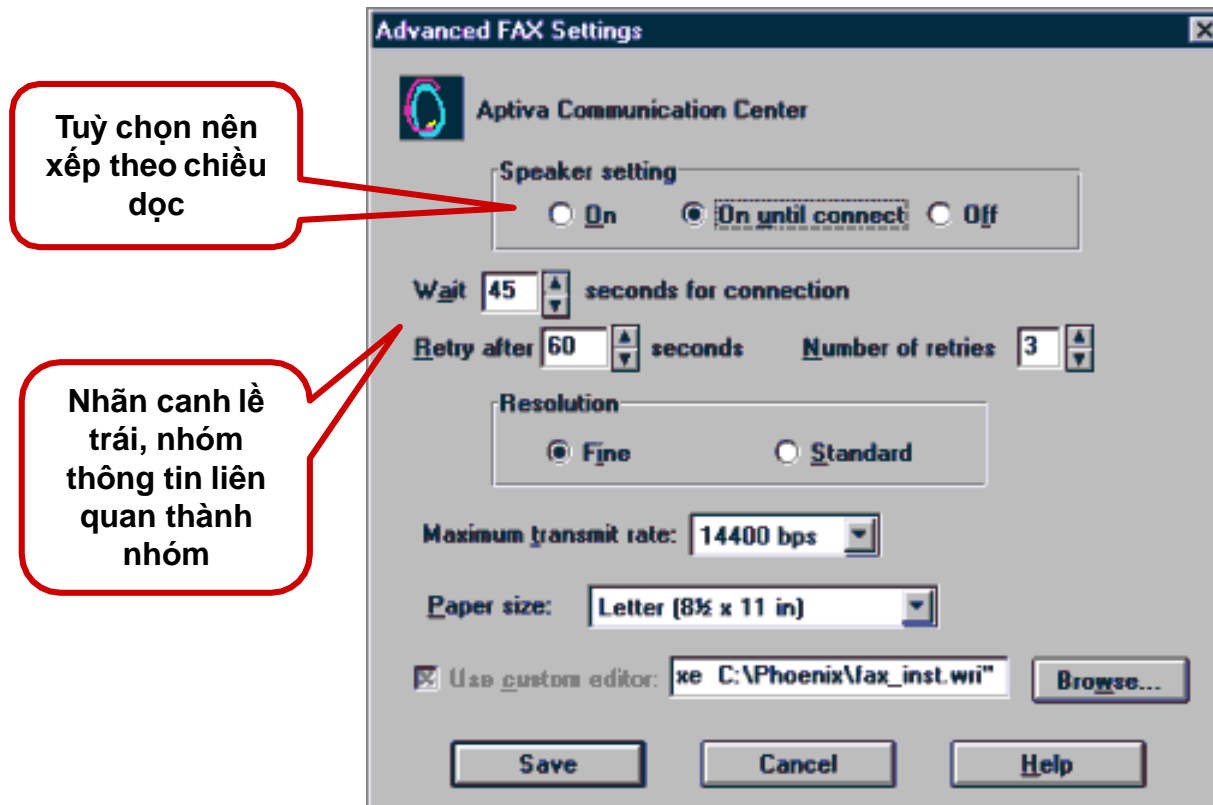


Đề xuất thiết kế



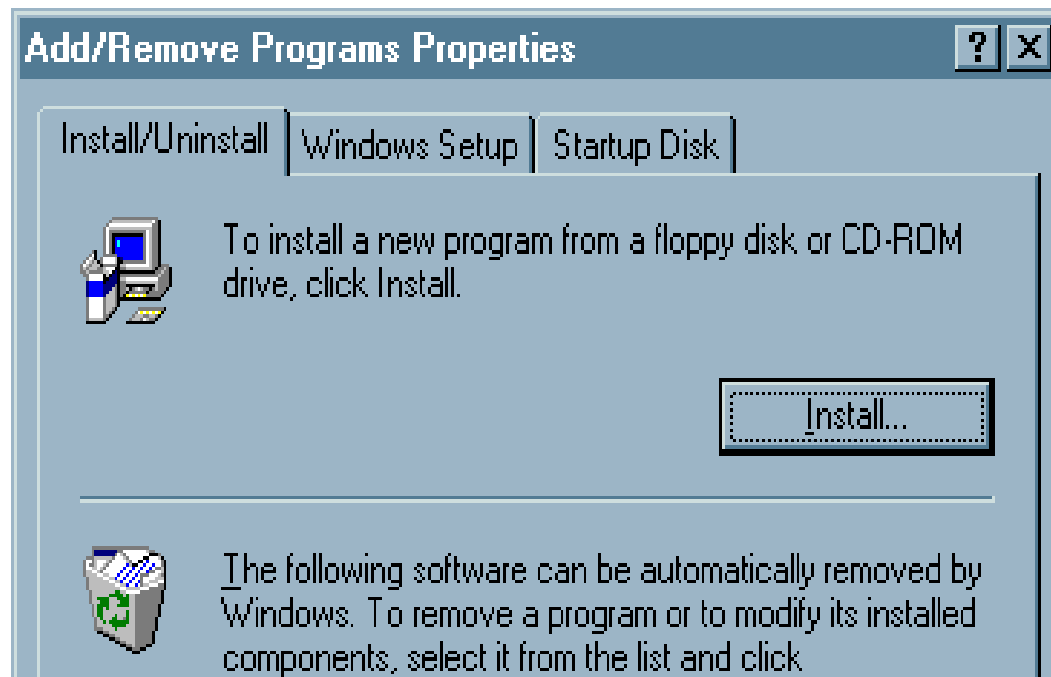
2.1. Định nghĩa tính sử dụng

□ Giao diện thiết lập máy Fax



2.1. Định nghĩa tính sử dụng

- Thêm/Loại bỏ chương trình trong Windows 9X



Các tab không liên quan với nhau

2.1. Định nghĩa tính sử dụng

□ Giao diện nhập thông tin bệnh nhân

Patient Account

Company 002	Invoice AK-3356	S.S.N. 044-52-8995	Trip Date 6/12/92	Operator T. Michaels	Date 6/15/92
Last Name Carlson		First Melissa	Medicare A 74493021	Medi-Cal	

(A) Patient's Address:

71194 El Camino Real
San Jose, CA 92344
Home: (714) 688-5433
Work:

(B) Birthday: 3/5/34
Sex: F
Sig/File:
POE:

(C) Charge To:
SHP Senior Plan
P.O. Box 3382
San Diego, CA 92445

(B) Physician:
Marshall
4550 Stetson Dr.
(409) 448-3922

From Location:
445 Elm St.
Sonoma

To Location:
General Hospital
Sonoma
Admit: C

(D) Diagnosis:
0054 Syncope Episode
9000 Head injury

Billing:
000 Private
Charge 345.90
Paid 290.00
Adjust -50.00
Balance -5.90

(E) Employer:

Caller:
Family
(405) 934-5596

(R) Reason...
Transport: 1
Ambulance: 5
Update

Flwd Disp Arrive Comp
03:45 03:45 03:53 04:13

R.T. (U) Auth: 9333
Inst Billing ID:

Notes
Payments
Insurance
Letters
Billing
Trans
Clear

2.1. Định nghĩa tính sử dụng

- Không thu nhỏ được Text của Microsoft Paint
- Microsoft Paint không tự động cuộn cửa sổ
- Phím left, right chức năng giống với các phím page up, page down trong Acrobat

2.2. Sáu thuộc tính của tính sử dụng

- ❑ Hiệu quả: Tính chính xác và đầy đủ mà với nó người sử dụng đạt được mục tiêu định trước
- ❑ Tính học được: Hệ thống có dễ học không?
- ❑ Năng suất: Một khi đã dễ học, có nhanh sử dụng không?
- ❑ Tính nhớ được: Có dễ nhớ những gì đã học?
- ❑ Các lỗi: Ít lỗi và dễ vượt qua lỗi?
- ❑ Thỏa mãn mục đích: Có thích thú sử dụng hệ thống?

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

- Sự rõ ràng
 - Phần hệ thống liên quan phải được nhìn thấy
 - Người sử dụng nhận biết trạng thái hiện hành của hệ thống và cần biết phải thực hiện thao tác nào
 - VD: Khi di chuột đến điểm bất kỳ trên màn hình phải biết được điều gì sẽ xảy ra nếu nhấn phím chuột

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

- Phản hồi
 - Khi có bất kỳ thay đổi gì, nó cần phải được nhìn thấy
 - VD: Khi xóa tệp
 - Các loại phản hồi: Thị giác, âm thanh, xúc giác

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

□ Gợi ý

- Là tập các thao tác hay thủ tục có thể thực hiện trên đối tượng
- Khả năng tưởng tượng giúp xác định cách sử dụng đối tượng chỉ bằng quan sát chúng
- VD đối tượng vật lý:
 - Ghế gợi ý ngồi
 - Tay cửa gợi ý xoay
 - Phím gợi ý nhấn
- Con chạy chuột gợi ý trỏ, phím chuột gợi ý nhấn, màn hình gợi ý sờ,

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

- VD gợi ý GUI:
 - Con chuột gợi ý trở
 - Phím chuột gợi ý nhấn
 - Màn hình gợi ý sờ
 - Bàn phím gợi ý gõ
 - Làm lõm phần tử khi không thể nhấn chuột lên chúng
 - Phần tử có nền trắng và con chuột nhấp nháy gợi ý có thể soạn thảo
 - Phần tử có nền xám gợi ý không thể sửa
 - Phần tử có màu xám gợi ý bị vô hiệu hóa
 - Những đường nổi gợi ý có thể dịch chuyển

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

□ Ánh xạ

- Là quan hệ giữa các điều khiển (đối tượng đồ họa trong giao diện,...) và ảnh hưởng của nó trên hệ thống
- Ánh xạ tự nhiên mang lại lợi thế của sự tương ứng vật lý và các chuẩn văn hóa
- Ví dụ:
 - Xoay tay lái ô tô theo chiều kim đồng hồ để rẽ phải
 - Sử dụng âm thanh to hơn khi chọn số lớn và ngược lại

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

□ Ràng buộc

- Liên quan đến tổng số khả năng. Là giới hạn vật lý, ngữ nghĩa, văn hóa và logic trên tổng số khả năng
- Ví dụ 1: Đồ chơi xe máy
 - Vật lý: Bánh trước chỉ lắp vừa vào một vị trí
 - Ngữ nghĩa: Tài xế ngồi trên ghế và quay mặt về phía trước
 - Văn hóa: Đèn đỏ lắp phía sau, đèn vàng lắp phía trước
 - Logic: Hai đèn xanh, hai mẫu trắng đi với nhau

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

- Ràng buộc
 - Ví dụ 2: Ổ cắm chuột và bàn phím trên case máy tính

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

□ Qui ước

- Là ràng buộc về văn hóa, phụ thuộc vào các nền văn hóa khác nhau
- Là tùy ý
- Được chấp nhận theo thời gian
- Công tắc đèn
 - Mỹ: Bật xuống là tắt
 - Anh: Bật lên là tắt
- Van vòi nước
 - Mỹ: Ngược chiều kim đồng hồ là mở
 - Anh: Ngược chiều kim đồng hồ là đóng

2.2. Nguyên lý thiết kế hệ thống có tính sử dụng

□ Qui ước

■ Màu đỏ

- Mỹ: Nguy hiểm
- Ai cập: Chết
- Ấn độ: Sống
- Trung quốc: Hạnh phúc

■ Bàn phím

- QWERTY: Tiếng anh
- AZERTY: Tiếng pháp

2.3. Tính tiện lợi

- Các nguyên tắc
 - ***Tính có thể học được:*** Các hệ thống tương tác phải dễ học
 - ***Tính dễ sử dụng:*** Các hệ thống tương tác phải hiệu quả trong việc giúp người sử dụng đạt được mục đích
 - ***Tính linh động:*** Các hệ thống tương tác phải thích ứng được những hoàn cảnh khác nhau
 - ***Tính cảm xúc:*** Các hệ thống tương tác phải làm cho người dùng cảm thấy thoải mái

2.3. Tính tiện lợi

- Tại sao phải theo các nguyên tắc này?
 - Những nguyên tắc này giúp tập trung vào mục tiêu đặt ra
 - Giúp có thể đo được mức độ mục tiêu đặt ra được hoàn thành đến đâu.
 - Đây là một phương pháp đã được dùng để đánh giá một thiết kế!!!!

2.3. Tính tiện lợi

- Tính có thể học được
 - Tính có thể học được thể hiện qua thời gian và công sức bỏ ra để có thể đạt được một trình độ sử dụng nhất định.

2.3. Tính tiện lợi

- Tính có thể tổng hợp được: Khi tôi làm thế này thì cái gì đó xảy ra!
 - Hệ thống cho phép người dùng tổng hợp các kiến thức mà họ thu được thông qua việc sử dụng hệ thống từ đó có thể xây dựng một hệ thống các nguyên tắc về các kiến thức này.

2.3. Tính tiên lợi

- Tính có thể đoán được: Khi tôi làm thế này thì điều đó sẽ xảy ra!
 - Người dùng thường dự đoán kết quả của một sự tương tác dựa vào hệ thống kiến thức mà họ thu được từ các lần tương tác trước. Hệ thống nên hỗ trợ các suy luận hay dự đoán này bằng cách luôn luôn đưa ra các thông tin phản hồi nhất quán.
 - Ví dụ: Mở tệp: cửa sổ hội thoại xuất hiện để chọn tên tệp.
- => Ghi tệp: chắc là một cửa sổ hội thoại cũng sẽ xuất hiện để chọn tên tệp

2.3. Tính tiện lợi

■ Tính quen thuộc

- Người dùng sử dụng các kiến thức có từ trước để quyết định xem sẽ phải làm gì trong các tình huống mới.
- Vận dụng các kiến thức theo quy luật, để người dùng đỡ phải suy nghĩ quá nhiều trước một tình huống mới

2.3. Tính tiện lợi

- Tính khái quát và tính kiên định
 - Người sử dụng khái quát những quy luật và kỹ năng để áp dụng vào những tình huống tương tự.
 - Hệ thống nên hỗ trợ việc tạo nên các kỹ năng bằng cách hoạt động tương tự trong những tình huống tương tự.

2.3. Tính tiện lợi

- Làm thế nào để có thể đánh giá “tính có thể học được?”
 - Thông qua thời gian sử dụng để học được cách hoàn thành một công việc nhất định ở một trình độ nhất định

2.3. Tính tiện lợi

□ Tính dễ sử dụng

- Tính dễ sử dụng được thể hiện qua mức độ dễ mà một nhiệm vụ có thể được hoàn thành một cách hiệu quả mà không có lỗi.
- Tính quan sát được: Tôi biết tôi đang làm gì!
 - Cho phép người dùng trực tiếp nhận biết được trạng thái của hệ thống.
 - Thiết kế hệ thống của bạn sao cho không chỉ hành động mà các trạng thái cũng nhìn thấy được

2.3. Tính tiện lợi

- Tính dễ sử dụng
 - Tính phản ứng nhanh
 - Cho phép người dùng nhận biết được phản ứng cho hành động của họ ngay lập tức
 - Tính có thể khôi phục được
 - Người dùng có thể gây ra lỗi
 - Các hệ thống nên được thiết kế sao cho các lỗi có thể được ngăn chặn và khôi phục được.

2.3. Tính tiện lợi

- Tính dễ sử dụng
 - Tính thích nghi với nhiệm vụ: Các tính năng của hệ thống có thích nghi với nhiệm vụ của người dùng không?
 - Các hệ thống phải được thiết kế sao cho tính năng của chúng đáp ứng nhiệm vụ của người dùng. Hay nói cách khác, các tính năng của hệ thống phải dựa trên nhiệm vụ của người dùng

2.3. Tính tiện lợi

□ Tính dễ sử dụng

- Làm thế nào để đánh giá được tính dễ sử dụng?
 - Mức hiệu suất công việc đạt được
 - Thời gian hoàn thành công việc ở mức cao nhất
 - Tần suất lỗi
 - Công sức đầu óc bỏ ra: thông qua câu hỏi điều tra hoặc các chỉ số sinh lý (v.d. nhịptim)

2.3. Tính tiện lợi

□ Tính linh động

- Tính linh động được thể hiện qua khả năng đáp ứng của một hệ thống đối với những người dùng khác nhau trong những trường hợp khác nhau.
- Cửa sổ khởi tạo
 - Người sử dụng thích khởi tạo. V.d. : Windows Wizard.
- Đa tuyến đoạn(Multithreading)
 - Người dùng thường làm nhiều việc một lúc. Hệ thống nên được thiết kế để cho phép điều này.

2.3. Tính tiện lợi

□ Tính linh động

- Khả năng chuyển giao giữa các nhiệm vụ
 - Có khả năng phân biệt được người dùng nào đang làm gì. V.d.: Tự động sửa chính tả.
- Tính thay thế được
 - Cho phép những giá trị vào ra tương đương có thể được thay thế thoải mái. V.d: kéo thả
- Tính tùy biến(Customizability)
 - Cho phép một hệ thống có thể thích nghi được với nhu cầu của người dùng

2.3. Tính tiện lợi

□ Tính linh động

- Làm thế nào để đánh giá được tính linh động?
 - Thông qua hiệu suất và số lỗi tạo ra trong các tình huống khác nhau

2.3. Tính tiện lợi

□ Tính cảm xúc

- Tính cảm xúc thể hiện qua mức độ ưa thích và thoải mái người sử dụng cảm thấy khi sử dụng một hệ thống.
- Giá trị cảm xúc: !!!!
- Tính thẩm mỹ
- Các yếu tố môi trường: Các hệ thống nên được thiết kế phù hợp với các môi trường vật lý

2.3. Tính tiện lợi

- Tính cảm xúc
 - Làm thế nào để có thể đánh giá được tính cảm xúc?
 - Thông qua câu hỏi điều tra hoặc các chỉ số sinh lý

CHƯƠNG 3

KHẢ NĂNG CON NGƯỜI

3.1. Hệ thống nhận thức

- ❑ Con người là nhân vật trung tâm trong mọi hệ thống tương tác
- ❑ Máy tính được thiết kế để phục vụ con người => Yêu cầu của người là ưu tiên số một.
- ❑ Để có thể phục vụ con người được tốt:
 - Cần phải biết một người nào đó có khả năng làm gì và không làm được gì
 - Cần biết con người nhận biết thế giới như thế nào, lưu trữ, xử lý thông tin và giải quyết vấn đề như thế nào, và thao tác với các đồ vật ra sao

3.1. Hệ thống nhận thức

- Bốn giai đoạn xử lý thông tin
 1. Thông tin vào từ môi trường được mã hóa
 2. Dạng thông tin “biểu diễn trong” được so sánh với thông tin biểu diễn trong bộ nhớ
 3. Quyết định lựa chọn trả lời
 4. Tổ chức trả lời và các hành động cần thiết

3.2. Mô hình bộ xử lý thông tin

- Mô hình bộ xử lý con người gồm các phân hệ sau:
 - Hệ thống cảm nhận
 - Hệ thống nhận thức
 - Hệ thống vận động
- Các hệ thống có thể hoạt động song song (VD: vừa lái ô tô vừa nghe nhạc) hoặc nối tiếp (VD: nhấn phím để trả lời)

3.2. Mô hình bộ xử lý thông tin

□ Quy trình:[1]

- Dữ liệu đầu vào từ tai, mắt,... được lưu trữ trong bộ nhớ cảm biến ngắn hạn
- Bộ xử lý cảm nhận lấy dữ liệu từ đầu vào cảm biến và thử nhận biết các biểu tượng trong đó: chữ, từ, âm vị, icon. Trong quá trình nhận dạng nó được sự hỗ trợ bởi bộ nhớ dài hạn, nơi lưu trữ các biểu tượng đã nhận dạng trước đó

3.2. Mô hình bộ xử lý thông tin

□ Quy trình:[2]

- Bộ xử lý nhận thức lấy các thông tin đã nhận biết bởi bộ xử lý cảm nhận và thực hiện so sánh, lập quyết định. Nó có thể trao đổi thông tin với bộ nhớ làm việc
- Bộ xử lý vận động thu nhận các hành động từ bộ xử lý nhận thức và ra lệnh cơ bắp thực hiện nó. Ở đây, phản hồi cho khả năng quan sát hành động bằng các cách cảm nhận, sử dụng để điều chỉnh chuyển động. Sự chú ý được xem như luồng điều khiển.

3.3. Thành phần vào ra

- Con người giao tiếp với thế giới thông qua nhận và gửi thông tin bằng các thành phần vào ra

3.3.1. Thành phần vào

- Thông qua năm giác quan: thị giác, thính giác, khứu giác, vị giác, xúc giác
- Hiện nay, ba giác quan là quan trọng cho HCl, hai giác quan vị giác và khứu giác vẫn chưa được quan tâm đến

3.3.1. Thành phần vào

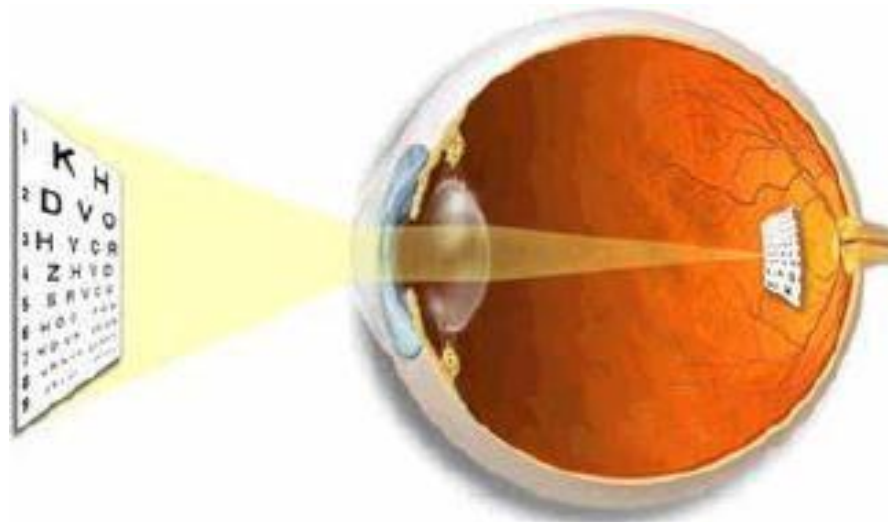
Thị giác

Lông mi

Đồng tử

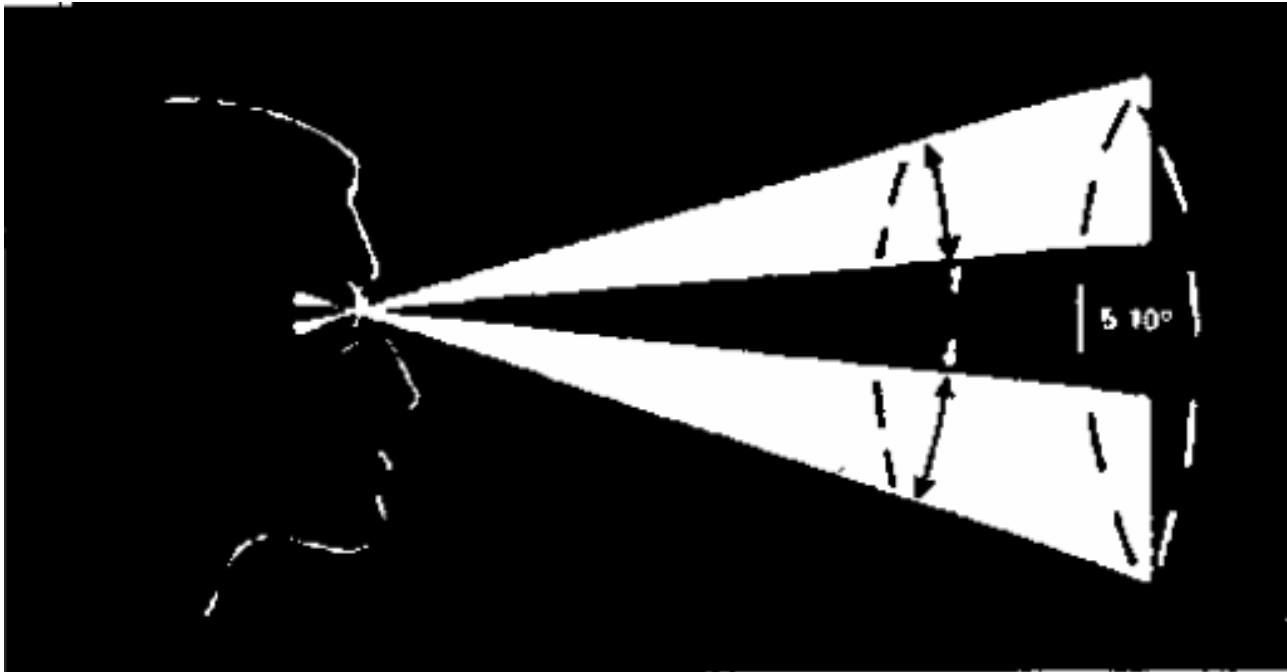
Màng cứng

Tròng đen



3.3.1. Thành phần vào

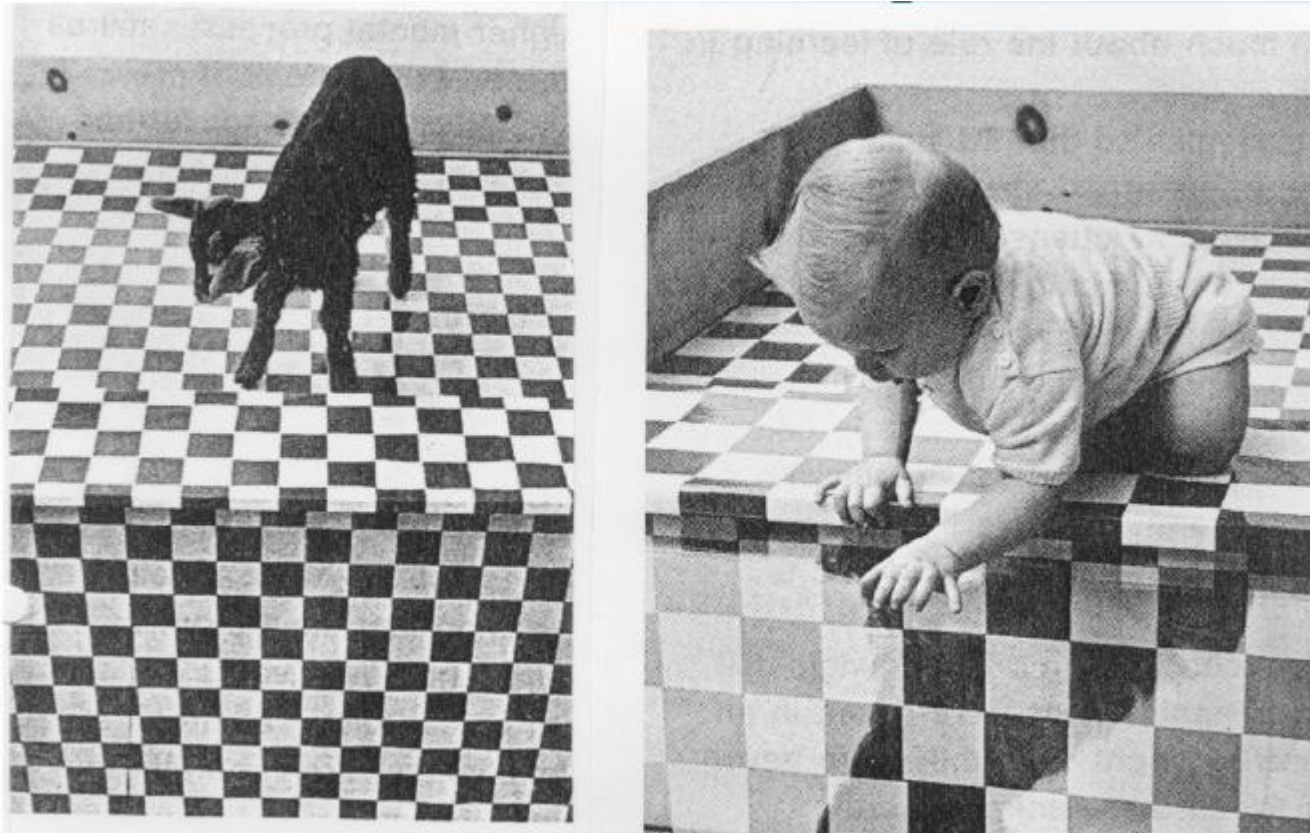
□ Thị giác



Điểm mù buổi tối

3.3.1. Thành phần vào

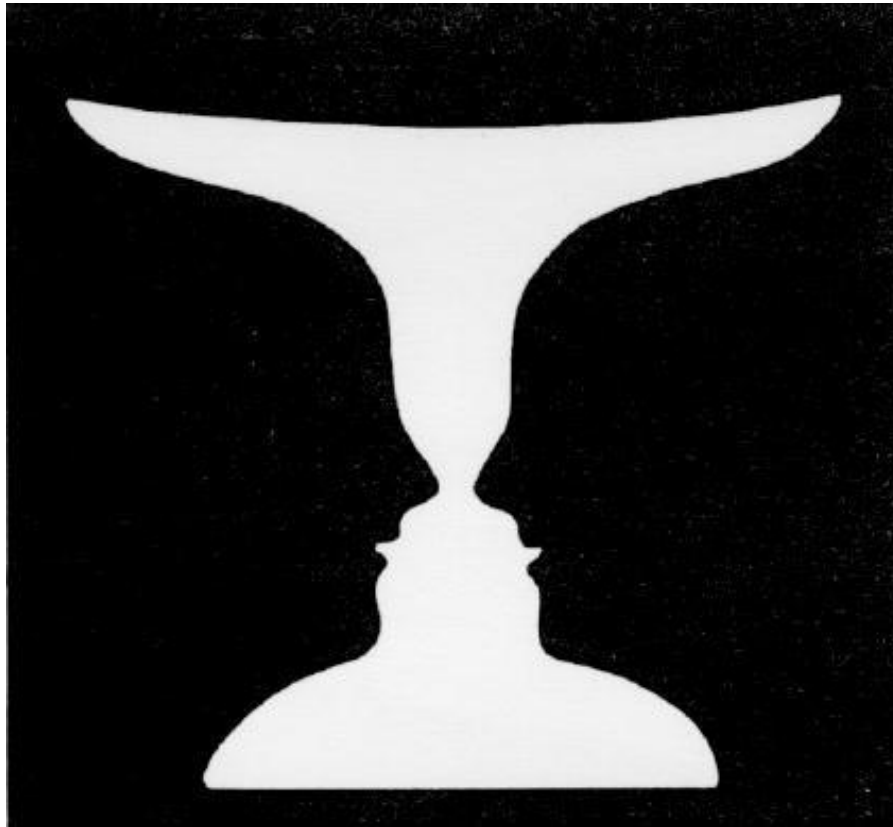
□ Thị giác



Vách ảo

3.3.1. Thành phần vào

▣ Thị giác



Đảo hình và nền – chiếc ly

3.3.1. Thành phần vào

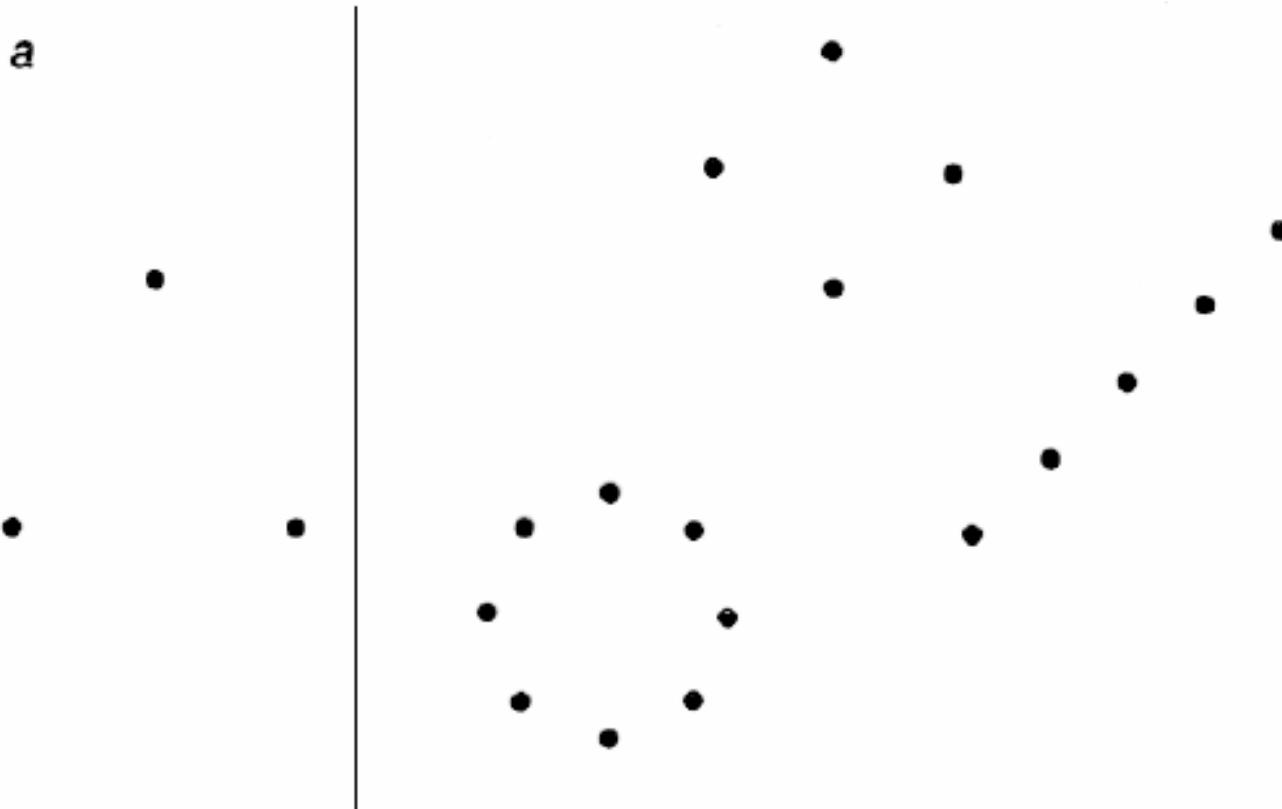
- Thị giác – Cảm nhận hình ảnh
 - Hình ảnh không chỉ là được cảm nhận một cách thụ động mà còn theo một số luật cơ bản để nhận được “hình đẹp”:
 - Luật Praegnanz: Trong số một số tổ chức hình ảnh được xuất hiện đồng thời, tổ chức hình ảnh nào đơn giản nhất, ổn định nhất sẽ được cảm nhận.
 - Luật kề cận: Hướng về nhóm những thành phần ở cạnh nhau

3.3.1. Thành phần vào

- ❑ Thị giác – Cảm nhận hình ảnh
 - ❑ Luật tương tự: Hướng về nhóm những thành phần tương tự nhau
 - ❑ Luật liên tục: hướng về nhóm những thành phần liên tiếp nhau tạo thành các đường cong mượt
 - ❑ Luật đóng: hướng về nhóm những thành phần tạo thành một hình đóng

