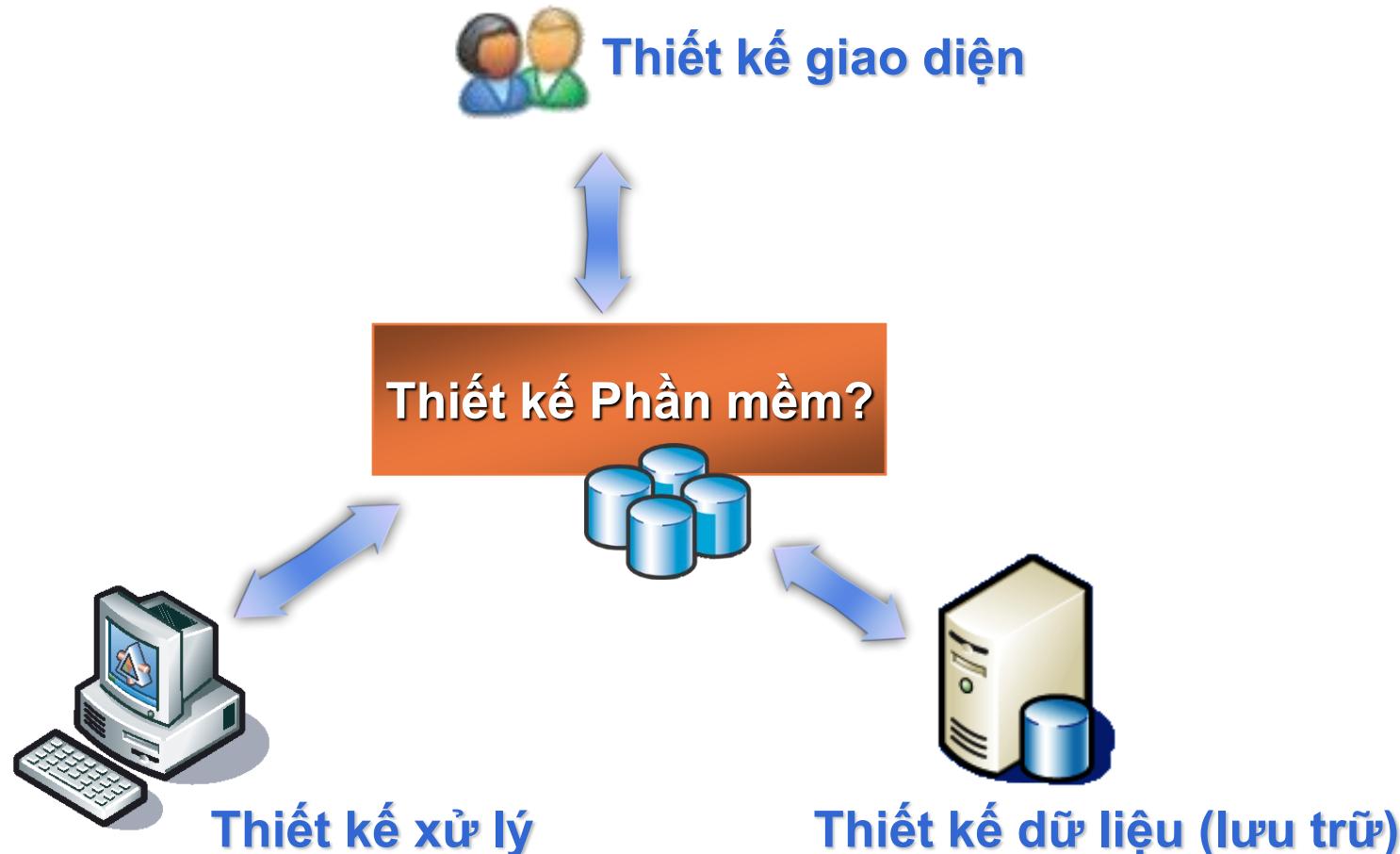


Chương 4: Thiết kế Phần mềm



GVLT: ThS Nguyễn Thị Thanh Trúc

Nội dung



KHÁI NIỆM

HỆ THỐNG LÀM GIÀU ?

- what to do ?
- mô hình dữ liệu
- mô hình chức năng
- yêu cầu phi chức năng

PHÂN TÍCH

CHƯƠNG TRÌNH

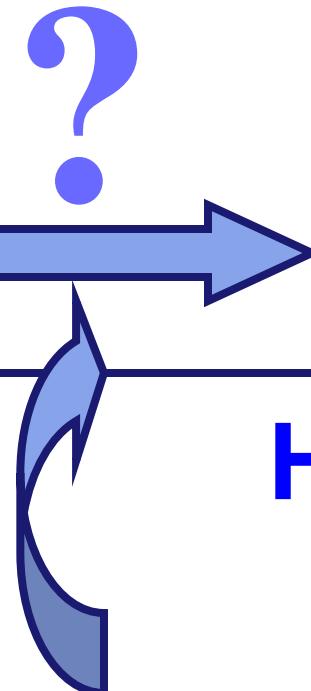
- Thực thi được
- Môi trường cụ thể
(Ngôn ngữ lập trình, Hệ điều hành,...)

LẬP TRÌNH

GIẢI PHÁP?

*làm như thế nào?
HOW TO DO?*

HOẠT ĐỘNG THIẾT KẾ





VAI TRÒ

- ❖ Tạo mô hình cài đặt của phần mềm
- ❖ Là phương tiện trao đổi thông tin để đảm bảo chất lượng
 - dễ hiểu, dễ sửa đổi hơn mã chương trình
 - có nhiều mức chi tiết; cung cấp cái nhìn tổng thể
- ❖ Nếu không có thiết kế; hoặc thiết kế tồi
 - làm tăng công sức mã hóa
 - làm tăng công sức bảo trì

