



Chương 3

Sự tương tác

<http://fit.lqdtu.edu.vn/~trangdld/info.aspx>

Mục tiêu

- Sự tương tác là gì?
- Các yếu tố tạo ra một sản phẩm có tính tương tác tốt
- Cơ bản về mô hình và loại hình tương tác (trong thiết kế giao diện)

Nội dung

1. Khái niệm tương tác
2. Các mô hình tương tác
3. Thiết kế hợp lý
4. Các loại hình tương tác
5. Giao diện WIMP
6. Tính tương tác
7. Một số yêu cầu khác

1. Khái niệm tương tác

- Tương tác (trong Tương tác người – máy) là quá trình trao đổi thông tin giữa người sử dụng và hệ thống máy tính (các thiết bị)
- Với con người: quá trình thu nhận thông tin thông qua các giác quan, cảm nhận, trải nghiệm và tác động.
- Với “máy tính”: quá trình nhập dữ liệu, xử lý và phản hồi (hiển thị, ...)

Tương tác là gì?



■ Mục tiêu là:

- Hiệu quả trong chuyển tải (chất và lượng thông tin, ..)
- Đảm bảo các yêu cầu của người dùng
- Phù hợp với khả năng của hệ thống

→ Yêu cầu với nhà phát triển?

2. Các mô hình tương tác

- Điều kiện của sự tương tác
- Mô hình Foley
- Mô hình Norman
- Khung tương tác

Một vài điều kiện

- Miền – Đó là phạm vi lĩnh vực hoạt động của chương trình, sản phẩm.
 - VD: thiết kế đồ họa, quản trị thông tin, ...
- Mục tiêu – Những gì mà chúng ta cần đạt đến – hay đúng hơn, là những gì mà khách hàng cần!
 - VD: Vẽ hình 3D, quản lý sản phẩm, ...
- Nhiệm vụ – Ta cần làm gì để đạt được mục tiêu đó?
 - VD: Ctr+cuộn, điền vào khung, ...

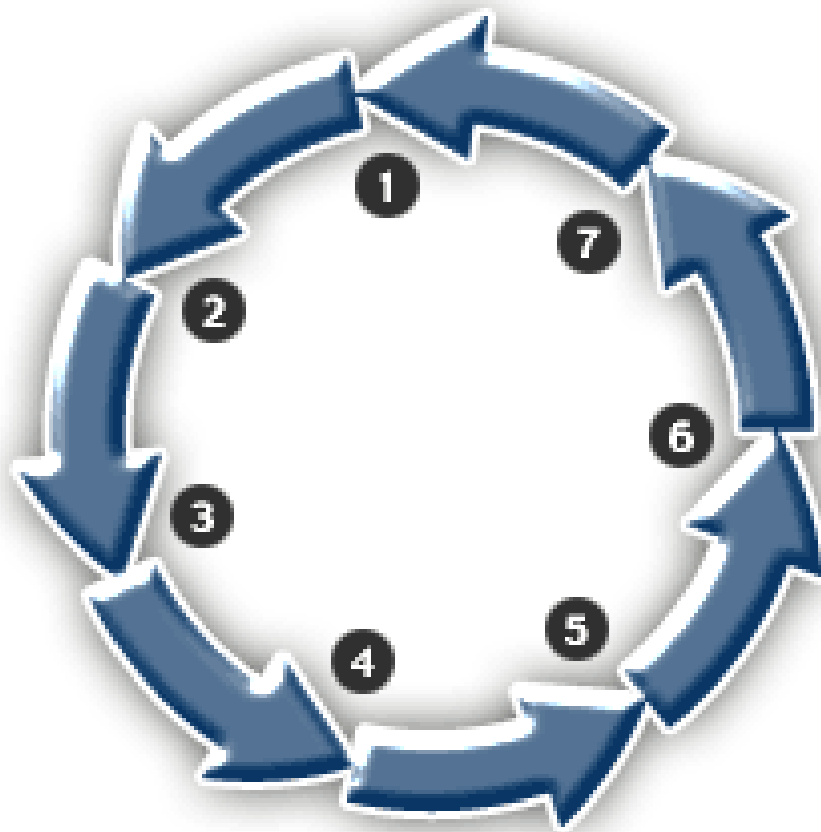
Mô hình Foley (1980)

- Mức độ khái niệm: Hiểu được mô hình tâm thần của người sử dụng
- Mức độ ngữ nghĩa: Ý nghĩa các thông tin mà người dùng nhập vào hệ thống và ngược lại.
- Mức độ cú pháp: Cách sắp xếp từ ngữ thành câu có nghĩa (với máy).
- Mức độ từ vựng: Hiểu biết về cơ chế của người dùng khi xây dựng cơ cấu tương tác người - máy

Mô hình Donald Norman(1988)

- 7 giai đoạn:
 - Người dùng thiết lập các mục tiêu
 - Mô hình hóa (công thức hóa) các mục tiêu đó
 - Cụ thể hóa các hành động trên giao diện
 - Thực hiện hành động
 - Cảm nhận các trạng thái của hệ thống
 - Diễn giải các trạng thái của hệ thống
 - Đánh giá trên cơ sở các mục tiêu đã đề ra
- Mô hình Norman tập trung chủ yếu vào giao diện người dùng