3.3.1. Thành phần vào

□ Thị giác

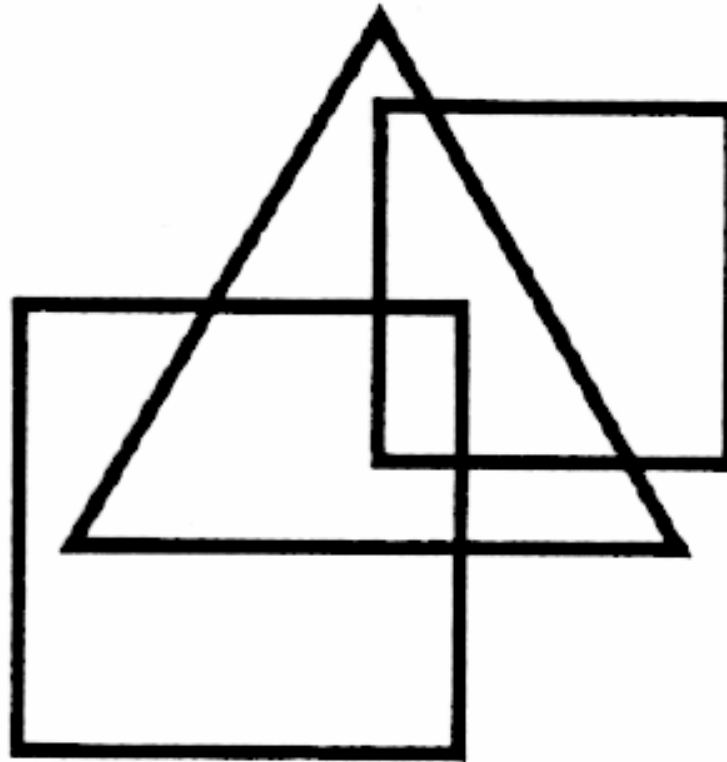
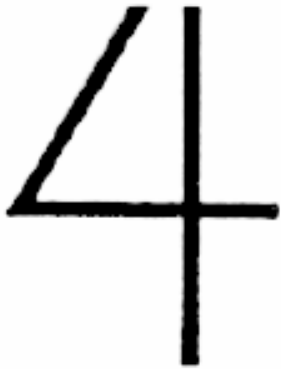


Hình ảnh ẩn –
Tam giác
trong hình
bên trái được
giấu trong
hình bên phải

3.3.1. Thành phần vào

□ Thị giác

b

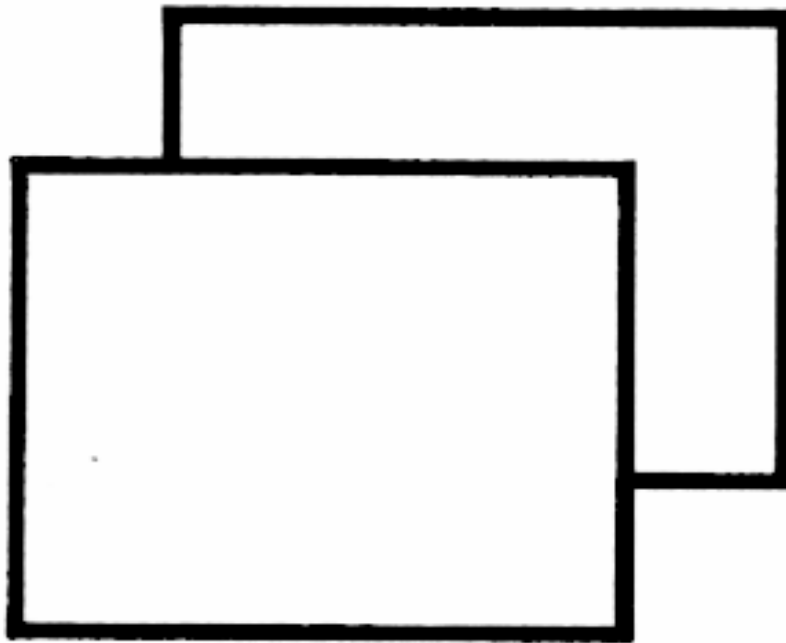


Hình ảnh ẩn –
Tam giác
trong hình
bên trái được
giấu trong
hình bên phải

3.3.1. Thành phần vào

□ Thị giác

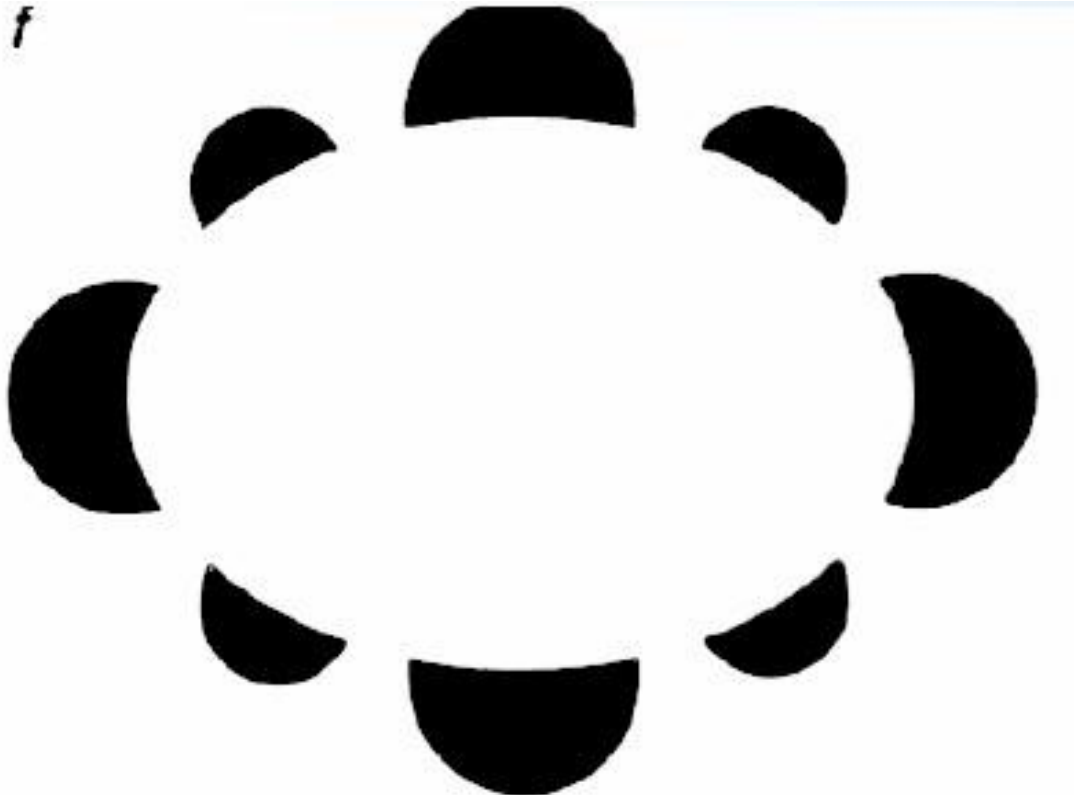
e



Ta cảm nhận được
rằng có hai hình
chữ nhật đè lên
nhau

3.3.1. Thành phần vào

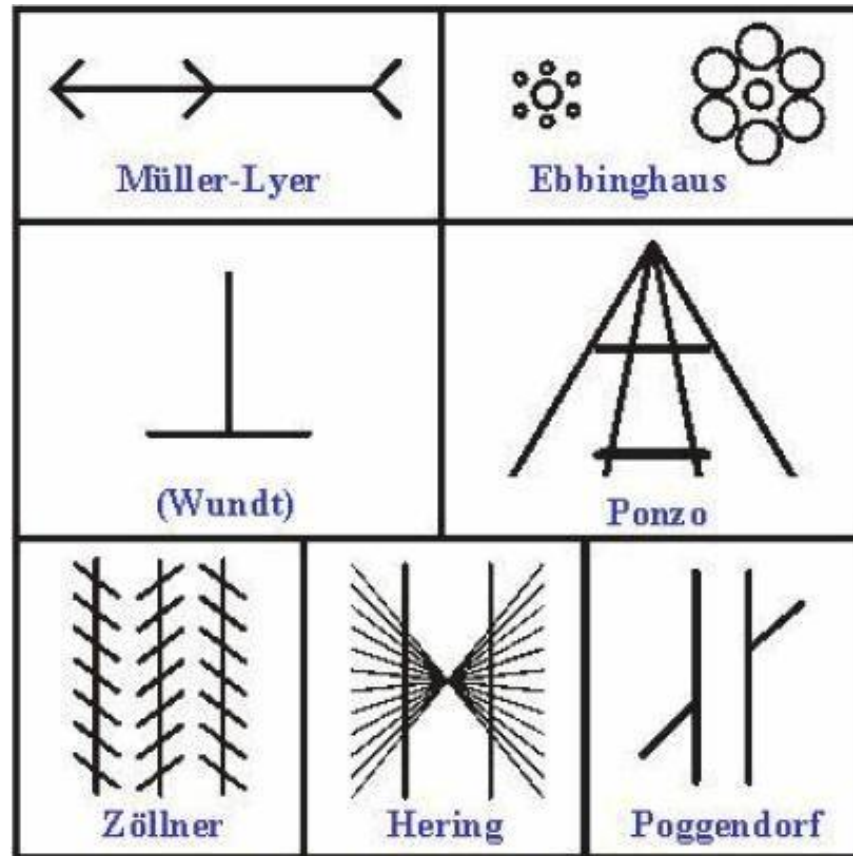
□ Thị giác



Ta vẫn cảm nhận được một hình oval mặc dù hình này không được vẽ rõ ràng

3.3.1. Thành phần vào

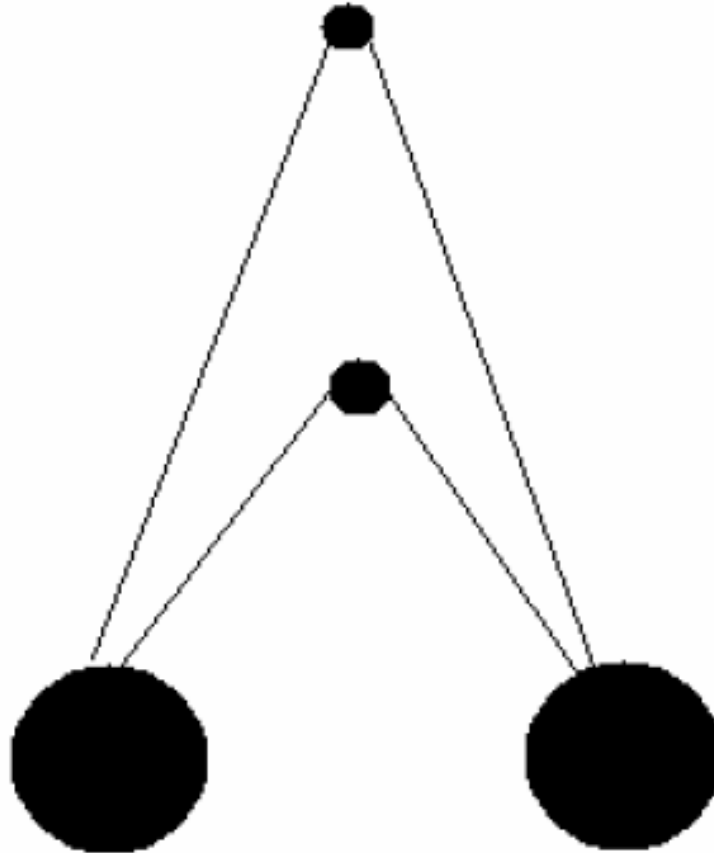
□ Thị giác



Ảo giác

3.3.1. Thành phần vào

▣ Thị giác



Cảm nhận về độ sâu thông qua hai mắt

3.3.1. Thành phần vào

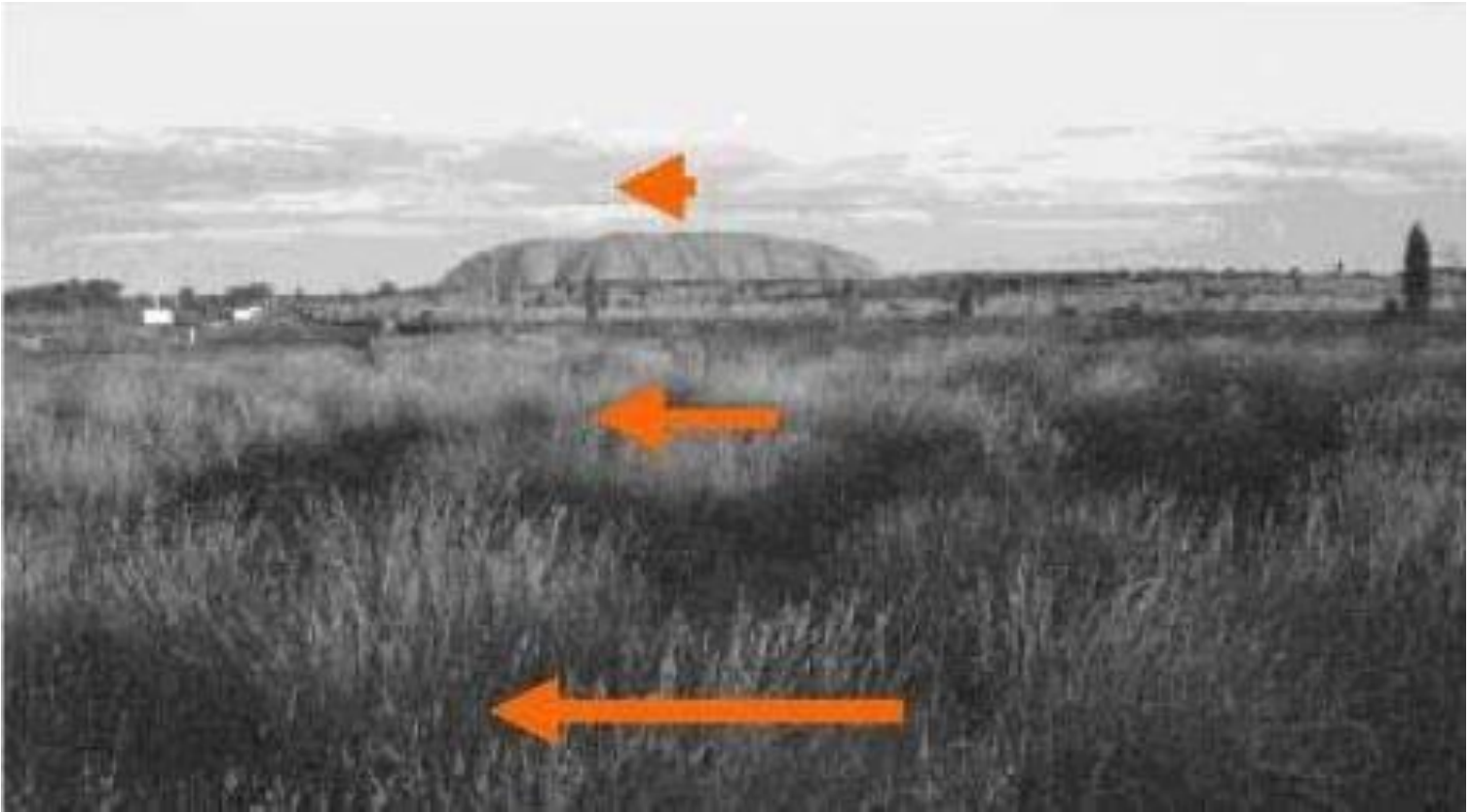
□ Thị giác



Cảm nhận về độ sâu thông qua chi tiết hình ảnh

3.3.1. Thành phần vào

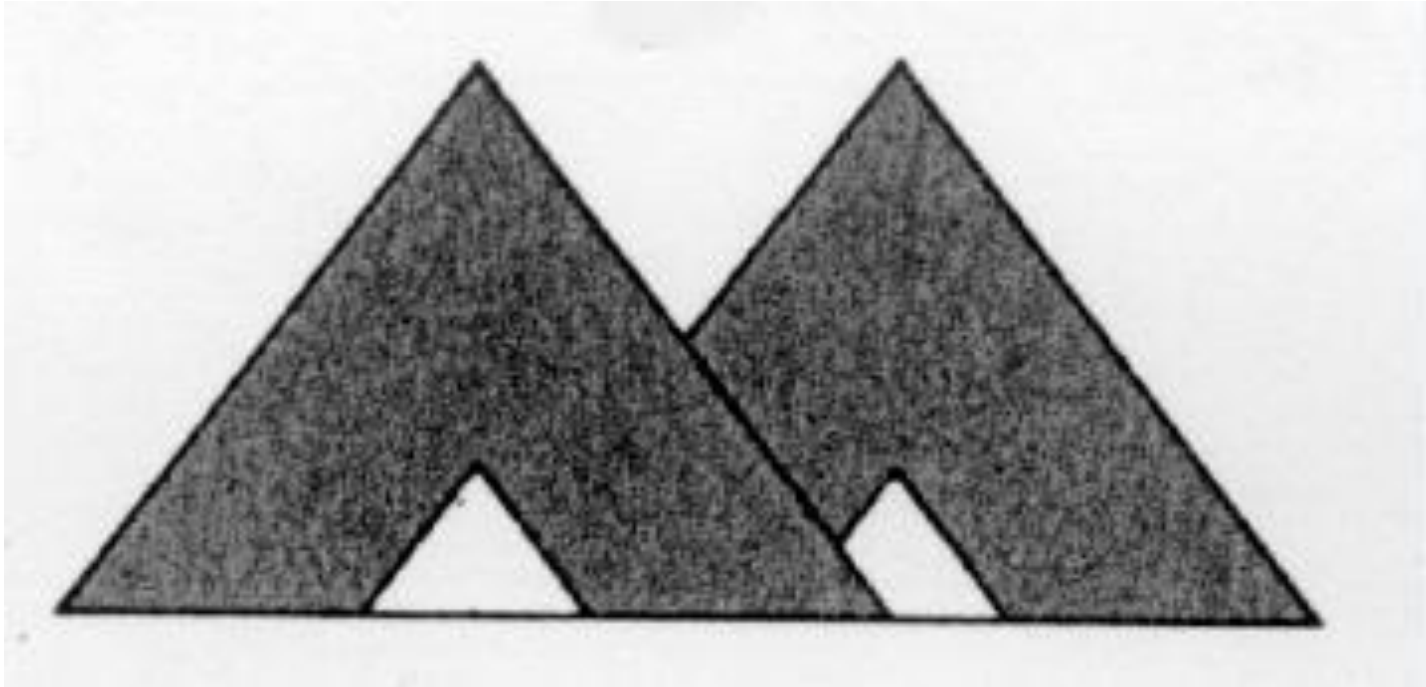
□ Thị giác



Cảm nhận về độ sâu thông qua chuyển động

3.3.1. Thành phần vào

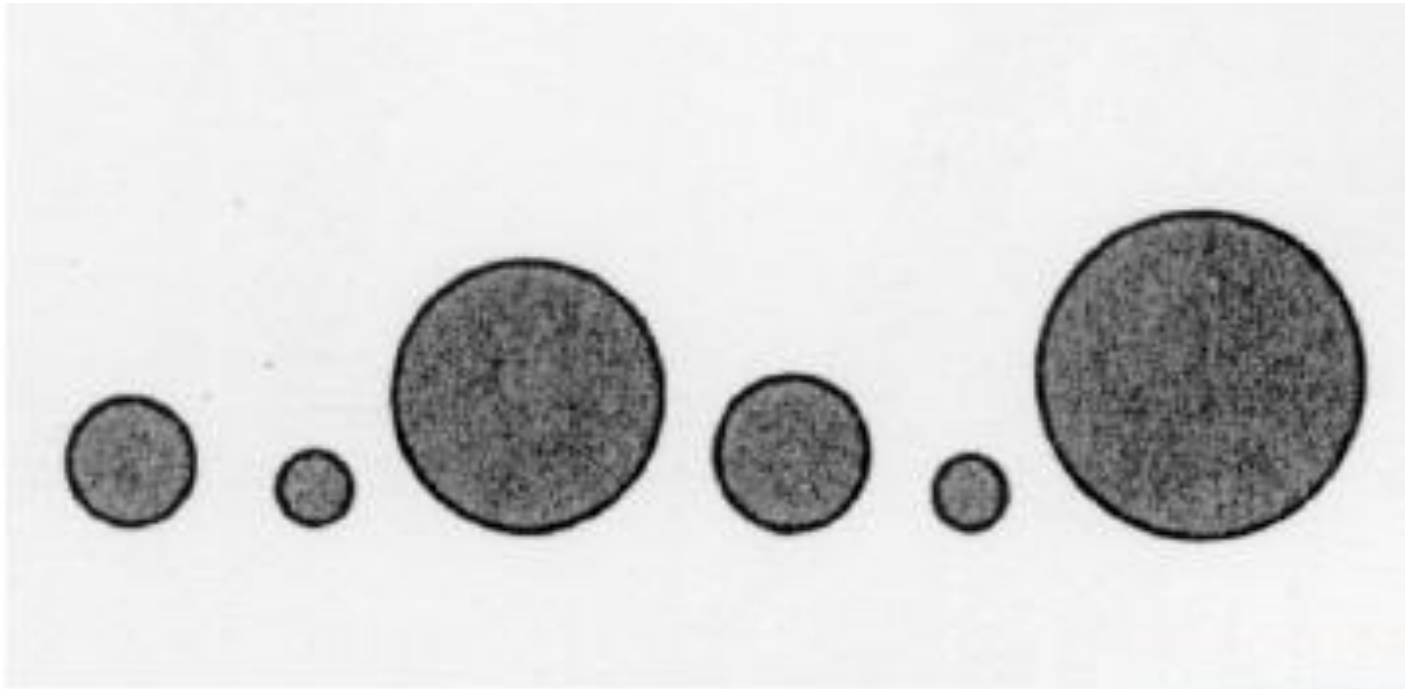
□ Thị giác



Cảm nhận về độ sâu theo vị trí

3.3.1. Thành phần vào

□ Thị giác



Cảm nhận về độ sâu theo kích thước

3.3.1. Thành phần vào

□ Thính giác

■ Thính giác quan trọng với chúng ta như thế nào:

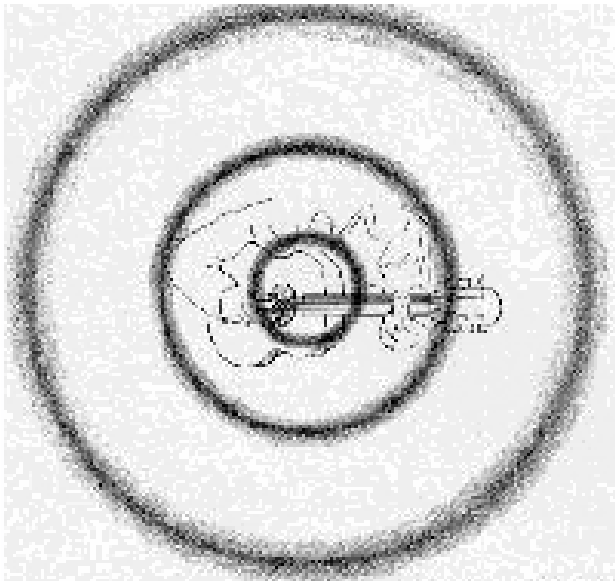
- Hệ thống báo động
- Cân bằng cảm xúc: Nghe nhạc
- Giao tiếp

■ Những vấn đề thú vị nhất về thính giác

- Cơ bản của hòa âm
- Sự ảnh hưởng của kiến thức tới việc tách các nguồn âm thanh khác nhau
- Nhận dạng các giọng nói

3.3.1. Thành phần vào

- Thính giác - Bản chất của âm thanh



Nguồn âm thanh phát ra các sóng tròn

3.3.1. Thành phần vào

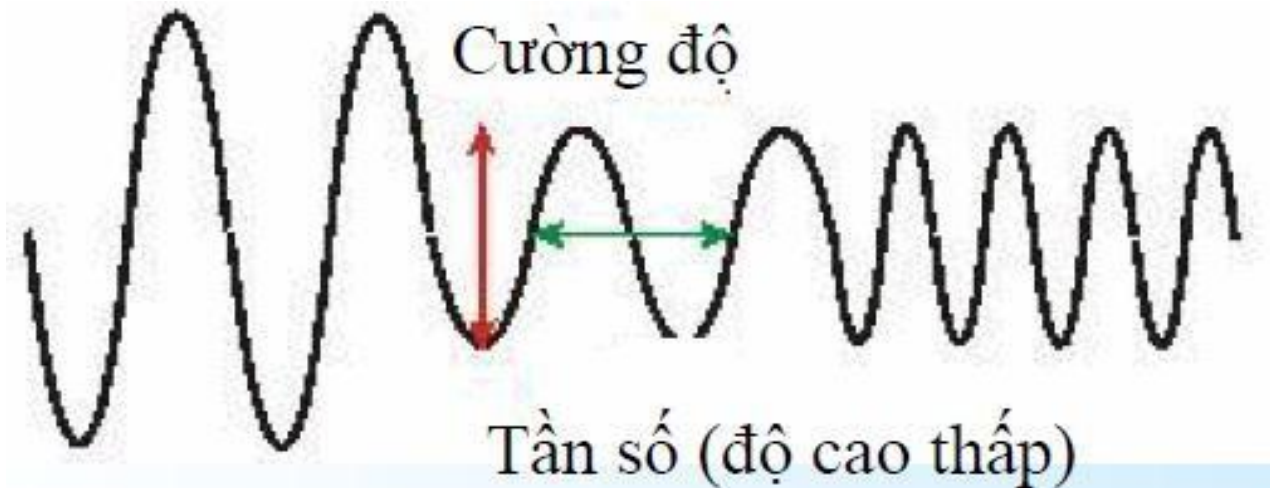
□ Thính giác - Bản chất của âm thanh



“Tiếng ồn trắng”(white noise) là sự tổng hợp của rất nhiều âm thanh với tần số và cường độ ngẫu nhiên (tiếng gợn sóng trên hồ)

3.3.1. Thành phần vào

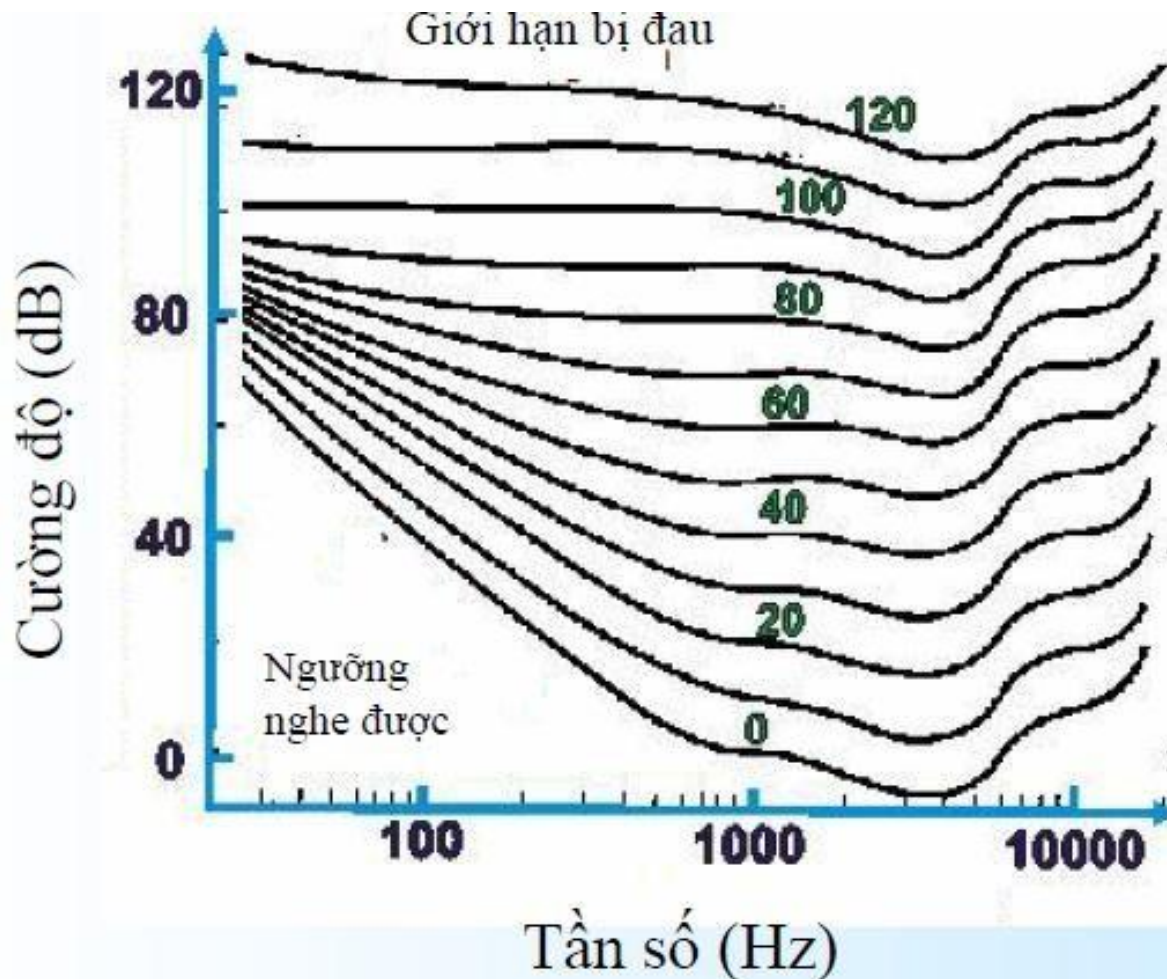
- Thính giác - Bản chất của âm thanh



Một âm thanh trong trẻo được thể hiện bằng dạng sóng hình sin (không có nhiễu) với biên độ là cường độ và tần số là độ cao thấp

3.3.1. Thành phần vào

□ Thính giác – Cảm nhận âm thanh



3.3.1. Thành phần vào

- Thính giác - Cảm nhận âm thanh
 - Con người chúng ta có thể cảm nhận được:
 - Tần số của âm thanh (độ cao thấp)
 - Cường độ (độ to nhỏ)
 - Âm sắc (khác về nguồn âm thanh – phát ra từ các dụng cụ khác nhau, mặc dù có cùng tần số và cường độ)
 - Chúng ta cũng có thể nhận ra vị trí của nguồn âm thanh
 - Dùng hai tai
 - Sự khác nhau về thời gian nhận và cường độ

3.3.1. Thành phần vào

- ❑ Thính giác - Cảm nhận âm thanh
 - Con người có thể nghe được âm thanh với các tần số từ khoảng 20Hz tới 15kHz
 - Con người có thể phân biệt được thay đổi khoảng 1.5Hz ở các tần số thấp. Càng tần số cao thì khả năng phân biệt càng kém hơn
 - Những tần số khác nhau thì ảnh hưởng đến những phần khác nhau trong hệ thống xử lý âm thanh của con người=> tạo ra những kích thích khác nhau đến hệ thống thần kinh

3.3.1. Thành phần vào

- Thính giác - Cảm nhận âm thanh
 - Hệ thống xử lý âm thanh thực hiện một số thao tác lọc âm thanh => cho phép bỏ qua những tiếng ồn nền => tập trung vào âm thanh quan trọng
 - “Cocktail party efect”– chúng ta vẫn có thể nghe thấy ai đó gọi chúng ta trong một căn phòng rất ồn

3.3.1. Thành phần vào

- Thính giác - HCI
 - Các âm thanh hiện vẫn đang được sử dụng chính vào việc
 - thông báo:
 - Thông báo khi gõ nhầm nút
 - Thông báo khi vào Windows
 - Thông báo khi máy sắp hết pin
 - Hiện nay âm thanh đang được nghiên cứu:
 - Tổng hợp tiếng nói => nghe đọc tài liệu thay vì nhìn tài liệu => phục vụ người khiếm thị
 - Dùng âm nhạc để tạo ra các hiệu ứng trong trình diễn nội dung

3.3.1. Thành phần vào

□ Xúc giác

- Xúc giác: Khả năng phát hiện và hiểu được những thông
 - tin cảm nhận được từ da
- Các nghiên cứu đã chỉ ra rằng khả năng cảm nhận bằng xúc giác của trẻ nhỏ có thể đã được phát triển trước khả năng phân biệt vật thể bằng thị giác
- Theo bạn, môn thể thao nào (không) cần xúc giác?

3.3.1. Thành phần vào

□ Xúc giác – Cảm nhận

■ Sức ép:

- Mạnh
- Nhẹ

■ Nhiệt độ:

- Lạnh
- Ấm

■ Mức độ đau:

- Nhọn
- Tù

3.3.1. Thành phần vào

- Xúc giác – HCI
 - Ta cảm nhận được các phím được nhấn xuống

3.3.2. Thành phần ra

- Thông qua một số các cơ quan như tay, chân, mắt, đầu và hệ thống tiếng nói.
- Khi giao tiếp với máy tính, hiện nay tay vẫn đóng vai trò chính trong việc gõ bàn phím và điều khiển chuột
- Tiếng nói, ánh mắt và chuyển động của đầu đang được nghiên cứu để điều khiển máy tính

3.4. Bộ nhớ

- Bộ nhớ giác quan (sensory memory)
- Bộ nhớ ngắn hạn (short-term memory)
- Bộ nhớ dài hạn (long-term memory)

3.4.1. Bộ nhớ giác quan

- Vùng đệm chứa các tín hiệu nhận vào bằng các giác quan:
 - Bộ nhớ hình tượng (iconic memory) cho thị giác
 - Bộ nhớ tượng thanh (echoic memory) cho thính giác
 - Bộ nhớ xúc giác (haptic memory) cho xúc giác

3.4.1. Bộ nhớ giác quan

- Các bộ nhớ này liên tục bị ghi đè bởi những tín hiệu mới
- Ví dụ 1: Hiện tượng lưu ảnh
- Ví dụ 2: Tai lưu thông tin trong một thời gian ngắn— 2 tai nhận được một âm thanh tại hai thời điểm khác nhau (rất gần nhau) → xác định âm thanh được phát từ đâu

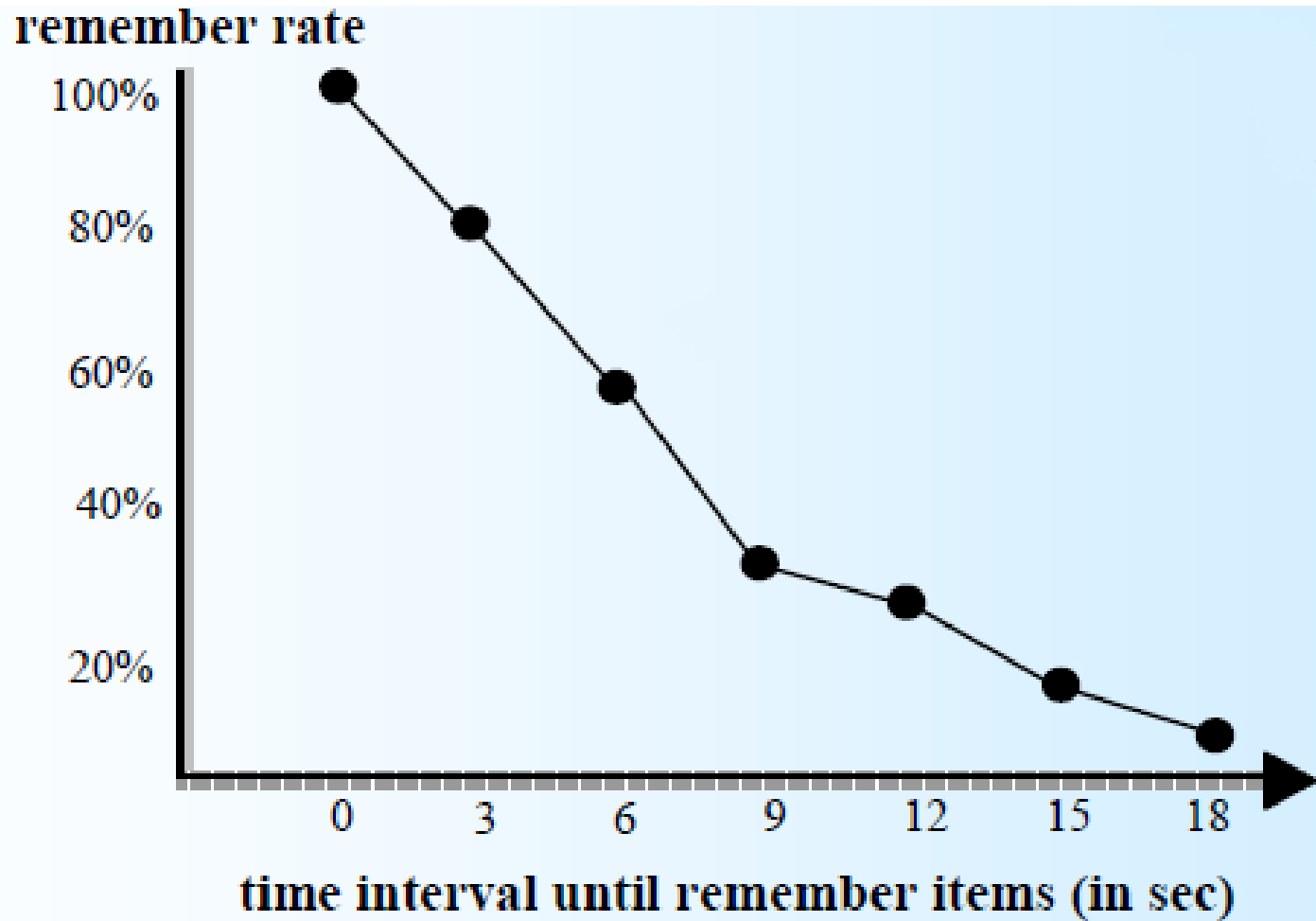
3.4.2. Bộ nhớ ngắn hạn

- Hay còn gọi là bộ nhớ làm việc
- Ví dụ 1: Để tính 35×6 , chúng ta có thể nhân 30×6 rồi cộng với 5×6 hoặc 35×2 ra 70 rồi lấy 70×3
- Ví dụ 2: Đọc sách, chúng ta phải nhớ một số thông tin thì mới hiểu được quyển sách: các từ trong câu đang đọc, một số câu trước đó, một số chi tiết trước đó

3.4.2. Bộ nhớ ngắn hạn

- ❑ Con người có thể nhớ 7 ± 2 mục liên tiếp:
- ❑ Bạn thử nhớ 2419406832 xem bạn nhớ được bao nhiêu số?
- ❑ Thế còn 764 321 5793 ?