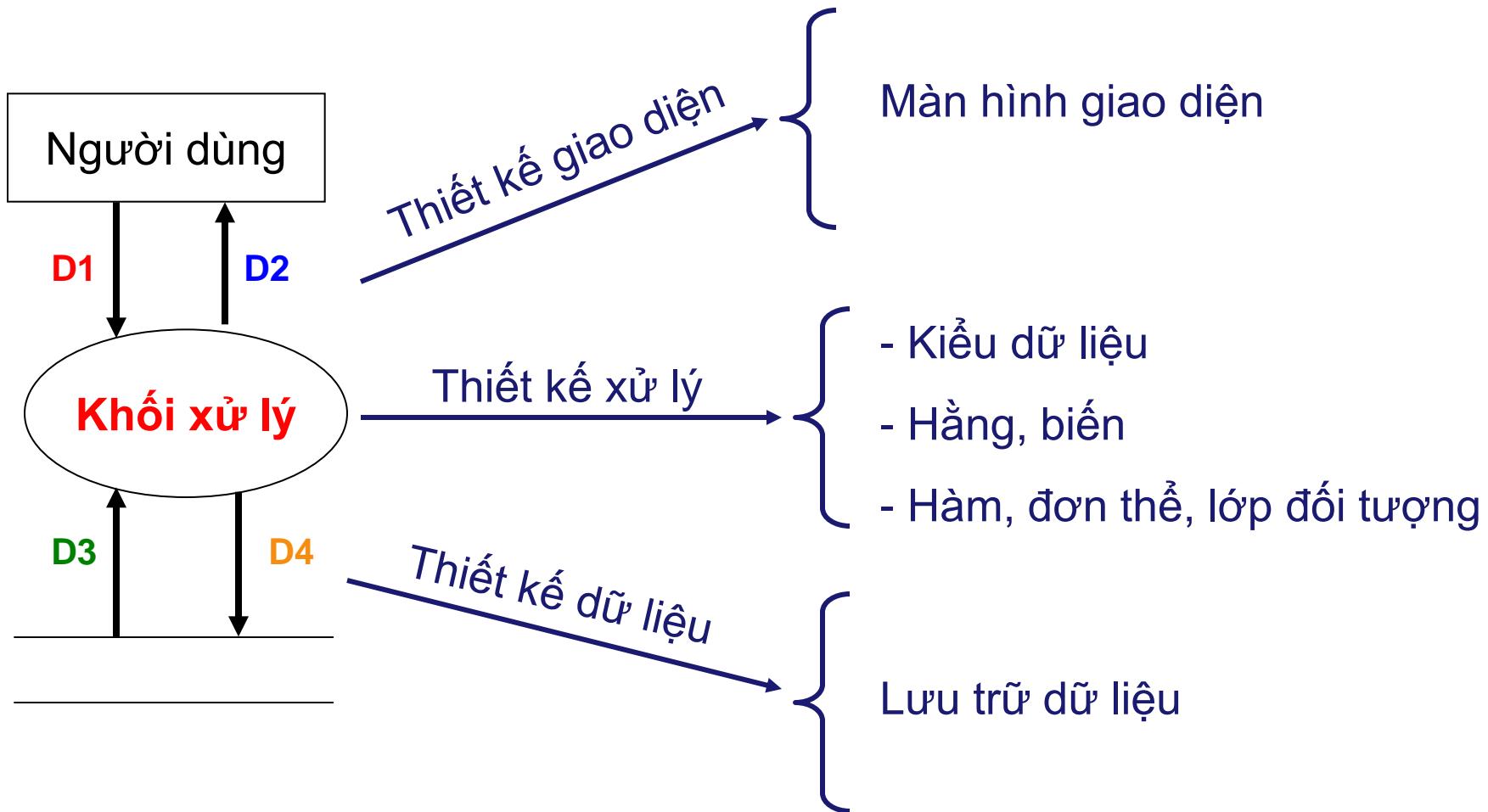
Tổng quan

❖ Khái niệm:

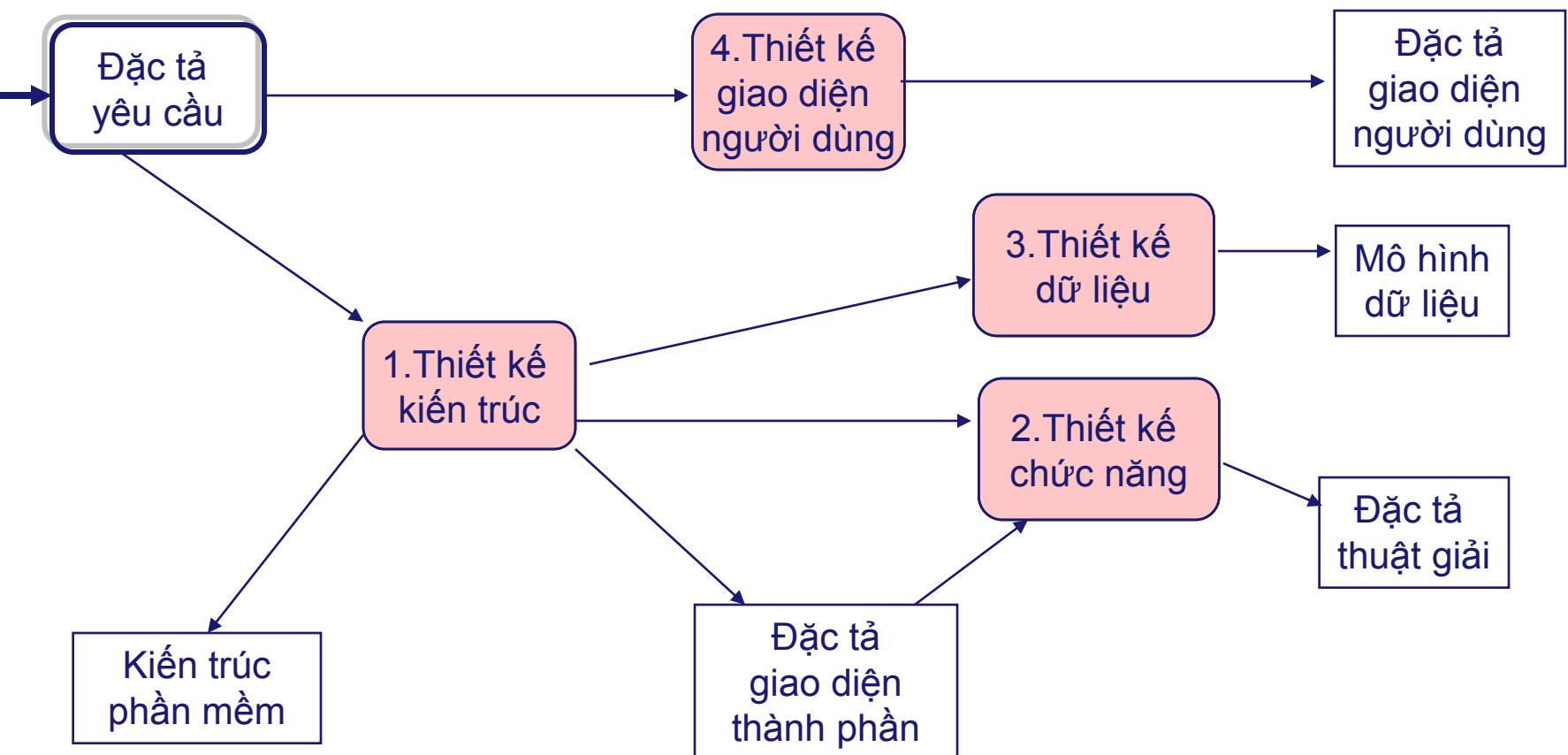
- Thiết kế phần mềm là mô tả chi tiết tổ chức, hoạt động các đơn vị xử lý của phần mềm
- Dựa trên kết quả của việc phân tích
- Là cơ sở cho việc thực hiện phần mềm



Tổng quan



HOẠT ĐỘNG



KHÁI NIỆM

- ❖ Kiến trúc hệ thống là khái niệm “*mức cao nhất*” về hệ thống trong môi trường của nó [IEEE 98]
 - “Mức cao nhất”: trừu tượng hóa từ các chi tiết về thiết kế, triển khai để tập trung vào tính tổng thể của hệ thống
- ❖ Là tập hợp các thành phần khác nhau được tích hợp để tạo nên khái niệm tổng thể
- ❖ Gồm các thành phần phần mềm, các thuộc tính bên ngoài của các thành phần này và mối quan hệ giữa chúng [BAS 98]