Mô hình Norman



1. Forming the goal of the HCI interaction

2. Forming the intention of the HCI interaction

3. Specifying the HCI action to be performed

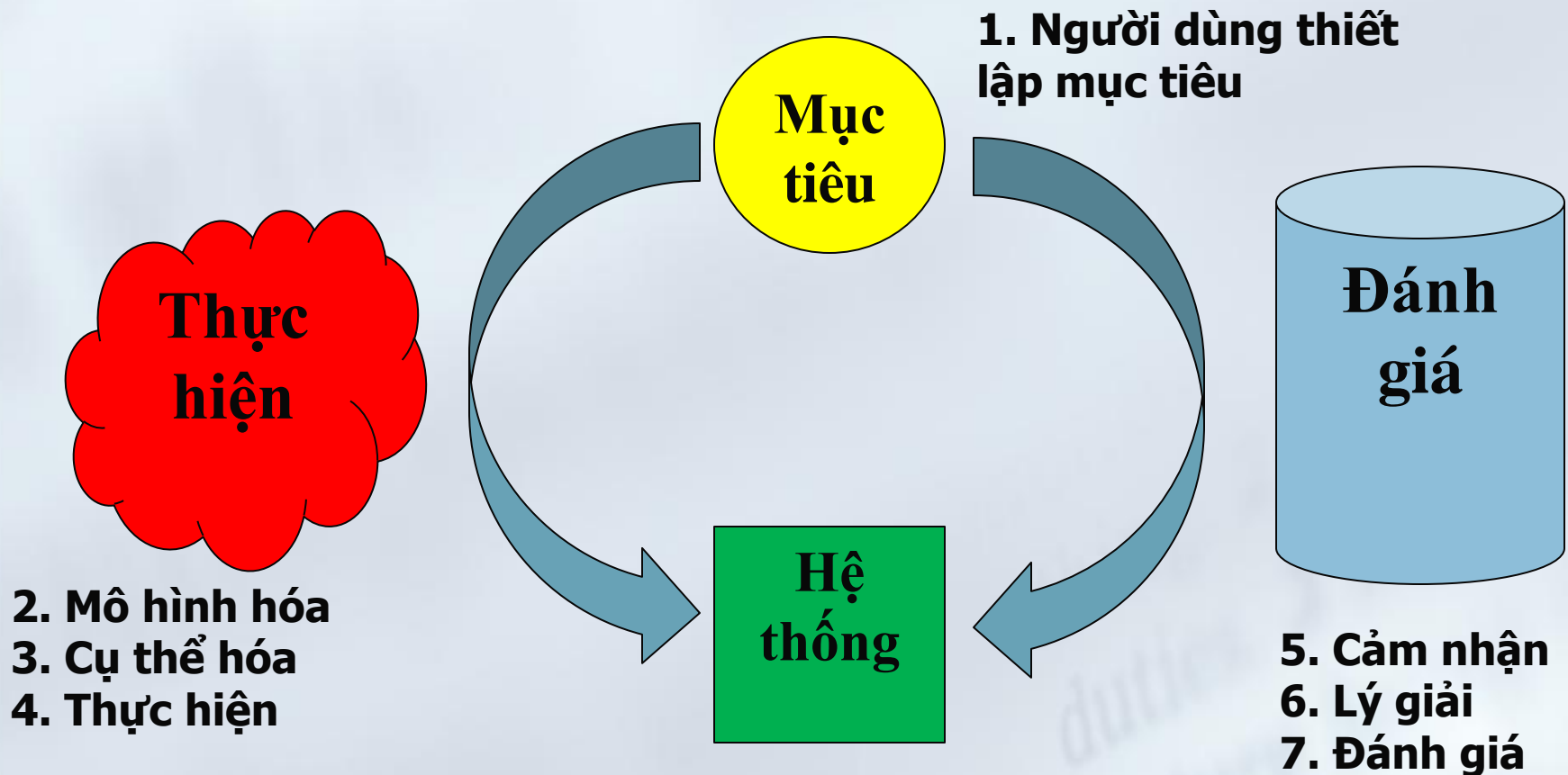
4. Executing an action with the computer

5. Perceiving the systems state through feedback

6. Interpreting the systems state through feedback

7. Evaluating the systems output

Chu trình thực hiện/đánh giá

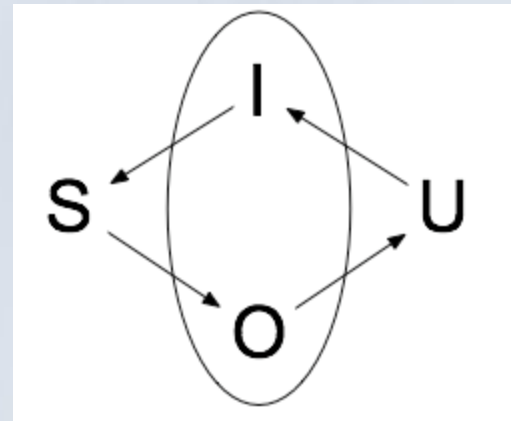


Sử dụng mô hình Norman

- Việc sử dụng mô hình Norman trong thiết kế giao diện không phải lúc nào cũng dễ dàng, do một số khó khăn:
 1. Khoảng cách trong quá trình thực thi:
Công thức các hành động của người sử dụng
 # Các hành động mà hệ thống cho phép.
Yêu cầu # Khả năng
 2. Khoảng cách trong quá trình đánh giá:
Mong muốn của người dùng (trạng thái hệ thống)
 # Sự biểu diễn chính xác của các trạng thái đó
Mục tiêu # Nhận thức

Abowd & Beale framework

- Coi quá trình tương tác người-máy giống như một bộ dịch ngôn ngữ
 - Khung tương tác
 - Bộ dịch
- 4 thành phần chính: Người dùng, hệ thống, đầu vào, đầu ra.