3.4.2. Bộ nhớ ngắn hạn



3.4.3. Bộ nhớ dài hạn

- ❑ Dung lượng lớn hơn nhiều so với bộ nhớ ngắn hạn
 - ❑ Thời gian truy cập lâu hơn
 - ❑ Nhiều thông tin:
 - Thông tin cũ nhiều thông tin mới được học
 - Thông tin mới nhiều các thông tin cũ
- => Học nhiều quên nhiều, học ít quên ít, không học không quên???

3.4.3. Bộ nhớ dài hạn

- Có hai loại bộ nhớ dài hạn:
 - Loại nhớ theo tình tiết (episodic)
 - Loại nhớ theo ngữ nghĩa (semantic)

3.4.3. Bộ nhớ dài hạn

- Loại nhớ theo tình tiết (episodic)
 - Bộ nhớ loại này ghi lại các sự kiện và kinh nghiệm theo
 - cấu trúc chuỗi
 - Giúp chúng ta nhớ lại các sự kiện đã xảy ra trong quá khứ

3.4.3. Bộ nhớ dài hạn

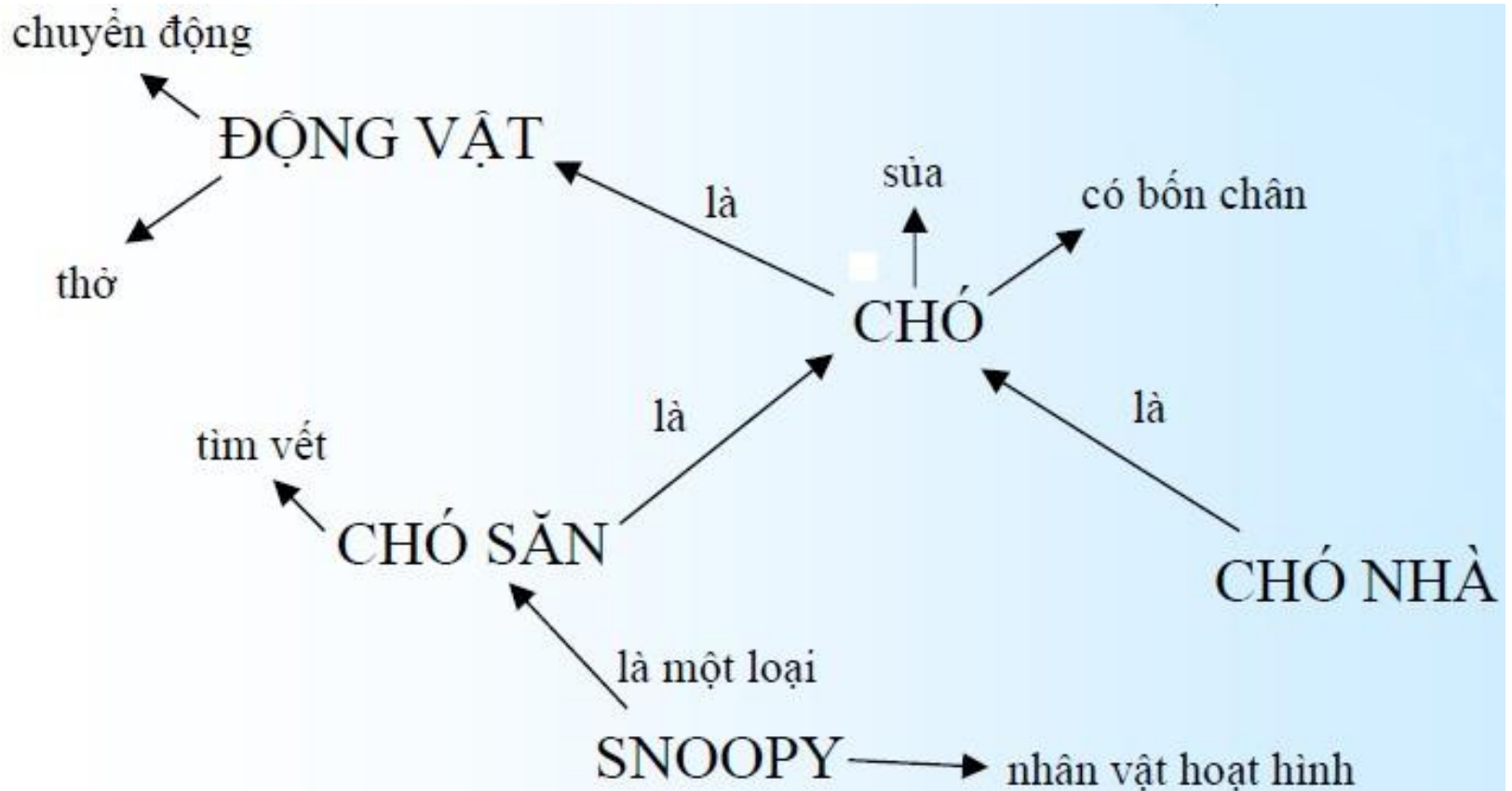
- Loại nhớ theo ngữ nghĩa
 - Bộ nhớ loại này ghi lại các khái niệm, sự thật và các kỹ
 - năng chúng ta học được theo cấu trúc liên kết
 - Các thông tin trong bộ nhớ loại này nhận được từ bộ nhớ theo tình tiết, cho phép chúng ta học được các khái niệm và sự thật mới từ kinh nghiệm

3.4.3. Bộ nhớ dài hạn

- Loại nhớ theo ngữ nghĩa
 - Bộ nhớ này được tổ chức để cho phép chúng ta truy cập thông tin, các mối quan hệ giữa các thông tin và cho phép chúng ta suy diễn
 - Bộ nhớ này thường được biểu diễn dưới dạng mạng lưới – mạng lưới ngữ nghĩa (semantic network)

3.4.3. Bộ nhớ dài hạn

□ Loại nhớ theo ngữ nghĩa



3.4.3. Bộ nhớ dài hạn

- Loại nhớ theo ngữ nghĩa
 - Cho phép chúng ta suy diễn: Vì chó có 4 chân => chó săn
 - có 4 chân
 - Lưu ý: Có những liên kết nối sang hẳn những lĩnh vực khác (ví dụ như Snoopy → phim hoạt hình)

3.4.3. Bộ nhớ dài hạn

□ Loại nhớ theo ngữ nghĩa

- Sự tồn tại của mạng ngữ nghĩa này trong con người chúng ta được chứng minh bởi Collins và Quillian (1969):
 - Một số người được hỏi về các thuộc tính của một số đối tượng. Thời gian suy nghĩ để trả lời được ghi lại. Kết quả là: người ta suy nghĩ lâu hơn khi được hỏi những câu hỏi kiểu như: “chó săn có thở không” so với các câu hỏi kiểu như: “chó săn có tìm vết được không?”
 - Lý do: Con người phải tìm kiếm thông qua mạng ngữ nghĩa, suy ngược lên để tìm ra câu trả lời

3.4.3. Bộ nhớ dài hạn

- Một số loại cấu trúc khác của bộ nhớ cũng được đưa ra để giải thích chúng ta lưu trữ kiến thức như thế nào:
 - Kiểu khung
 - Kiểu kịch bản
 - Kiểu quy tắc

3.4.3. Bộ nhớ dài hạn

□ Bộ nhớ dài hạn kiểu khung

CHÓ

Tính chất cố định: Có 4 chân

Tùy biến:

-Kích cỡ

-Màu sắc

3.4.3. Bộ nhớ dài hạn

□ Bộ nhớ dài hạn kiểu kịch bản

KỊCH BẢN ĐƯA CHÓ ĐI KHÁM

Điều kiện:

- Chó bị ốm
- Bệnh viện thú y mở cửa
- Người chủ có tiền

Kết quả:

- Chó khỏi ốm
- Người chủ nghèo hơn
- Bác sĩ giàu hơn

Cảnh:

- Đến phong khám
- Ngồi chờ
- Bác sĩ khám
- Trả tiền

3.4.3. Bộ nhớ dài hạn

- Bộ nhớ dài hạn kiểu quy tắc

NẾU chó vẫy đuôi

THÌ lại gần

NẾU chó nhe răng

THÌ chạy

3.4.3. Bộ nhớ dài hạn

- Xử lý
 - Ba hoạt động chính:
 - Ghi nhớ
 - Quên
 - Truy cập thông tin

3.4.3. Bộ nhớ dài hạn

□ Ghi nhớ

- Do quá trình nhắc đi nhắc lại từ bộ nhớ ngắn hạn
- Ebbinghaus(1885): Tự thí nghiệm và rút ra kết luận: lượng thông tin nhớ được tỷ lệ với thời gian học
- Baddeley và Longman (1978): Việc ghi nhớ hay học tập sẽ đạt hiệu quả tốt hơn nếu nó được dàn trải đều theo thời gian

3.4.3. Bộ nhớ dài hạn

□ Ghi nhớ

- Nhớ các từ mô tả các đối tượng dễ hơn các từ mô tả các khái niệm:
 - Chuỗi 1: Nhà - cửa – cây – mèo – chó - ô tô
 - Chuỗi 2: Tuổi tác - logic - lạnh - im lặng – quá khứ - chủ nghĩa
- Những thông tin có ý nghĩa và quen thuộc thì dễ nhớ hơn: đọc thần thoại Hy Lạp thì khó nhớ hơn thần thoại Việt Nam, châu Á

3.4.3. Bộ nhớ dài hạn

□ Quên

- Nếu quá trình ghi nhớ thông tin được trợ giúp bằng các cấu trúc, sự quen thuộc và tính cụ thể, làm thế nào để chúng ta có thể mất thông tin, có thể quên?

3.4.3. Bộ nhớ dài hạn

□ Quên

■ Có hai học thuyết chính về sự quên:

- Sự phân rã (decay) : Thông tin trong bộ nhớ sẽ dần dần bị mất đi (Ebbinghaus, 1885)
- Sự can thiệp – nhiễu (interference): Các thông tin cũ bị mất đi do có sự can thiệp của các thông tin mới. Các thông tin cũ nhiều khi cũng có thể can thiệp lại thông tin mới
 - Ví dụ 1.: Bạn có số ĐT di động mới, việc nhớ số di động mới sẽ làm bạn quên đi số di động cũ
 - Ví dụ 2: thỉnh thoảng bạn rẽ nhầm vào đường nhà người yêu cũ thay vì đi thẳng để đến nhà người yêu mới

3.4.3. Bộ nhớ dài hạn

□ Quên

- Cảm xúc cũng ảnh hưởng đến sự quên –các sự kiện có
 - nhiều cảm xúc sẽ ít bị quên hơn các sự kiện có ít cảm xúc
- Câu hỏi đặt ra: Chúng ta có thực sự quên hay chỉ là thông tin khó có thể (không thể) truy cập được? –không ai chứng minh được – tuy nhiên có những bằng chứng là thông tin có thể không mất đi hẳn

3.4.3. Bộ nhớ dài hạn

□ Truy cập thông tin

■ Hai loại truy cập thông tin:

- Nhớ lại: Các thông tin được sao chép lại từ bộ nhớ
- Nhận dạng: So sánh thông tin với các thông tin trong bộ nhớ. Quá trình nhận dạng đơn giản hơn quá trình nhớ lại vì có thông tin làm gợi ý. Ví dụ nhớ ra mặt một người khó hơn nhận ra người đó; câu hỏi lựa chọn dễ hơn câu hỏi thông thường

3.5. Bộ xử lý

- ❑ Suy nghĩ
- ❑ Rèn luyện kỹ năng
- ❑ Xử lý lỗi

3.5.1. Suy nghĩ

- ❑ Điểm khác biệt giữa người và loài vật, giữa người
 - và máy
- ❑ Suy nghĩ: Giải quyết cả những vấn đề chúng ta chưa thấy bao giờ
- ❑ Suy nghĩ:
 - Suy luận (Reasoning)
 - Giải quyết vấn đề (Problem solving)

3.5.1. Suy nghĩ

□ Suy nghĩ – Suy luận

- Dùng các kiến thức đã có để suy ra kết luận hoặc suy ra một điều gì đó mới về lĩnh vực đang quan tâm
- Ba loại suy luận:
 - Suy diễn (deductive)
 - Quy nạp (inductive)
 - ??? (abductive)

3.5.1. Suy nghĩ

- Suy nghĩ – Suy luận – Suy diễn
 - Đưa ra kết luận tất yếu từ một số giả thiết

Nếu là chiều thứ 2 thì chúng ta có tiết HCl

Hôm nay là thứ 2

=> chúng ta có tiết HCl

3.5.1. Suy nghĩ

- Suy nghĩ – Suy luận – Suy diễn
 - Chúng ta cũng có khi suy diễn sai:

Một số học sinh đại học đi làm

Một số người đi làm là sếp

=> Một số học sinh là sếp

3.5.1. Suy nghĩ

- Suy nghĩ – Suy luận – Quy nạp
 - Tổng quát hóa từ những trường hợp chúng ta đã thấy để suy ra những trường hợp chúng ta chưa gặp
 - Ví dụ: Mọi người đàn ông chúng ta gặp đều sợ vợ => mọi người đàn ông đều sợ vợ
 - Tất nhiên, suy luận này có thể sai! Chứng minh là sai thì dễ - chỉ ra một ví dụ sai ! Không thể chứng minh đúng -> tìm càng nhiều bằng chứng càng tốt để hỗ trợ cho ý kiến

3.5.1. Suy nghĩ

- Suy nghĩ – Suy luận – abductive
 - Suy luận từ một thực tế ra hành động hay trạng thái gây ra thực tế đó

Nếu ông A uống rượu thì ông A đi xe máy nhanh
Hôm nay ông A đi xe máy nhanh

→ ông A uống rượu

- Không đáng tin cậy! Nhưng vẫn được dùng nhiều

3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề
 - Tìm giải pháp cho một nhiệm vụ mới dùng những kiến thức chúng ta đã có
 - Có nhiều quan điểm về việc chúng ta giải quyết vấn đề thế nào:
 - Nguyên lý Gestalt
 - Nguyên lý “không gian vấn đề”
 - Nguyên lý “phép loại suy trong giải quyết vấn đề”

3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề - Nguyên lý Gestalt
 - Chúng ta giải quyết vấn đề bằng phương pháp mò mẫm (trial and error)
 - Thí nghiệm: Cho một số cọc, một số chiếc kìm—làm thế nào để nối 2 đoạn dây điện lại với nhau?

3.5.1. Suy nghĩ

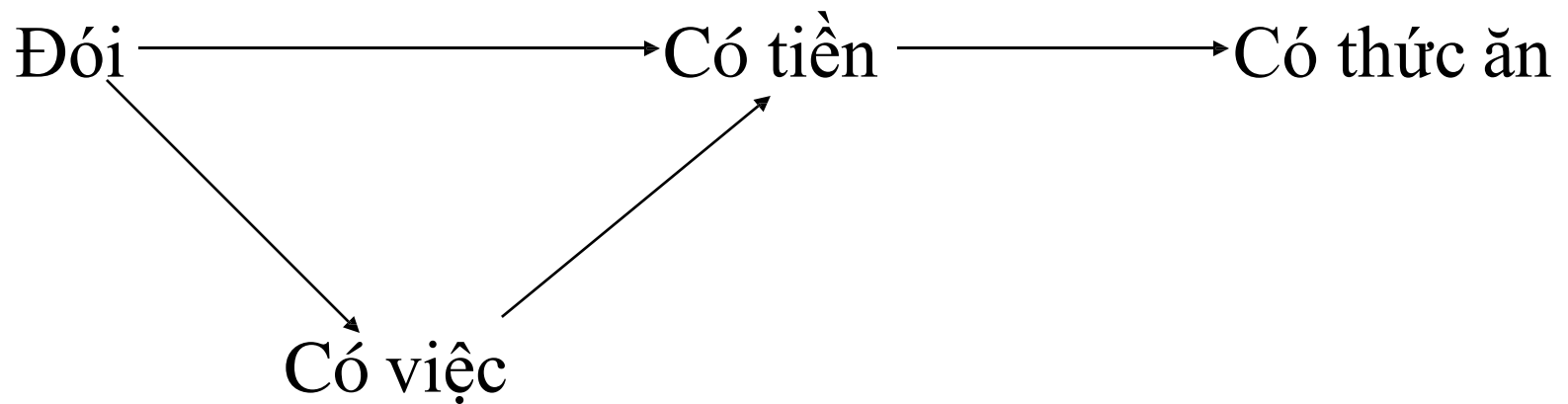
- Suy nghĩ - giải quyết vấn đề - Nguyên lý Gestalt
 - Kết quả:
 - Có nhiều lời giải khác nhau
 - Trong đó có một lời giải: Khi người thí nghiệm đang dùng kìm để cắt vỏ dây điện – dây lắ => Buộc kìm vào dây điện - tạo ra con lắ – chạy sang phía dây kia – bắt con lắ – nối 2 dây với nhau
 - Nguyên lý này nghe rất hấp dẫn – nhưng chưa có đủ bằng chứng để chứng minh

3.5.1. Suy nghĩ

- ❑ Suy nghĩ - giải quyết vấn đề - “Nguyên lý không gian vấn đề”
 - Tâm điểm của việc giải quyết vấn đề chính là một “không gian vấn đề”(Newell và Simon, 1972).
 - Không gian vấn đề chứa những trạng thái vấn đề, giải quyết vấn đề được thực hiện thông qua việc sinh ra những trạng thái vấn đề bằng các toán tử (operator) chuyển trạng thái
 - Một vấn đề có một trạng thái ban đầu và một trạng thái đích. Con người dùng các toán tử để đi đến trạng thái đích.

3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề - “Nguyên lý không gian vấn đề”
 - Tuy nhiên không gian vấn đề rất lớn nên con người dùng phương pháp tìm kiếm *heuristics*



3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề - “phép loại suy trong giải quyết vấn đề”
 - Phép loại suy (analogy): Quá trình suy luận dựa trên sự giống nhau, sự tương tự
 - Trong giải quyết vấn đề, các lĩnh vực cũ cùng các cách giải quyết cũ được đưa sang các vấn đề mới tương tự

3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề - “phép loại suy trong
 - giải quyết vấn đề”
 - Ví dụ: Đưa cho các bác sỹ giải quyết một khối u bằng chiếu tia. Tuy nhiên nếu chiếu tia mạnh quá –phá cả các mô lành. Nếu chiếu tia yếu quá – không đủ phá khối u.
 - Giải pháp: Chiếu các tia yếu từ mọi phía vào khối u –vừa không phá các mô lành, vừa hợp thành tia đủ mạnh để phá khối u

3.5.1. Suy nghĩ

- Suy nghĩ - giải quyết vấn đề - “phép loại suy trong
 - giải quyết vấn đề”
 - Chỉ có 10% số bác sỹ đưa ra giải pháp đó
 - Tuy nhiên sau khi cho các bác sỹ xem tình huống:
 - Để đánh một pháo đài, người ta cần cử một sư đoàn đến. Tuy nhiên trên đường đi, có rất nhiều mìn và rất dễ nổ nếu cả sư đoàn đi qua. Người ta đành chia sư đoàn thành các nhóm nhỏ đi theo những ngã đường khác nhau.
- ⑩ → Đã có đến 80% các bác sỹ đưa ra giải pháp nói trên để xử lý khối u

3.5.1. Rèn luyện kỹ năng

- ❑ Giải quyết vấn đề tập trung vào việc giải quyết những vấn đề mới
- ❑ Trên thực tế, chúng ta rất hay gặp lại các vấn đề cũ – thông qua việc lặp đi lặp lại giải quyết các vấn đề cũ → rèn luyện kỹ năng → tăng hiệu suất công việc
- ❑ Ví dụ: Nấu ăn, chơi cờ....

3.5.1. Xử lý lỗi

- ❑ Trong cuộc sống, chúng ta không phải là không mắc sai lầm! Có sai lầm nhỏ, có sai lầm lớn
- ❑ Lỗi khi sử dụng kỹ năng: Khi tình huống đã thay đổi, chúng ta vẫn sử dụng kỹ năng đó => gây ra lỗi.
Ví dụ: Kỹ năng đi xe từ chỗ làm về nhà. Hôm nay vợ dặn qua siêu thị mua đồ - vẫn dùng kỹ năng cũ
→ đi quá mất siêu thị

Xin chân thành cảm ơn!

- **CHƯƠNG 4**
- **MÁY TÍNH TRONG HCI**

4.1. Tổng quan

- Để có thể hiểu được tương tác giữa người và máy chúng ta cũng phải hiểu được máy tính, khả năng và hạn chế của máy tính
- Máy tính là (Thimbleby, 1990):
 - “Phía tham dự vào sự tương tác mà chạy chương trình”
 - ⑩ → Theo định nghĩa này, một chiếc công tắc đèn cũng có thể coi là máy tính???

4.1. Tổng quan

- Một hệ thống máy tính điển hình



4.1. Tổng quan

- Nhập dạng bó và dạng tương tác
 - Nhập vào dạng bó: Khi có khối lượng lớn dữ liệu, theo định dạng nhất định => nhập dữ liệu nhanh – khó có thể được chuẩn bị bởi người sử dụng – giao tiếp giữa máy và máy – không được HCI quan tâm
 - Nhập dạng tương tác : Người sử dụng ngồi trước màn hình và nhập dữ liệu - vấn đề được HCI quan tâm

4.2. Thiết bị nhập

□ Bàn phím:

- QWERTY: Dựa theo bàn phím của máy đánh chữ - các từ được kết hợp từ các chữ cái ở vị trí cách xa nhau – tránh tắc nghẽn máy đánh chữ - tránh mỏi tay



4.2. Thiết bị nhập

□ Bàn phím:

- Alphabetic: Sắp xếp theo thứ tự bảng chữ cái A, B, C – dễ cho người mới sử dụng – không tiện lợi cho người đã sử dụng lâu

4.2. Thiết bị nhập

□ Bàn phím:

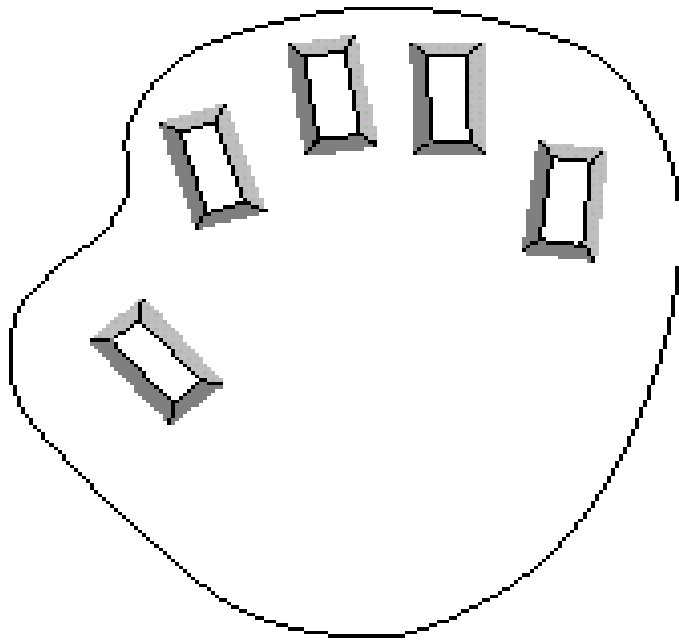
- DVORAK: Tăng tốc độ đánh máy – dành cho người thuận tay trái – không thể thay thế được QWERTY

~ `	! 1	@ 2	# 3	\$ 4	% 5	^ 6	& 7	* 8	(9) 0	{ [}]	← Backspace
Tab ↔	" "	< ,	> .	P	Y	F	G	C	R	L	? /	+ =	 \ _
Caps Lock ⬆	A	O	E	U	I	D	H	T	N	S	- _	Enter ⬇	
Shift ⬆	:	Q	J	K	X	B	M	W	V	Z	Shift ⬆		
Ctrl	Win Key	Alt								Alt Gr	Win Key	Menu	Ctrl

4.2. Thiết bị nhập

□ Bàn phím:

- Chord: 5 phím, nhẹ, gọn, muốn gõ một chữ –gõ một vài phím một đến vài lần – nhàm chán



4.2. Thiết bị nhập

- Nhận dạng chữ viết:
 - Không chính xác lắm tuy nhiên ngày càng được cải tiến
 - Tự nhiên hơn
 - Ví dụ: PDA
- Nhận dạng tiếng nói

4.2. Thiết bị nhập

- Chuột
 - Chuột hai bánh xe
 - Chuột bi
 - Chuột quang

4.2. Thiết bị nhập

□ Trackball

- Con chuột lật ngược – không cần diện tích



4.2. Thiết bị nhập

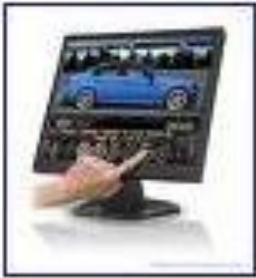
□ Joystick

- Một số laptop dùng loại này thay chuột (IBM Thinkpad)



4.2. Thiết bị nhập

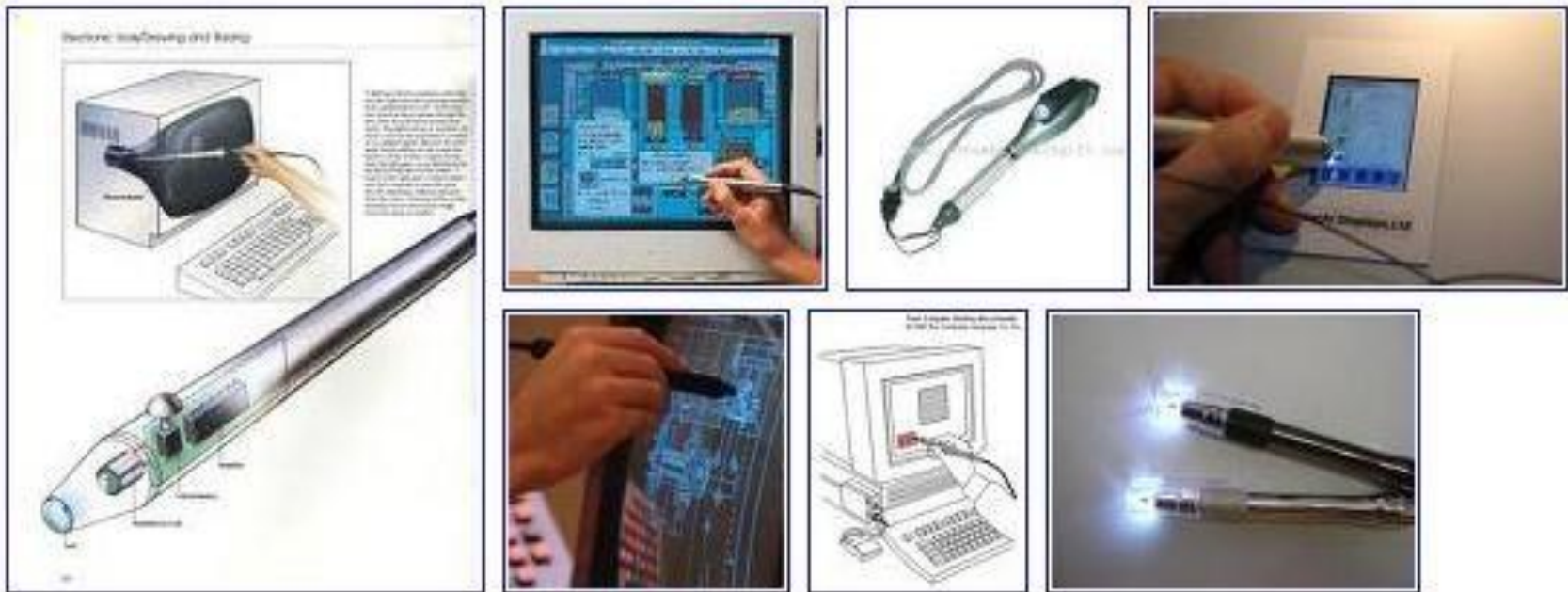
□ Màn hình cảm ứng



4.2. Thiết bị nhập

□ Light pen

- Giống súng trong các trò chơi bắn súng



4.2. Thiết bị nhập

- Bảng cảm ứng
 - Dùng nhiều trong các laptop



- Nhận tín hiệu từ mắt
- Dùng một số phím của bàn phím: lên, xuống, trái, phải

4.2. Thiết bị nhập

□ Chuột3D:

- Di chuyển trong không gian: vị trí
- Thêm 3 chiều nữa để điều khiển: góc lên xuống, hướng trái phải, góc quay quanh trục



4.2. Thiết bị nhập

□ Găng tay dữ liệu (dataglove)

- Tự nhiên
- Đắt



4.2. Thiết bị nhập

- ❑ Mũ thực tại ảo: Vừa là thiết bị ra, vừa là thiết bị vào – dựa vào chuyển động của đầu – thay đổi góc quay



- ❑ Thiết bị theo dõi chuyển động của cả người: Dùng các sensor dính vào người

4.2. Thiết bị xuất

- ❑ Màn hình CRT (Cathode Ray Tube)
- ❑ Màn hình LCD (Liquid Crystal Display)
- ❑ Màn hình 3 chiều
- ❑ Các thiết bị đơn giản khác như LED
- ❑ Các thiết bị phát âm thanh
- ❑ Máy in

Xin chân thành cảm ơn!