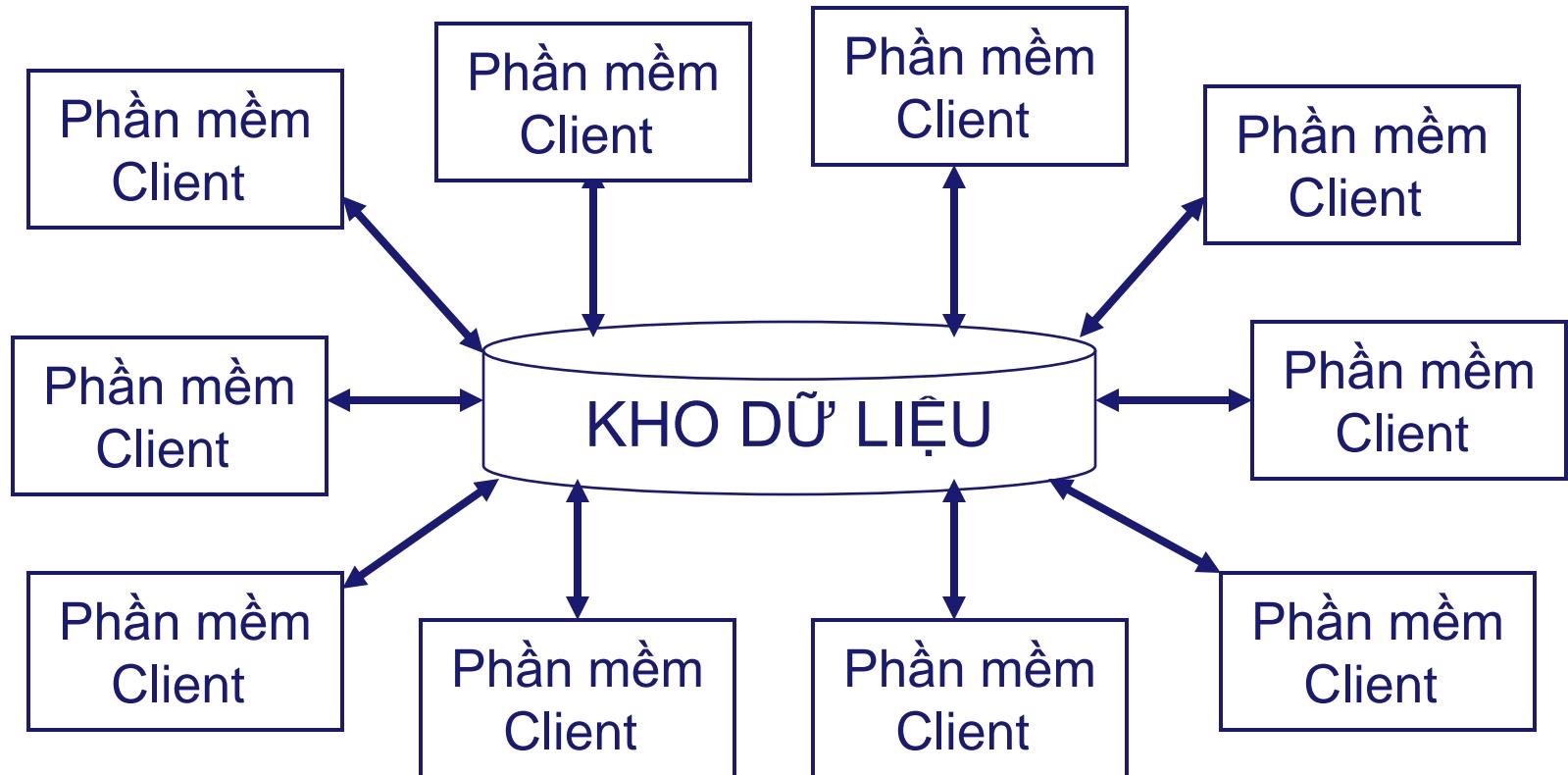


CÁC KIỂU KIẾN TRÚC

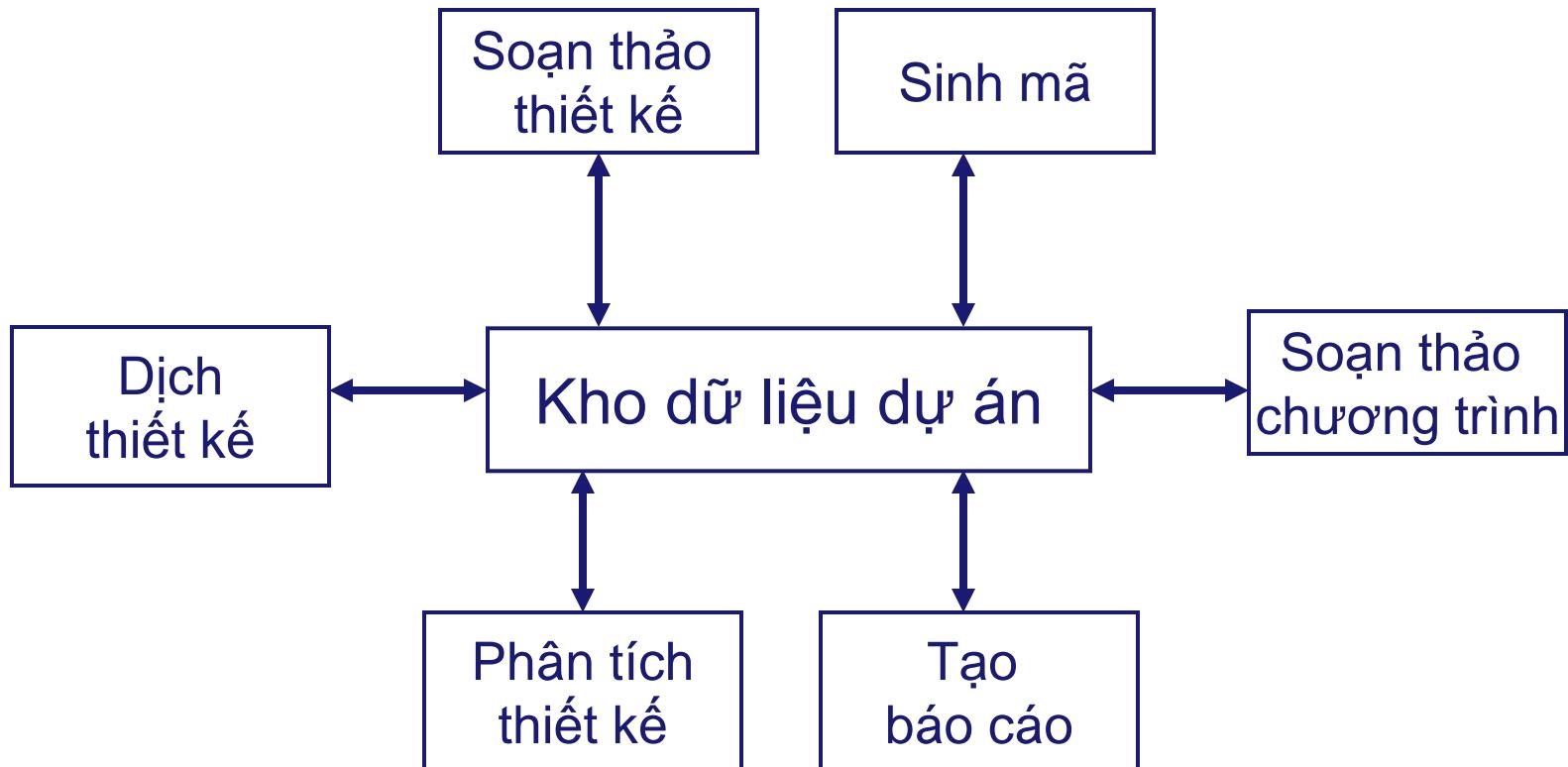
- ❖ Dữ liệu tập trung
- ❖ Khách – dịch vụ
- ❖ Luồng dữ liệu
- ❖ Gọi và trả lại
- ❖ Hướng đối tượng
- ❖ Kiến trúc tầng

KIẾN TRÚC DỮ LIỆU TẬP TRUNG

(Data-Centered Architecture)



KIẾN TRÚC DỮ LIỆU TẬP TRUNG (Data-Centered Architecture)



Bộ công cụ CASE



KIẾN TRÚC DỮ LIỆU TẬP TRUNG

(Data-Centered Architecture)

❖ Ưu điểm

- Tiện lợi cho chia sẻ dữ liệu lớn
- Các phân hệ không cần quan tâm tổ chức dữ liệu

❖ Nhược điểm

- Các phân hệ phải thống nhất mô hình dữ liệu
- Khó thay đổi cấu trúc dữ liệu
- Các phân hệ không thể đưa ra chính sách riêng
- Khó khăn trong quản lý giao dịch