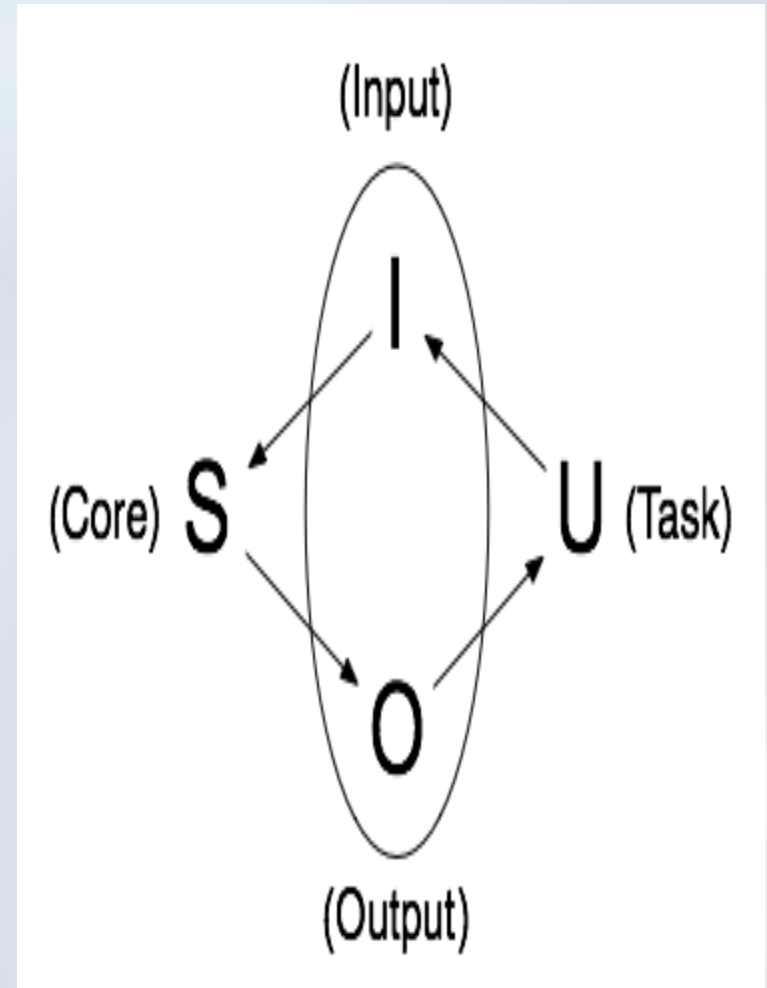


Abowd & Beale framework

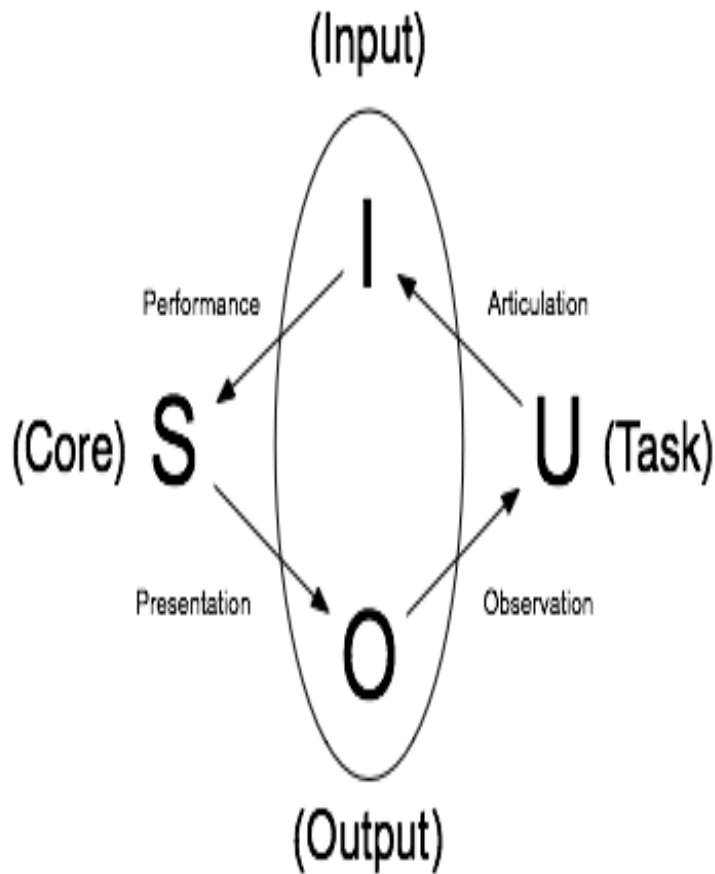
- Mỗi một đối tượng trong mô hình có một ngôn ngữ riêng của nó
Quá trình tương tác được hiểu là quá trình dịch giữa các ngôn ngữ
- Các vấn đề gặp phải trong tương tác cũng giống như các vấn đề gặp trong dịch ngữ

Abowd & Beale framework

- Người dùng (User) chịu trách nhiệm mô hình hóa các ý định cũng như thiết lập các mục tiêu. Điều khiển thông qua các nhiệm vụ
- Hệ thống chịu trách nhiệm xử lý và trả lại các dữ liệu. Hệ thống được coi là tồn tại trong lõi (core)



Abowd & Beale framework



- User kết nối với hệ thống thông qua Input → “phát âm” *articulation*
Biểu diễn yêu cầu của User thông qua “task language”
- Hệ thống nhận lệnh quá quá trình thực thi (*performance*) trên lỗi ngôn ngữ. Và trình bày lại cho người dùng

Sử dụng mô hình Abowd&Beale

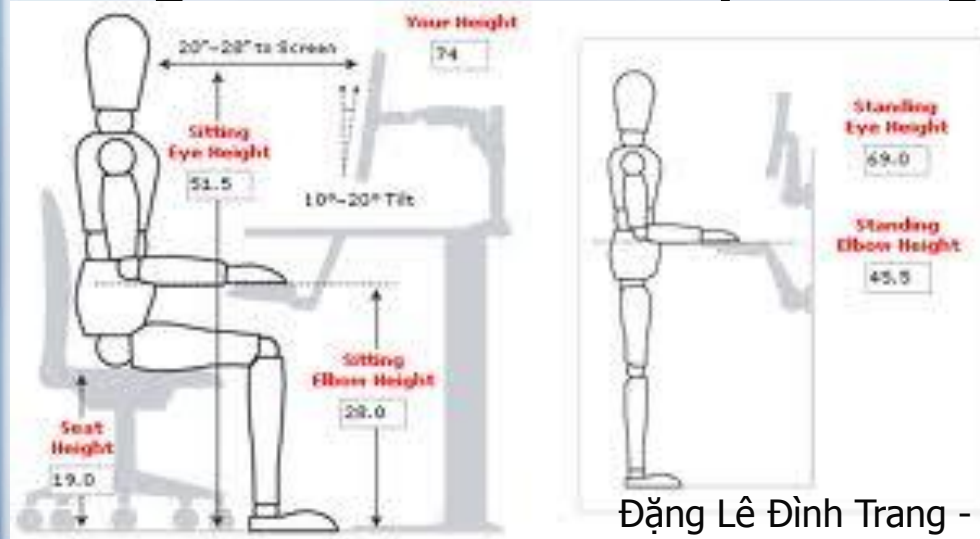
- Mục tiêu của người sử dụng
 - Hành động trên giao diện
 - Các thay đổi trạng thái của hệ thống
 - Hiển thị trên màn hình
 - Diễn giải bởi người sử dụng
- Bộ khung chung để hiểu về tương tác
 - Các giới hạn về vật lý của hệ thống
 - Xác định các thành phần chính của tương tác
 - So sánh đánh giá hệ thống
 - Trừu tượng hóa

3. Thiết kế hợp lý

- Công thái học
(khía cạnh vật lý của giao diện)
- Giao diện công nghiệp

Công thái học (Ergonomics)

- Nghiên cứu về các yếu tố vật lý trong tương tác
- Các yếu tố về con người
- CTH đưa ra các tiêu chuẩn, guidelines cho người thiết kế hệ thống.



Công thái học – Ví dụ

- Sắp xếp các thiết bị điều khiển và hiển thị - nhóm các công cụ điều khiển theo chức năng, mật độ sử dụng, quy trình
- Môi trường xung quanh – sắp xếp chỗ ngồi phù hợp với kích thước người dùng
- Các vấn đề liên quan tới sức khỏe - nhiệt độ, độ ẩm, ánh sáng, độ ồn
- Tâm lý người dùng – màu sắc, hình dáng, ...



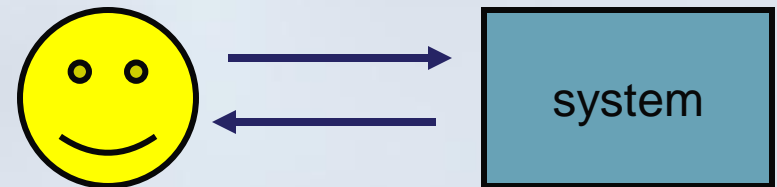
- Giao diện trong công nghiệp và trong văn phòng khác nhau như thế nào?