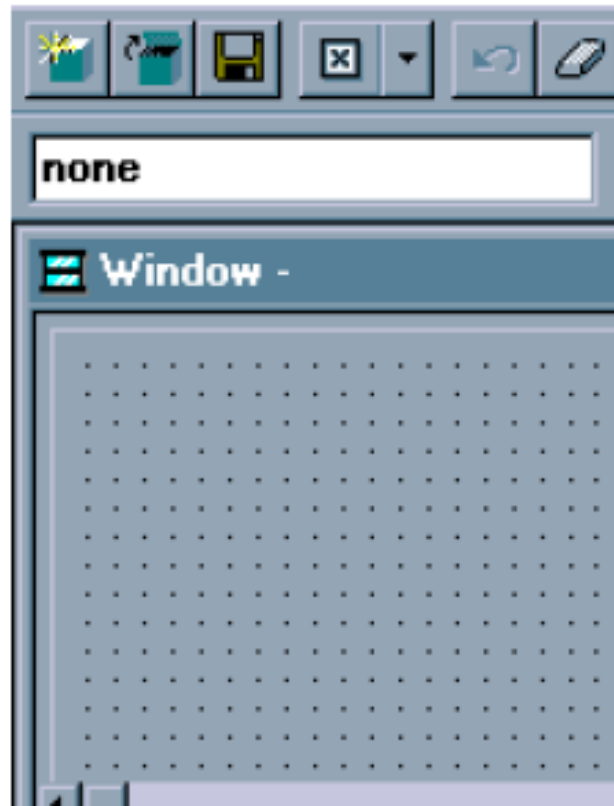
CHƯƠNG 5

THIẾT KẾ HCI HƯỚNG

NGƯỜI SỬ DỤNG

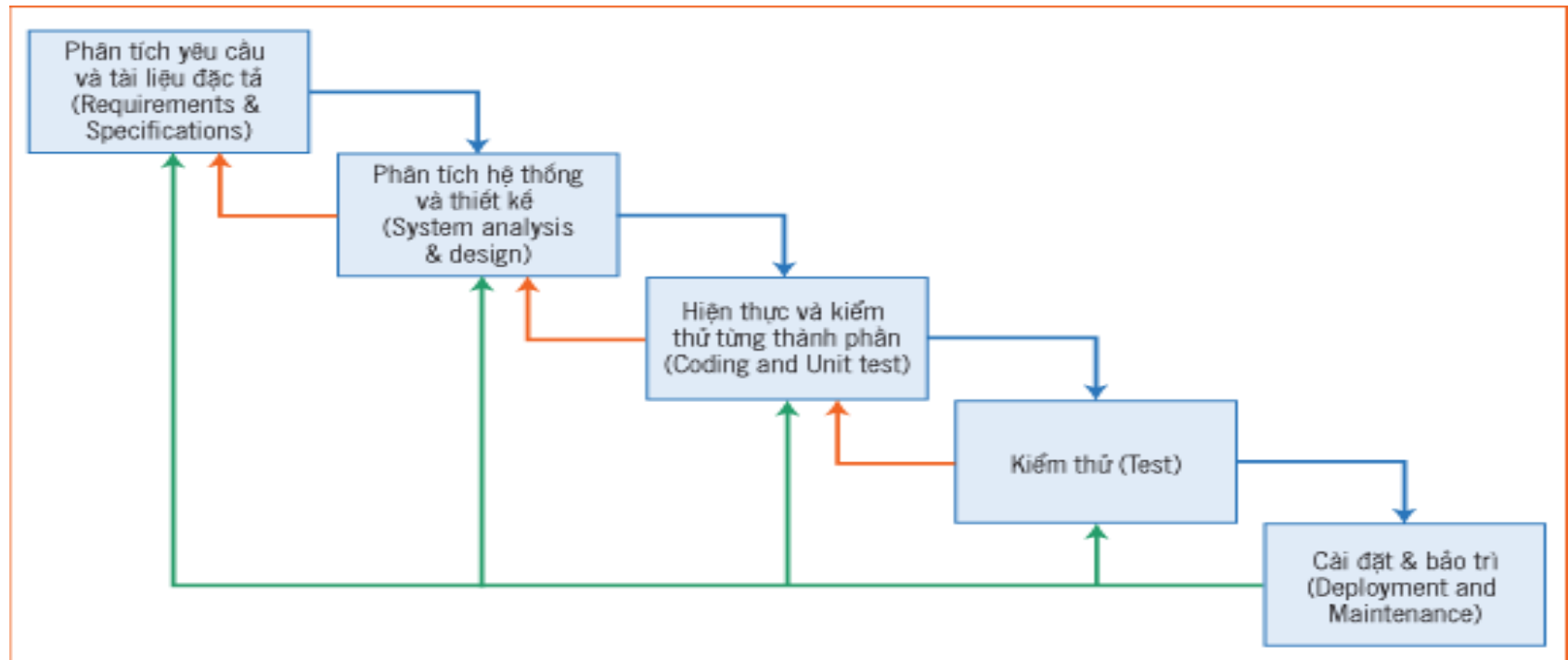
5.1. Ví dụ giao diện

□ Sybase PowerBuilder



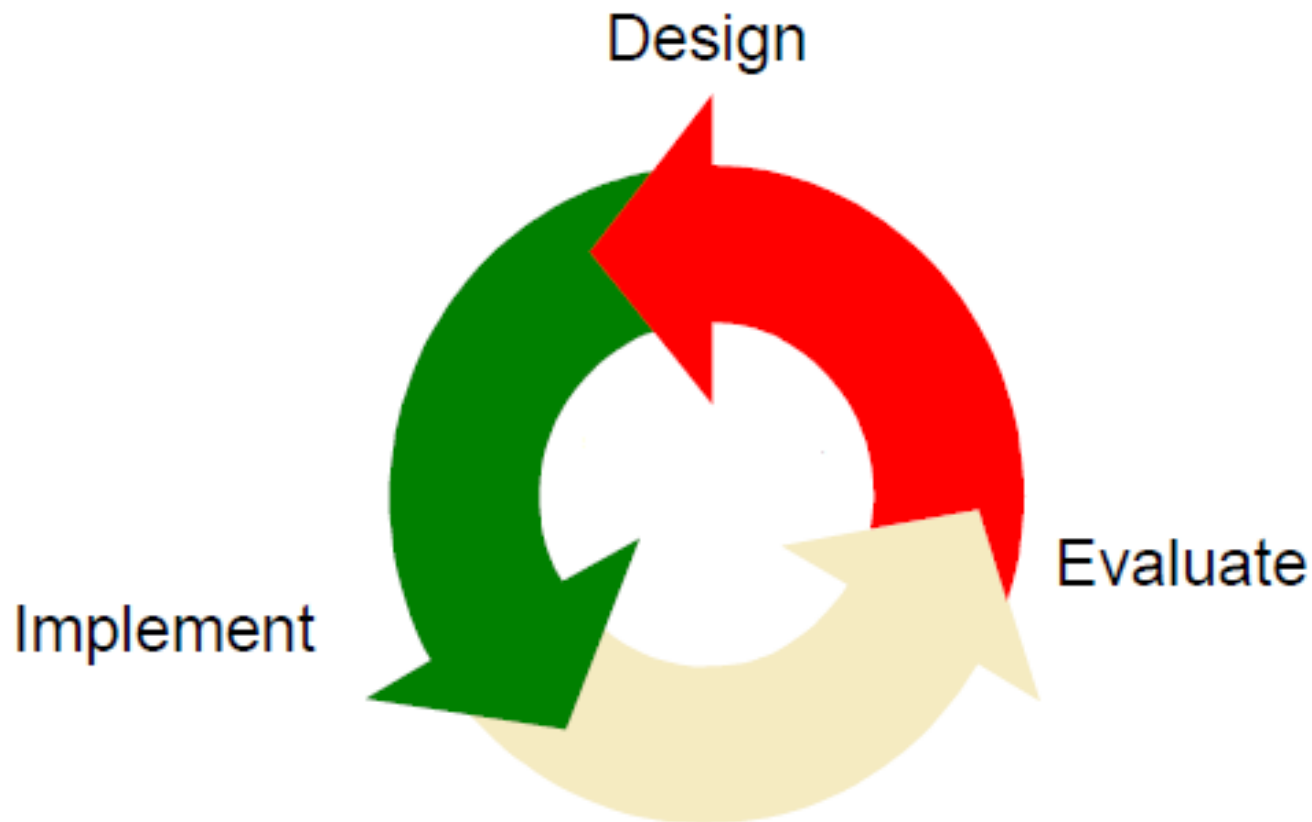
5.2. Các mô hình phát triển phần mềm

□ Mô hình thác nước



5.2. Các mô hình phát triển phần mềm

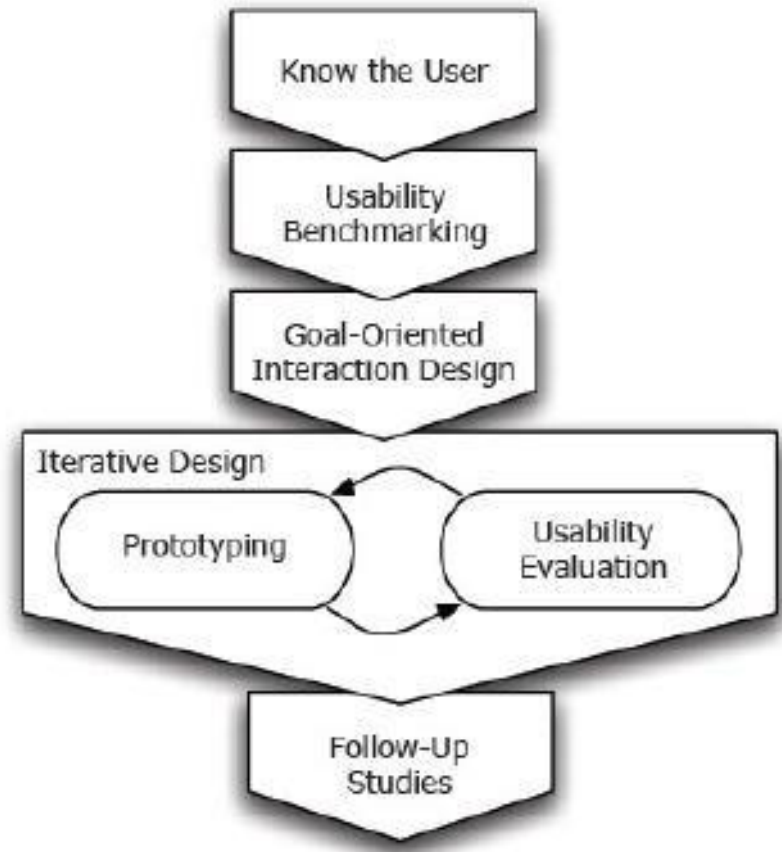
□ Mô hình thiết kế lặp



5.2. Vòng đời kỹ nghệ GDNSD

□ Các bước:

- Nhận biết ai là người sử dụng
- Đo tính sử dụng được
- Thiết kế tương tác hướng mục tiêu
- Thiết kế lặp(làm bản mẫu – đánh giá tính sử dụng được)
- Nghiên cứu phản hồi



5.2. Vòng đời kỹ nghệ GDNSD

- Nhận biết ai là người sử dụng
 - Quan sát, phỏng vấn số đông
 - Hình thành “user profile” cho các nhóm
 - Nhận biết mục tiêu, quan điểm của người sử dụng
 - Hình thành kịch bản người sử dụng điển hình

5.2. Vòng đời kỹ nghệ GDNSD

□ Đo tính sử dụng được

- Phân tích sản phẩm (giao diện) cạnh tranh theo kinh nghiệm
- Thiết lập đích tính sử dụng cho giao diện

5.2. Vòng đời kỹ nghệ GDNSD

- Thiết kế tương tác hướng mục tiêu
 - Thiết kế giao diện ban đầu hướng mục tiêu

5.2. Vòng đời kỹ nghệ GDNSD

□ Thiết kế lặp

- Thực hiện “thiết kế, kiểm thử, tái thiết kế”
- Xây dựng và đánh giá các bản mẫu
- Các yếu tố cần quan tâm
 - Phát hiện các vấn đề liên quan đến tính sử dụng được
 - Sửa lỗi, hình thành giao diện mới
 - Thu thập, lưu trữ các lý do thay đổi thiết kế
 - Đánh giá phiên bản giao diện mới theo giới hạn thời gian hay kinh phí

5.2. Vòng đời kỹ nghệ GDNSD

□ Nghiên cứu phản hồi

- Thu thập dữ liệu về tính sử dụng sau khi phân phát sản phẩm dùng thử
- Phương pháp tiến hành: quan sát, phỏng vấn, lấy ý kiến từ báo chí, thư từ, file log của phần mềm,...

5.3. Thiết kế hướng người sử dụng

- Mục đích
 - Thiết kế hệ thống sử dụng thuận tiện
- Tập trung vào
 - Mục tiêu, mô hình khái niệm, khả năng, ràng buộc của người sử dụng

5.3. Thiết kế hướng người sử dụng

□ Cách thực hiện

- Người thiết kế phải hiểu về người sử dụng: nhiệm vụ, yêu cầu về tính sử dụng,...
- Người sử dụng nên tham gia tích cực trong giai đoạn phân tích thiết kế
- Lập tiến trình thiết kế
- Người thiết kế và người sử dụng cùng nhau đánh giá hệ thống và đưa ra giải pháp cải tiến

5.4. Phân tích NSD và nhiệm vụ

- Mục tiêu của phân tích người sử dụng
 - Nhận biết ai là người sử dụng hệ thống?
 - Họ có kỹ năng gì, mức độ đến đâu?
 - Họ sử dụng hệ thống như thế nào?

5.4. Phân tích NSD và nhiệm vụ

- Nhiệm vụ của phân tích người sử dụng
 - Nhận biết các yếu tố quan trọng của người sử dụng
 - Tuổi, nam/nữ, dân tộc
 - Học vấn
 - Khả năng vật lý
 - Kinh nghiệm sử dụng máy tính
 - Các kỹ năng (gõ phím, đọc,...)
 - Kinh nghiệm chuyên môn, nghiệp vụ
 - Tần suất sử dụng hệ thống
 - Môi trường làm việc, xã hội
 - Quan hệ giao tiếp, nền văn hóa
 - ...

5.4. Phân tích NSD và nhiệm vụ

- Nhiệm vụ của phân tích người sử dụng
 - Phân nhóm người sử dụng theo các yếu tố
 - Các mức kỹ năng sử dụng máy tính và nghiệp vụ: Người mới, người học việc, người thành thạo, chuyên gia,...
 - Các mức yếu tố tần suất sử dụng hệ thống: Thường xuyên, thỉnh thoảng,...

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 1: Hệ thống cho phép mua tạp hóa và trả tiền
 - Ai là người sử dụng
 - Người mua tạp hóa
 - Tuổi từ 10 đến 80, khả năng vật lý (cao, tính năng động, sức mạnh,...)
 - Không có kinh nghiệm máy tính
 - Không được đào tạo
 - Có hiểu biết về thức ăn, không hiểu biết về nghiệp vụ kiểm kê của siêu thị
 - Khách hàng trong siêu thị thường hỏi nhau về cái mình cần

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 1: Hệ thống cho phép mua tạp hóa và trả tiền
 - Nhóm người sử dụng chính
 - Phụ nữ thường mua bán cho gia đình, thường dẫn theo trẻ em
 - Nhân viên bán hàng giúp khách hàng

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 2: Hệ thống kế toán cho doanh nghiệp nhỏ
 - Mức độ kỹ năng sử dụng máy tính
 - Bắt đầu: 15%
 - Học việc: 30%
 - Kinh nghiệm: 45%
 - Thành thạo: 10%

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 2: Hệ thống kế toán cho doanh nghiệp nhỏ
 - Mức độ kỹ năng lĩnh vực máy tính
 - Bắt đầu: 5%
 - Học việc: 15%
 - Kinh nghiệm: 50%
 - Thành thạo: 30%

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 2: Hệ thống kế toán cho doanh nghiệp nhỏ
 - Tần suất sử dụng máy tính
 - Người ít dùng: 20%
 - Người hay dùng: 80%

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 2: Hệ thống kế toán cho doanh nghiệp nhỏ
 - Sở thích môi trường đồ họa
 - Windows: 50%
 - MacIntosh: 35%
 - Khác: 10%
 - Không biết: 5%

5.4. Phân tích NSD và nhiệm vụ

- Ví dụ 2: Hệ thống kế toán cho doanh nghiệp nhỏ
 - → Tính toán để tìm ra nhóm người sử dụng chính dựa trên xác suất. Hệ thống này thiết kế chủ yếu cho nhóm người có kinh nghiệm sử dụng máy tính, có kinh nghiệm trong nghiệp vụ kế toán, thường xuyên sử dụng hệ thống trên môi trường Windows (có xác suất lớn)

5.4. Phân tích NSD và nhiệm vụ

□ Phân tích nhiệm vụ

- Thu thập dữ liệu về nhiệm vụ mà người sử dụng thực hiện
- Phân tích để hiểu sâu sắc nhiệm vụ
- Mô tả nhiệm vụ một cách rõ ràng
- Đảm bảo giao diện phù hợp với nhiệm vụ của người sử dụng

5.4. Phân tích NSD và nhiệm vụ

□ Phân tích nhiệm vụ

- Sự khác nhau giữa phân tích hệ thống và phân tích nhiệm vụ
 - Phân tích nhiệm vụ hướng tới hành động bên ngoài của user
 - Phân tích hệ thống hướng tới thiết kế bên trong của hệ thống
- Các câu hỏi cần trả lời khi phân tích nhiệm vụ
 - Người sử dụng làm cái gì
 - Họ làm việc bằng công cụ gì
 - Họ cần phải hiểu biết những gì khi làm việc

5.4. Phân tích NSD và nhiệm vụ

□ Phân tích nhiệm vụ

- Ví dụ: Với hệ thống quản lý thư viện, để đạt được mục tiêu mượn sách:
 - Đi đến thư viện
 - Tìm sách cần trên máy tính
 - Đến giá và lấy sách
 - Mang sách đến đăng ký với thủ thư và rời khỏi thư viện

5.4. Phân tích NSD và nhiệm vụ

□ Phân tích nhiệm vụ

■ Cách thực hiện: Phân rã chức năng

- Mục tiêu
- Tiền điều kiện
- Phân rã nhiệm vụ
- Nhiệm vụ phải được “đóng”, nếu không thì người sử dụng phải có khả năng kết thúc nhiệm vụ

5.4. Phân tích NSD và nhiệm vụ

- Phân tích nhiệm vụ
 - Phương pháp
 - Quan sát
 - Phỏng vấn

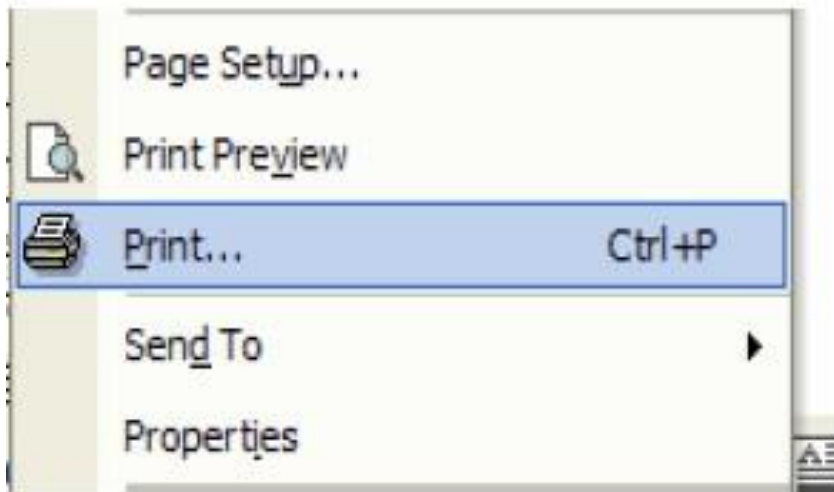
Xin chân thành cảm ơn!

CHƯƠNG 6

NGUYÊN LÝ THIẾT KẾ

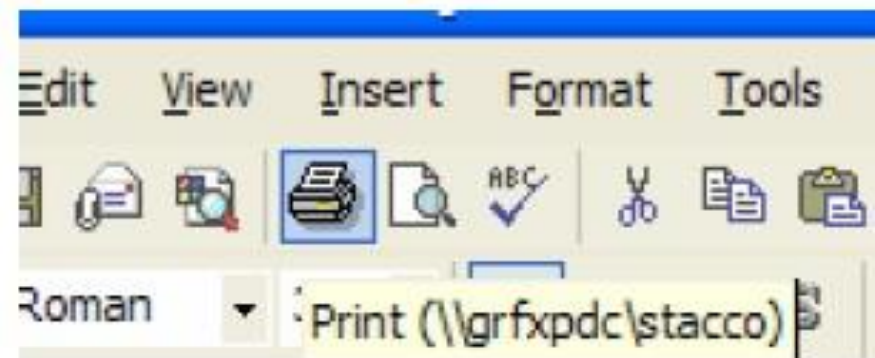
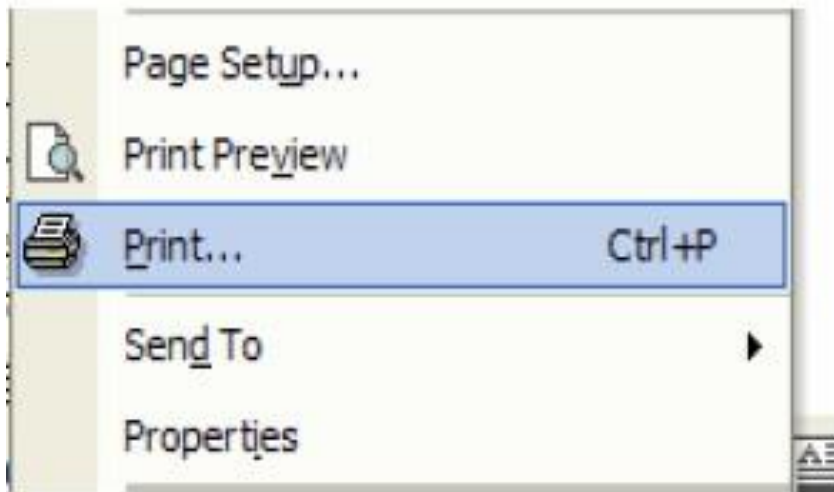
6.1. Ví dụ giao diện

- ❑ In ấn trong Microsoft Office
 - Sử dụng menu
 - Sử dụng thanh công cụ
 - Sử dụng tổ hợp phím



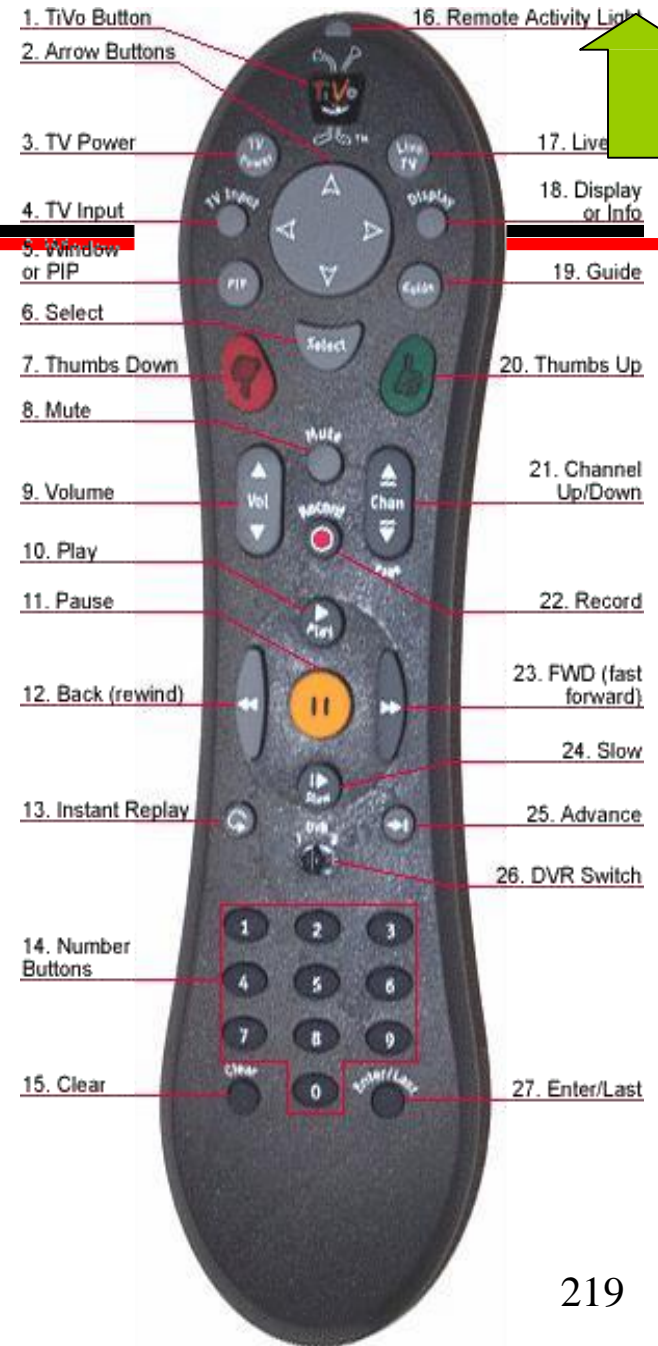
6.1. Ví dụ giao diện

- ❑ In ấn trong Microsoft Office
 - Hỗ trợ các dạng người dùng khác nhau
 - “Phi nhất quán”
- Đây là thiết kế tốt hay tồi?



6.1. Ví dụ giao diện

□ Điều khiển từ xa của tivi



6.1. Ví dụ giao diện

□ Điều khiển từ xa của tivi

- Tính đơn giản: Không quá nhiều phím, sử dụng phần mềm trên màn hình
 - Các phím quan trọng phải lớn và có hình dạng duy nhất
 - Các phím liên quan phải ánh xạ tự nhiên; kênh, tiếng ánh xạ đứng
 - Phím Pause phải lớn.
 - Thiết kế đồ họa tốt: Sử dụng ít màu, chỉ nhấn mạnh các phím quan trọng
 - Thiết kế công nghiệp: Hình dáng, kích thước phù hợp
- Tại sao phím Play lại nhỏ?

6.2. Hướng dẫn TK tính sử dụng

- Thiết kế hướng người sử dụng
 - Xác định ai là người sử dụng
 - Hiểu được nhiệm vụ của họ
- Luật Fitts
 - Kích thước của các điều khiển liên quan đến tầm quan trọng của nó
 - Kích thước điều khiển nhỏ rất khó chọn
 - Cạnh màn hình là quý giá

6.2. Hướng dẫn TK tính sử dụng

□ Bộ nhớ

- Sử dụng chia đoạn để đơn giản hóa trình diễn thông tin
- Tối ưu hóa bộ nhớ làm việc

□ Sử dụng màu

- Không phụ thuộc mạnh vào việc phân biệt các màu
- Tránh màu đỏ trên chữ xanh
- Tránh chi tiết là màu xanh nhỏ

6.2. Hướng dẫn TK tính sử dụng

- Nguyên lý Norman về thao tác trực tiếp
 - Sự gợi ý
 - Ánh xạ tự nhiên
 - Rõ ràng
 - Phản hồi

6.3. Hướng dẫn của Nielsen

- ❑ Phù hợp với thế giới thực
 - Phù hợp nhiều nhất với kinh nghiệm người sử dụng trong thế giới thực
- ❑ Nhất quán và chuẩn
 - Người sử dụng không bị bất ngờ với cách mà lệnh và giao diện làm việc
 - Việc tương tự thì hành động theo cùng cách như nhau
 - Các loại nhất quán
 - ❑ Nhất quán trong: Nhất quán ngay trong từng ứng dụng
 - ❑ Nhất quán ngoài: Nhất quán giữa các ứng dụng trong cùng platform
 - ❑ Nhất quán ẩn dụ: Nhất quán trong việc chọn ẩn dụ trong UI

6.3. Hướng dẫn của Nielsen

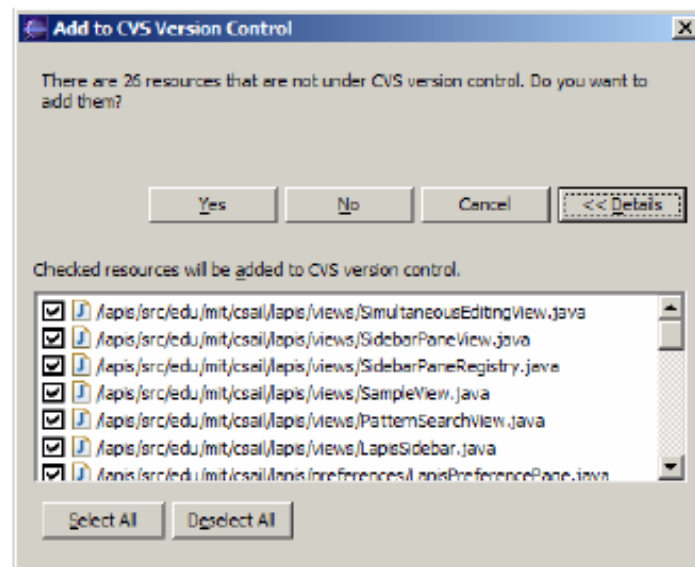
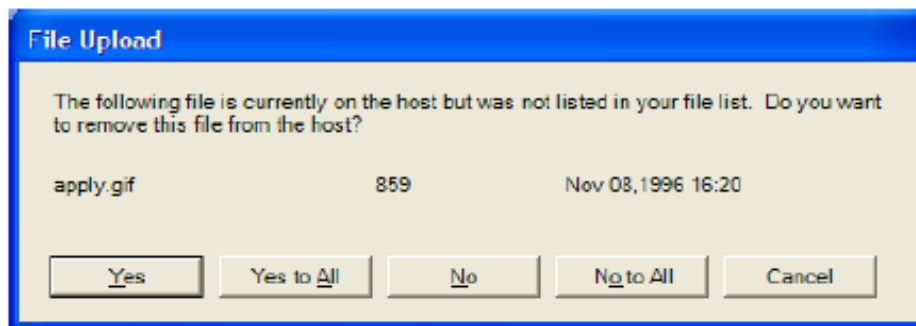
- Trợ giúp và tài liệu
 - Nhảy ngữ cảnh
 - Hướng nhiệm vụ
 - Cụ thể
 - Ngắn gọn
- Người sử dụng làm chủ

6.3. Hướng dẫn của Nielsen

- ❑ Quan sát trạng thái hệ thống
 - Thay đổi hình dạng con chạy
 - Sử dụng highlight
 - Sử dụng thanh trạng thái
 - Khuyến cáo
 - ❑ Hành động $<0.1s$, cảm giác xảy ra tức thì
 - ❑ Hành động $0.1 - 1s$, người sử dụng nhận ra hành động nhưng không cần phản hồi
 - ❑ Hành động $1 - 5s$, hiển thị con chạy busy
 - ❑ Hành động $>5s$, hiển thị progressbar

6.3. Hướng dẫn của Nielsen

❑ Mềm dẻo và hiệu quả

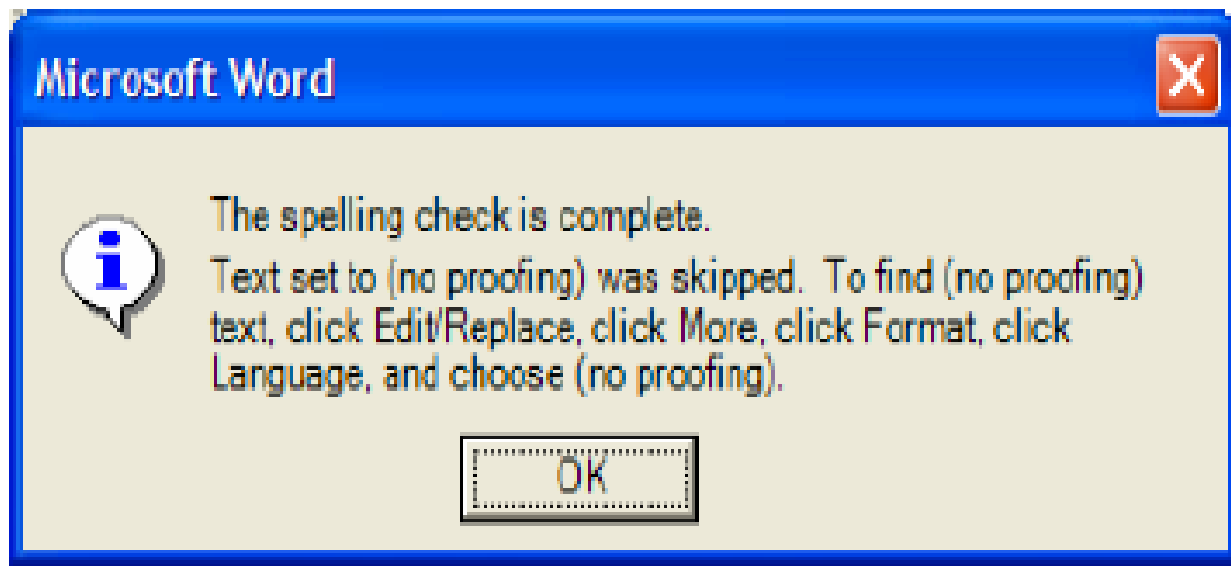


6.3. Hướng dẫn của Nielsen

- Quản lý lỗi
 - Tránh lỗi
 - Lỗi mô tả
 - Lỗi thu hút
 - Lỗi phương thức
 - Thông báo lỗi cần
 - Chính xác
 - Sử dụng ngôn ngữ người sử dụng
 - Có tính xây dựng
 - Lịch sự

6.3. Hướng dẫn của Nielsen

- ❑ Nhận dạng, không hồi tưởng
 - Giảm gánh nặng bộ nhớ của người sử dụng
 - Lựa chọn thay cho nhập
 - Dùng menu thay cho ngôn ngữ lệnh

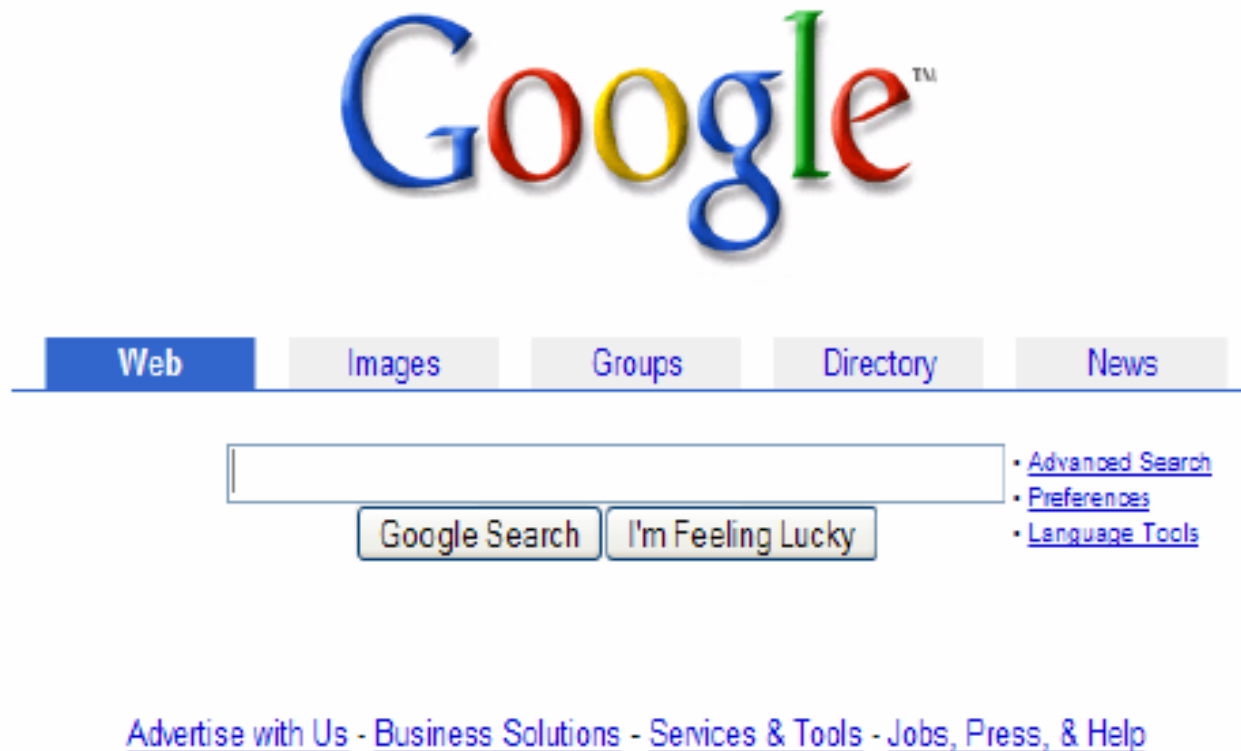


6.3. Hướng dẫn của Nielsen

- Thiết kế mỹ thuật và tối thiểu
 - Loại bỏ những thứ không có lý do có mặt
 - Không để trợ giúp quá dài ở cửa sổ chính
 - Loại bỏ những đồ họa không liên quan
 - Loại ra các đặc trưng không cần thiết

6.3. Hướng dẫn của Nielsen

□ Thiết kế mỹ thuật và tối thiểu



6.3. Hướng dẫn của Nielsen

- Thiết kế mỹ thuật và tối thiểu
 - Sử dụng ít màu và chọn màu phù hợp



6.4. Các quy tắc vàng của Shneiderman

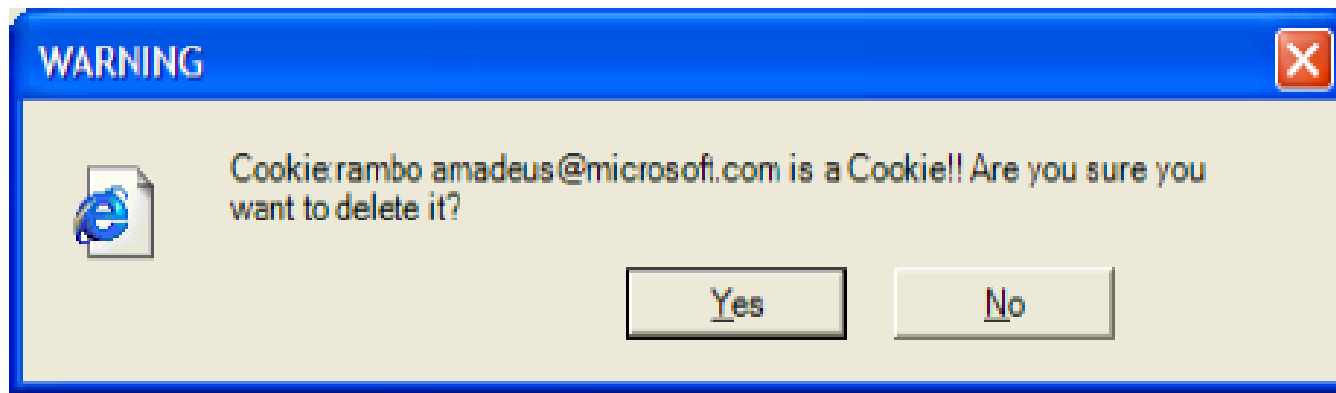
1. Nhất quán
2. Cho phép người dùng thường xuyên sử dụng phím tắt
3. Cung cấp thông tin phản hồi
4. Thiết kế các hộp thoại khi kết thúc
5. Đề phòng và xử lý lỗi đơn giản
6. Cho phép dễ dàng thực hiện hành động trở về
7. Người sử dụng làm chủ
8. Giảm thiểu tải bộ nhớ ngắn hạn

Xin chân thành cảm ơn!