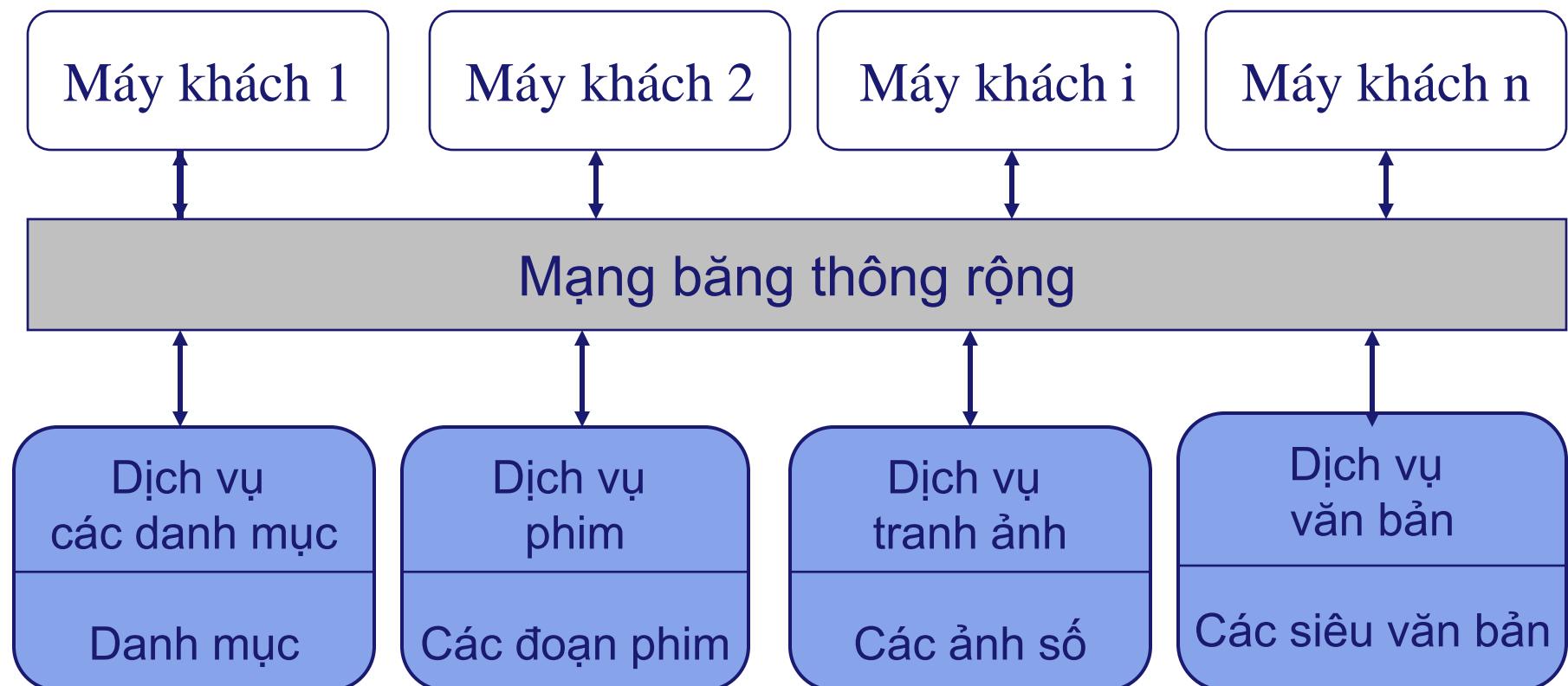
KIẾN TRÚC KHÁCH-DỊCH VỤ

(Client-Server Architecture)

- ❖ Hệ thống được phân tán về dữ liệu và xử lý thông qua quan hệ các thành phần
- ❖ Các máy dịch vụ (server) độc lập cung cấp các dịch vụ đặc biệt như: in ấn, quản lý dữ liệu
- ❖ Các máy khách (client) sử dụng dịch vụ server
- ❖ Hệ thống mạng để client truy cập dịch vụ server

KIẾN TRÚC KHÁCH-DỊCH VỤ (Client-Server Architecture)

Kiến trúc hệ thống thư viện phim & tranh ảnh





KIẾN TRÚC KHÁCH-DỊCH VỤ (Client-Server Architecture)

❖ Ưu điểm

- Sử dụng hiệu quả mạng
- Dễ dàng mở rộng, thêm dịch vụ

❖ Nhược điểm

- Khó tích hợp dữ liệu
- Cần cơ chế bảo toàn dữ liệu cho từng server

Là mô hình phát triển ứng dụng phổ biến



Mô hình Client-Server

❖ Mô hình Thin-Client

- Server=Dữ liệu + Xử lý
- Client=Giao diện

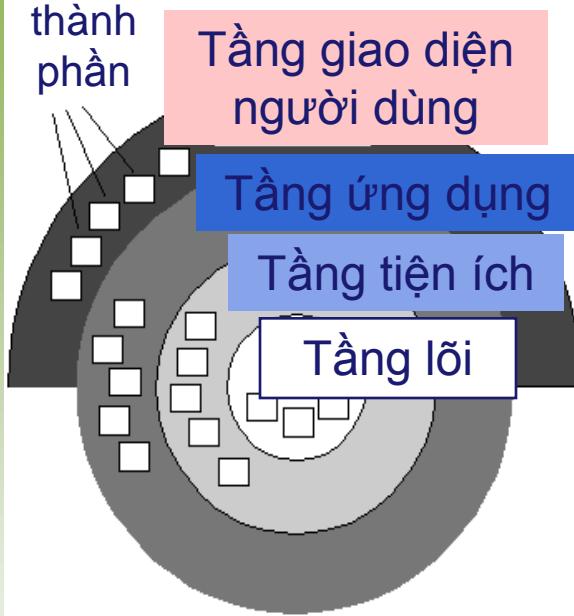
❖ Mô hình Fat-Client

- Server=Dữ liệu
- Client=Xử lý + Giao diện

KIẾN TRÚC PHÂN TẦNG

(Layered Architecture / Abstract machine model)

Các
thành
phần

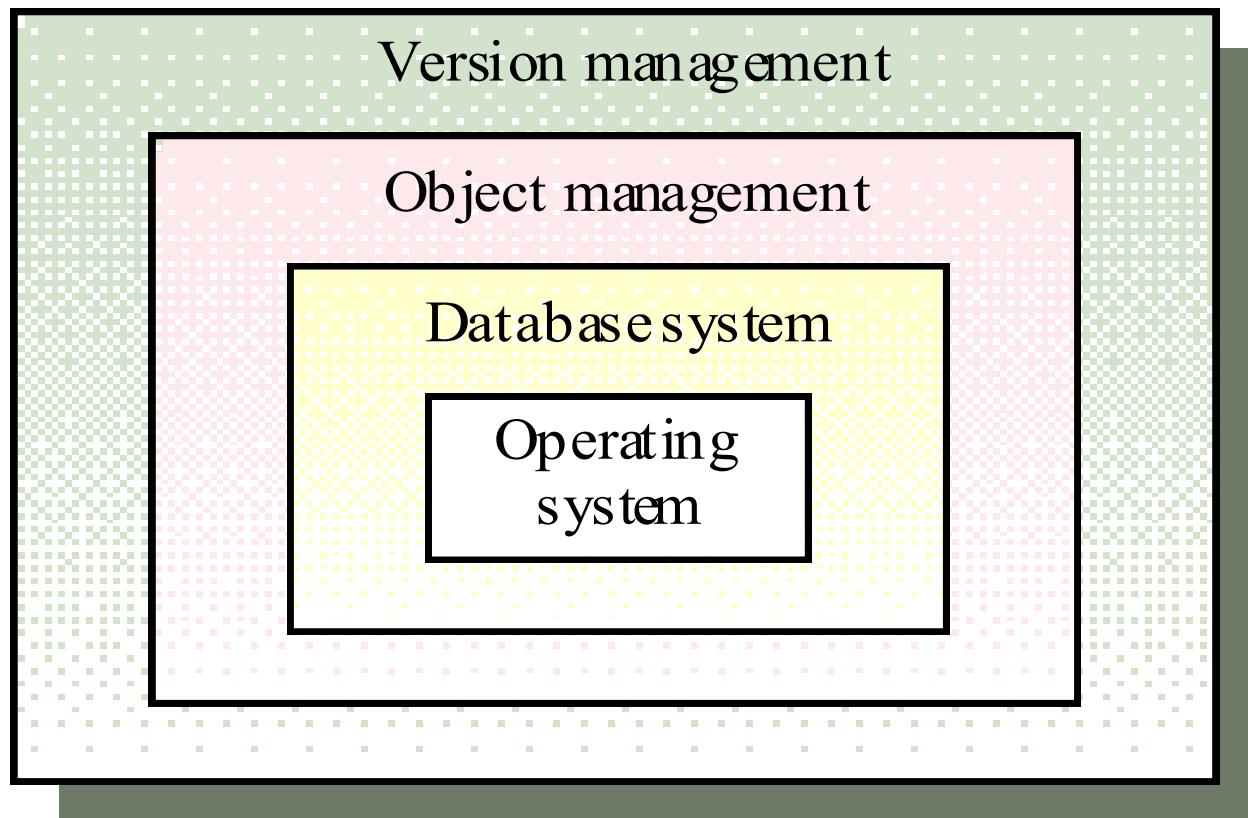


- Mô hình hóa tương tác các hệ con
- Phân rã hệ thống thành các tầng, mỗi tầng cung cấp một tập các dịch vụ
- Mỗi tầng thay đổi thì chỉ ảnh hưởng tầng liền kề => phát triển tăng trưởng
- Khó xây dựng cấu trúc hệ thống theo kiến trúc này

KIẾN TRÚC PHÂN TẦNG

(Layered Architecture / Abstract machine model)

Hệ thống quản lý phiên bản





Mô hình 3-Tiers

- ❖ Đặc điểm: phân làm 3 phân hệ
 - Data:
 - Dịch vụ dữ liệu
 - Data Server
 - Business
 - Thư viện xử lý
 - Application server
 - Client
 - Giao diện người dùng
 - Tương tác theo quy tắc “Thanh máy” **không tương tác vượt tầng**
- ❖ Ưu điểm tương tự mô hình Client-Server, Xử lý chia nhỏ và dùng chung
- ❖ Mô hình đa tầng
 - Mở rộng mô hình 3-Tiers
 - Phân làm nhiều tầng xử lý