Giao diện công nghiệp (tiếp)

- Văn phòng– thao tác trực tiếp

- Người dùng tương tác với thế giới ảo

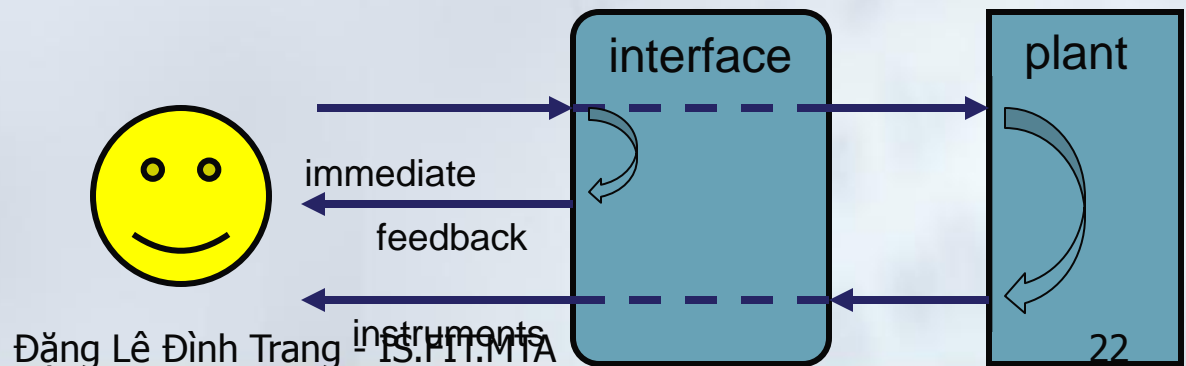


- Công nghiệp – thao tác trực tiếp

- Người dùng tương tác với thế giới thực thông qua giao diện

- Vấn đề đặt ra

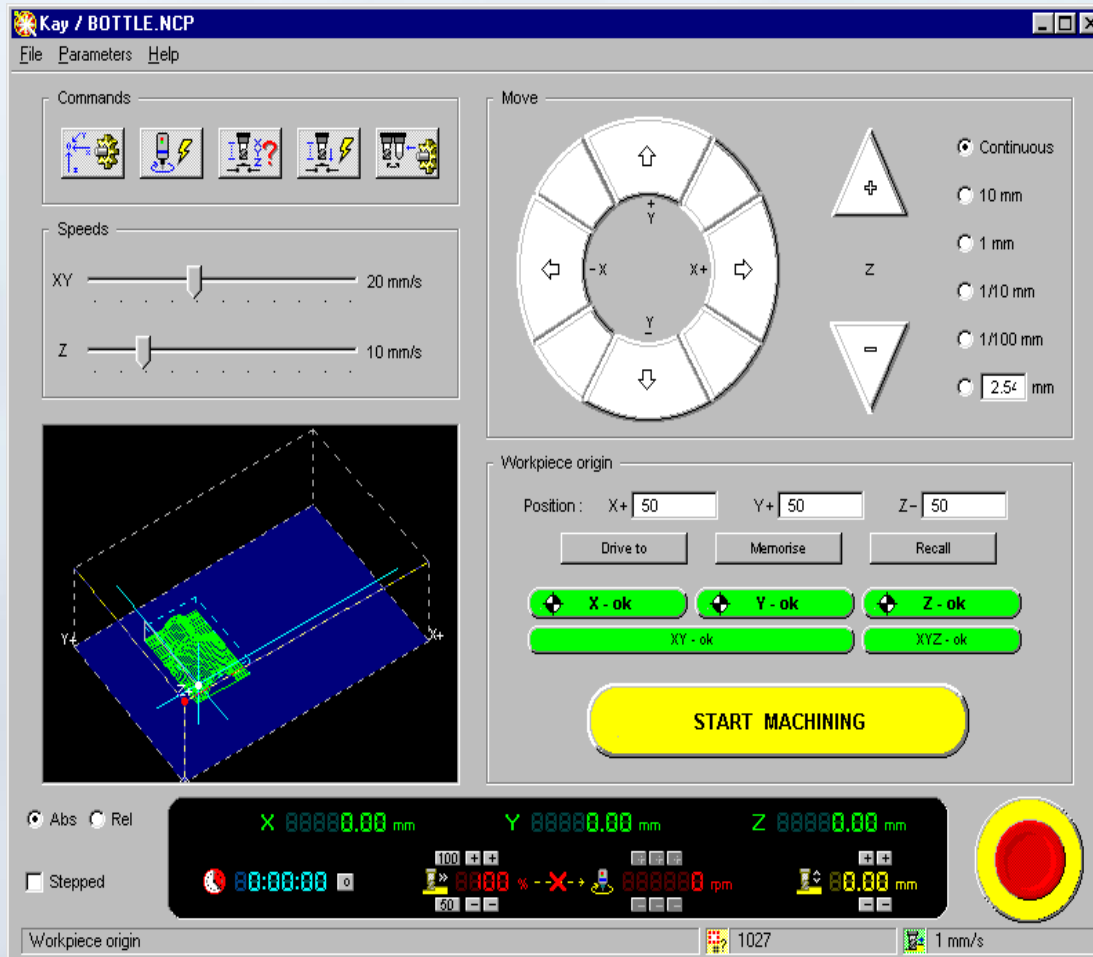
- Phản hồi
- Độ trễ



Giao diện công nghiệp (tiếp)

- Đơn giản, dễ học, dễ vận hành
 - Rõ ràng, đơn chức năng
 - Thể hiện theo quy trình (vận hành, sản xuất)
 - Chọn và bấm
 - Không đòi hỏi về đồ họa
 - Không đòi hỏi tính động (popup, animation,..)
- Giao diện cho ô tô ?

VD: máy CNC vs Car (android)