- **CHƯƠNG 7**
- **MÔ HÌNH VÀO RA**

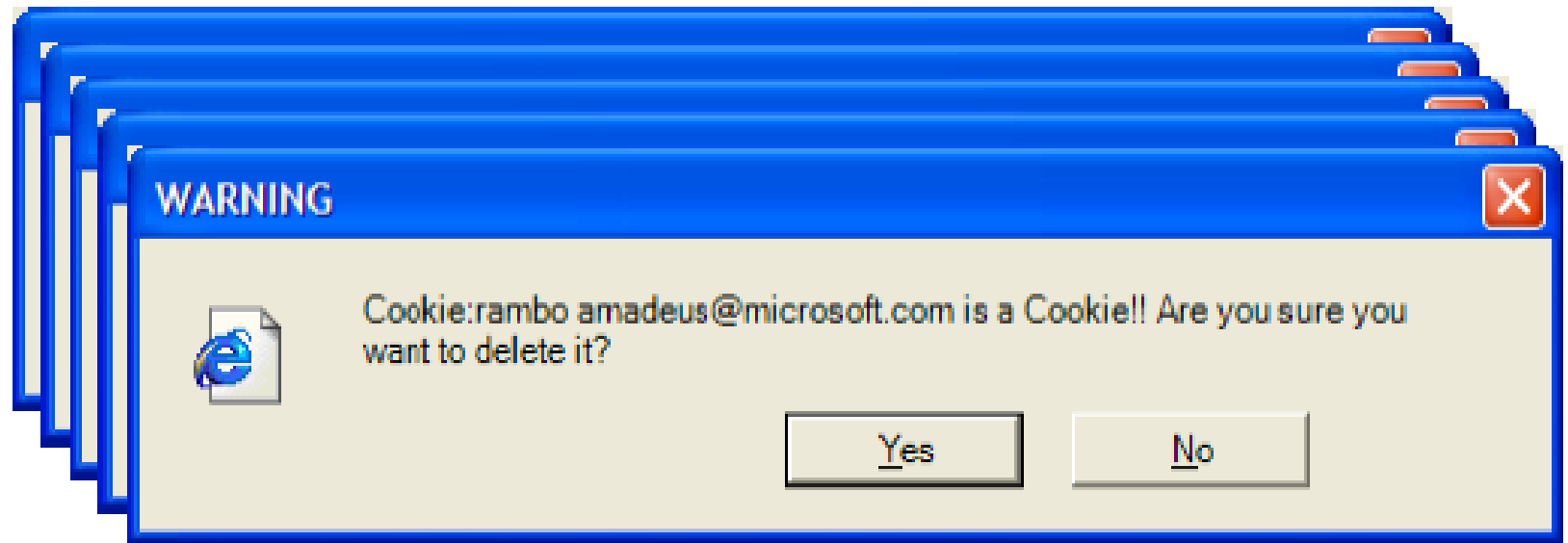
7.1. Ví dụ giao diện

- ❑ Thông điệp xuất hiện khi hủy nội dung của IE cache trong Windows



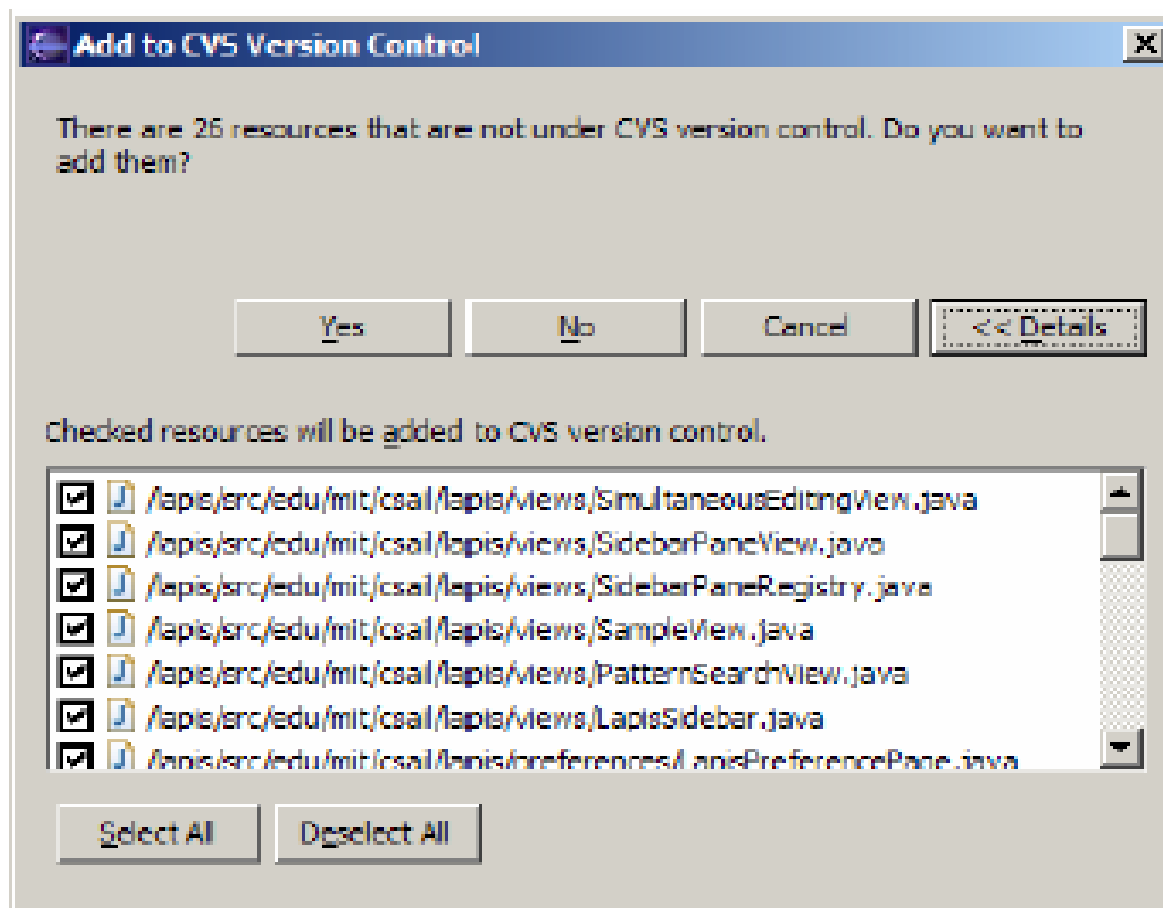
7.1. Ví dụ giao diện

- ❑ Thông điệp xuất hiện khi hủy nội dung của IE cache trong Windows



7.1. Ví dụ giao diện

□ IBM Eclipse



7.2. Mô hình ra

- ❑ Các thành phần (components)
 - Đối tượng đồ họa sắp xếp theo hình cây, có khả năng tự vẽ
 - lại
- ❑ Nét (strokes)
 - Vẽ đầu ra bằng cách gọi các hàm vẽ mức cao drawLine,
 - drawRectangle, drawArc, và drawText
- ❑ Pixel
 - Xem màn hình như mảng 2D các điểm ảnh và vẽ trực tiếp từng điểm ảnh.

7.2. Mô hình ra

→ Các ứng dụng GUI hiện đại đều sử dụng cả 3 mô hình trên: HTML (component); máy in laser Postscript (stroke input, pixel output); plotter (stroke input and output); PDF (stroke); LCD panel (pixel).

7.3. Phong cách tương tác-Metaphor

□ Ví dụ



7.3. Phong cách tương tác-Metaphor

□ Phong cách tương tác

■ Ngôn ngữ lệnh

- Trước đây là ngôn ngữ điều khiển công việc, tham gia vào dòng lệnh
- Ngày nay được nhúng vào phong cách tương tác khác
- Khi thiết kế cần chú ý thiết kế cú pháp

■ Thực đơn và form

- Trình diễn dãy thực đơn hay form tới người sử dụng
- Được nhúng vào phong cách tương tác khác (hộp thoại)
- Vấn đề quan trọng là thiết kế cấu trúc đường dẫn

7.4. Mô hình vào

□ Ví dụ: Google



7.4. Mô hình vào

□ Ví dụ: Google

■ Ưu điểm

- Thiết kế đẹp và tối thiểu
- Đặc trưng và liên kết không cần thiết bị loại bỏ
- Nạp xuống nhanh và dễ sử dụng

■ Hạn chế: Với người sử dụng lần đầu

- Google thực sự làm gì?
- Phải gõ gì vào hộp Text?
- Cụm từ “Google Search”, “I’m Feeling Lucky” có nghĩa gì?
- Trợ giúp ở đâu?

7.4. Mô hình vào

□ Sử dụng sự kiện cho việc vào dữ liệu

■ Giao diện người sử dụng trước đây

- Hiện thị dấu nhắc chờ người sử dụng nhập vào câu trả lời

→ Hệ thống làm chủ hoàn toàn đối thoại, làm chủ trình tự mà input và output xảy ra

■ Giao diện ngày nay

- Người sử dụng chủ động và tự do
 - Tự do nhấn chuột tại bất kỳ đâu trên giao diện
 - Kích hoạt lệnh bất kỳ nếu nó sẵn sàng
 - Tương tác với bất kỳ khung nhìn nào nếu nhìn thấy
- Đối thoại nghiêng mạnh về phía người sử dụng

7.4. Mô hình vào

□ Các loại sự kiện vào

■ Sự kiện thô(raw)

- Xuất phát từ trình điều khiển thiết bị
- Di chuột, nhấn, nhả phím chuột, bàn phím
- Nếu không hỗ trợ các yếu tố này thì là thiết kế tồi

■ Sự kiện chuyển đổi(translated)

- Chuyển sự kiện thô ở mức thấp thành sự kiện ở mức cao
- Sự kiện nhấn, nhả phím, chuột đã chuyển thành Keypressed, KeyDown, KeyUp, Click, MouseDown, MouseUp,...
- Tăng tính mềm dẻo cho lập trình viên

7.4. Mô hình vào

- Tính chất của sự kiện vào
 - Vị trí con chạy chuột
 - Trạng thái của phím chuột
 - Sự kiện vào bao gồm trạng thái phím bổ sung Shift, Alt, Control,...
 - Thời điểm nhận dữ liệu vào
 - Các sự kiện được lưu trữ trong hàng đợi

7.4. Mô hình vào

□ Hàng đợi sự kiện

- Đầu vào từ người sử dụng thường xuất hiện đột ngột
- Hàng đợi sự kiện cung cấp vùng đệm giữa người sử dụng và ứng dụng → ứng dụng không phải xử lý sự kiện ngay khi nó xuất hiện
- Sự kiện sườn như nhấn, nhả phím không thay đổi
- Sự kiện mô tả trạng thái liên tục có thể kết hợp thành sự kiện đơn với trạng thái cuối cùng
- Ví dụ: Di chuyển hình đồ họa lớn trên màn hình, vẽ đường bất kỳ bằng di chuột.

7.4. Mô hình vào

□ Vòng lặp sự kiện

- Đọc các sự kiện trong hàng đợi, phân phát chúng đến các thành phần phần mềm tương ứng trong phân cấp quan sát
- Các toolkit mức cao cung cấp vòng lặp ngay bên trong chúng (Java, VB, C#,...)
- Các toolkit mức thấp yêu cầu ứng dụng làm việc này

7.4. Mô hình vào

- Phân phát và truyền đạt sự kiện
 - Chọn thành phần phần mềm để nhận sự kiện
 - Nếu thành phần phần mềm từ chối quản lý sự kiện thì nó được truyền lên phân cấp quan sát cao hơn (cha) cho đến khi được quản lý. Nếu không được thành phần phần mềm nào quản lý thì sẽ bỏ qua

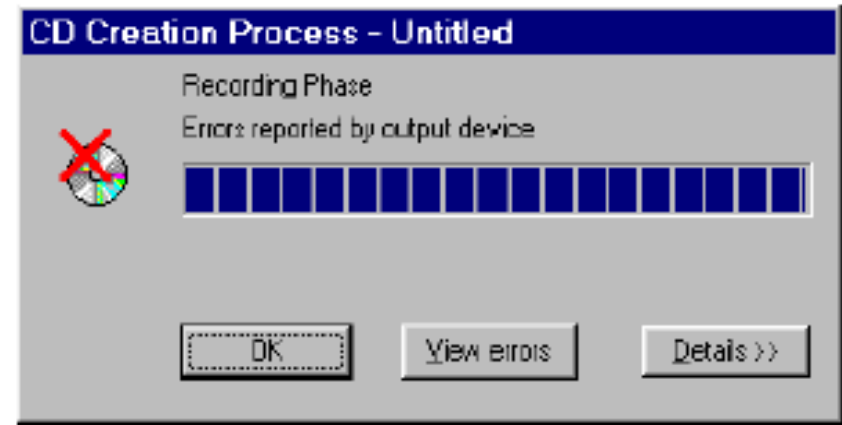
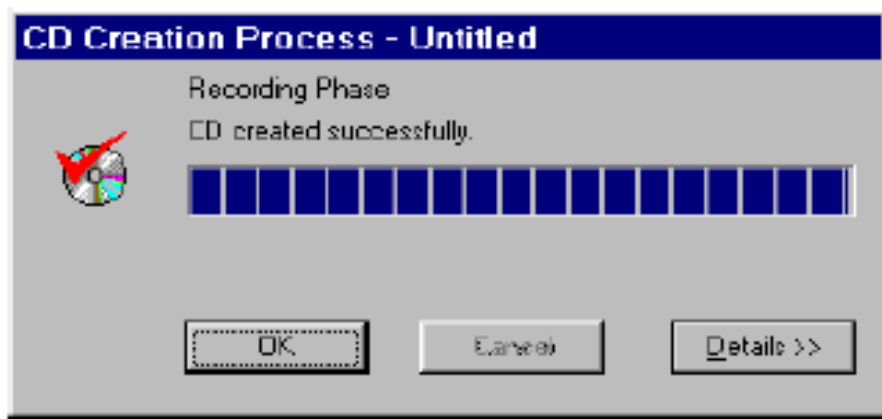
Xin chân thành cảm ơn!

CHƯƠNG 8

XÂY DỰNG BẢN MẪU

7.1. Ví dụ giao diện

❑ Phần mềm ghi đĩa CD Adaptec Easy CD Creator



7.2. Prototype là gì?

□ Khái niệm

- Là mô hình thu nhỏ
- Là bản thảo hay mô phỏng với khả năng dễ thay đổi của một phần UI
- Sử dụng trong nhiều lĩnh vực thiết kế khác nhau
- Độ trung thực khác nhau

7.2. Prototype là gì?

□ Khái niệm



7.2. Prototype là gì?

- Tại sao phải xây dựng prototype
 - Prototype được xây dựng nhanh hơn nhiều so với cài đặt cuối cùng → Đánh giá, nhận phản hồi sớm về ưu, nhược điểm của thiết kế
 - Với thiết kế khó giải quyết → Xây dựng nhiều prototype chứa các giải pháp khác nhau
 - Hỗ trợ giao tiếp giữa các đội ngũ phát triển và sử dụng UI

7.2. Prototype là gì?

□ Các loại prototype

- Dây các phác họa màn hình
- Storyboard (phim hoạt hình như dây các phác họa)
- Powerpoint slideshow
- Video mô phỏng sử dụng hệ thống
- Makét bằng bìa cứng
- Đoạn chương trình với chức năng hạn chế được viết bằng ngôn ngữ lập trình C#, Java...
- Mô hình vật lý

7.2. Prototype là gì?

□ Độ trung thực

- Là mức độ tương tự của prototype so với giao diện UI cuối cùng
- Prototype mức độ trung thực thấp thiếu nhiều chi tiết, sử dụng nguyên liệu rẻ tiền,...
- Prototype mức độ trung thực cao có mức độ tương tự cao với sản phẩm cuối cùng

7.3. Prototype giấy

□ Tổng quan

- Là lựa chọn tốt cho các vòng lặp thiết kế đầu tiên
- Là mô hình của giao diện chủ yếu được làm từ giấy
- Được phác họa bằng tay trên các mẫu giấy. Mỗi mẫu giấy biểu diễn các phần tử khác nhau
- Đáp ứng tương tác bằng cách
 - Xếp các mẫu giấy, viết các đáp ứng lên giấy
 - Dùng tay trở thay chuột
 - Viết trên mẫu giấy thay cho gõ phím

7.3. Prototype giấy

□ Tổng quan

■ Kiểm thử

- Giao nhiệm vụ cho người dùng, quan sát người dùng thực hiện

■ Đánh giá

- Độ trung thực thấp cả về hình dáng và cảm giác
- Có thể có độ trung thực cao theo chiều sâu với chi phí thấp

7.3. Prototype giấy

- Tại sao phải xây dựng prototype giấy?
 - Phác họa bằng tay nhanh hơn phác họa bằng máy tính, lập trình
 - Dễ thay đổi
 - Giúp người thiết kế tập trung trí tuệ vào bức tranh tổng thể, tránh chú ý vào chi tiết nhỏ nhặt
 - Tăng phản hồi nhận được từ người sử dụng
 - Không đòi hỏi các kỹ thuật đặc biệt

7.3. Prototype giấy

□ Một số ví dụ

A hand-drawn sketch of a user interface for a 'Computer Telephone'. The title 'Computer Telephone' is at the top. Below it are three input fields: 'Last name:', 'First name:', and 'Phone:'. At the bottom are two buttons: 'Place call' and 'Help'.

A clean, digital version of the 'Computer Telephone' interface. It features the title 'Computer Telephone' at the top. Below the title are three labels: 'Last Name:', 'First Name:', and 'Phone:'. At the bottom are two buttons: 'Place Call' and 'Help'.

7.3. Prototype giấy

□ Một số ví dụ

What to Do
Touch a different color or scan another item.

What you selected

JPG Stroller

☒ Green
☐ Red
☐ blue

Item	Style	Cost
JPG stroller	Green	98.00
~~~~~	~~~~~	~~~~~
~~~~~	~~~~~	~~~~~

tax: 10.00
Total: 124.97

All done?

What to do
Touch a different color, or scan another item.

What you selected

JPG Stroller
For children between 1-3 years old ...\$98.

☒ Green
☐ Blue
☐ Red (out of stock)

Item	Style	Cost
JPG Stroller	Green	98.00

tax: 6.98
Total: \$104.98

All done?

7.3. Prototype giấy

□ Một số ví dụ

The image shows a hand-drawn prototype of a music application interface on a blue background. The interface is divided into several sections:

- Top Section:**
 - On the left, a box labeled "IICM on AIR home".
 - Next to it, a login section with fields for "LOGIN" and "PASSWORD", and buttons for "WELL USER" and "SUBMIT".
 - To the right, the status "Status: running" is displayed.
 - Below the status, there's a field for "actual song:" with a wavy line underneath.
 - Further right are two buttons: "search song" and "request song".
 - On the far right, there's a "PLAYLIST" section with a "Song history" button.
 - At the top right corner, a button labeled "uploading Songs" is visible.
- Middle Section:**
 - A box labeled "Songarchive" is centered at the top of this section.
 - Below it is a table with the following headers: "#", "Songname", "Duration", "Rating:", "good", "vote!", and "Request".
 - A dropdown menu is open under the "Rating:" header, showing options: "great", "good", "fair", "bad", and "shitty".
- Bottom Section:**
 - A table with 5 rows, numbered 1 to 5 in the first column. The rest of the rows are empty, representing a list of songs.

7.3. Prototype giấy

- ❑ Các công cụ xây dựng maket giao diện
 - Bảng áp phích trắng (11”x14”)
 - ❑ Làm nền, khung cửa sổ
 - Số lượng các thẻ chỉ mục (4”x6”, 5”x8”)
 - ❑ Làm menu, window contents, dialog boxes,...
 - Hồ dán
 - ❑ Dán các mẫu giấy
 - Bảng giấy trắng
 - ❑ Text fields, checkbox, short message,...
 - Giấy bóng kính
 - ❑ Hilight, user “typing”
 - Máy sao chụp
 - ❑ Tạo nhiều blank
 - Bút, kéo,...

7.3. Prototype giấy

- ❑ Kinh nghiệm xây dựng prototype giấy
 - Kích thước nên lớn hơn thực tế
 - ❑ Ngón tay to hơn chuột
 - ❑ Chữ viết thường to hơn chữ in
 - ❑ Đặt trên bàn, quan sát từ xa
 - Không quan tâm nhiều đến màu sắc
 - ❑ Sử dụng màu đơn
 - Không cần trình diễn mọi hiệu ứng quan sát trên giấy
 - Tổ chức tốt các mẫu giấy nhỏ
 - Sử dụng kẹp giấy, phong bì thư

7.3. Prototype giấy

- ❑ Có thể biết gì, không biết gì từ prototype giấy
 - Người sử dụng cần phải hiểu ẩn dụ, mô hình khái niệm
 - Có thể nhận biết các vấn đề
 - ❑ Mô hình khái niệm: Người sử dụng có hiểu UI không?
 - ❑ Chức năng: Nó có là cái cần phải làm không? Có thiếu đặc trưng không?
 - ❑ Đường dẫn và luồng nhiệm vụ: Người sử dụng có thể tìm ra cách sử dụng? Phù hợp với thông tin tiên điều kiện?
 - ❑ Thuật ngữ: Người sử dụng có hiểu các nhãn không?
 - ❑ Nội dung màn hình: Cái gì sẽ xuất hiện trên màn hình

7.3. Prototype giấy

- Có thể biết gì, không biết gì từ prototype giấy
 - Khó có thể nhận biết các vấn đề
 - Mọi vấn đề về tính sử dụng được
 - Tính trung thực của đồ họa
 - Nhiệm vụ tương tác mà nó phụ thuộc vào cảm giác độ trung thực cao (luật fitts,...)
 - Thời gian đáp ứng

7.3. Prototype máy tính

□ Tổng quan

- Có độ trung thực cao về hình dáng và cảm giác
- Có độ trung thực thấp theo chiều sâu

7.3. Prototype máy tính

- ❑ Có thể biết gì từ prototype máy tính
 - Phản hồi động từ giao diện
 - ❑ Rõ ràng, phức tạp hay rối bời
 - ❑ Có thể tìm thấy các phần tử quan trọng
 - Bố trí màn hình
 - ❑ Lựa chọn phù hợp chưa?
 - Phản hồi tương tác
 - ❑ Có khả năng phản hồi dưới nhiều dạng khác nhau: trạng thái, thông điệp, hình dạng con trỏ,...
 - Luật fitts
 - ❑ Các điều khiển đủ lớn chưa? Có quá gần nhau không?...

7.3. Prototype máy tính

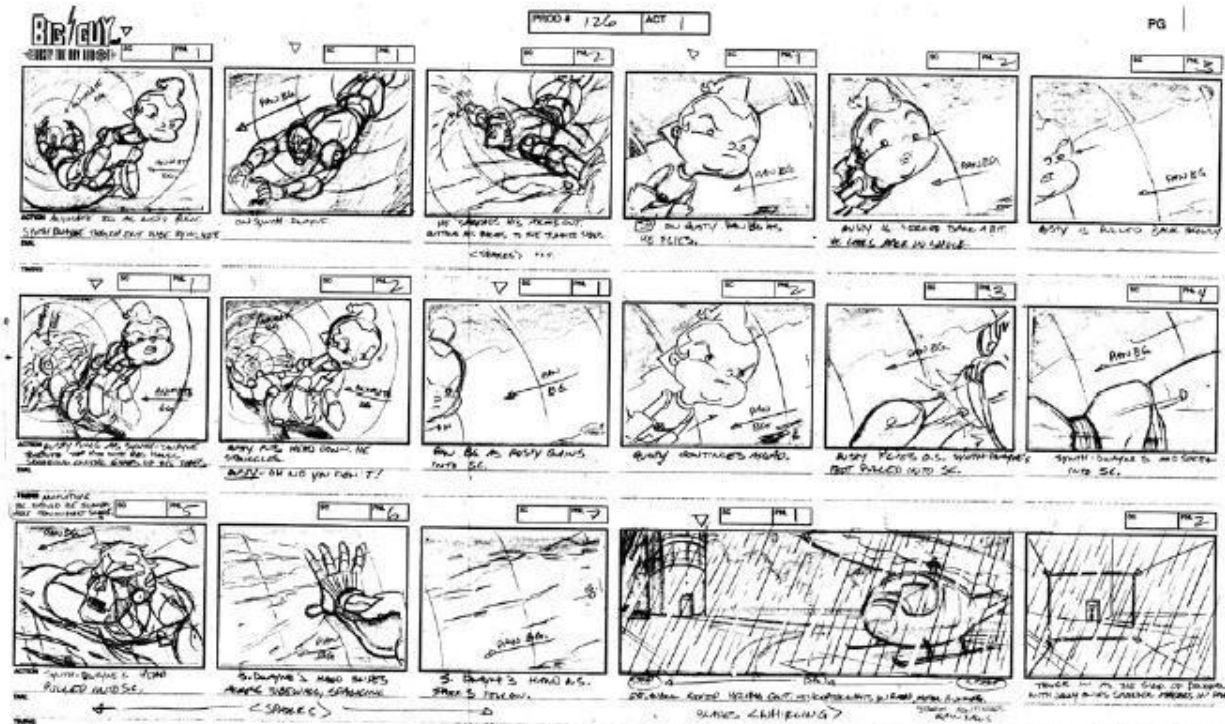
- Tại sao sử dụng prototype tool?
 - Nhanh hơn lập trình
 - Dễ sửa lỗi
 - Dễ thay đổi, dễ “vứt đi”

7.3. Prototype máy tính

□ Các kỹ thuật xây dựng prototype

■ Storyboard

- Là trình tự của màn hình cố định. Mỗi màn hình có một hoặc nhiều hotspots(hyperlink) có thể bấm chuột để nhảy tới màn hình khác



7.3. Prototype máy tính

□ Các kỹ thuật xây dựng prototype

■ Form builder

- Là công cụ để vẽ các giao diện thực sử dụng các widgets (button, label,...)
- Giảm công sức mã hóa, nhất quán ngoài
- Mang theo “suy nghĩ” của người thiết kế

7.3. Prototype máy tính

□ Công cụ xây dựng Storyboard

- Trình diễn PowerPoint chỉ là slide show. Nên tạo hyperlink để nhảy tới slide bất kỳ
- Macromedia Flash giúp xây dựng giao diện đa phương tiện. Hữu ích cho các thông tin phản hồi động.
- HTML. Mỗi màn hình là một bản đồ ảnh

7.3. Prototype máy tính

- Nhận xét Storyboard
 - Ưu điểm: Giống prototype giấy
 - Nhược điểm: Tĩnh

7.3. Prototype máy tính

- Form Builder

- HTML

- .NET Windows Forms, Eclipse Visual Editor, Borland Jbuilder, Sun NetBeans,...

Xin chân thành cảm ơn!

CHƯƠNG 9

THIẾT KẾ BIỂU TƯỢNG

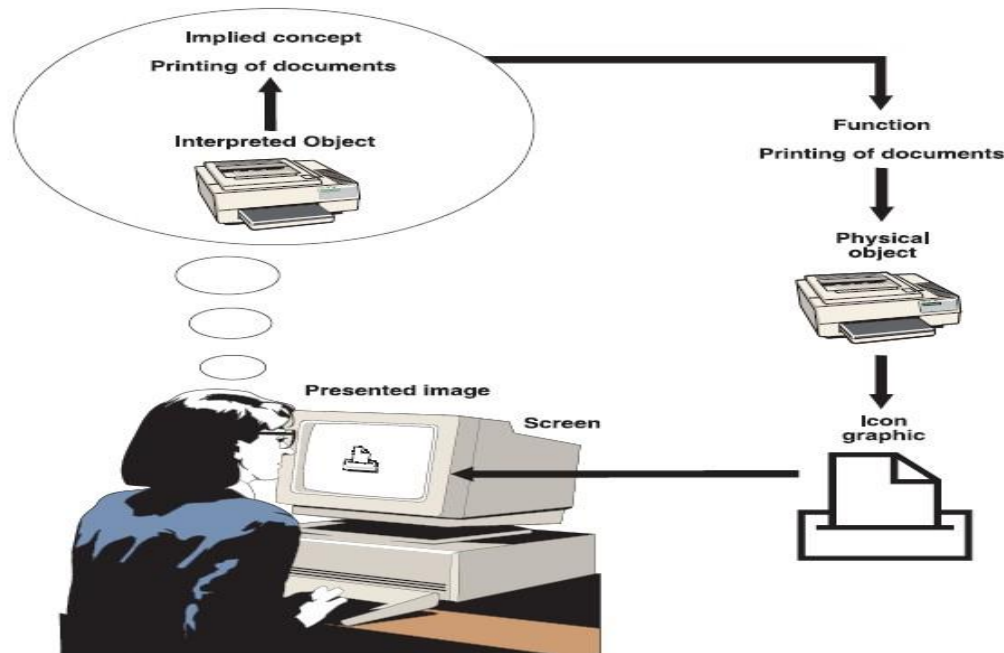
9.1. Tổng quan

- ❑ Các icon (biểu tượng/ hình vẽ) được sử dụng trên màn hình để thực hiện sự tương tác giữa các ứng dụng và ND.
- ❑ Các biểu tượng đồ hoạ có thể cung cấp sự độc lập về ngôn ngữ trong khi trao đổi thông tin với ND.
- ❑ Là một phần của giao diện đồ hoạ cung cấp khả năng học, hiểu và ghi nhớ các phần tử chức năng của hệ thống và trợ giúp ND khi thao tác với những phần tử đó.
- ❑ Thường một giao diện người dùng đồ hoạ – GUI sẽ cung cấp một biểu diễn ẩn dụ cho các nhiệm vụ của ND. Các biểu tượng có thể biểu diễn các ẩn dụ một cách trực tiếp, hay một đối tượng vật lí một cách gián tiếp.

9.1. Tổng quan

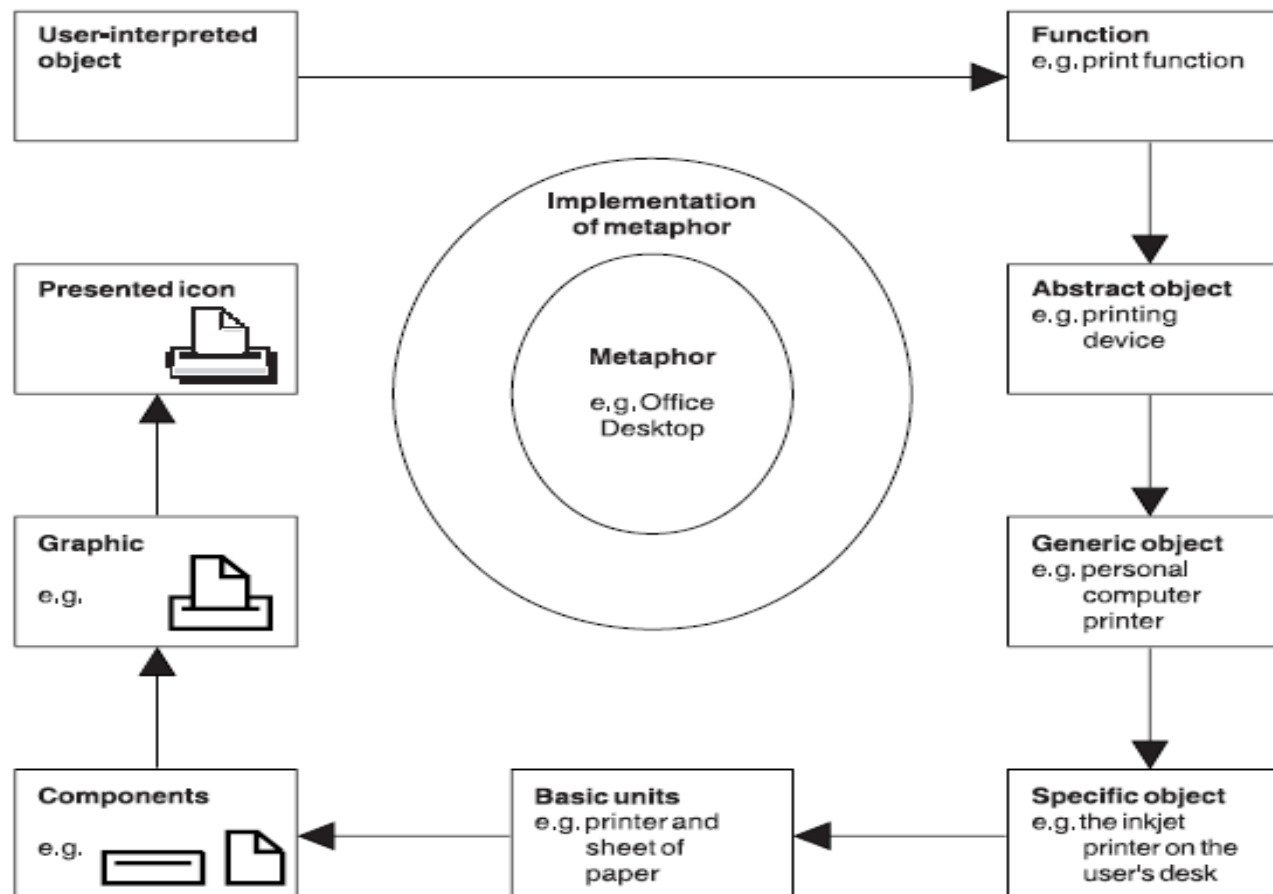
□ Các loại

- Biểu diễn các đối tượng vật lý như ổ đĩa, máy in,...
- Biểu diễn các chức năng của HT (action Icons)
- Biểu diễn các đối tượng điều khiển (control datas)



9.1. Tổng quan

□ Cấu trúc khái niệm của đối tượng biểu tượng



9.2. Các chuẩn thiết kế

- Có nhiều chuẩn cho thiết kế đảm bảo tính tiện dụng (Usability)

	Những nguyên lý và khuyến cáo	Các đặc tả
Sử dụng trong ngữ cảnh	ISO/IEC 9126-1: Kỹ nghệ phần mềm – chất lượng sản phẩm. Phần 1: chất lượng mẫu.	ISO 20282: Tính dùng được của sản phẩm quen thuộc.
	ISO/IEC TR 9126-4: Kỹ nghệ phần mềm – chất lượng sản phẩm. Phần 4: Chất lượng theo đơn vị đo sử dụng.	
	ISO/IEC 9241-11: Hướng dẫn về tính dùng được.	
Giao diện và tương tác phần mềm	ISO/IEC 9126-2: Kỹ nghệ phần mềm – chất lượng sản phẩm. Phần 2: Đo lường bên ngoài.	ISO/IEC 10741-1: Tương tác đối thoại – điều khiển con trỏ cho soạn thảo văn bản.
	ISO/IEC TR 9126-3: Kỹ nghệ phần mềm – chất lượng sản phẩm. Phần 3: Đo lường bên trong.	ISO/IEC 11581: Các ký hiệu và các hàm biểu tượng.
	ISO 9241: Những yêu cầu công thái học cho văn phòng làm việc với các thiết bị đầu cuối hiển thị trực quan. Các phần 10-17.	ISO/IEC 18021: Giao diện công nghệ thông tin – người dùng cho các công cụ di động.
	ISO 14915: Công thái học phần mềm cho các giao diện người dùng đa phương tiện.	