Mô hình Peer-to-Peer

❖ Đặc điểm

- Là mô hình phân tán, triển khai trên nhiều máy (nút)
- Các nút tương tác với nhau, mỗi nút đóng vai trò Client-Server
- Chia sẻ dữ liệu xử lý

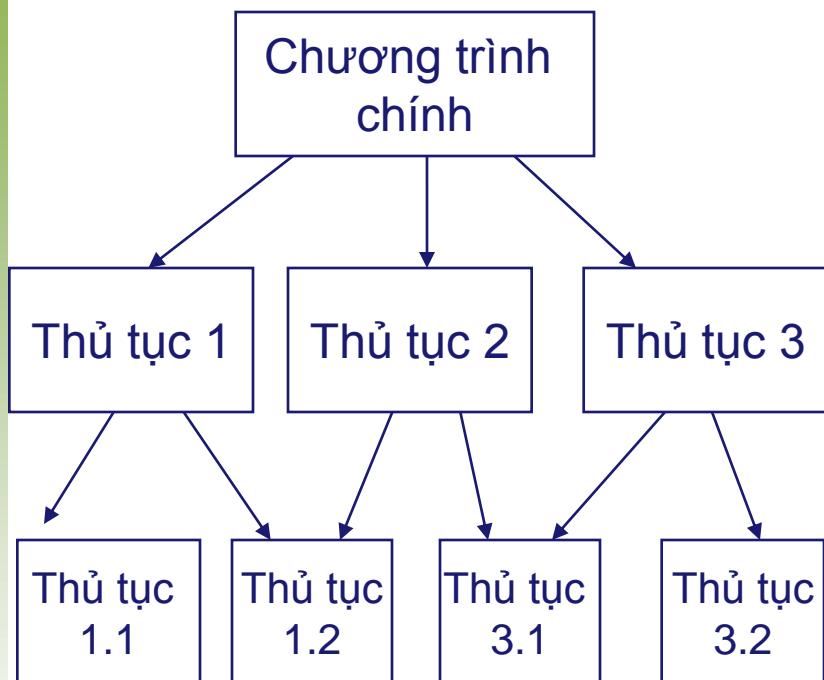
❖ Ưu điểm

- Dễ triển khai, không cần server trung tâm
- Không gian lưu trữ và khả năng xử lý dàn trải

❖ Khuyết điểm

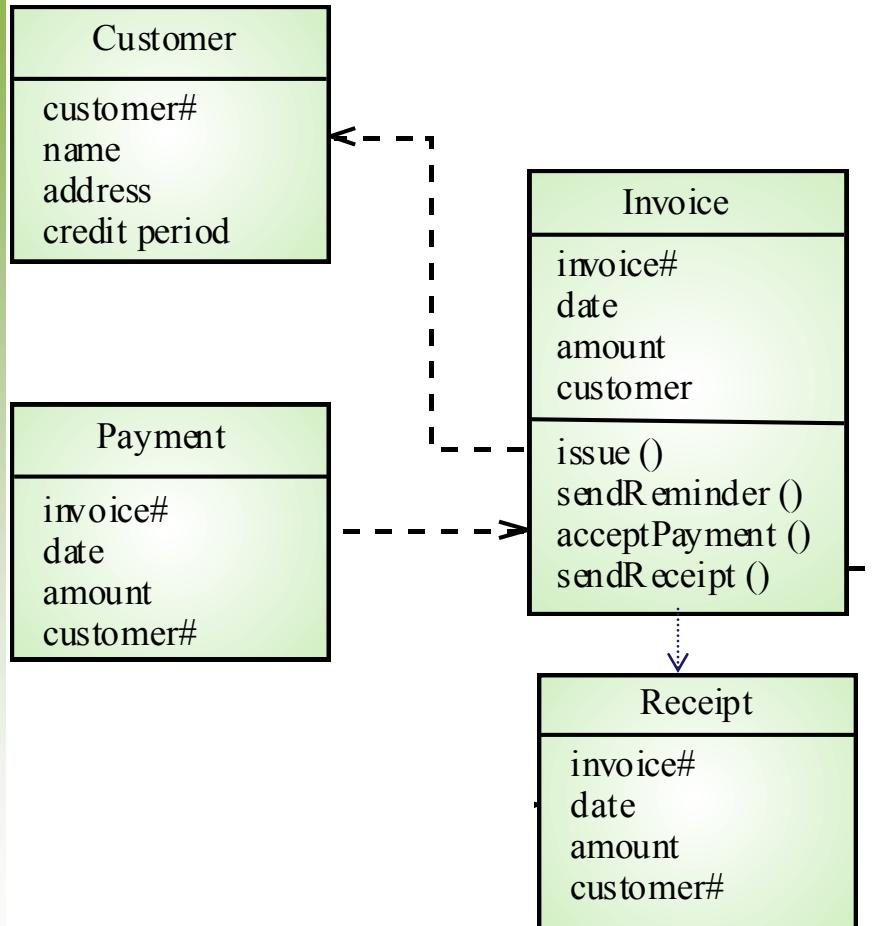
- Khó lập trình và quản lý dữ liệu

KIẾN TRÚC GỌI VÀ TRẢ LẠI (Call and Return Architecture)



- Sử dụng một hệ con để điều khiển, khởi động và dừng các hệ thống con khác
- Truyền điều khiển trên xuống
- Dùng cho các hệ thống tuần tự

MÔ HÌNH HƯỚNG ĐỐI TƯỢNG



- ❖ Là tập hợp đối tượng ghép nối lồng lèo qua giao diện
- ❖ Xác định: lớp đối tượng, thuộc tính, phương thức
- ❖ Xác định mô hình điều khiển để phối hợp thao tác đối tượng



Thiết kế hướng đối tượng

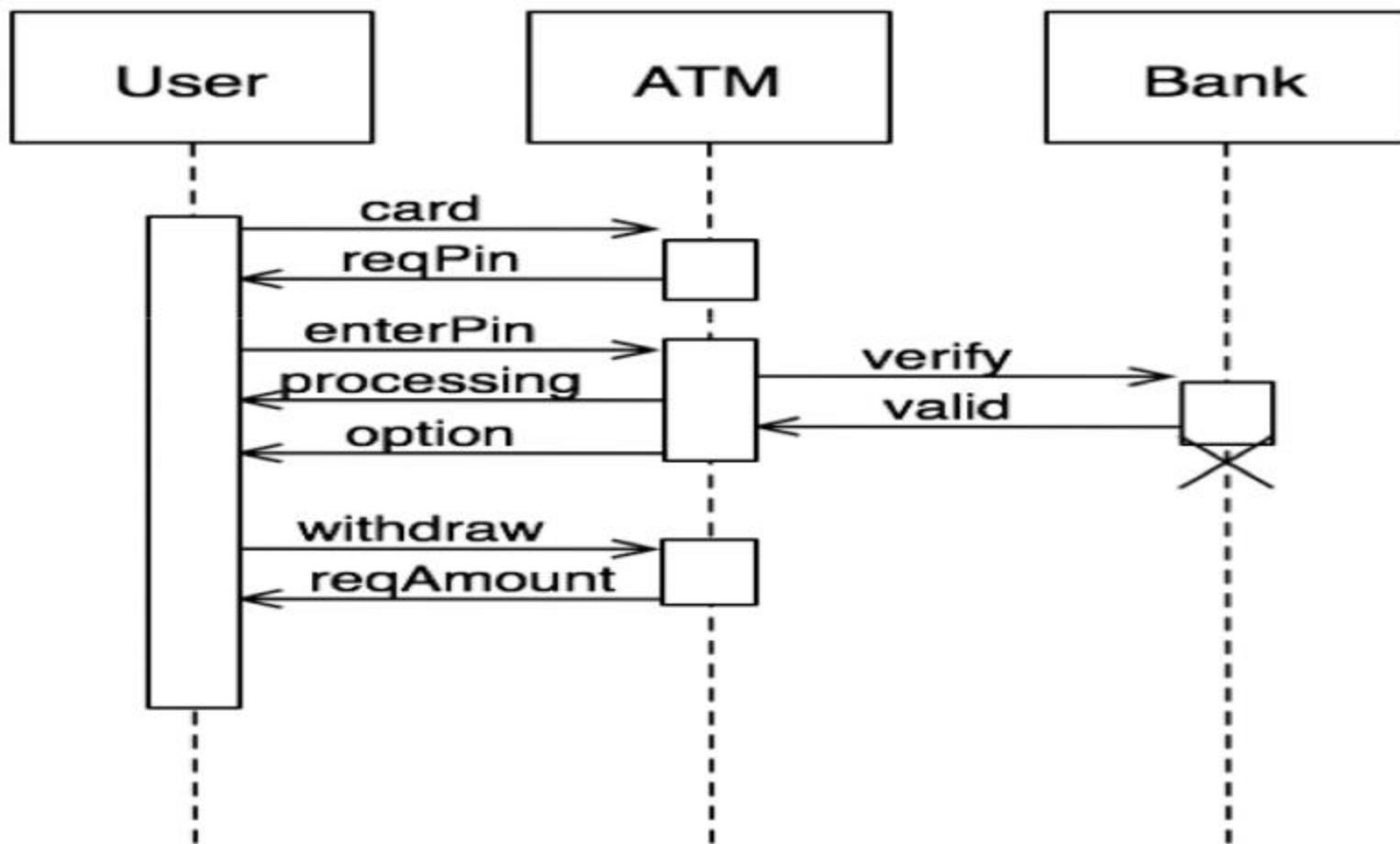
- ❖ Đối tượng= Dữ liệu + Xử lý
 - Dữ liệu \leftrightarrow Thuộc tính
 - Xử lý \leftrightarrow Phương thức
- ❖ Đối tượng là thể hiện cụ thể của lớp đối tượng
- ❖ Phân loại đối tượng
 - Đối tượng nghiệp vụ: business/domain object
 - Dữ liệu + xử lý nghiệp vụ
 - Đối tượng hệ thống
 - Dữ liệu + xử lý hỗ trợ
 - Đối tượng giao diện người dùng: UI object
 - Dữ liệu + xử lý tương ứng người dùng