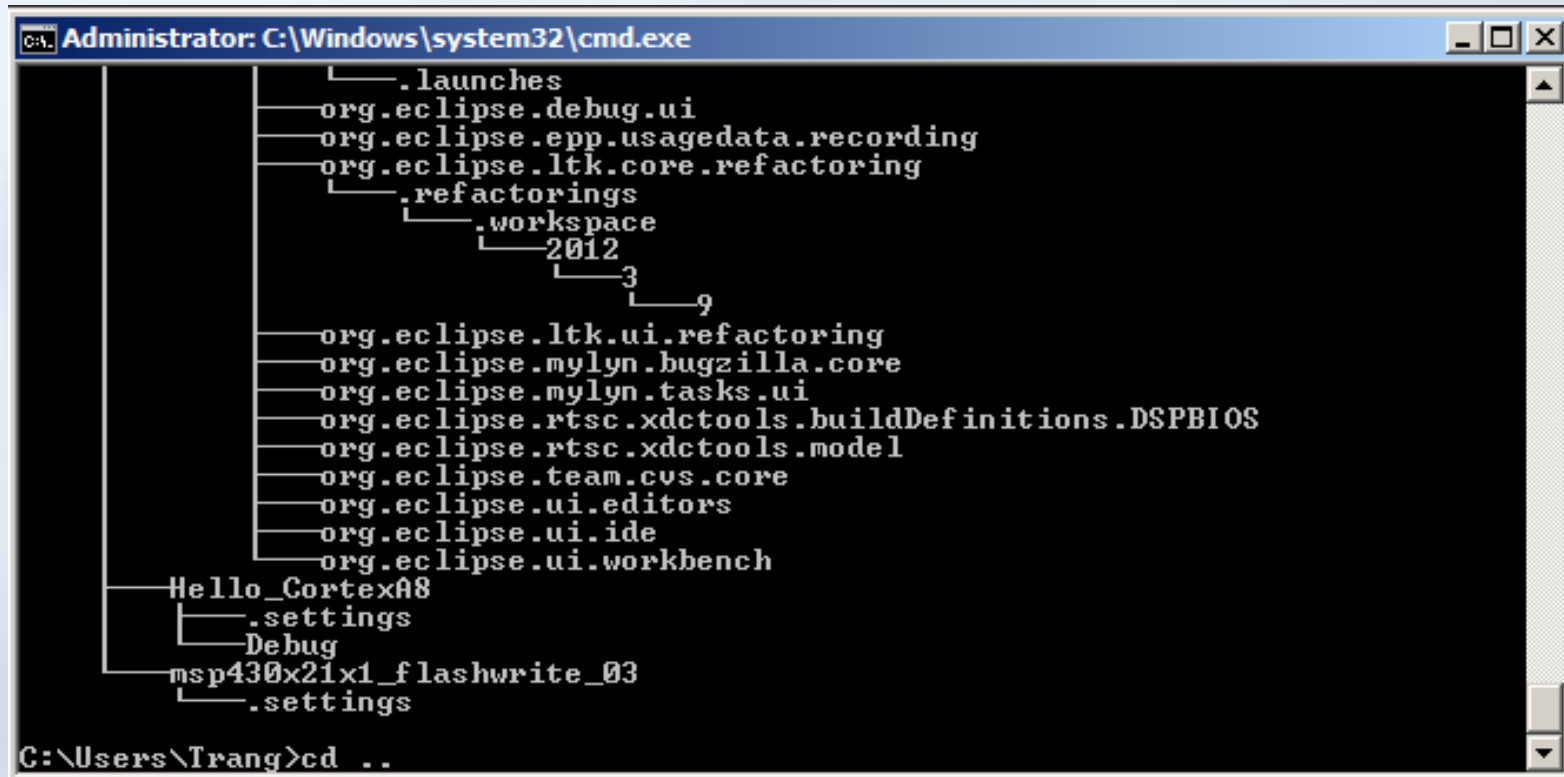
4. Các loại hình tương tác

- Đối thoại giữa máy và người
- Các loại hình tương tác

Các loại tương tác thông thường

- Dòng lệnh
- Menu
- Ngôn ngữ tự nhiên
- Hỏi, trả lời và hội thoại truy vấn
- Điền form và bảng tính
- WIMP
- Trỏ và kích
- Giao diện 3 chiều

Giao diện dòng lệnh



```
Administrator: C:\Windows\system32\cmd.exe
.
├── .launches
├── org.eclipse.debug.ui
├── org.eclipse.epp.usagedata.recording
├── org.eclipse.ltk.core.refactoring
│   └── refactorings
│       └── workspace
│           ├── 2012
│           │   ├── 3
│           │   └── 9
│           └── org.eclipse.ltk.ui.refactoring
├── org.eclipse.mylyn.bugzilla.core
├── org.eclipse.mylyn.tasks.ui
├── org.eclipse.rtsc.xdctools.buildDefinitions.DSPBIOS
├── org.eclipse.rtsc.xdctools.model
├── org.eclipse.team.cvs.core
├── org.eclipse.ui.editors
├── org.eclipse.ui.ide
├── org.eclipse.ui.workbench
├── Hello_CortexA8
│   ├── .settings
│   └── Debug
├── msp430x21x1_flashwrite_03
│   └── .settings
└── C:\Users\Trang>cd ..
```

Ưu, nhược điểm của giao diện này?

Giao diện dòng lệnh (tiếp)

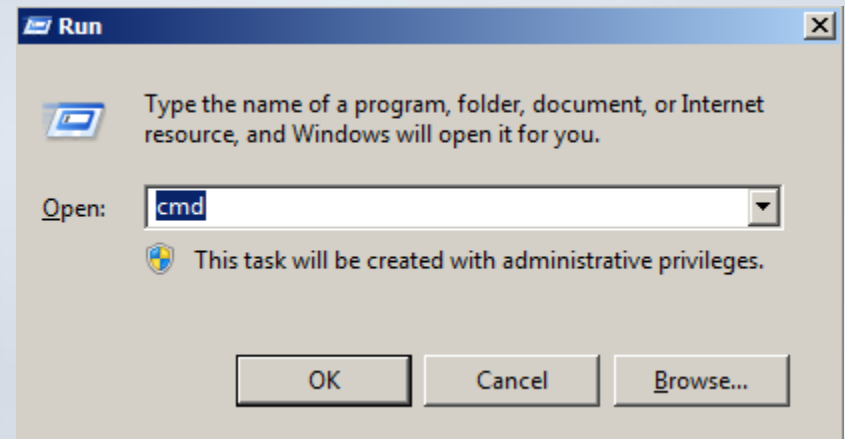
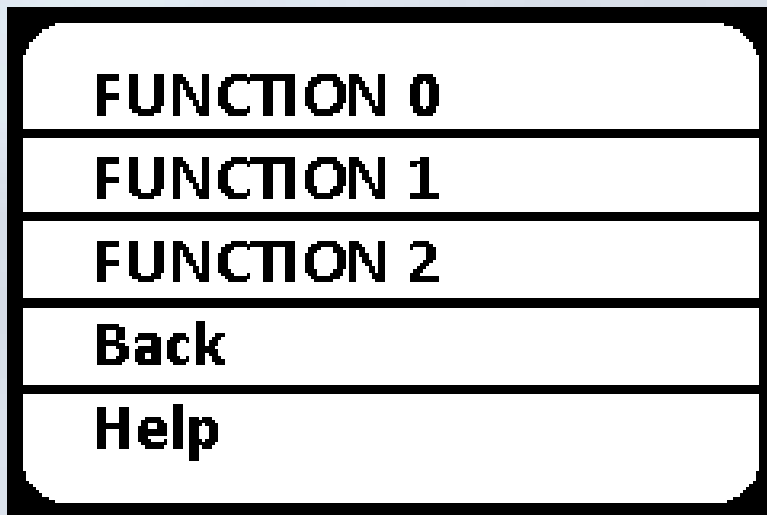
- Diễn giải trực tiếp các chỉ lệnh tới cho máy tính – Có thể nhập lệnh thông qua: phím chức năng, 1 ký tự, từ viết tắt, toàn bộ từ, hoặc tổ hợp
- Phù hợp với các nhiệm vụ có tính lặp lại
- Tốt hơn với người dùng chuyên nghiệp
- Cho phép truy cập trực tiếp tới các chức năng của hệ thống
- Ví dụ điển hình là các hệ Unix