9.2. Các chuẩn thiết kế

Giao diện phần cứng	ISO 11064: Thiết kế công thái học của các trung tâm điều khiển.	ISO 9241: Những yêu cầu công thái học cho văn phòng làm việc với các thiết bị đầu cuối hiển thị trực quan. Các phần 3-9.
		ISO 13406: Các yêu cầu công thái học cho các công việc với hiển thị trực quan dựa trên các bảng phẳng.
		ISO/IEC 14754: Các giao diện Pen-based – các hành động chung để soạn thảo văn bản trong một hệ Pen-based.
		ISO 18789: các yêu cầu công thái học, các kỹ thuật đo lường cho các màn hình trực quan điện tử.
Tài liệu	ISO/IEC 18019: Các nguyên tắc chỉ đạo (guidelines) cho thiết kế và sự chuẩn bị của tài liệu người dùng phần mềm.	ISO/IEC 15910: quá trình dẫn chứng bằng tài liệu người dùng phần mềm.
Quá trình phát triển	ISO 13407: Các quá trình thiết kế hướng con người cho các hệ tương tác.	ISO/IEC 14598: Công nghệ thông tin – Sự đánh giá các sản phẩm phần mềm.
	ISO TR 16982: Các phương pháp khả dụng hỗ trợ thiết kế hướng con người.	

9.2. Các chuẩn thiết kế

□ Cách tiếp cận chuẩn

■ Các chuẩn ISO hướng xử lý (Process-Oriented ISO Standards)

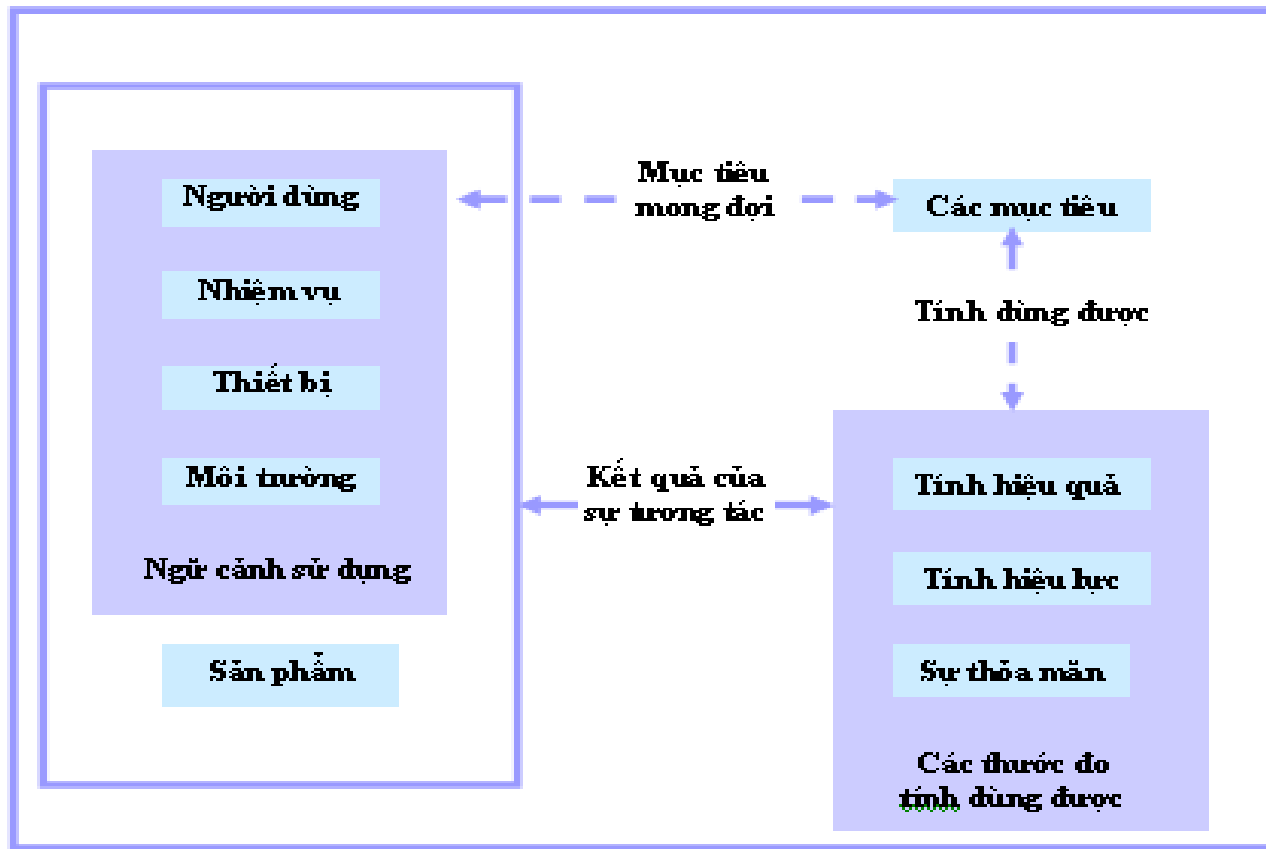
- ISO 9241-11 Hướng dẫn tính dùng được
- ISO 13407 : Các quá trình thiết kế người dùng trung tâm cho các hệ thống tương tác
- ISO/TR 18529 : Công thái học cho tương tác người dùng – hệ thống . Các mô tả quá trình , chu trình , vòng đời hướng người dùng
- ISO/TR 16982 : Công thái học cho tương tác người dùng – hệ thống. Các phương pháp tính dùng được hỗ trợ cho thiết kế hướng người dùng

■ Các chuẩn ISO hướng sản phẩm (Produit-Oriented ISO Standards)

- ISO/IEC 9126 – 1 : mô hình cho chất lượng của sản phẩm dựa trên công nghệ , kỹ thuật
- ISO/IEC 9126 – 2 : tính dùng được trong các thước đo chất lượng bên trong và bên ngoài

9.2. Các chuẩn thiết kế

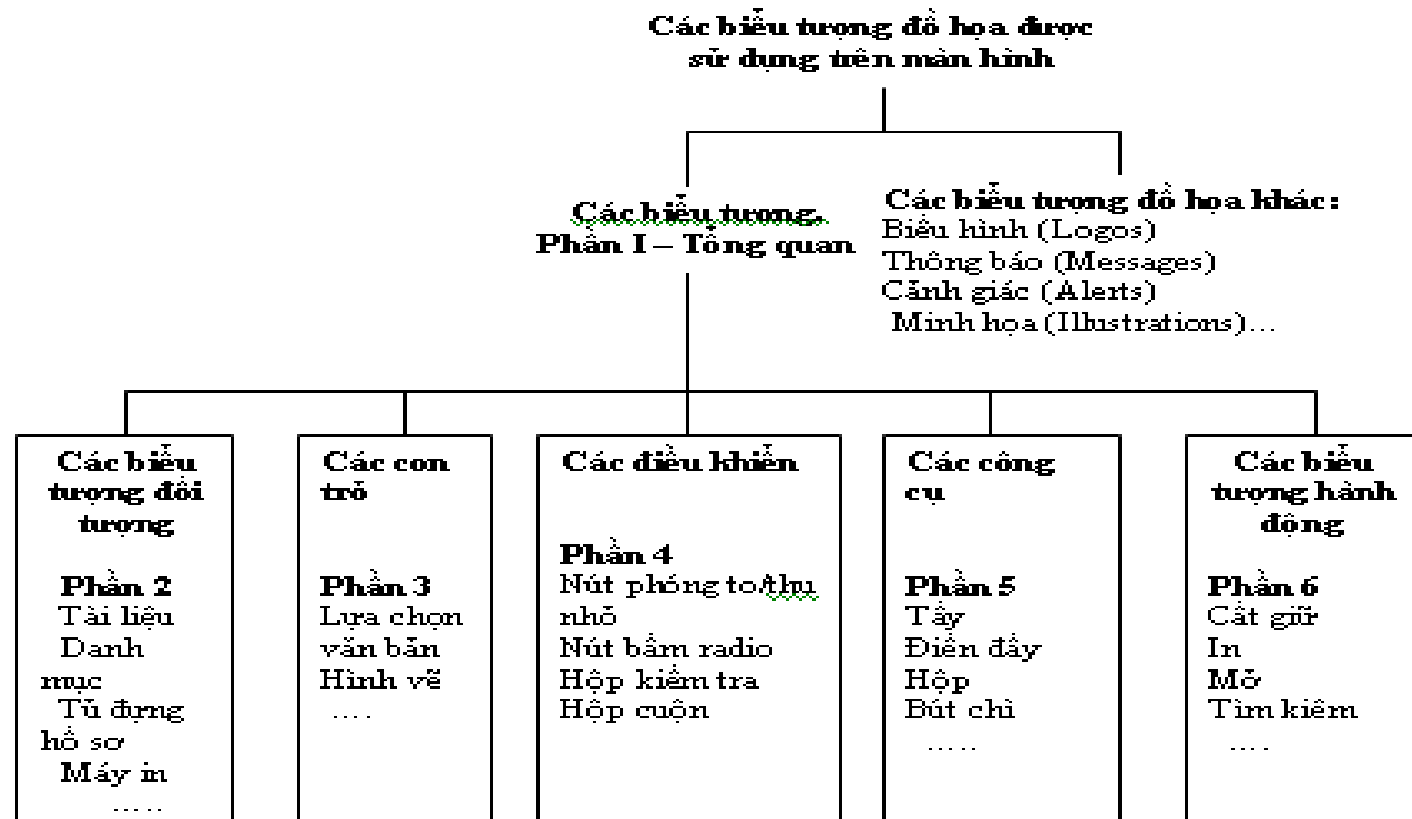
- Cách tiếp cận chuẩn
 - Chuẩn ISO 9241 cho tính tiện dụng



9.2. Các chuẩn thiết kế

□ Cách tiếp cận chuẩn

■ Chuẩn ISO 11581 cho thiết kế ICON



9.2. Các chuẩn thiết kế

▣ 7 nguyên tắc thiết kế

1. Approach Icon Design Holistically

"If you need to draw several icons, you need to think over images for the whole set of icons before proceeding with illustrating activities." This is one of two major points made in this article on icon design. He goes on to explain how failing to plan how the whole set of icons will work together from the beginning will ensure a huge waste of time, as redesign will be inevitable.



Add invoice



Delete invoice



Send invoice



Show
unsent invoices



Show
sent invoices



Show
paid up invoices



Show
all invoices

9.2. Các chuẩn thiết kế

▣ 7 nguyên tắc thiết kế

2. Consider Your Audience

Symbols may differ for common elements you may use for your designs.

An important aspect here is national characteristics. Cultural traditions, surroundings and gestures can differ radically from country to country."



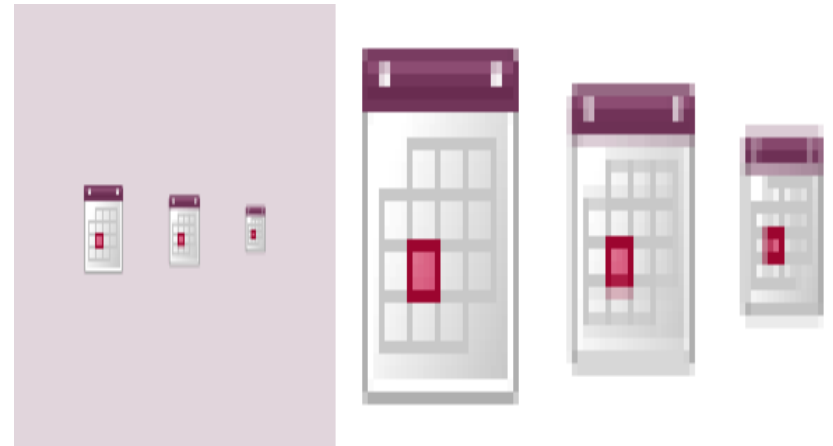
Mail

9.2. Các chuẩn thiết kế

▣ 7 nguyên tắc thiết kế

3. Design for the Size the Icon will be Used At

The approach taken for small icons and large icon design is immensely different. Firewheel has a good article that covers the scaling subject called [Icon Design: Bitmap vs Vector](#). Also, review this article on [Icon Design Sizing](#) over at Mezzoblue. It covers some inherent issues with designing icons for small sizes.



9.2. Các chuẩn thiết kế

□ 7 nguyên tắc thiết kế

4. Keep Icons Simple and Iconic

The icons below look really cool. It requires a judgment, though, as to whether the loss of some of the quick recognition of the symbol is worth the added design around the symbol. At a large size it works just fine, as they function similar to illustrations. At smaller sizes though, a less-dressed solution may be preferable.



9.2. Các chuẩn thiết kế

□ 7 nguyên tắc thiết kế

5. Cast Consistent Lighting, Reflections, and Shadows

The guidebook gives really specific rules for the Vista Icon set. This gives more exacting standards for icon designers and ensures a unified icon system.

Following is a specific rule to see an example, "Use shadows to lift objects visually from the background, and to make 3D objects appear grounded, rather than awkwardly floating in space." There are many more rules in this guide.

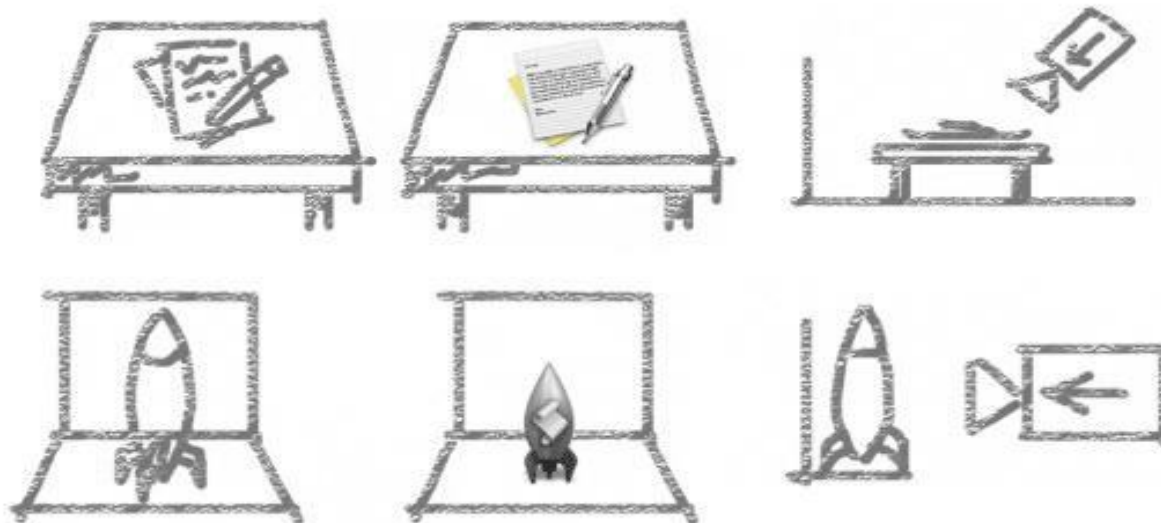


9.2. Các chuẩn thiết kế

□ 7 nguyên tắc thiết kế

6. Utilize a Limited Perspective

"The various perspectives are achieved by changing the position of an imaginary camera capturing the icon." The image below shows the difference in perspective between an Application Icon (*Top*) and a Toolbar Icon (*Bottom*).



9.2. Các chuẩn thiết kế

□ 7 nguyên tắc thiết kế

7. Create Consistent Icon Set Styles

Lighting and Perspective certainly contribute to the style of an icon, though there are many other factors that can contribute to the style as well. If you're trying to fit your icon into a grunge-style Web site design, you'll likely be adding texture to the style of the icon's design.

Echo



9.2. Các chuẩn thiết kế

▣ 10 lỗi hay mắc phải khi thiết kế

#1 Insufficient differentiation between icons

Sometimes within one set of icons, we have icons that look alike and it is very hard to understand what is what. If you miss the legends, you can very easily get the icons mixed up.



Terminal



Activity Monitor



Console

Icons from the Utilities section in Mac OS X. I am always getting them confused and launching the wrong application.

9.2. Các chuẩn thiết kế

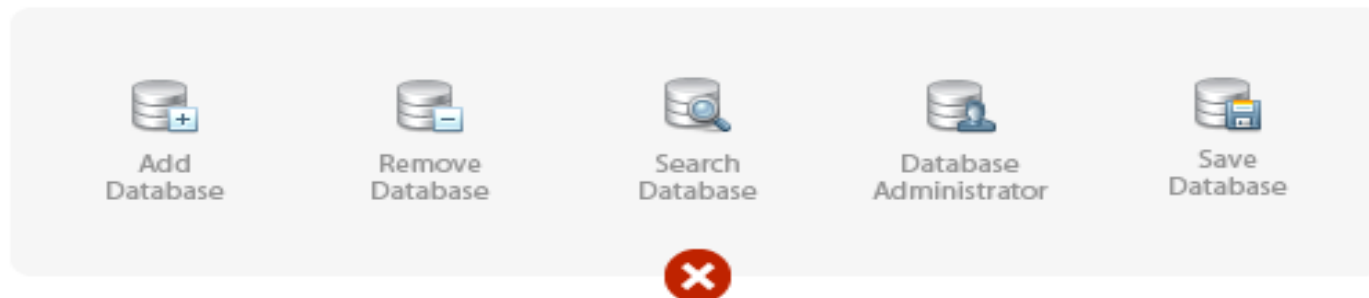
❑ 10 lỗi hay mắc phải khi thiết kế #2 Too many elements in one icon

- ❑ The simpler and more laconic the icon, the better. It is preferable to keep the number of objects in a single icon to a minimum.
- ❑ Nevertheless, Microsoft's designers, inspired by the new format of icons featured in Windows Vista, decided to go big and drew bloated icons to justify their bloated budget:



9.2. Các chuẩn thiết kế

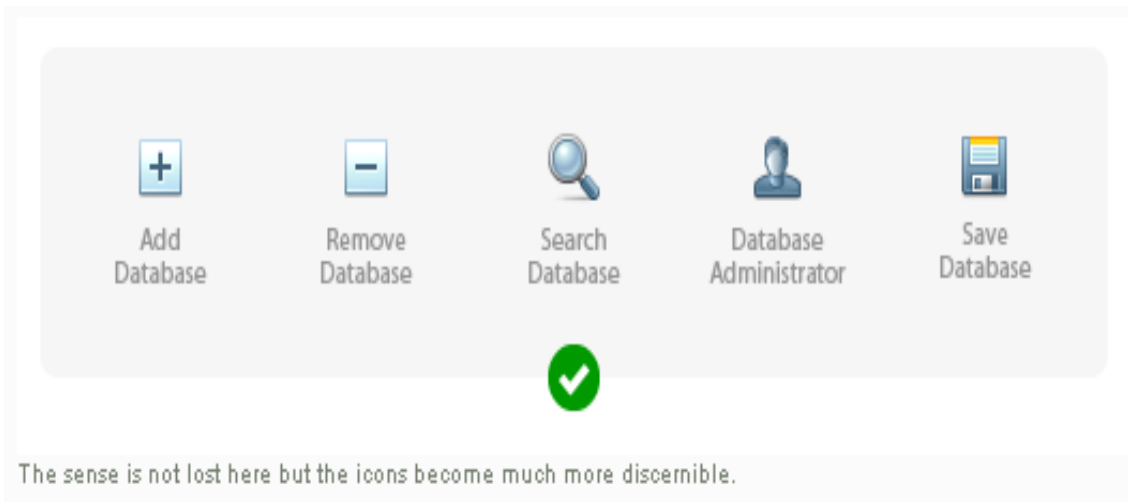
- ❑ 10 lỗi hay mắc phải khi thiết kế
- ❑ #3 Unnecessary elements
 - ❑ An icon should be easy to read. The fewer elements it has, the better. It is better if the whole image is relevant and not only part of it. Therefore, you have to pay attention to the context of using icons.
 - ❑ Let us take for instance some database icons: The simpler and more laconic the icon, the better. It is preferable to keep the number of objects in a single icon to a minimum.



At first glance everything looks alright.

9.2. Các chuẩn thiết kế

- 10 lỗi hay mắc phải khi thiết kế
 - But if this application (or a separate toolbar) deals only with databases, we can (and should) remove the unnecessary part:



- The sense is not lost here but the icons become much more discernible (nhận thức được rõ hơn).

9.2. Các chuẩn thiết kế

▣ 10 lỗi hay mắc phải khi thiết kế

#4 Lack of unity of style within a set of icons

It is a unity of style that unites several icons into a set. The uniting property can be any of the following: color scheme, perspective, size, drawing technique or a combination of several such properties. If there are only a few icons in the set, the designer can keep some rules in his head. If there are many icons in the set and there are several designers working on them (for instance, icons for an operating system), then special instructions are created. Such instructions describe in detail how to draw an icon so that it fits straight into the set.

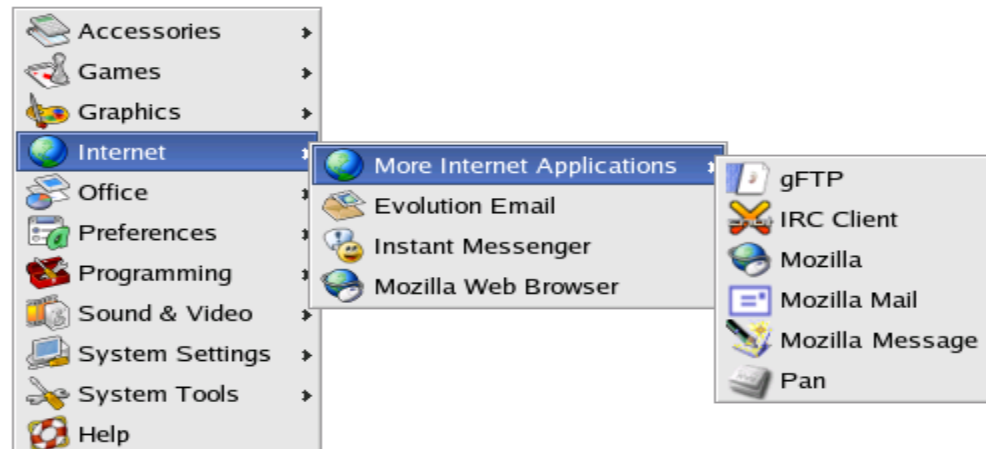


9.2. Các chuẩn thiết kế

▣ 10 lỗi hay mắc phải khi thiết kế

#5 Unnecessary perspective and shadows in small icons

- ▣ Progress does not stand still: interfaces have gained the potential to display semi-transparent objects, lost the limitation on the number of colors and there is now a trend towards 3D icons. But is it really all that useful? Not always! Especially if we are talking about icons sized 16×16 or smaller.
- ▣ For example, let us take the application manager from GNOME 2.2.0 (RedHat 9):

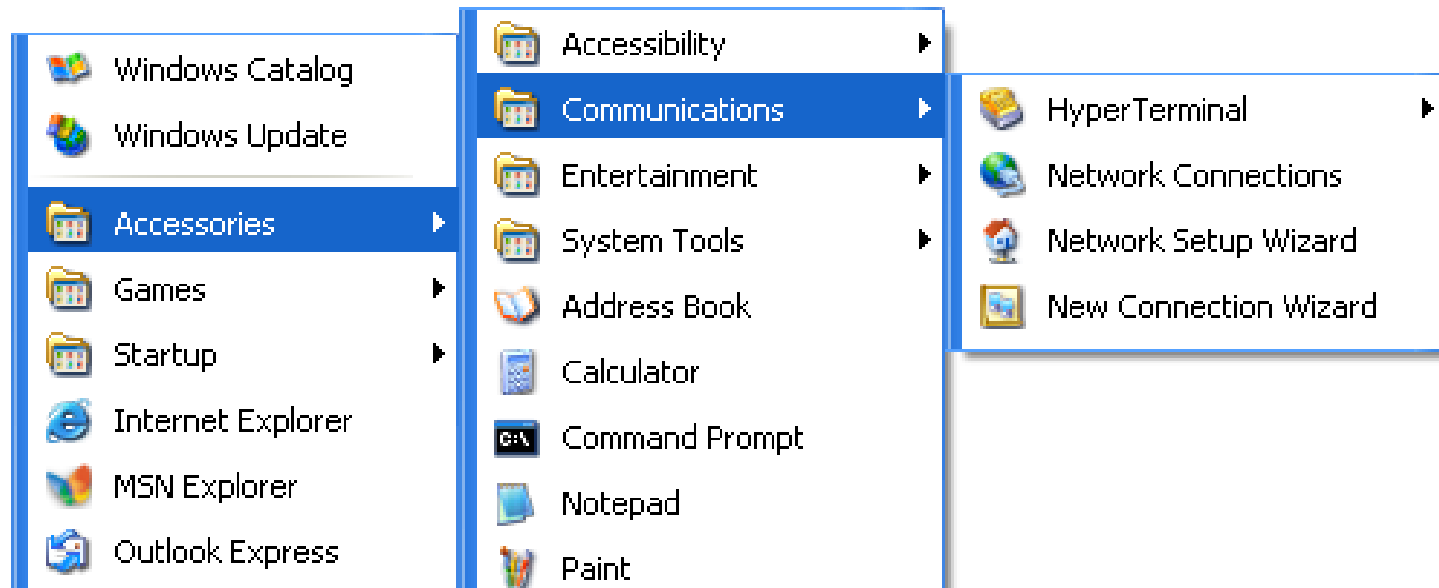


9.2. Các chuẩn thiết kế

❑ 10 lỗi hay mắc phải khi thiết kế

Perspective in icons of such minute size is unnecessary and even counter-productive.

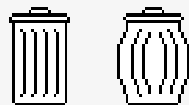
And here is the application manager from Windows XP:



9.2. Các chuẩn thiết kế

■ 10 lỗi hay mắc phải khi thiết kế #6 Overly original metaphors

- Selecting what is to be displayed in an icon is always a compromise between recognizability and originality. Before a metaphor (image) is developed for an icon it is wise to see how it is done in other products. Maybe the best solution lies not in coming up with something original but rather in adopting the existing solution.
- An example of excessive originality is the bin icon in OS/2 Warp 4, which is not actually a bin at all but a shredder.



Mac OS System 4



Lisa Office System



Windows 98



OS/2 Warp 4



9.2. Các chuẩn thiết kế

❑ 10 lỗi hay mắc phải khi thiết kế

#7 National or social characteristics not being taken into account

- ❑ It is always necessary to take into account the conditions in which your icon is going to be used. An important aspect here is national characteristics. Cultural traditions, surroundings and gestures can differ radically from country to country.
- ❑ Let us suggest that we need to draw an icon for working with e-mail. It makes perfect sense to use a metaphor of real paper mail. A mailbox for example.

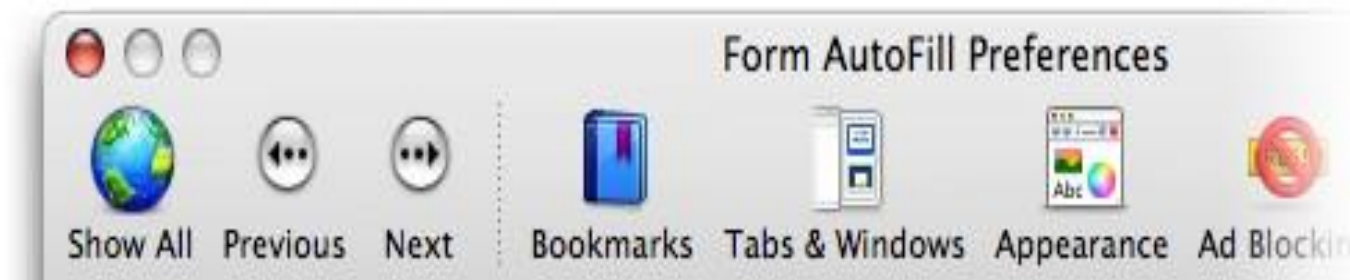
9.2. Các chuẩn thiết kế

❑ 10 lỗi hay mắc phải khi thiết kế

#8 Images of real interface elements in icons

- ❑ [The manual on creating icons for Mac OS X](#) warns us: “Avoid using Aqua interface elements in your icons; they could be confused with the actual interface.” But all in vain! For instance, take a look at the following icon:

- ❑ Here is an interesting example from the OmniWeb browser interface:



9.2. Các chuẩn thiết kế

❑ 10 lỗi hay mắc phải khi thiết kế #9 Text inside icons

- ❑ This mistake is commonly seen in application icons. Clearly the first thing that comes to mind when working on an application icon is to adapt the application's logo. What is so bad about the text inside the icon? Firstly, it is directly language-related and so impedes localization. Secondly, if the icon is small, it is impossible to read the text. Thirdly, in the case of application icons, this text is repeated in the name of the application.



Flickr Uploadr



iTunes-playlist.icns

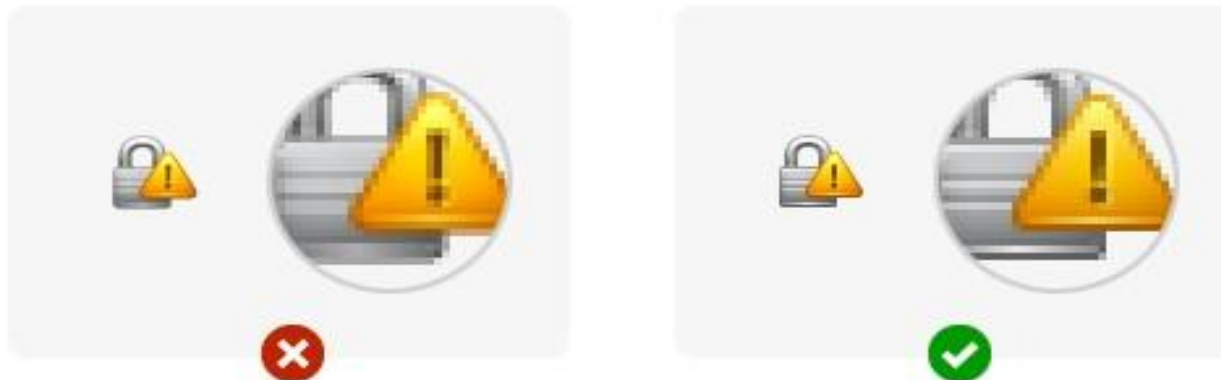


Jets'n'Guns

9.2. Các chuẩn thiết kế

❑ 10 lỗi hay mắc phải khi thiết kế #10 Outside the pixel framework

As a rule, this problem occurs if you use a vector editor for drawing icons. In large size everything looks pretty and clear; but in reality the icons are small, and under rasterization anti-aliasing frets the objects' borders.



9.2. Các chuẩn thiết kế

❑ Lời khuyênGet Started with Icon Design

- Designing icons for Web sites is a good way to get started with icon design. Often there are only a few icons needed for a site design. Start simple with a small Web site design project where you are required to design only a handful of icons or less. This is a good way to gain some experience with icon design.
- Start the icon design process with research. Consider the common symbolic metaphor used to describe the icon you're looking to make. Sketch as much as necessary to lock down the concept. Compliment the style of the icon designs with the Web site design you'll be using them on. Consider the color, perspective, and graphic look of the site.

9.3. Quy trình thiết kế

□ Các bước

- Thu thập các yêu cầu , tìm hiểu môi trường
- Xây dựng ý tưởng
- Xác định cỡ của Icon
- Sử dụng màu
- Tính hợp pháp

9.3. Quy trình thiết kế

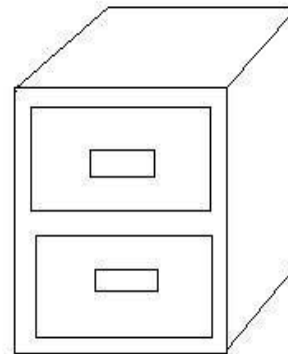
□ Các bước

- Thu thập các yêu cầu , tìm hiểu môi trường
- Xây dựng ý tưởng
- Xác định cỡ của Icon
- Sử dụng màu
- Tính hợp pháp

9.3. Quy trình thiết kế

□ Thu thập yêu cầu

- Đây là bước đầu tiên. Icon cần biểu diễn được môi trường
- Thí dụ: Khi phải thiết kế kho lưu trữ thông tin thì với KSPM đó có thể là ổ đĩa; ngược lại với nhân viên Văn phòng đó lại là tủ đựng tài liệu.



9.3. Quy trình thiết kế

□ Xây dựng ý tưởng

- Liệt kê danh sách các biểu tượng
- Lựa chọn hình ảnh để thể hiện nội dung
- Thí dụ nếu là Icon cho Web thì có thể chọn quả địa cầu
- Chú ý: Hình ảnh có thể chiếm đến 80% thông tin
- Cuối cùng là kết hợp Logo của hãng với hình ảnh đã chọn

9.3. Quy trình thiết kế

□ Xác định kích cỡ

- Kích cỡ có ảnh hưởng lớn: Sự thu hút phụ thuộc vào kích cỡ.
- Kích cỡ còn phụ thuộc vào độ phân giải của màn hình: 1600 x 1200 , 1280 x 1024, 1024x768
- Cỡ là 40 x 40 hoặc 48 x 48 vừa thể hiện hình ảnh chi tiết hơn đồng thời giúp Icon đẹp hơn

9.3. Quy trình thiết kế

□ Màu sắc

- ISO đã qui định khá rõ. TD gam màu nóng như đỏ thể hiện sự nguy hiểm.
- Bộ màu của Icon sẽ phụ thuộc vào số bit màu thể hiện mà hệ thống dùng sao cho hài hoà.

9.3. Quy trình thiết kế

□ Tính hợp pháp

- Thông thường thì Icon có thể được chúng ta dùng lại miễn phí tuy nhiên đối với một số nước thì luật pháp quy định việc trả tiền bản quyền khi sử dụng. Vd như hãng IBM của Mỹ

9.3. Quy trình thiết kế

□ Các yêu cầu

- Giao diện của một biểu tượng khi biểu diễn một trạng thái hoặc một chế độ của hệ thống máy tính sẽ được phân biệt rõ ràng với biểu diễn của một trạng thái hoặc chế độ khác
- Một icon sẽ vẫn dễ hiểu và dễ phân biệt khi có bất kỳ sự thay đổi nào về giao diện do sự thay đổi về trạng thái hoặc chế độ, trong môi trường mà nó được sử dụng.
- Tất cả các biểu tượng sẽ tuân theo mệnh đề 4, 5 trong ISO 9241-3:1992

9.3. Quy trình thiết kế

□ Các yêu cầu

- Bất cứ khi nào một biểu tượng được di chuyển đè lên một biểu tượng khác, nhưng không phải là kích hoạt bất kỳ một vùng nhạy cảm nào, thì vùng nhạy cảm xếp chồng của icon dịch chuyển sẽ ở trên các icon khác.
- Sự tương tác với các icon sẽ không xoá đi bất cứ một dữ liệu nào mà không được phép của người dùng.
- Màu sắc sẽ không được sử dụng như là các phần tử thông tin duy nhất để phân biệt giữa các icon trừ khi biểu diễn của phần tử chức năng chính là màu đó.