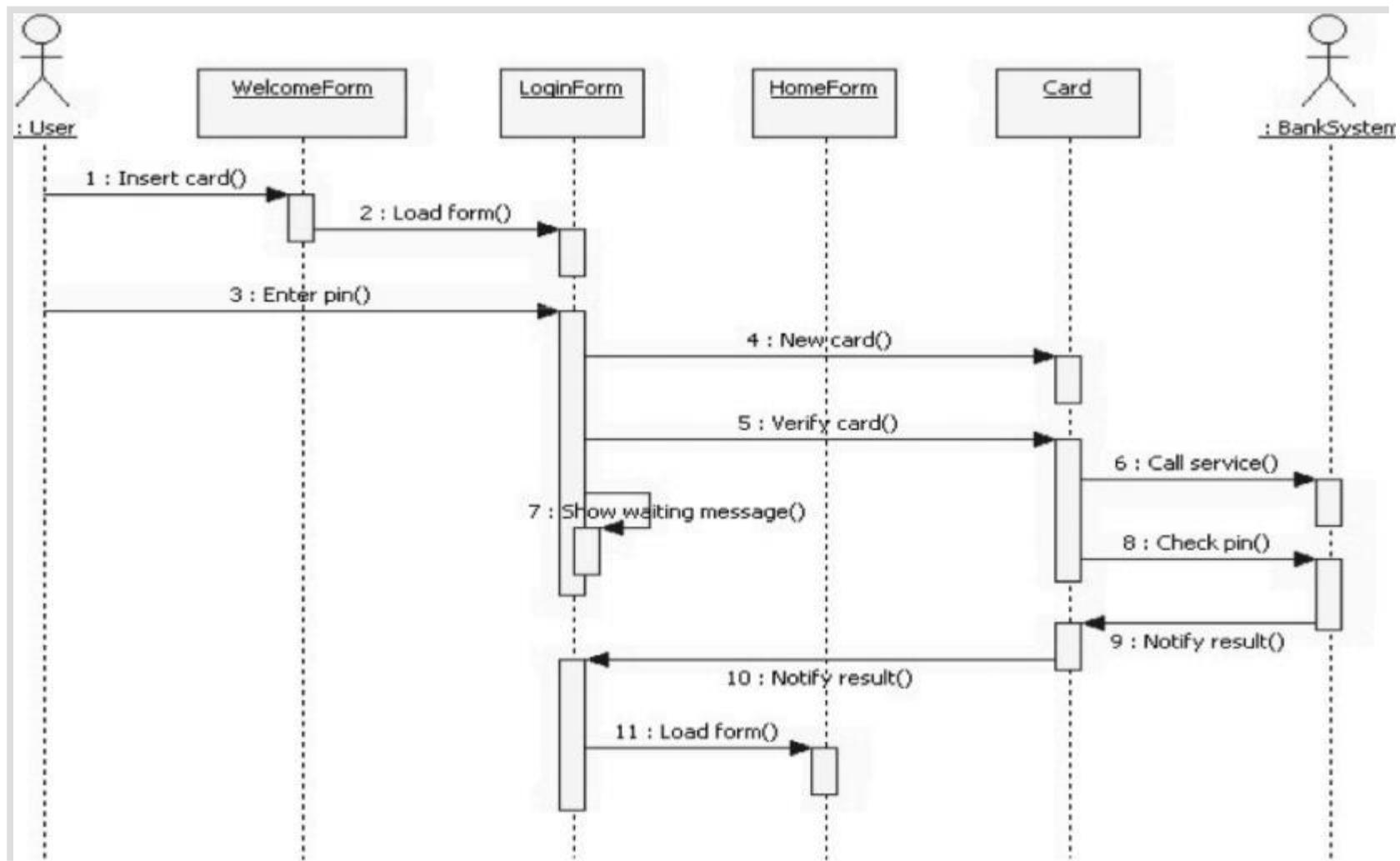
Các bước thiết kế đối tượng

- ❖ Xây dựng kịch bản tương tác ban đầu
- ❖ Xác định đối tượng **dựa trên các tương tác** của kịch bản ban đầu
- ❖ Chi tiết hóa kịch bản ban đầu **có sự tham gia** của các đối tượng **đã xác định**:
 - Phân tích các tương tác thành các xử lý
 - Đối tượng nhận xử lý sẽ chứa các phương thức tương ứng với các xử lý này

Ví dụ



Ví dụ

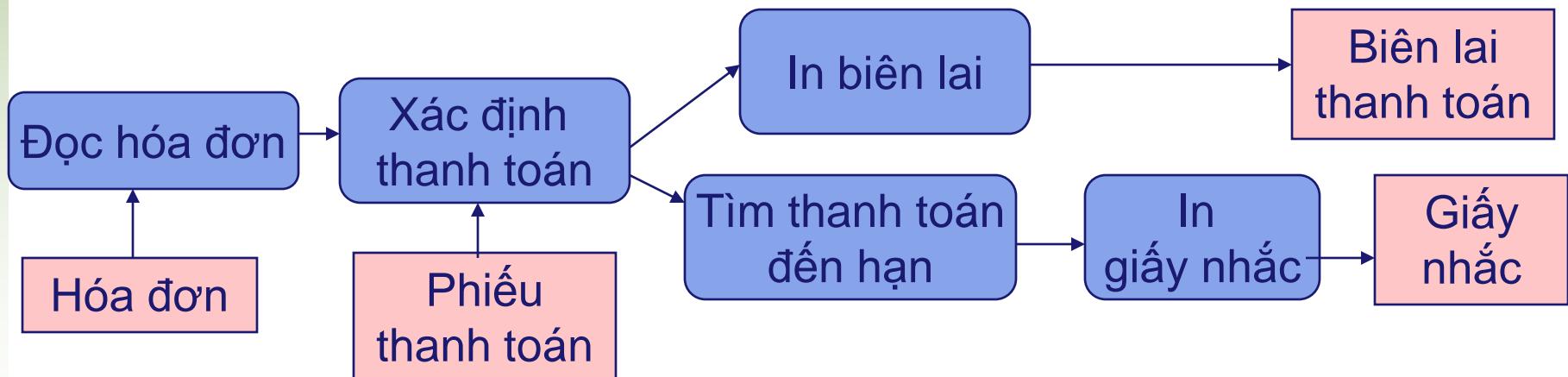


Ví dụ

Tương tác	Đối tượng	Loại
card	WelcomeForm	UI
reqPin	LoginForm	UI
enterPin	LoginForm	UI
verify	Card	Domain
processing	LoginForm	UI
valid	Card	Domain
option	HomeForm	UI
withdraw	HomeForm	UI
reqAmount	WithdrawForm	UI