Menu

- Là phương pháp hiển thị tập hợp các lựa chọn
- Ưu điểm:
 - Bao quát các lựa chọn
 - Dễ hiểu, dễ nhớ
 - Giới hạn số lượng đầu vào (không cần xử lý các trường hợp ngoại lệ do người dùng nhập vào)
 - Giảm số lượng phím cần để thao tác

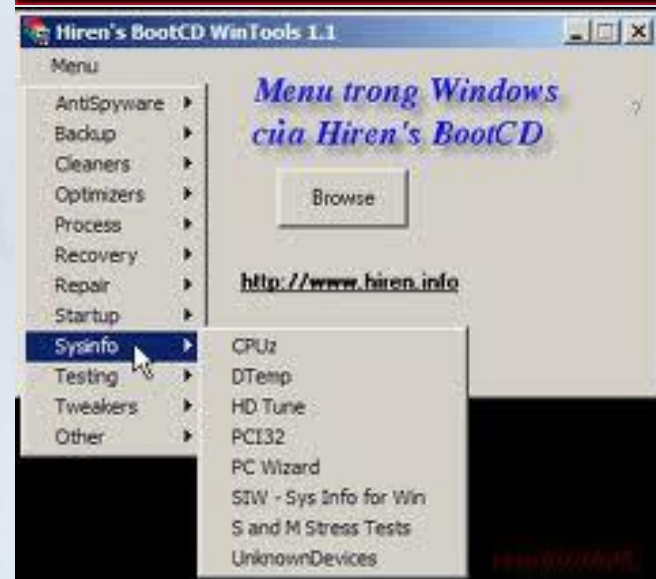
Menu (tiếp)

- Hạn chế:
 - Tốc độ chậm hơn so với comandline
 - Số lượng lệnh là có giới hạn



Menu (tiếp)

- Các phương pháp lựa chọn trong menu:
 - Mũi tên
 - Kí tự, con số
 - Con trỏ (kéo, cuộn, kích)
- Cho phép thực hiện nhiều menu chồng nhau (xếp các chức năng theo nhóm)
- Là nền tảng của giao diện WIMP



Ngôn ngữ tự nhiên

- Là phương pháp nhận và thực thi lệnh thông qua ngôn ngữ tự nhiên của người (text hoặc tiếng nói)
- Thân thiện với người sử dụng
- Một số vấn đề cần giải quyết:
 - Ngôn ngữ tự nhiên là mơ hồ (phụ thuộc nhiều vào ngữ cảnh)
 - Khó để thực hiện tốt
- Giải pháp:
 - Giới hạn số lệnh
 - Bắt từ khóa
 - Kết hợp truy vấn

SEND an E-MAIL

Ok, Who you want to send?

My TEACHER

I don't know him!

Giao diện truy vấn

- Giao diện hỏi và trả lời
 - Người sử dụng thực hiện tương tác thông qua một loạt các câu hỏi
 - Phù hợp với người sử dụng mới, nhưng số chức năng là rất hạn chế
 - Thường được sử dụng trong các hệ thống thông tin
 - Đòi hỏi sự chuẩn bị công phu của các nhà phát triển
- Ngôn ngữ truy vấn (vd SQL)
 - được sử dụng để lấy thông tin từ cơ sở dữ liệu
 - đòi hỏi sự hiểu biết về cấu trúc cơ sở dữ liệu và cú pháp ngôn ngữ, do đó đòi hỏi một số chuyên gia

Điền vào mẫu

- Được sử dụng trong nhập dữ liệu, khôi phục dữ liệu
- Rõ ràng, trực quan
- Yêu cầu
 - Thiết kế tốt, trong sáng
 - Tính bảo mật

Thông tin cần phải nhập

Tên tài khoản:

Xin mời nhập Tên tài khoản bạn muốn dùng trong diễn đàn.

Mật Khẩu:

Nhập lại Mật Khẩu:

Nhập Mật khẩu cho tài khoản của bạn. Chú ý: Mật khẩu phân biệt chữ hoa và chữ thường.

Địa chỉ Email:

Nhập lại Email:

Xin mời nhập địa chỉ Email của bạn.

Ảnh xác nhận



reCAPTCHA™
stop spam.
read books.

Thông tin thêm

Múi giờ:

(GMT +7:00) Bangkok, Hanoi, Jakarta

Tất cả thời gian hiển thị trên diễn đàn có thể tự động điều chỉnh để phù hợp với múi giờ của bạn. Chọn múi giờ gần đúng nhất từ danh sách trên.

Thiết lập quy ước giờ mùa hè (DST):

Tự động phát hiện thiết lập giờ mùa hè

In addition, you may set the appropriate option to allow for daylight saving time in your world.

Nhận Email:

☒ Nhận Email từ Ban Quản Trị

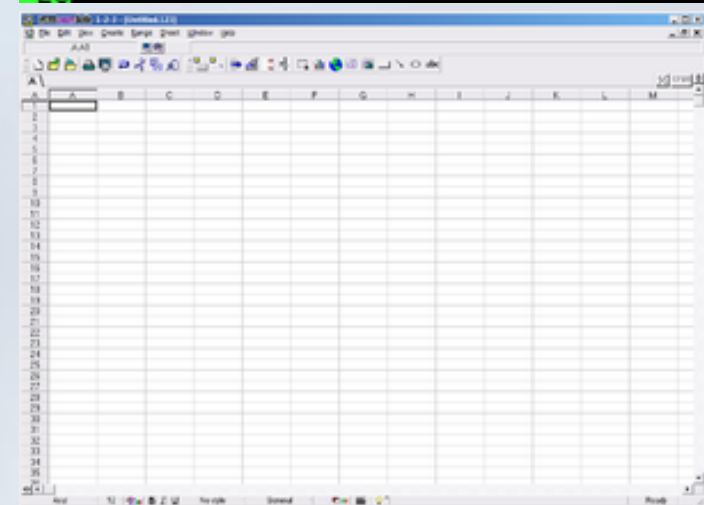
☒ Nhận Email từ thành viên khác

Bảng tính

- Bảng tính đầu tiên là VISICALC, sau đó là Lotus 1-2-3. Ngày nay chương trình bảng tính phổ biến nhất là MS Excel
- Là một biến thể tính vi vừa Form-filling
 - Lưới các ô chứa dữ liệu hoặc công thức
 - Công thức có thể truy cập tới dữ liệu của ô khác
 - Người dùng có thể nhập, thay đổi dữ liệu mà vẫn duy trì tính nhất quán.