9.3. Quy trình thiết kế

□ Các yêu cầu

- Khi một icon đồ hoạ được sử dụng như là các thành phần của các icon khác, ý nghĩa tạo nên bởi các phần tử sẽ nhất quán với tất cả các công dụng của từng phần tử



9.3. Quy trình thiết kế

□ Các yêu cầu

- Các chi tiết phụ có thể được kết hợp vào các icon để thể hiện một cách chi tiết hơn các chức năng, tuy nhiên vẫn đảm bảo tính dễ nhận biết của nó



9.3. Quy trình thiết kế

□ Các khuyến nghị

- Giao diện của các icon nên nhất quán trong một tập các icon, nghĩa là trong một tập các icon nên được hiển thị cùng một kiểu đồ họa giống nhau.
- Nếu các icon được biểu diễn ở các kích cỡ khác nhau trên màn hình, thì thiết kế của icon nên vẫn đảm bảo được tính dễ hiểu và dễ nhìn, đảm bảo các thành phần chính của nó vẫn xuất hiện.

9.3. Quy trình thiết kế

□ Các khuyến nghị

- Nếu các icon được sử dụng trên các màn hình khác nhau làm cho các icon được hiển thị với những tỷ lệ 2 cạnh khác nhau, thì các phương pháp thiết kế nên quan tâm đến việc tạo ra giao diện của icon sao cho nó luôn tương tự như hình ảnh thiết kế ban đầu.
- Tất cả các icon đã có sẵn nên dễ hiểu. Khi tính dễ hiểu từ lần quan sát đầu tiên không phải là một yêu cầu về tính dùng được, thì các icon nên có khả năng học và dễ nhận biết.

9.3. Quy trình thiết kế

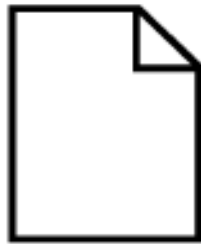
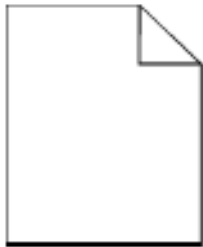
□ Các khuyến nghị

- Vị trí của các nhãn liên kết với icon có thể thay đổi bởi người dùng nên nhất quán bên trong một môi trường hoặc một tập các môi trường được thiết kế để sử dụng cùng nhau.
- Hoạt hình không được làm giảm tính dễ hiểu và dễ nhận biết của một icon (mục 5 of ISO 9241-3:1992)

9.3. Quy trình thiết kế

□ Các biến thể

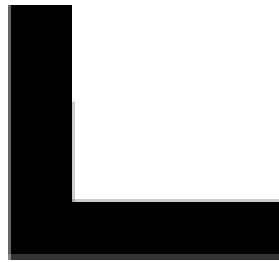
- Các biến thể toàn cục cho các thuộc tính đường thẳng như: kiểu, độ rộng, điểm kết thúc, liên thông, mẫu,...



9.3. Quy trình thiết kế

□ Các biến thể

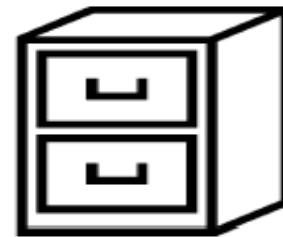
- Các biến thể toàn cục cho các thuộc tính canh góc là: cong, vuông, và liên thông



9.3. Quy trình thiết kế

□ Các biến thể

- Mức độ chi tiết có thể được tăng lên để thêm vào tính thực tế.



- Bề mặt các mẫu hoặc màu sắc có thể khác nhau toàn bộ nhưng không làm giảm tính dễ hiểu của các icon.
- Các phần tử đồ họa được thêm vào không nên làm giảm tính dễ nhận dạng của icon.

9.3. Quy trình thiết kế

□ Một số hình ảnh về Icons

Mac OS X Icons (128 px)



Icons for Touch Screen Buttons (128 px)



Large Icons (128px)



9.3. Quy trình thiết kế

□ Một số hình ảnh về Icons

Large Icons (72 px)



Avatars (60 px)



Emoticons in 2 Sizes



9.3. Quy trình thiết kế

▣ Một số hình ảnh về Icons

Toolbar Icons (32 px)



Toolbar Icons (32 px)



Toolbar Icons (32 px)



9.3. Quy trình thiết kế

■ Một số hình ảnh về Icons

Toolbar Icons (32 px)



Icons for a Touch Screen (30 px)



Toolbar Icons (24 px)



9.3. Quy trình thiết kế

- Một số hình ảnh về Icons

Toolbar Icons (24 px)

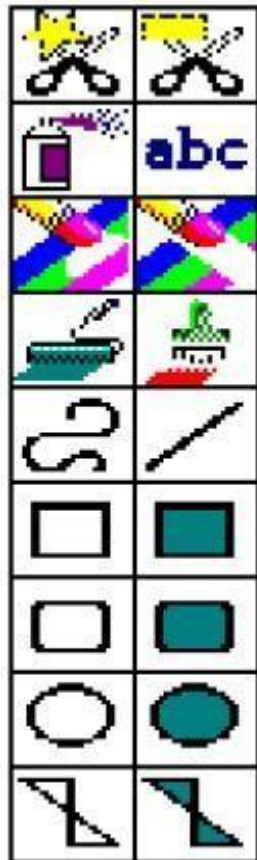


Tree and Status Icons (16 px)



9.4. Đánh giá

□ Thường dùng kỹ thuật Heuristic

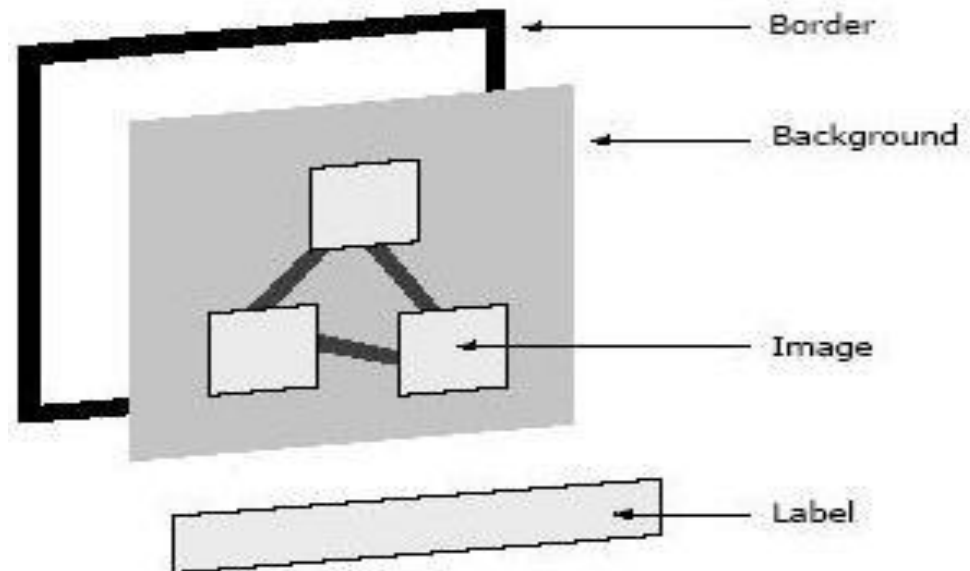


9.4. Đánh giá

□ Đánh giá Icon theo thuộc tính

- Để đánh giá được Icon ta sẽ đánh giá dựa trên hình ảnh của nó, mà đã là đánh giá theo hình ảnh của Icon thì chúng ta sẽ dựa vào 4 tính chất cơ bản sau:

- Viền (border)
- Nền (Background)
- Ảnh (Image)
- Nhãn (label)



Xin chân thành cảm ơn!

CHƯƠNG 10

THIẾT KẾ TRỢ GIÚP

10.1. Tổng quan

□ Khái niệm

- Là hệ thống giúp đỡ người dùng, có mặt trong các ứng dụng phần mềm, ứng dụng Web-based, intranet
- “Help” đôi khi còn được gọi là “online Help”
- Hệ trợ giúp được thiết kế tốt giúp người dùng sử dụng tốt phần mềm.
- Là nơi đầu tiên người dùng tìm đến khi cần giúp đỡ.

10.1. Tổng quan

□ Mục đích

- Mục đích chính của Help là để trả lời các câu hỏi người dùng gặp phải trong quá trình sử dụng
- Mục đích xa hơn của Help là đưa ra một tài liệu toàn diện để người dùng tham khảo và nghiên cứu sâu hơn

10.1. Tổng quan

- ❑ Sự hữu ích của một hệ trợ giúp phụ thuộc vào cách phân phối, tổ chức thông tin
- ❑ Người dùng muốn hệ trợ giúp trực tuyến đưa ra những câu trả lời nhanh cho các câu hỏi, và họ không muốn phải đào sâu trong hệ trợ giúp để tìm ra nó

10.1. Tổng quan

- ❑ Các cơ chế trong các định dạng Help cho phép chỉ ra cho người dùng những thông tin phù hợp cho công việc hiện thời của họ
- ❑ Cung cấp một cách tiếp cận nhanh tới những thông tin bổ sung trong tập các tài liệu lớn

10.2. Phân loại

- Yêu cầu đối với hệ trợ giúp
 - Tính sẵn dùng – Availability.
 - Tính chính xác và đầy đủ - Accuracy and Completeness.
 - Tính nhất quán – Consistency.
 - Tính khỏe – Robustness.
 - Tính linh hoạt – Flexibility.
 - Tính không tương tranh – Unobtrusiveness

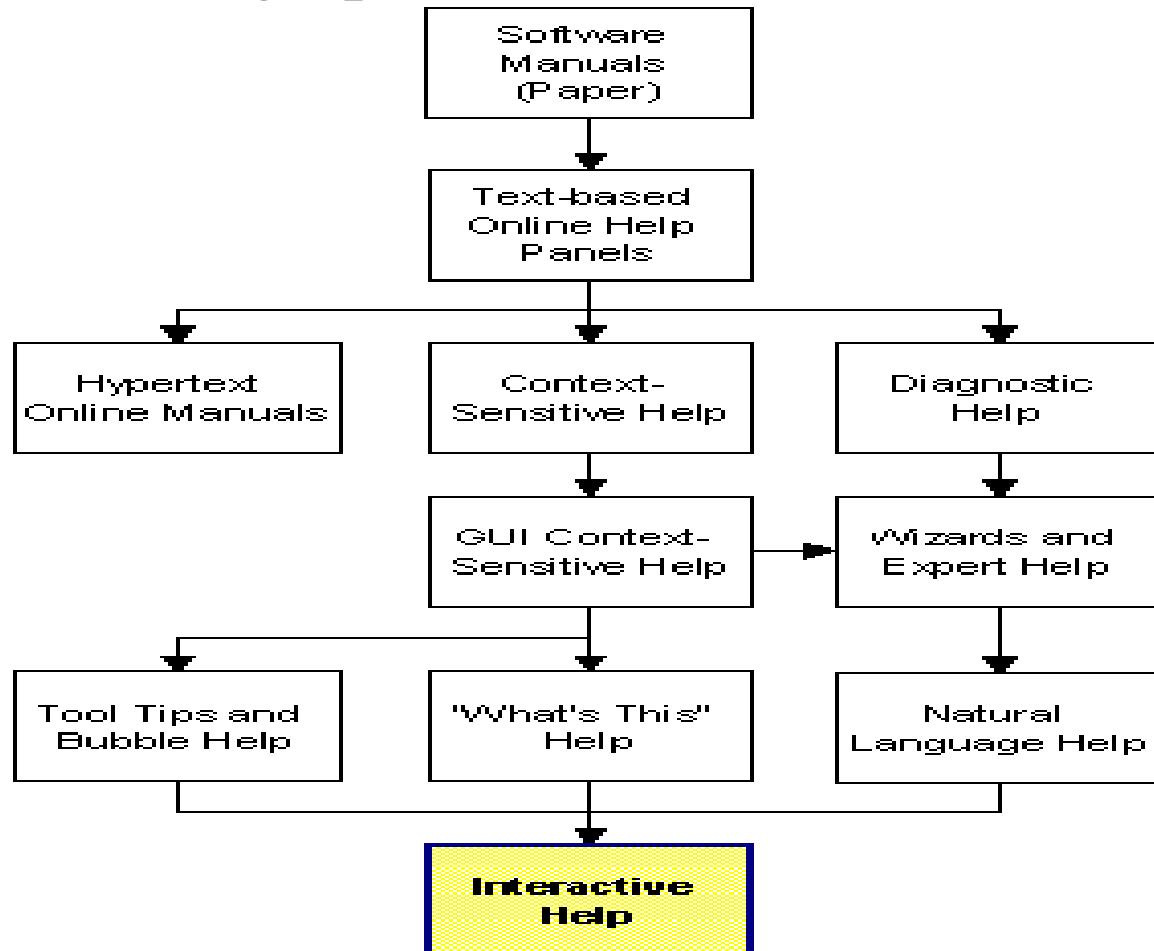
10.2. Phân loại

□ Các loại trợ giúp

- Tham khảo nhanh - *Quick reference*
- Trợ giúp cho từng công việc - *Task specific help*
- Giải thích đầy đủ - *Full explanation*
- Hướng dẫn – *Tutorial*

10.2. Phân loại

▣ Các loại trợ giúp



10.2. Phân loại

□ Các loại trợ giúp

- Hệ trợ giúp là các bản hướng dẫn sử dụng (User guide) được viết hoàn toàn trên giấy
- Các hệ trợ giúp trực tuyến đầu tiên có lẽ là các panel trợ giúp - các mô tả ngữ cảnh ngắn gọn của các màn hình ứng dụng

10.2. Phân loại

□ Các loại trợ giúp

- Hệ trợ giúp chẩn đoán là một chương trình nhỏ (được thêm vào ứng dụng phần mềm) hướng dẫn người dùng thông qua một loạt các câu hỏi để có thể đi đến một chỉ dẫn hoặc giải pháp
- Phát triển trong các hệ trợ giúp ra quyết định

10.2. Phân loại

□ Các loại trợ giúp

- Trợ giúp cảm ngữ cảnh (Context – sensitive Help): Là chế độ trợ giúp cho người sử dụng hiển thị lên màn hình những tài liệu có liên quan với lệnh, chế độ, hoặc động tác mà họ đang thực hiện
- Giảm bớt thời gian và số lần gõ phím để có được sự trợ giúp trên màn hình

10.2. Phân loại

□ Các loại trợ giúp

- CUA (Common User Access) - IBM lần đầu tiên đưa ra khái niệm các khuôn dạng trợ giúp cơ bản được chuẩn hoá cho tất cả các ứng dụng phần mềm
- Tập hợp tiêu chuẩn về các đề mục của trình đơn cơ sở, về cách tổ chức các đề mục trên trình đơn đó, và về các tổ hợp phím cơ bản

10.2. Phân loại

□ Các loại trợ giúp

- Các trợ giúp chuyên gia hay còn gọi là wizard
- Tool tip, What's This
- Point-and-click
- Trợ giúp bằng ngôn ngữ tự nhiên: AnswerWork trong WinHelp cho phép người dùng gọi đến trợ giúp bằng tiếng nói (thông qua phần mềm nhận biết tiếng nói)

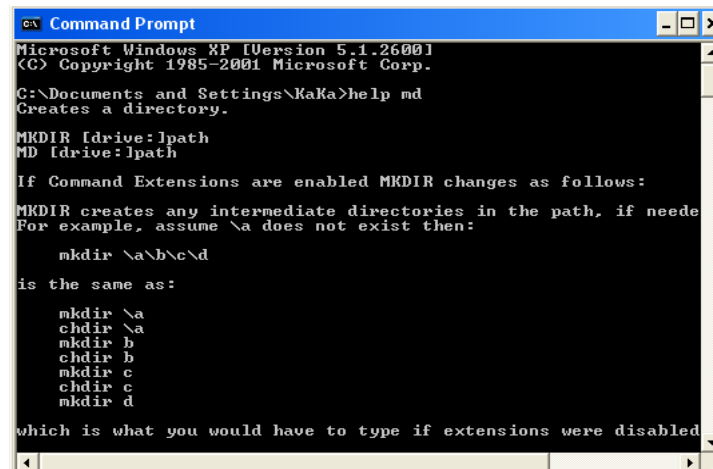
10.2. Phân loại

□ Các loại trợ giúp

- Trợ giúp tương tác (Interactive Help): không đợi đến khi được hỏi mới đưa ra trợ giúp

10.2. Phân loại

- ❑ Trợ giúp theo lệnh – Command assistance
 - Cung cấp trợ giúp qua các dòng lệnh.
 - Được sử dụng trong UNIX với lệnh *man* khi muốn hướng dẫn và lệnh *help* trong DOS
 - Đơn giản và khá hiệu quả
 - Giới hạn từ khi người dùng không thể biết trước hết tất cả các câu lệnh.



```

C:\Documents and Settings\KaKa>help md
Creates a directory.

MKDIR [drive:]lpath
MD [drive:]lpath

If Command Extensions are enabled MKDIR changes as follows:
MKDIR creates any intermediate directories in the path, if needed.
For example, assume \a does not exist then:

    mkdir \a\b\c\d
is the same as:

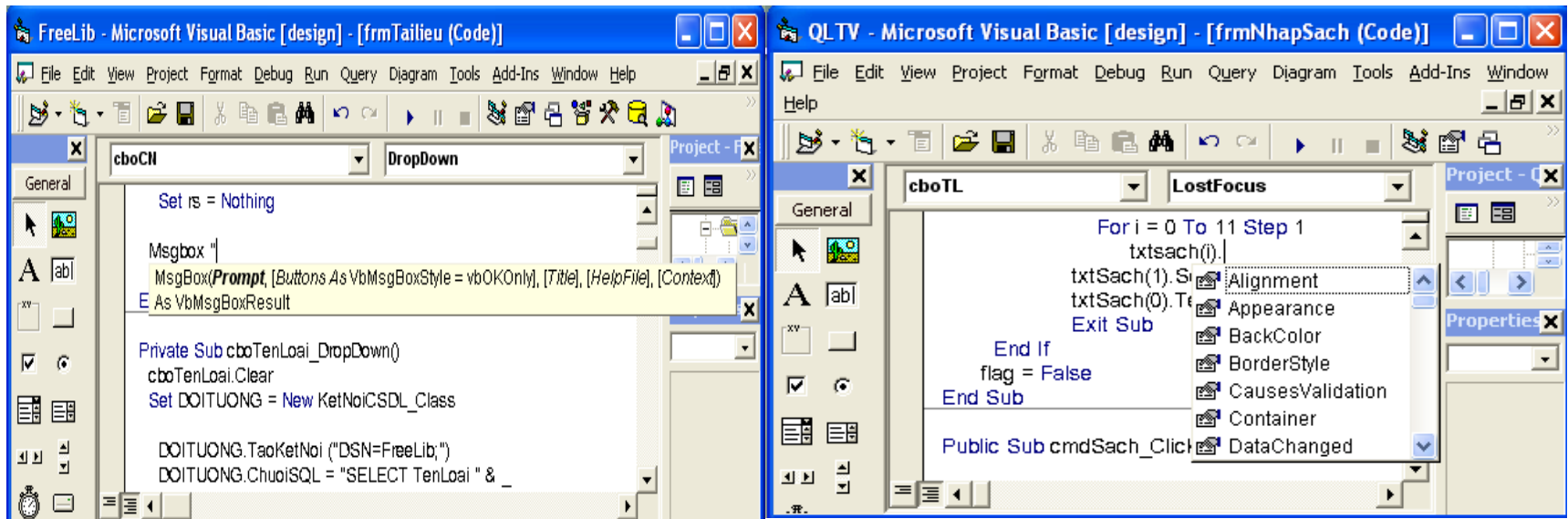
    mkdir \a
    chdir \a
    mkdir b
    chdir b
    mkdir c
    chdir c
    mkdir d

which is what you would have to type if extensions were disabled

```

10.2. Phân loại

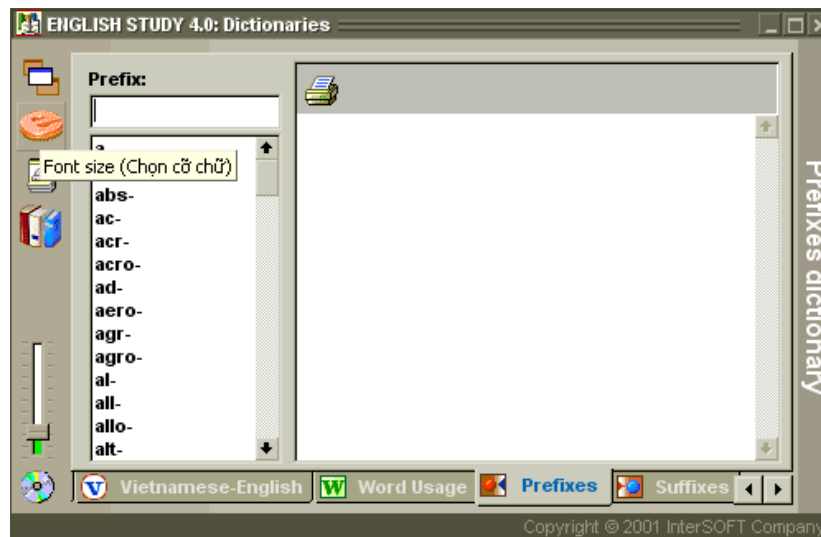
- ❑ Các dấu nhắc câu lệnh – Command Prompts
 - Cung cấp sự trợ giúp khi người dùng bắt gặp một lỗi
 - Thường ở trong dạng dấu nhắc sửa lỗi.
 - Thường có trong lập trình



10.2. Phân loại

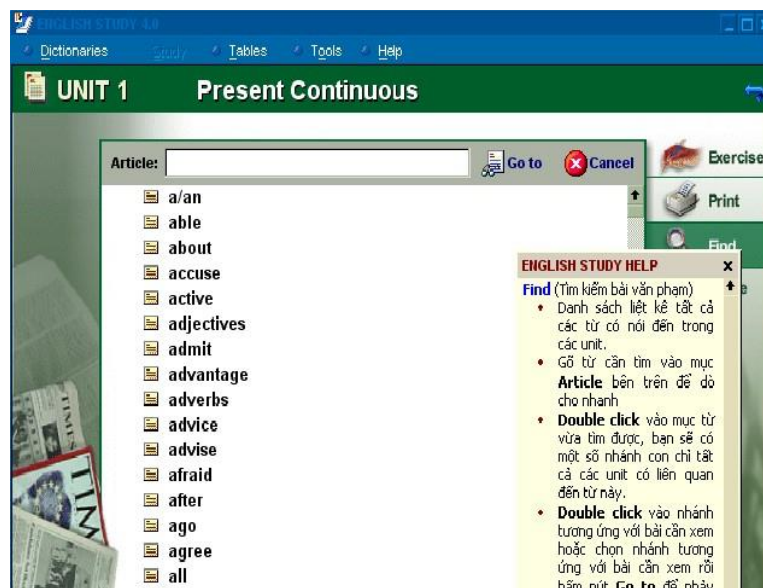
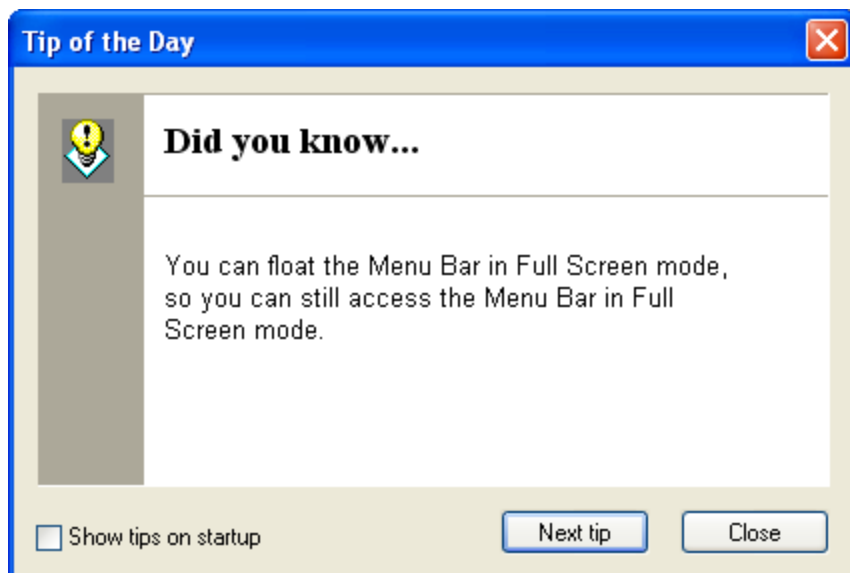
□ Trợ giúp ngữ cảnh

- Cung cấp các khóa hay chức năng mà được thông dịch
- Trợ giúp mức window Hiển thị các Topic khi ấn F1 hay các nút trợ giúp
- Trợ giúp mức vùng Hiện các ToolTip



10.2. Phân loại

- ❑ Các hướng dẫn trực tuyến OnLine Tutorial
 - Cho phép người dùng làm việc với những trợ giúp ngay trong ứng dụng.
 - Hệ hướng dẫn thông minh (Cung cấp các đề mục linh hoạt)



10.2. Phân loại

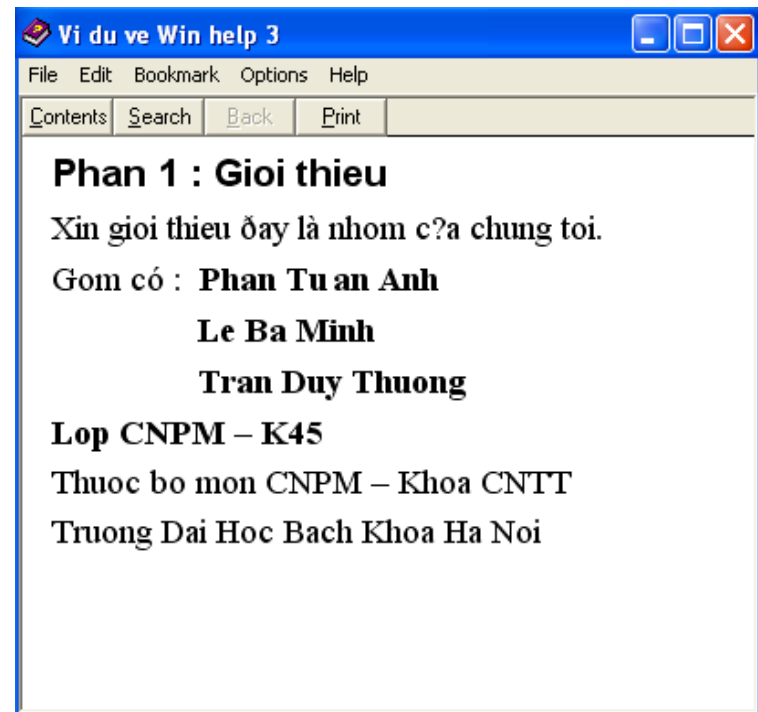
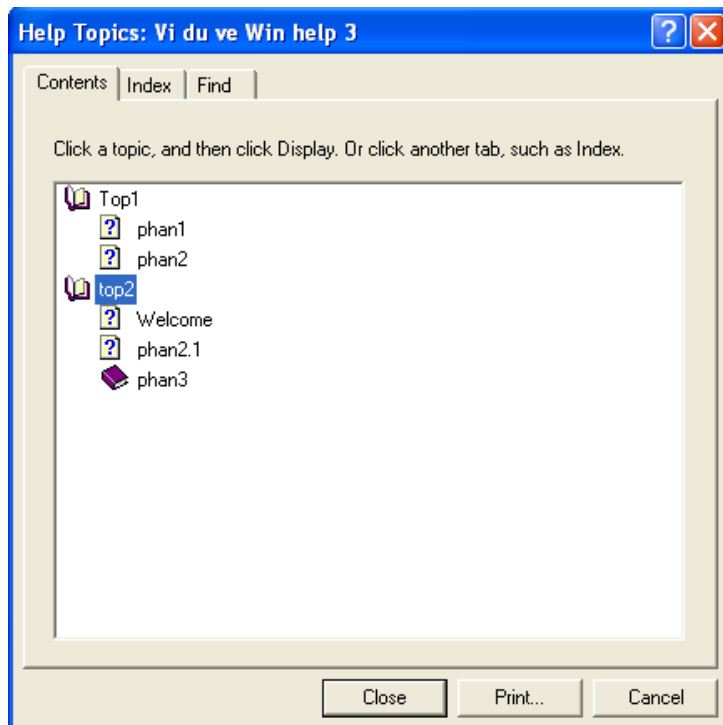
□ Các tài liệu trực tuyến

- Bao gồm các tài liệu và tài nguyên trực tuyến
- Cung cấp một lượng lớn các thông tin không phụ thuộc vào vấn đề riêng nào.
- Như là nguồn tham khảo lớn và đầy đủ
- Ngày càng được phát triển với rất nhiều chủng loại.

10.2. Phân loại

□ WinHelp

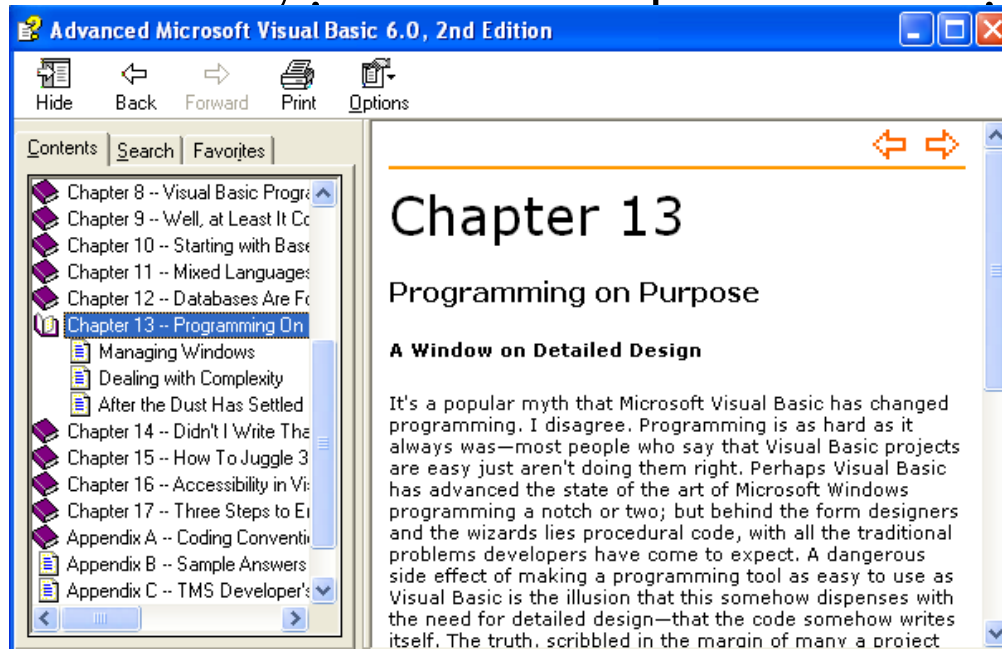
- Là hệ thống trợ giúp ban đầu cho Microsoft Windows
- Tất cả các version của Microsoft Windows vẫn tiếp tục hỗ trợ trợ giúp theo định dạng WinHelp



10.2. Phân loại

□ HTML Help

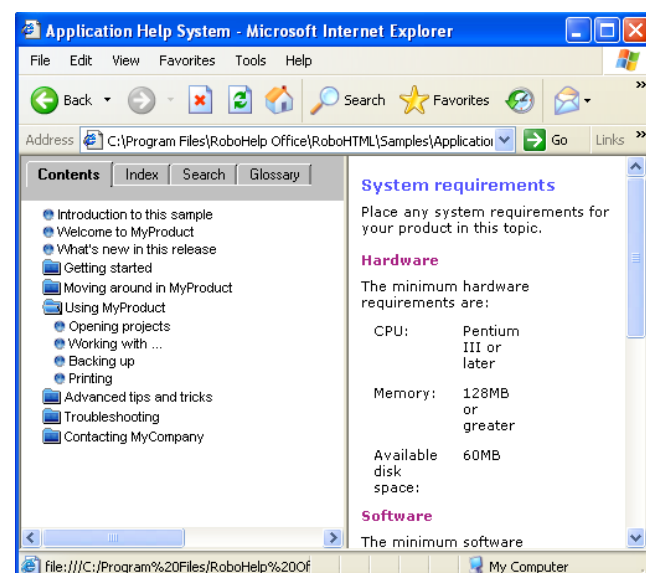
- Được ra đời vào năm 1997
- Chạy trên nền Window 32-bit
- Giao diện sử dụng sẽ rất thân thiện với người dùng
- Phải cài trình duyệt Internet Explorer 4.0 hoặc cao hơn nữa



10.2. Phân loại

□ Web Help

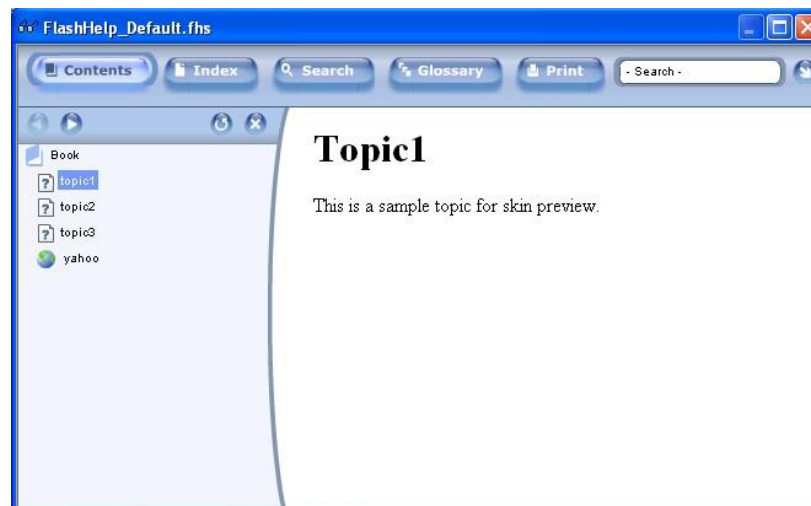
- Là một dạng HTML Help được thiết kế để chạy trên những trình duyệt rộng và đa dạng
- Sử dụng trên rất nhiều hệ điều hành khác nhau như Window , UNIX , MACINTOSH hay LINUX
- WEB help sẽ tạo ra các file định dạng HTML , XML . . . và các file image, chúng sẽ được phân bố trên các thư mục của người sử dụng hoặc trên một server



10.2. Phân loại

□ Flash Help

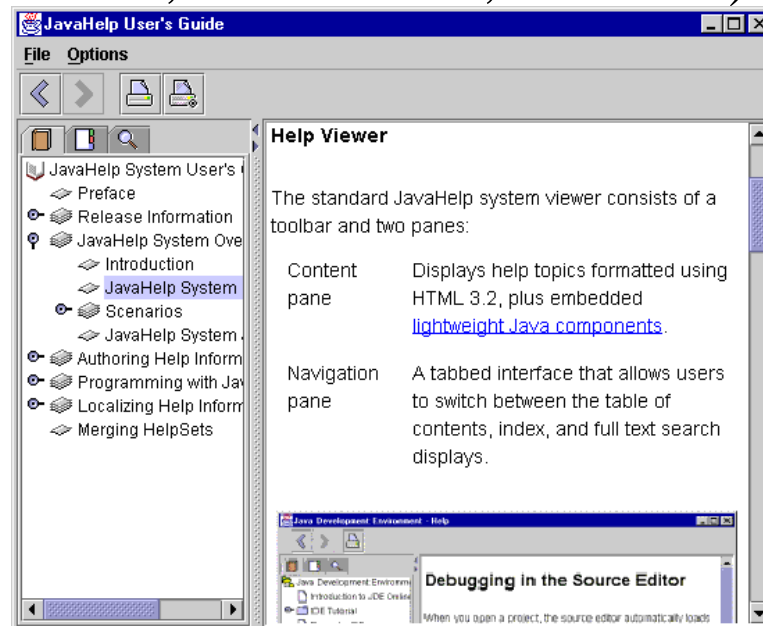
- Nâng cấp từ WEB Help
- Tìm kiếm nhanh , hiệu quả khi phải qua firewalls và băng thông kết nối thấp
- Tìm theo yêu cầu của người sử dụng
- Giao diện Help động, có tích hợp các Flash sinh động bao gồm cả âm thanh