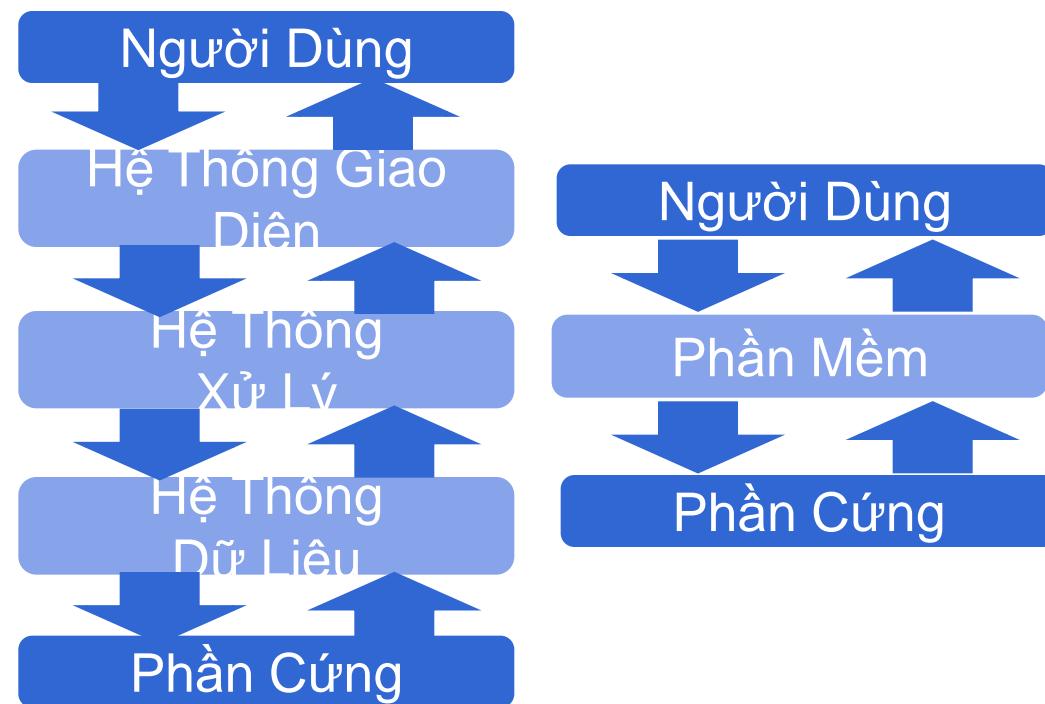
MÔ HÌNH LUỒNG DỮ LIỆU

- ❖ Mỗi chức năng là một tiến trình chuyển hóa các đầu vào thành các đầu ra
- ❖ Các biến thể: pipe/filter model, batch model



KIẾN TRÚC PHẦN MỀM

Mô Hình 3 Lớp





KIẾN TRÚC PHẦN MỀM

- Hệ thống giao diện
 - Nội dung và hình thức trình bày các màn hình giao tiếp của phần mềm.
 - Hệ thống các thao tác mà người dùng thực hiện trên màn hình giao tiếp và xử lý tương ứng của phần mềm.
- Hệ thống xử lý
 - Hệ thống các hàm xử lý của phần mềm tương ứng với các yêu cầu phần mềm.
- Hệ thống dữ liệu
 - Hệ thống các thành phần lưu trữ cùng với quan hệ giữa chúng.
 - Cách thức tổ chức lưu trữ dữ liệu trên bộ nhớ phụ.



Thiết kế phần mềm

- Thiết kế dữ liệu -> sơ đồ logic
- Thiết kế giao diện -> danh sách các màn hình của phần mềm (*màn hình thực đơn, màn hình nhập liệu, màn hình tra cứu...*).
- Thiết kế xử lý -> danh sách các xử lý của phần mềm.



Thiết kế giao diện

- ❖ Thiết kế giao diện: Mô tả chi tiết cách thức **giao tiếp** giữa người sử dụng và phần mềm
- ❖ Màn hình giao diện
 - Nội dung
 - Hình thức trình bày
 - Biến cố phải xử lý
 - Lập danh sách các biến cố:

STT	Điều kiện kích hoạt	Xử lý	Ghi chú
1			
...			

Thiết kế giao diện

- ❖ Mô tả danh sách các thành phần của giao diện:

STT	Tên	Kiểu	Ý nghĩa	Miền giá trị	Giá trị mặc định	Ghi chú
1						
...						



Thiết kế dữ liệu (lưu trữ)

- ❖ Thiết kế dữ liệu: Mô tả chi tiết cách thức tổ chức, lưu trữ các dữ liệu của phần mềm
- ❖ Tổ chức lưu trữ
 - Bảng/Tập tin
 - Thuộc tính/Cấu trúc
 - Liên kết giữa các bảng/Tập tin
- ❖ Danh sách các bảng

STT	Bảng	Ý nghĩa	Ghi chú
1			
...			



Thiết kế dữ liệu (lưu trữ)

- ❖ Danh sách các thuộc tính bảng X

STT	Thuộc tính	Kiểu	Ràng buộc	Giá trị Khởi động	Ghi chú
1					
...					

Thiết kế Xử lý

- ❖ Thiết kế xử lý: Mô tả chi tiết hệ thống các hàm xử lý (cùng với các hằng, biến, kiểu liên quan) của phần mềm
- ❖ Danh sách các kiểu dữ liệu xử lý

STT	Kiểu dữ liệu	Ý nghĩa	Ghi chú
...

- ❖ Danh sách các thuộc tính kiểu dữ liệu X

STT	Thuộc tính	Kiểu	Ràng buộc	Giá trị khởi động	Ghi chú
...

Thiết kế Xử lý

❖ Danh sách các biến

STT	Biến	Kiểu	Ý nghĩa	Ghi chú
...

❖ Danh sách các hằng

STT	Hằng	Kiểu	Giá trị	Ý nghĩa	Ghi chú
...

❖ Danh sách các hàm xử lý

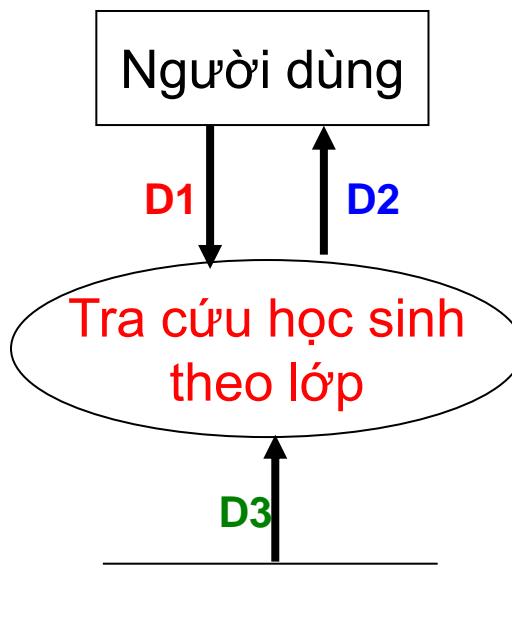
STT	Hàm	Tham số	Kết quả trả về	Thuật giải	Ý nghĩa	Ghi chú
...