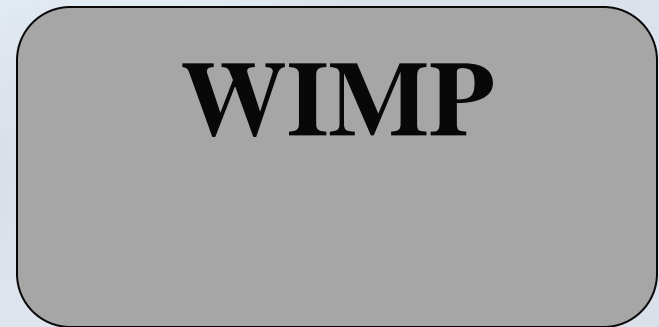
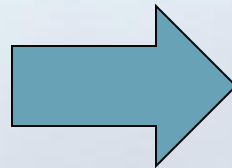
C11 (L) TOTAL C1 25

| 1 | A | B | C | D |
|-----------|-----------|-----|-------|----------|
| | ITEM | NO. | UNIT | COST |
| 1 | MUCK RAKE | 43 | 12.95 | 556.05 |
| 2 | BUZZ CUT | 150 | 4.95 | 742.50 |
| 3 | TOE TONER | 250 | 49.95 | 12487.50 |
| 4 | EYE SNUFF | 2 | 4.95 | 9.90 |
| SUBTOTAL | | | | 13155.50 |
| 9.75% TAX | | | | 1282.66 |
| TOTAL | | | | 14438.16 |



Giao diện WIMP

- Windows
- Icons
- Menus
- Pointers



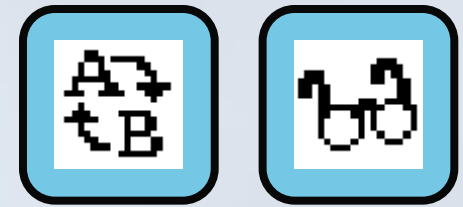
Đây là loại giao diện mặc định, phổ biến trong phần lớn các hệ thống máy tính ngày nay, đặc biệt là các hệ máy tính cá nhân.

Giao diện trở và kích

- Sử dụng trong các ứng dụng:
 - Đa phương tiện
 - Trình duyệt web
 - Siêu liên kết
- Đơn giản – chỉ cần con trỏ chuột / touch pad, nhấn vào biểu tượng, đường link hoặc vị trí trên bản đồ
- Giảm thiểu tối đa việc nhập text

Giao diện 3 chiều

- Thực tại ảo
- Trong các hệ thống bình thường
 - Dùng để làm nổi bật
 - Hiệu quả trực quan
 - Sử dụng quá nhiều có thể gây bối rối
- Không gian làm việc 3 chiều
 - Dùng để mở rộng không gian ảo
 - Ánh sáng và hình khối tạo ra độ sâu
 - Các hiệu ứng về khoảng cách



flat buttons ...



Giao diện 3 chiều (tiếp)



5. Các thành phần của WIMP

- Cửa sổ
- Biểu tượng
- Menu
- Con trỏ
- ...

Cửa sổ chương trình

- Một vùng màn hình hoạt động “độc lập”
 - Nội dung có thể là text, graphic
 - Có thể di chuyển hoặc thay đổi kích thước
 - Có thể xếp chồng, đặt kề bên nhau
- Thanh cuộn – cho phép người dùng di chuyển trong phạm vi cửa sổ
- Thanh tiêu đề - tên của cửa sổ chương trình

Biểu tượng

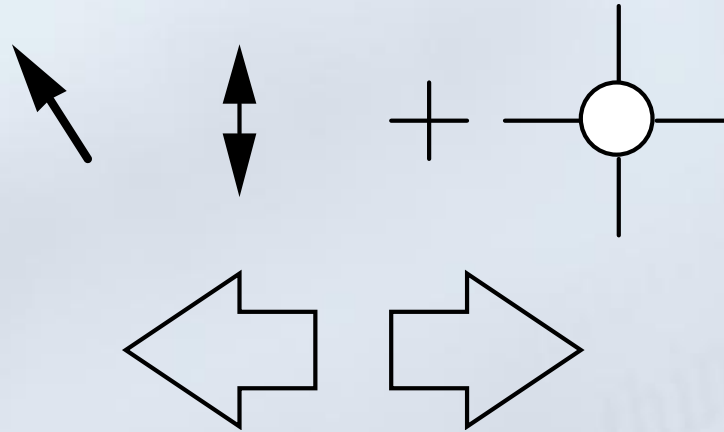
- Các biểu tượng (icon) là các hình ảnh nhỏ, thể hiện nội dung nào đó liên quan tới nội dung ứng dụng
- Đại diện cho một số đối tượng trong giao diện – thường là cửa sổ hoặc hoạt động
- Biểu tượng có thể nhiều và phong phú
 - Tính cách điệu cao
 - Bao hàm nội dung chính của chương trình
 - Tạo ấn tượng với người dùng

Con trỏ

- Là một thành phần quan trọng trong WIMP đại diện cho hành động của người sử dụng, đồng thời phản ánh vùng chức năng mà người dùng đang làm việc.
- Vị trí, cách di chuyển
- Hình dáng tĩnh hoặc động
 - Tính tương tác của chương trình

Con trỏ (tiếp)

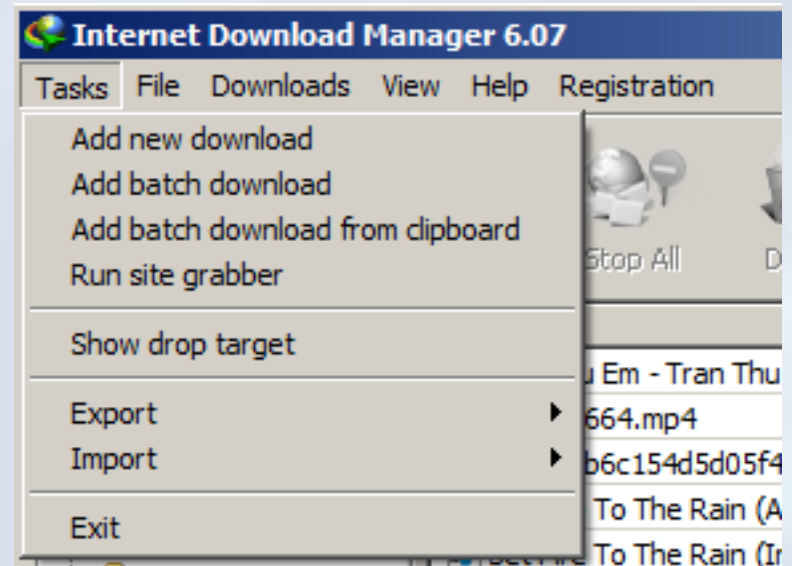
- Bạn biết bao nhiêu cách điều khiển con trỏ?
- Một số hình ảnh con trỏ:



Lưu ý: con trỏ phản ánh nội dung, hành động đang được thực thi

Menu

- Menu trong WIMP như thế nào?
- Dùng để chọn các hàm, chức năng cung cấp trên màn hình
- Yêu cầu lựa chọn với con trỏ
- Vấn đề: chiếm không gian màn hình.
- Phương án: Pop-up?