10.2. Phân loại

□ Java Help

- Được thiết kế cho các ứng dụng viết bằng ngôn ngữ lập trình Java
- Có thể chạy được trên hầu hết các hệ điều hành khác nhau (Window, UNIX, Macintosh , LINUX)



10.2. Phân loại

□ Robo Help

- Robo Help là một công cụ tạo ra các Help trợ giúp cho các ứng dụng chạy trên desktop cũng như web-based
- Hỗ trợ 10 loại ngôn ngữ khác nhau
- Bộ công cụ Robo Help do hãng Macromedia sản xuất
- Robo Help sử dụng bộ công cụ soạn thảo Microsoft Word để tạo và hiệu chỉnh các file topic

10.3. Quy trình thiết kế

□ Các bước

- Tạo các Help Topic
- Xác định các cửa sổ để hiển thị Help Topic
- Xác định phương pháp định hướng các Help Topic

10.3. Quy trình thiết kế

□ Tạo các Help Topic

- Help Topic là đơn vị tổ chức cơ bản trong một hệ trợ giúp,
 - chứa đựng tất cả những thông tin mà người dùng tìm kiếm
- **Welcome Topic:** Chủ đề đầu tiên trong một hệ trợ giúp,
 - biểu diễn mục đích tổng thể của một hệ trợ giúp
- **Overview style topic:** cung cấp thông tin mang tính khái niệm và nền tảng về một chủ đề

10.3. Quy trình thiết kế

□ Tạo các Help Topic

- ***Procedure style topic:*** biểu diễn một chuỗi các bước giúp người dùng hoàn thành một nhiệm vụ cụ thể, bắt đầu với một tiêu đề mô tả một vài loại hoạt động, ví dụ: “Creating a Topic”, hoặc “To create a Topic inside the Project Tab”
- ***Definition style topics:*** là các mô tả ngắn gọn thường được hiển thị trong một cửa sổ pop-up

10.3. Quy trình thiết kế

□ Tạo các Help Topic

- ***What's This? Style topics:*** là những topic pop-up nhỏ có thể hiển thị bằng cách kích chuột vào một tính năng giao diện trong một ứng dụng để cung cấp một mô tả ngắn gọn về tính năng, nó đơn giản chỉ là mô tả các chức năng cụ thể trên giao diện ứng dụng
- ***Một số các loại chủ đề khác:*** bao gồm các thông điệp lỗi (error message), các chủ đề xử lý sự cố (troubleshooting), các chủ đề hiển thị trong các ứng dụng trình diễn đa phương tiện

10.3. Quy trình thiết kế

□ Xác định cửa sổ hiển thị

- Không bắt buộc các cửa sổ Help cảm ngữ cảnh phải hoàn thành quá nhiều nhiệm vụ
- Thiết kế các cửa sổ cảm ngữ cảnh “Always On Top”
- Không nhét quá nhiều thông tin vào một màn hình Help
- Tránh dùng hình ảnh nền, hình mờ, màu sắc làm cho văn bản chủ đề Help khó đọc

10.3. Quy trình thiết kế

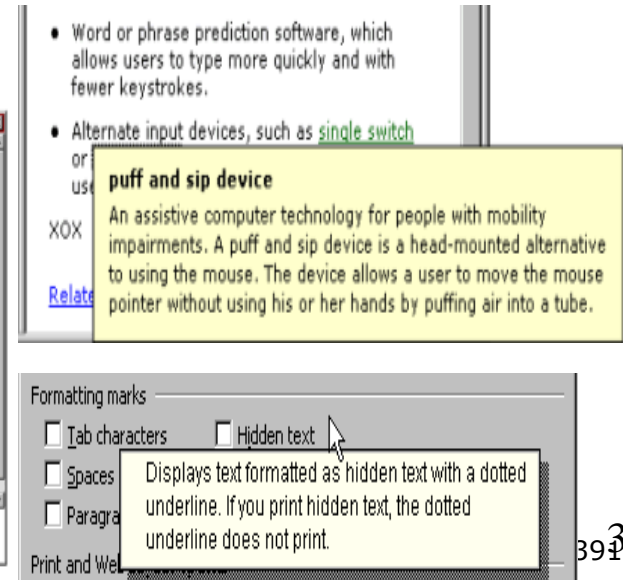
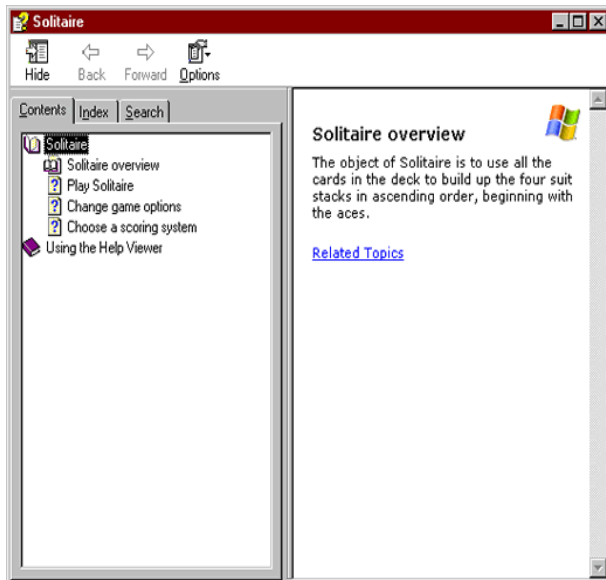
□ Xác định cửa sổ hiển thị

- Cố gắng đơn giản thiết kế Help, cần rất ít các kiểu mẫu trong một hệ thống Help, giữ cho quy ước về kiểu mẫu nhất quán; tạo các khoảng trống theo chiều dọc và căn lề để tổ chức các thông tin trợ giúp sao cho dễ đọc
- Dùng các bảng khi tổ chức các thông tin phức tạp và khi so sánh
- Dùng các định nghĩa pop-up để giải thích các thuật ngữ, gom các định nghĩa ngắn gọn trong một glossary hoặc trong một tab glossary

10.3. Quy trình thiết kế

❑ Xác định cửa sổ hiển thị

- Các loại cửa sổ: các cửa sổ chính, cửa sổ thứ cấp, và cửa sổ pop-up.
- Cửa sổ chính và cửa sổ thứ cấp: cửa sổ tĩnh
- Cửa sổ pop-up: cửa sổ tạm thời



10.3. Quy trình thiết kế

□ Định hướng các Help Topic

■ Các siêu liên kết dẫn tới các chủ đề liên quan

- Các siêu liên kết thường khó bảo trì
- Tránh tạo ra nhiều hơn 4 hoặc 5 siêu liên kết tới các chủ đề liên quan
- Sử dụng các siêu liên kết để hiển thị thông tin bổ sung trong các cửa sổ pop-up hoặc các cửa sổ thứ cấp
- Các siêu liên kết không được sâu quá 3 mức

10.3. Quy trình thiết kế

- Định hướng các Help Topic
 - Siêu liên kết và các điều khiển chủ đề liên quan
 - Các siêu liên kết chuẩn
 - Các điều khiển See Also
 - Các điều khiển liên kết từ khoá
 - Các điều khiển chủ đề liên quan

Xin chân thành cảm ơn!

CHƯƠNG 11

ĐÁNH GIÁ

11.1. Tổng quan

- ❑ Đánh giá không phải là một giai đoạn trong quá trình thiết kế mà là nhiệm vụ trung tâm của quá trình thiết kế. Nó diễn ra trong suốt vòng đời của HCI
- ❑ Đánh giá là thu thập dữ liệu kiểm tra về tính dùng được của thiết kế
- ❑ Ba mục đích chính của đánh giá:
 - Khẳng định tính mở rộng các chức năng của hệ thống
 - Khẳng định tính hiệu quả của giao tiếp
 - Xác định một số vấn đề đặc biệt nảy sinh trong quá trình sử dụng

11.1. Tổng quan

- Chức năng của HT là quan trọng: có đáp ứng với đặc tả yêu cầu ND? Thiết kế HT phải có khả năng đáp ứng các nhiệm vụ đặt ra một cách dễ dàng:
 - Hành động mà ND cần để thực hiện nhiệm vụ
 - Đánh giá khả năng sử dụng của HT với cái mà ND mong đợi

11.1. Tổng quan

- Tính hiệu quả của giao tiếp ND:
 - Lượng hoá được các tiêu chí của thiết kế với ND
 - Tính dễ học
 - Tính sử dụng lại
 - Trạng thái của ND
 - Xác định các lĩnh vực của TK chất lên người dùng theo một cách nào đó

11.1. Tổng quan

- Một số vấn đề nảy sinh trong quá trình TK
 - Sắc thái của TK khi chưa được sử dụng có tạo nên kết quả
 - không mong đợi?
 - Về cả 2 vấn đề: chức năng và tính dùng được

11.2. Các kiểu đánh giá

- Các kiểu đánh giá
 - Đánh giá trong PTN
 - Đánh giá thực nghiệm
 - Đánh giá trong giai đoạn TK
 - Đánh giá trong giai đoạn cài đặt

11.2. Các kiểu đánh giá

□ Đánh giá trong PTN

- Nơi tiến hành: trong PTN
- Dừng trong giai đoạn TK
- Người TK thực hiện một số khẳng định không cần/có sự hiện diện của ND.
- Các trang bị cần thiết có thể được yêu cầu như thiết bị nghe nhìn, thiết bị ghi,...

11.2. Các kiểu đánh giá

□ Đánh giá trong TN

- Dùng trong giai đoạn TK hay cài đặt
- Việc đánh giá thực hiện trong môi trường ND nhằm đánh giá HT trong hoạt động và trạng thái ND.

11.2. Các kiểu đánh giá

□ Đánh giá trong TN

- Dùng trong giai đoạn TK hay cài đặt
- Việc đánh giá thực hiện trong môi trường ND nhằm đánh giá HT trong hoạt động và trạng thái ND
- Hạn chế: tiếng ồn, chuyển động,... => mất tập trung
- Do môi trường tự nhiên nên khách quan
- Đánh giá được các hoạt động thường ngày, tháng, => được ưa chuộng.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Mục đích

- Dùng trong giai đoạn TK => tránh được một số sai sót đáng tiếc
- Các sai sót phát hiện được trong giai đoạn cuối sẽ làm giảm chi phí
- Một số phương pháp đánh giá sẽ được dự đoán

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn thiết kế
 - Các kỹ thuật
 - Cognitive WalkThrough
 - Đánh giá kiểu Heuristic
 - Đánh giá dựa vào mô hình
 - Đánh giá dựa vào xem xét lại

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Kỹ thuật Cognitive WalkThrough

- Là phương pháp có tính dự đoán kiểu Review nhằm phát hiện vấn đề từ rất sớm
- Xây dựng các định nghĩa từ các nhiệm vụ đặc tả HT hoặc từ màn hình "mock-up": từ màn hình này qua màn hình khác
- Người đánh giá duyệt qua một chuỗi hành động để kiểm tra tính dùng được. Mục đích chính của kỹ thuật này là thiết lập tính dễ học và dễ dùng của hệ thống.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – 4 bước tiến hành

- Một mô tả về nguyên mẫu của HT, không cần đầy đủ song cũng nên khá chi tiết thí dụ như vị trí và ngôn từ cho menu
- Một mô tả về nhiệm vụ mà ND phải thực hiện. Nhiệm vụ phải mang tính tiêu biểu, cái mà ND hay làm
- Một danh sách chi tiết các hành động cần thiết để hoàn thành nhiệm vụ theo nguyên mẫu
- Một chỉ dẫn về ND là ai và các tri thức, kinh nghiệm mà người đánh giá có thể giả định.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – 4 câu hỏi

- ND cố tạo ra bất cứ tác động gì trên hành động đó? Những khẳng định về nhiệm vụ mà hành động được hỗ trợ đúng theo kinh nghiệm và tri thức của ND trên tương tác đó.
- ND có khả năng để ý hành động đúng là có?. Thí dụ, ND có nhìn thấy phím hay một mục của menu, qua đó hành động tiếp là đang thực hiện bởi HT. Điều này có nghĩa là ND sẽ thấy phím đó (hay mục đó) tại thời điểm họ cần (Không nhất thiết phải biết).

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – 4 câu hỏi

- Khi ND tìm thấy một hành động đúng trên giao tiếp liệu họ có biết đó là cái duy nhất đúng cho mục đích mà họ cố tạo ra? Có thể đó là một phím lệnh/ mục menu quan sát được nhưng liệu ND có biết đó là cái mà họ cần để hoàn thành nhiệm vụ.
- Sau khi hành động tiến hành, ND sẽ hiểu phản hồi của HT? Giả sử rằng ND đã chọn đúng hành động, họ sẽ biết tiếp cái gì? Đó là việc hoàn thành chu trình tương tác đánh giá/thực hiện.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – hình thức tiến hành

Cognitive Walkthrough start-up sheet:

Interface

Task

Evaluator: Date

Task Description:

Action Sequence:

Anticipated User:

User's initial Goal:

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Hình thức tiến hành

Next action #: ----- Description -----

1. Correction goal

2. Problem forming Correction goals

-

-

3. Problem identifying Action

-

-

4. Problem performing the action

-

-

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

- Mục tiêu: Thiết kế điều khiển từ xa cho Video nhằm cho phép ND lập trình chọn lựa thời gian ghi
- Thiết kế ban đầu: Hình xxx
- Hình một: Tập các phím trong trạng thái sử dụng bình thường
- Hình hai: Phím ghi thời gian được nhấn. Video cho phép ND lập trình 3 chế độ ghi theo chuỗi khác nhau. Số hiệu chuỗi tiếp theo có thể được tự động thiết lập. Chúng ta muốn biết liệu thiết kế của mình có hỗ trợ nhiệm vụ ND không?

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn thiết kế
 - Cognitive WalkThrough – Ví dụ

Time: 21:45
Chanel:3

1	2	3	⊖
4	5	6	
7	8	9	0
⏪	●	⏩	
<<	>	>>	

Start: 1800
End: 1915
Chanel:4
Date:

1	2	3	⊖
4	5	6	
7	8	9	0
⏪	●	⏩	
<<	>	>>	

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

- Giả sử ND quen thuộc với ghi hình nhưng không biết thiết kế chuyên dụng này.
- Bước tiếp theo của Walkthrough là xác định chuỗi hành động cần phải tiến hành. Chúng ta đặc tả điều này theo thuật ngữ hành động ND (Action) và đáp ứng của HT (System Response):
 - Action A: Nhấn phím thời gian ghi
 - Đáp ứng A: Chuyển về hiển thị theo mode thời gian. Con trỏ xuất hiện sau
 - chữ Start

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

Action B: Nhấn số 1800

Đáp ứng B: Mỗi chữ số được hiện ngay sau khi nhấn và con trỏ chuyển về vị trí tiếp.

Action C: Nhấn phím ghi thời gian

Đáp ứng C: Con trỏ chuyển về sau vị trí End.

Action D: Nhấn số 1915

Đáp ứng D: Mỗi chữ số được hiện ngay sau khi nhấn và con trỏ chuyển về vị trí tiếp.

Action E: Nhấn phím ghi thời gian

Đáp ứng E: Con trỏ chuyển về sau vị trí Chanel.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

Action F: Nhấn số 4

Đáp ứng F: Mỗi chữ số được hiện ngay sau khi nhấn và con trỏ chuyển về vị trí tiếp.

Action G: Nhấn phím ghi thời gian

Đáp ứng G: con trỏ chuyển về sau vị trí Date.

Action H: Nhấn số 010199

Đáp ứng H: Mỗi chữ số được hiện ngay sau khi nhấn và con trỏ chuyển về vị trí tiếp.

Action I: Nhấn phím ghi thời gian

Đáp ứng I: Số hiệu chuỗi hiện lên góc phải trên màn hình

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

Action J: Nhấn phím truyền

Đáp ứng J: Dữ liệu được truyền đi và chuyển về chế độ bình thường

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

- Sau khi đã thiết lập chuỗi hành động, tiến hành đánh giá bởi Walkthrough. Với mỗi action A, B,..., ta phải trả lời 4 câu hỏi sau và nói về tính đúng đắn của HT. Bắt đầu với hành động A:
- Action A: Nhấn phím ghi thời gian
- Câu hỏi 1: ND có cố gắng tạo nên bất cứ tác động nào khi hành động xảy ra?
- Giao tiếp không cung cấp chỉ dẫn nào khi ND nhấn phím ghi thời gian. Tuy nhiên, điều đó chấp nhận được vì giả thiết là ND quen thuộc với máy ghi Video

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

□ Câu hỏi 2: ND có khả năng để ý hành động đúng là có?

→ Phím ghi thời gian có trên điều khiển từ xa

□ Câu hỏi 3: Khi ND tìm thấy một hành động đúng trên giao tiếp liệu họ có biết đó là cái duy nhất đúng cho mục đích mà họ cố tạo ra?

→ Không rõ ràng với phím ghi thời gian. Biểu tượng đồng hồ trên đ/k từ xa có thể là ứng cử viên, tuy nhiên nó có thể hiểu là phím dùng để thay đổi thời gian. Ứng cử viên khác có thể như là phím thứ 4 bên trái (phím này liên quan tới ghi). Tóm lại, biểu tượng đồng hồ là lựa chọn đúng, song rất có thể ND bị nhầm tại điểm này

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Cognitive WalkThrough – Ví dụ

□ Câu hỏi 4: Sau khi hành động tiến hành, ND sẽ hiểu phản hồi của HT?

→ Ngay sau khi hành động được làm, việc hiển thị đã thay đổi chế độ và các dòng hiện lên là khá thân thuộc với ND (Start, End,...)

Cứ theo cách như vậy tiến hành cho các hành động.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Kỹ thuật Heuristic

- Là các nguyên tắc chung hay các guideline có thể trợ giúp một quyết định trong TK
- Do Jacob Nielsen và Rolf Molich đề xuất
- Dùng trong giai đoạn đầu của TK nhằm cố định tính dùng được
- Ý tưởng chính là nhiều đánh giá độc lập được tiến hành để kiểm tra tính dùng được
- 9 mưu mẹo và 10 qui định

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Heuristic – 9 mưu mẹo

1) Đối thoại đơn giản và tự nhiên

- Đơn giản: Sử dụng ít thông tin
- Tự nhiên: Lệnh gần với yêu cầu

2) Nói ngôn ngữ ND

- Sử dụng cách nói của ND
- Không dùng thuật ngữ công nghệ đặc biệt

3) Giảm tải tối thiểu cho bộ nhớ ND

- Không bắt ND phải nhớ cho hành động tiếp sau
- Lưu lại thông tin trên màn hình cho đến khi ND không cần nữa

11.2. Các kiểu đánh giá

▣ Đánh giá trong giai đoạn thiết kế

■ Heuristic – 9 mưu mẹo

4) Tính nhất quán

5) Phản hồi thông tin

6) Xử lý tình huống thoát: khi ND gặp 1 tình huống mà họ

- không cần => cung cấp lối thoát

7) Tạo lối tắt: Giúp đỡ người dùng có kinh nghiệm tránh những đối thoại và thông tin không cần thiết

8) Có thông báo lỗi tốt

9) Dự báo lỗi.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Heuristic – 10 quy định

- 1) Thấy được các trạng thái của HT: HT luôn cho ND thấy cái họ sẽ làm qua thông tin phản hồi trong thời gian hợp lý
- 2) Sự tự do và quản lý ND: ND có thể gặp lỗi, do vậy phải có thoát khẩn cấp
- 3) Dự báo lỗi
- 4) Chuẩn hoá và nhất quán
- 5) Đối sánh HT với thế giới thực

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Heuristic – 10 quy định

6) Nhận biết chứ không phải nhớ lại: Tạo cho đối tượng, hành động và các lựa chọn dễ dàng nhận biết bởi ND chứ không cần nhớ

7) Sử dụng hiệu quả và mềm dẻo. Dùng được cho cả ND có/không có kinh nghiệm

8) Thiết kế đơn giản và có thẩm mỹ

9) Trợ giúp ND nhận biết, đối thoại và phục hồi từ trạng thái lỗi

10) Trợ giúp và tài liệu

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Kỹ thuật Review

- Đánh giá kiểu Heuristic do Molich & Nelson đưa ra 1990: dùng trong các UD nhỏ, thời gian đánh giá chừng 2h. Tuy nhiên có thể nhiều hơn nếu UD lớn
- Đánh giá kiểu Discount usability: chi phí thấp, thời gian và tài nguyên không nhiều.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Kỹ thuật dựa vào mô hình

- Đặc tả chức năng của HT phần có liên quan
- Một phân tích nhiệm vụ chứa danh sách nhiệm vụ và gán chúng thành các thành phần
- Cấu trúc nhiệm vụ từ đơn giản đến phức tạp
- Các thao tác ND có thể đánh giá bằng phương pháp giải tích
- Mô hình vật lý Keyrock hay được áp dụng

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn thiết kế
 - Kỹ thuật dựa vào mô hình – Ví dụ
 - " Save File trong MS Word"
 - *Phương tiện*: Dùng chuột và menu pull down
 - *Giả thiết*: Nhiệm vụ bắt đầu bằng thao tác Homming khi ND đặt tay lên chuột. Thời gian nhấn phím là 0.35s, thời gian đáp ứng của HT là 1,2s.

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn thiết kế

■ Kỹ thuật dựa vào mô hình – Ví dụ

- *Dãy thao tác:*

1) Bắt đầu TH

2) Chuyển con trỏ đến menu trên đỉnh màn hình: TP cộng với thời gian suy nghĩ TM

tổng thời gian: TP+TM

3) Chọn trên menu File (nhấn File), dịch chuyển con trỏ rồi chọn Save:

tổng thời gian: TM+ TK(File)+TP+TK(Save)

4) MS Word: nhắc nhập tên tệp: TR , ND nhấn “File-2.4”,TK(return):
TR+ TM+TK (File-2.4)+TK(return).

Với TH = 0.4, TM = 1.35, TP = 1.1, TK = 0.35, TR = 1.2

=> tổng thời gian: 13.05s

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Có sự hiện diện của ND và hệ thống đã được cài đặt
 - Có thể sử dụng mô phỏng hay mẫu thử
 - 3 kỹ thuật chính
 - Đánh giá thực nghiệm
 - Kỹ thuật quan sát
 - Kỹ thuật hỏi đáp

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm
 - Khó thực hiện trong phòng thí nghiệm
 - Là một trong các phương pháp quan trọng để đánh giá TK hay khía cạnh nào đó của TK.
 - Cách thức tiến hành: Đề xuất giả thiết đánh giá thông qua một số biến độc lập/phụ thuộc và nhằm đánh giá một số khía cạnh đặc trưng của TK=> Đ/k đánh giá được thay đổi thông qua giá trị một số biến
 - Ba thông số: chủ đề (subjects), biến (variables) và giả thiết (hypothesis).

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm – Chủ đề
 - Việc lựa chọn chủ đề là quyết định tới thành công của đánh giá
 - Chủ đề chọn phải sánh được với kỳ vọng của lớp ND. Lý tưởng là kiểm tra với sự hiện diện của ND. Tuy nhiên không phải luôn có thể. Nếu ND không phải là ND thực sự thì phải chọn lớp tương tự: độ tuổi, tri thức, ...
 - Mẫu chọn phải đủ lớn và đặc trưng (ít nhất là 10).

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn cài đặt

■ Đánh giá thực nghiệm – Biến

- Thực nghiệm quản lý và đo các đại lượng dưới điều kiện được kiểm soát nhằm kiểm tra giả thiết.
- Hai loại biến: loại quản lý và loại đo. Loại thứ nhất gọi là biến độc lập, loại thứ hai là phụ thuộc
- Biến độc lập: là các đặc trưng của thực nghiệm, nhằm tạo ra các đ/k so sánh như: kiểu giao tiếp, mức độ trợ giúp, số mức menu, . . .
- Với các test phức tạp => cần nhiều biến độc lập.
- Thí dụ: đưa ra 2 sơ đồ kiểm tra để đánh giá việc chọn 1 đoạn VB trong mét hệ STVB: định vị, chọn, mở rộng => trên cơ sở đó đánh giá.
- Biến phụ thuộc: Là các biến có thể đo đếm được theo nhiều cách. Thí dụ tốc độ lựa chọn menu

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm – Giả thiết
 - Giả thiết là một số dự đoán kết quả của thực nghiệm được hình thành từ biến độc lập và biến phụ thuộc: thí dụ như thay đổi giá trị biến độc lập sẽ tạo nên sự thay đổi trong biến độc lập
 - Mục đích của thực nghiệm là chỉ ra rằng các dự đoán là chính xác. Người ta thường dùng các số đo thống kê để so sánh với các mức độ có nghĩa khác nhau

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm – Phương pháp giữa các nhóm
 - Mỗi một chủ đề được gán cho một điều kiện khác nhau. Phải có ít nhất 2 điều kiện: điều kiện thực nghiệm (các biến được điều khiển) và điều kiện kiểm tra. Việc kiểm tra này dùng để khẳng định rằng điều khiển sẽ chịu trách nhiệm về những cái khác nhau sẽ được đo đếm.
 - Số lượng nhóm (≥ 2) phụ thuộc số lượng biến độc lập
 - Ưu điểm: tác động của việc học thu được từ kết quả thực hiện của ND trong 1 đ/k. Mỗi ND thực hiện trong 1 đ/k
 - Nhược điểm: đòi hỏi nhiều chủ đề \Rightarrow sự khác nhau giữa các nhóm có thể phủ nhận bất cứ kết quả nào.

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm – Thiết kế thực nghiệm
 - Để tạo ra kết quả tin cậy được và có tính khái quát, một thực nghiệm phải được thiết kế cẩn thận
 - Qui trình:
 - 1. Xem xét các yếu tố mà thực nghiệm phải xem xét như chủ đề, biến và giả thiết
 - 2. Lựa chọn phương pháp: giữa các nhóm và trong nội bộ nhóm

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Đánh giá thực nghiệm – Phương pháp trong nhóm
 - Mỗi ND sẽ thực hiện trong điều kiện khác nhau. Thiết kế này có thể phải chịu đựng từ việc truyền tác động học. Tuy nhiên có thể hạn chế được nếu thứ tự trong đó điều kiện được ND xem xét bị thay đổi.
 - Ưu điểm: chi phí thấp do cần ít ND và sẽ hiệu quả nếu có đào tạo.
 - Việc lựa chọn phương pháp phụ thuộc vào tài nguyên, chủ đề tiêu biểu, . . .
 - Các test khác nhau sẽ tạo nên các khẳng định khác nhau về dữ liệu và nếu như test lựa chọn không đúng, thì kết quả có thể không hợp lý.

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Kỹ thuật quan sát
 - Kỹ thuật không hình thức, có thể quan sát hay dùng video
 - Quan sát trực tiếp: quan sát ND ở trạng thái đang làm việc => hiệu quả, thời gian. Nhược điểm: ND có thể mất tập trung
 - Quan sát gián tiếp: Quay video => phân tích kết quả thu nhận được.
 - Hay sử dụng kỹ thuật “think aloud” và đánh giá tập thể (cooperative evaluation)

11.2. Các kiểu đánh giá

□ Đánh giá trong giai đoạn cài đặt

■ Kỹ thuật quan sát – Think aloud

- Kỹ thuật nhằm mô tả cái ND tin sẽ xảy ra, tại sao lại hành động như vậy và họ sẽ cố làm cái gì?
- Ưu điểm của kỹ thuật này là đơn giản, đòi hỏi ít tri thức để thực hiện và cho sự hiểu thấu đáo về các vấn đề với mét giao tiếp => hệ thống đang được sử dụng thế nào.
- Thông tin thường là chủ quan và có chọn lọc
- Mỗi hành động miêu tả cái mà bạn sẽ làm thường thay đổi theo cách bạn làm

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Kỹ thuật quan sát – Đánh giá tập thể
 - Là một biến thể của think aloud, trong đó ND cố gắng xem mình như một thành viên trong quá trình đánh giá chứ không đơn thuần là chủ thể bị thực nghiệm.
 - Khi bắt đầu tiến hành, người đánh giá có thể hỏi ND các câu hỏi như: tại sao, cái gì nếu...
 - Ưu điểm:
 - Qui trình là ít bị ràng buộc và dễ học bởi người đánh giá
 - ND cố gắng phê phán HT
 - Người đánh giá có thể chỉ rõ điểm nhầm lẫn ở thời điểm xảy ra và cực đại hóa hiệu quả.

11.2. Các kiểu đánh giá

- ❑ Đánh giá trong giai đoạn cài đặt
 - Kỹ thuật quan sát – Cách thức tiến hành
 - ❑ Bút và giấy
 - ❑ Ghi âm
 - ❑ Ghi hình
 - ❑ Nhật ký máy tính
 - ❑ Sổ tay ND
 - Thường sử dụng phối hợp các phương pháp. Sử dụng các công cụ ghi tự động: Experimental Video Annotator, WorkPlace project,...

11.2. Các kiểu đánh giá

- Đánh giá trong giai đoạn cài đặt
 - Kỹ thuật hỏi đáp
 - Kỹ thuật phỏng vấn: có cấu trúc, mềm dẻo
=> cần xác định một số thông số như số câu hỏi, các gợi ý.
 - Hỏi và giám sát : dùng câu hỏi dạng đóng hay dạng mở.
Câu hỏi đóng có thể dùng như dạng Multichoice; câu hỏi mở phải có gợi ý.
 - Các dạng trả lời có thể dùng bảng, mẫu điền.

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

■ Tiêu chí

- Phụ thuộc sự tham gia của ND trong ngữ cảnh thực hiện nhiệm vụ hay đánh giá. Mỗi phương pháp có những ưu điểm và nhược điểm riêng.
- Có nhiều yếu tố cần phải xem xét khi lựa chọn các kỹ thuật đánh giá

11.2. Các kiểu đánh giá

- Lựa chọn phương pháp đánh giá
 - Các yếu tố phân biệt các kỹ thuật đánh giá
 - Giai đoạn trong vòng đời mà đánh giá thực hiện
 - Hình thức đánh giá
 - Mức độ chủ quan hay khách quan của kỹ thuật
 - Kiểu số đo cung cấp
 - Thông tin cung cấp
 - Mức độ đan xen
 - Tài nguyên yêu cầu

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

■ Đánh giá ngược với cài đặt

- Trong giai đoạn cài đặt các yếu tố vật lý đã tồn tại thí dụ như các bản mock-up cho việc cài đặt đầy đủ và có cái gì đó cụ thể có thể kiểm tra được
- Đánh giá trong giai đoạn thiết kế có khuynh hướng chỉ liên quan đến các chuyên gia và mang tính phân tích, trong khi đó đánh giá cài đặt xem xét ND như là chủ thể và là thực nghiệm

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

■ Phòng TN ngược với thực nghiệm

- Đánh giá trong PTN cho phép kiểm tra thực nghiệm và quan sát không có mặt ND, không tự nhiên
- Đánh giá Thực nghiệm khắc phục điểm yếu trên.
- Lý tưởng là tiến hành cả 2 kiểu trong giai đoạn TK

■ Khách quan ngược với chủ quan

- Nhiều kỹ thuật mang tính chủ quan như Walkthrough hay Think aloud dựa vào tri thức và kinh nghiệm của các nhà đánh giá - người phải nhận thức được và phải hiểu cái ND sẽ làm.

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

■ Định lượng ngược với định tính

- Kiểu số đo cung cấp bởi các kỹ thuật đánh giá là một yếu tố quan trọng
- Hai loại số đo: định tính và định lượng. Số đo định lượng thường là số và dễ phân tích khi dùng các kỹ thuật như thống kê; số đo định tính là phi số và khó phân tích nhưng có thể cung cấp các chi tiết quan trọng mà không thể xác định bằng số.
- Định lượng hay định tính có liên quan đến tính chủ quan hoặc khách quan của kỹ thuật
- Kỹ thuật chủ quan có khuynh hướng cung cấp số đo định tính; kỹ thuật khách quan cung cấp các số đo định lượng. Thông tin cung cấp.
- Có thể ánh xạ giữa 2 kiểu số đo, thí dụ như đánh giá kiểu hỏi đáp là định tính, song định lượng theo tỉ lệ.

11.2. Các kiểu đánh giá

- Lựa chọn phương pháp đánh giá
 - Thông tin cung cấp
 - Mức độ thông tin yêu cầu cho đánh giá là khá đa dạng
 - Đáp ứng tức thời
 - Một số kỹ thuật như Think aloud ghi lại hành vi của ND tại thời điểm tương tác. Một số kỹ thuật khác như Walkthrough lại liên quan đến tập sự kiện của ND.

11.2. Các kiểu đánh giá

- Lựa chọn phương pháp đánh giá
 - Tính xâm phạm
 - Một số kỹ thuật đặc biệt là các kỹ thuật cung cấp số đo trực tiếp có ảnh hưởng đến cách thức ứng xử của ND
 - Tài nguyên
 - Thiết bị, thời gian, tiền bạc, kinh nghiệm chuyên gia, ngữ cảnh
 - Một số quyết định bị bắt buộc do hạn chế về tài nguyên thí dụ như phải dùng camera để đánh giá song lại không có.

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

	Cognitive Walkthrough	Heuristic Evaluation	Review Based	Model Based
Giai đoạn	Toàn bộ	Toàn bộ	Thiết kế	Thiết kế
Hình thức	PTN	PTN	PTN	PTN
Mục đích	Không	Không	Nh TN	Không
Số đo	Định tính	Định tính	Nh TN	Định tính
Thông tin	M/ độ thấp	M/ độ cao	Nh TN	M/ độ thấp
Trực tiếp	N/A	N/A	Nh TN	Nh TN
Tính Gò bó	Không	Không	Không	Không
Thời gian	TB	Thấp	TB-thấp	TB
Thiết bị	Thấp	Thấp	Thấp	Thấp
Hàn lâm	Cao	TB	Thấp	Cao

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

	Thực nghiệm	Phỏng vấn	Hỏi đáp
Giai đoạn	Toàn bộ	Toàn bộ	Toàn bộ
Hình thức	PTN	PTN/TN	PTN/TN
Mục đích	Có	Không	Không
Số đo	Định tính	Định tính/ Định lượng	Định tính/ Định lượng
Thông tin	M. độ thấp/cao	M/ độ cao	M/ độ cao
Trực tiếp	Có	Không	Không
Tính Gò bó	Có	Không	Không
Thời gian	Cao	Thấp	TB-thấp
Thiết bị	TB	Thấp	Thấp
Hàn lâm	TB	Thấp	Thấp

11.2. Các kiểu đánh giá

□ Lựa chọn phương pháp đánh giá

	Suy nghĩ chủ yếu	Thể thức phân tích	Post-Task Walkthrough
Giai đoạn	Cài đặt	Cài đặt	Cài đặt
Hình thức	PTN/TN	PTN/TN	PTN/TN
Mục đích	Không	Không	Không
Số đo	Định tính	Định tính	Định tính
Thông tin	M/ độ thấp hoặc cao	M/ độ thấp hoặc cao	M/ độ thấp hoặc cao
Trực tiếp	Có	Có	Không
Tính Gò bó	Có	Có	Không
Thời gian	Cao	Cao	TB
Thiết bị	Thấp	Cao	Thấp
Hàn lâm	TB	Cao	TB

Xin chân thành cảm ơn!