Các loại menu

- Menu bar ở phía trên của khung cửa sổ chương trình, và menu được hạ xuống khi người dùng yêu cầu
 - Pull-down – giữ và kéo menu
 - Drop-down – kích chuột để menu xuất hiện
 - Fall-down – Đưa chuột đến biểu tượng của menu
- Menu ngữ cảnh, là các loại menu xuất hiện phụ thuộc vào vị trí của con trỏ
 - Pop-up menu – các hành động với đối tượng được chọn
 - Pie menu – sắp xếp theo vòng tròn. Ưu nhược điểm?

Menu mở rộng

- Các menu xếp chồng
 - Cấu trúc menu thừa kế
 - Một menu được lựa chọn lại tiếp tục mở ra các menu mới
 - Tạo ra các danh mục dài vô tận?
- Tăng tốc nhờ bàn phím
 - Tổ hợp phím – tác dụng giống như đối tượng trên menu
 - Tổ hợp phím hoạt động khi menu mở hoặc đóng

Các vấn đề trong thiết kế menu

- Chọn loại menu nào cho phù hợp với ứng dụng?
- Đưa tất cả vào menu được không?
- Ngôn ngữ của menu
- Phương pháp nhóm các đối tượng menu
 - Theo chức năng
 - Theo đối tượng
- Cách chọn tổ hợp phím cho các menu
 - Chữ cái đại diện cho chức năng của menu
 - Thừa kế từ phiên bản trước, nền tảng mà chương trình đang hoạt động

Một số thành phần khác

- Nút bấm – đại diện cho hành động nào đó
- Check box, radio button – dùng để đánh dấu lựa chọn của người dùng
- Thanh công cụ (toolbar) – tập hợp các chức năng, hành động trên các thanh dài, hỗ trợ truy cập nhanh các chức năng hay sử dụng

Text + biểu tượng + hiệu ứng → trực quan

➔ Hiệu quả tương tác tốt hơn, nhưng đừng lạm dụng!

Hộp hội thoại

- Một thành phần quan trọng trong giao diện tương tác.
- Dạng cửa sổ hiển thị thông tin cùng một vài lựa chọn đưa ra cho người dùng về sự kiện quan trọng hay các yêu cầu.
- Ứng dụng:
 - Thông báo
 - Xác nhận(save, delete, ...)
 - Truy vấn dữ liệu

Giao diện sử dụng tiếng nói

- Sử dụng tiếng nói để ra lệnh cho thiết bị
- Phát triển rất nhanh, nhưng vẫn còn thiếu chính xác, bị giới hạn bởi:
 - Ngôn ngữ
 - Ngữ điệu
 - Ngữ cảnh
- VD: Siri

6. Tính tương tác

- Dễ dàng tiếp cận
- Quan sát và cảm nhận

Quan sát và cảm nhận

- Tất cả các hệ thống sử dụng giao diện WIMP đều bao gồm các thành phần giống nhau như: cửa sổ, biểu tượng, menu, ...
- Nhưng mỗi hệ thống lại có cách hoạt động khác nhau

VD: Windows và Ubuntu 11 menu

Hiển thị + hoạt động = Quan sát + cảm nhận

Tính chủ động

- Tính chủ động hay khả năng dự đoán là một tính năng tương đối phổ biến hiện nay trong các phần mềm thiết kế, lập trình.
- Mô tả: đó là khả năng dự đoán trước hành động của người dùng, đưa ra các gợi ý hướng người dùng tới các thao tác chính xác, nhanh chóng
VD: eclipse IDE, MS Excel, Solidwork, ...

Tính chủ động (tiếp)

- Tính chủ động mang lại hiệu quả tương tác đặc biệt cao, khác với các vấn đề khác chủ yếu liên quan tới giao diện, data in/output, tính chủ động đi vào các hành động trên nội dung chương trình → sự thông minh
- Đòi hỏi:
 - Nhà phát triển phải nghiên cứu kỹ về nhu cầu, thói quen, tâm lý người dùng
 - Chuẩn bị chu đáo về mặt nội dung chương trình
 - ...

Lỗi và sửa lỗi

- Vấn đề lỗi trong tương tác là không thể tránh khỏi, đặc biệt là các lỗi của người dùng do giao diện.
- Giao diện càng phức tạp (tính tương tác tốt) càng dễ gặp lỗi. → dự đoán lỗi → sửa lỗi

VD: giao diện hội thoại luôn cần quá trình xác nhận lệnh + một vài câu để “từ chối”

Ngữ cảnh

- Ngữ cảnh ảnh hưởng rất nhiều tới hành động của hệ thống khi xảy ra tương tác
 - Ngữ cảnh chương trình (đang thực thi)
 - Ngữ cảnh hệ thống (Hệ điều hành)
 - Môi trường hoạt động của hệ thống (địa lý, vắng hóa, khí hậu, thời gian, ...)
- Ảnh hưởng tới cảm nhận của người dùng
- Thao tác sử dụng chương trình
- Bối rối, phân vân?

7. Một số yêu cầu khác

- Tính kinh nghiệm
- Yếu tố vật lý
- Các tiêu chuẩn và yếu tố văn hóa
- Thiết kế có tính cân bằng

Tính kinh nghiệm

- Trong quá trình thiết kế, các nhà phát triển luôn phải quan sát, thực hiện các thử nghiệm thực tiễn (kinh nghiệm) để đưa ra các thiết kế trực quan, gần gũi hơn với người dùng

→ Thực tại ảo!

VD: Khi thiết kế phần mềm đọc sách điện tử

+ Thói quen đọc sách

+ Chuyển động của trang sách, ...

Yếu tố vật lý

- Đặc biệt quan trọng khi thiết kế các chương trình mô phỏng thiết bị thực
 - Công thái học – kích thước tối thiểu của nút bấm, ...
 - Hình dạng kết hợp âm thanh, ánh sáng → tạo cảm giác thật
 - Tính động của hình ảnh – khi ấn phím ảo
- Quá nhiều các yếu tố vật lý
 - Cổng kênh, gây áp lực lên phần cứng
 - Tác dụng phụ.

Tiêu chuẩn, văn hóa

- Tùy theo ứng dụng, môi trường, lĩnh vực mà phần mềm được sử dụng, người thiết kế cần quan tâm tới các tiêu chuẩn của ngành nghề đó.

VD: Mạch điện: - màu đỏ: dây nóng, có điện
- màu đen: dây mát

- Yếu tố văn hóa (quốc gia, dân tộc) cũng cần quan tâm.

Thiết kế cân bằng

| Con người | Máy móc |
|--------------------|---------------------|
| Hiệu ứng chân thực | Hiệu năng hoạt động |
| Dễ dàng sử dụng | Thiết kế phức tạp |
| Tương tác tốt | Chi phí thiết kế |
| Thông minh | Thời gian |

Bạn cần tìm hiểu gì?

1. Lựa chọn một phần mềm nào đó (trên PC, smartphone, hay máy giặt) và phân tích tại sao lại có được thiết kế như vậy?
2. Tự mình thiết kế một giao diện bất kỳ, sử dụng các kiến thức đã học (mô hình, phong cách, quy tắc, ...).