Ví dụ

- ❖ Phân tích, thiết kế chức năng tra cứu học sinh theo lớp

1. Phân tích

Sơ đồ luồng dữ liệu



❖ Giải thích:

- D1: Lớp được chọn
- D2: Danh sách lớp, Danh sách học sinh trong lớp được chọn
- D3: Danh sách lớp, Danh sách học sinh

❖ Thuật toán xử lý

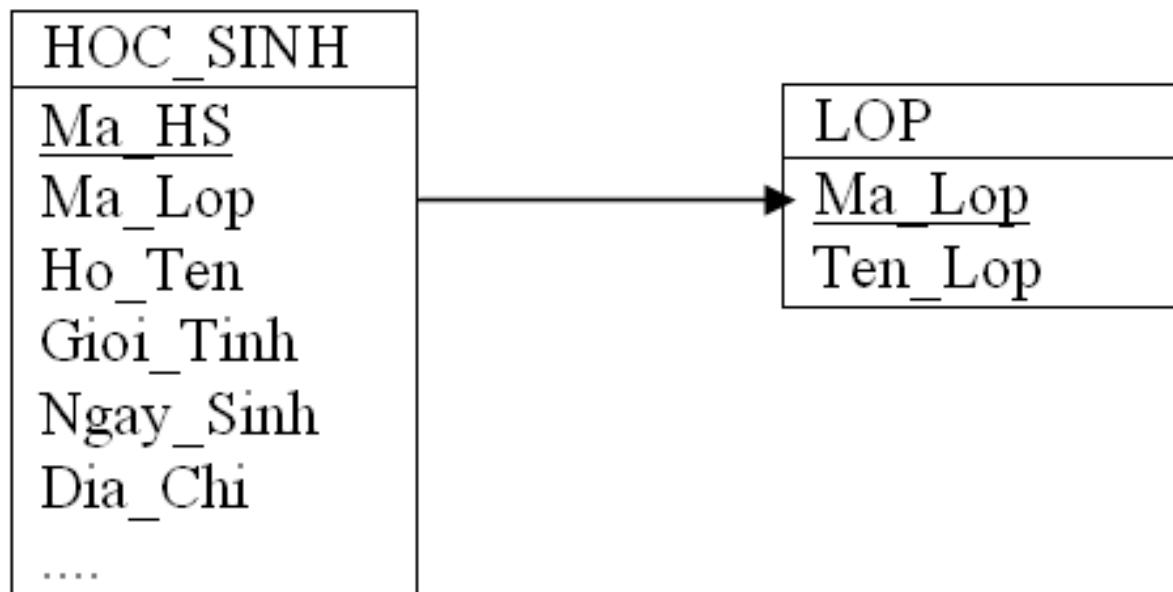
- Đọc D3, Xuất danh sách lớp
- Nhập D1
- Tính D2 (DS học sinh trong lớp được chọn)
- Xuất D2 (DS học sinh trong lớp được chọn)

Ví dụ (tt)

2. Thiết kế

a. Thiết kế dữ liệu (lưu trữ)

Dùng cơ sở dữ liệu quan hệ gồm hai bảng: HOC_SINH, LOP



Ví dụ (tt)

2. Thiết kế

b. Thiết kế giao diện

Màn hình tra cứu học sinh theo lớp

Danh sách lớp

STT	Mã lớp	Tên lớp

Danh sách học sinh

STT	Họ tên	Giới tính	Ngày sinh	Địa chỉ

(0)

(1)

Cách 1

Ví dụ (tt)

2. Thiết kế

b. Thiết kế giao diện

Màn hình tra cứu học sinh theo lớp

Tên lớp (0) 

Danh sách học sinh

STT	Họ tên	Giới tính	Ngày sinh	Địa chỉ

(1) 

Cách 2

Ví dụ (tt)

2. Thiết kế

b. Thiết kế giao diện

Cách 3

Ví dụ (tt)

2. Thiết kế

b. Thiết kế giao diện

Danh sách các biến cố

STT	Điều kiện kích hoạt	Xử lý
0	Khởi động màn hình	???
1	Chọn lớp	???
2	Chọn học sinh	???

Ví dụ (tt)

2. Thiết kế

b. Thiết kế xử lý

STT	Hàm	Tham số	Kiểu trả về	Thuật giải	Ghi chú
1	Xuất danh sách lớp			?	
2	Xuất danh sách học sinh			?	
...				

3. Bài tập: Mở rộng khi Trường quản lý phân cấp theo khối

Thiết kế dữ liệu



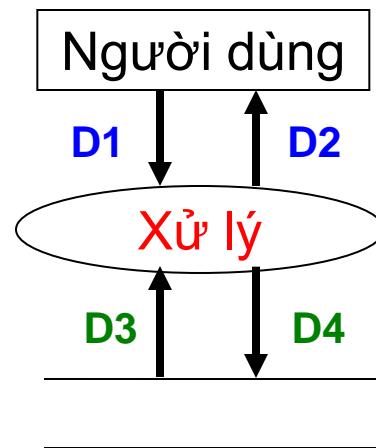


Thiết kế dữ liệu

- **Mục tiêu của việc thiết kế dữ liệu là nhằm mô tả cách thức tổ chức lưu trữ dữ liệu của phần mềm bên trong máy tính.**
- **Kết quả của quá trình thiết kế dữ liệu là xây dựng được sơ đồ logic.**
- **Khi thiết kế dữ liệu, ta quan tâm đến ba vấn đề sau:**
 - Thiết kế dữ liệu với tính đúng đắn
 - Thiết kế dữ liệu với tính tiến hóa
 - Thiết kế dữ liệu với yêu cầu hiệu quả về mặt truy suất và lưu trữ

Thiết kế dữ liệu

- ❖ **Mục tiêu:** Mô tả cách thức **tổ chức lưu trữ** dữ liệu của phần mềm/hệ thống dựa trên phần mềm.



- ❖ D1, D2 được lưu trữ theo cách thức nào trong **bộ nhớ chính**?
- ❖ D3, D4 được lưu trữ theo cách thức nào trong **bộ nhớ phụ**?



Thiết kế dữ liệu

❖ Cách thức lưu trữ dữ liệu trên **bộ nhớ chính**:

- Kiểu cấu trúc
- Kiểu mảng
- Kiểu xâu
- Kiểu cây
- ...

❖ Cách thức lưu trữ dữ liệu trên **bộ nhớ phu**:

- Tập tin (Có cấu trúc/Không có cấu trúc)
- Cơ sở dữ liệu (Quan hệ/Đối tượng)



Thiết kế dữ liệu

- ❖ Kết quả về cách thức lưu trữ trên bộ nhớ phụ (CSDL)
 - Danh sách các bảng (Table)
 - Mối quan hệ giữa các bảng.
 - Thông tin chi tiết của từng bảng với danh sách các thuộc tính, khóa chính



Thiết kế dữ liệu

❖ Các yêu cầu chất lượng:

- Tính đúng đắn: Lưu trữ đầy đủ và đúng nghĩa các thông tin có trong nghiệp vụ liên quan
- Tính tiến hóa: Lưu trữ thông tin về **tổ chức** và **qui định** có trong nghiệp vụ liên quan
- Tính hiệu quả: Lưu trữ **tiết kiệm** nhất không gian bộ nhớ, **truy xuất nhanh** nhất thông tin cần thiết
- Tính bảo mật: Lưu trữ thông tin về các người sử dụng phần mềm cùng với **quyền hạn** tương ứng



Thiết kế dữ liệu

❖ Thuộc tính

- Là các đặc trưng mô tả về đối tượng

❖ Thực thể

❖ Sơ đồ logic

- Công cụ cho phép **mô tả trực quan** cách thức lưu trữ dữ liệu trên bộ nhớ phụ với việc sử dụng cơ sở dữ liệu quan hệ
- Liên kết giữa các bảng (**mối kết hợp**)

Phân loại thuộc tính

- ❖ Thuộc tính **khóa**
- ❖ Thuộc tính có **giá trị rác**
- ❖ Thuộc tính **đa trị**
- ❖ Thuộc tính là **đối tượng phụ**
- ❖ Thuộc tính **tính toán**

GiangVien			
GVID	<pi>	TNBID	<M>
MaGV		TNBCODE	
HocVi		TNBNAME	
ChucDanh		TNBNAME	
TenVT		TNBNAME	
Ho		TNBNAME	
Ten		TNBNAME	
DiaChi		TNBDESCRIPTION	
DienThoai		TNBPHONE	
Email		TNBPHONE	

Phân loại thuộc tính

- ❖ Thuộc tính **khóa**
- ❖ Thuộc tính có **giá trị rời rạc**:
 - Thuộc tính phái của Sinh viên chỉ có 2 giá trị: **Nam hoặc Nữ**
 - Thuộc tính phái của Sinh viên chỉ có 2 giá trị: **0 hoặc 1 (0: Nam, 1: Nữ)**
 - Thuộc tính điểm học phần chỉ có các giá trị **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.**
 - Thuộc tính loại nhân viên chỉ có các giá trị: **Nhân viên văn phòng, nhân viên bán hàng, nhân viên sản xuất.**

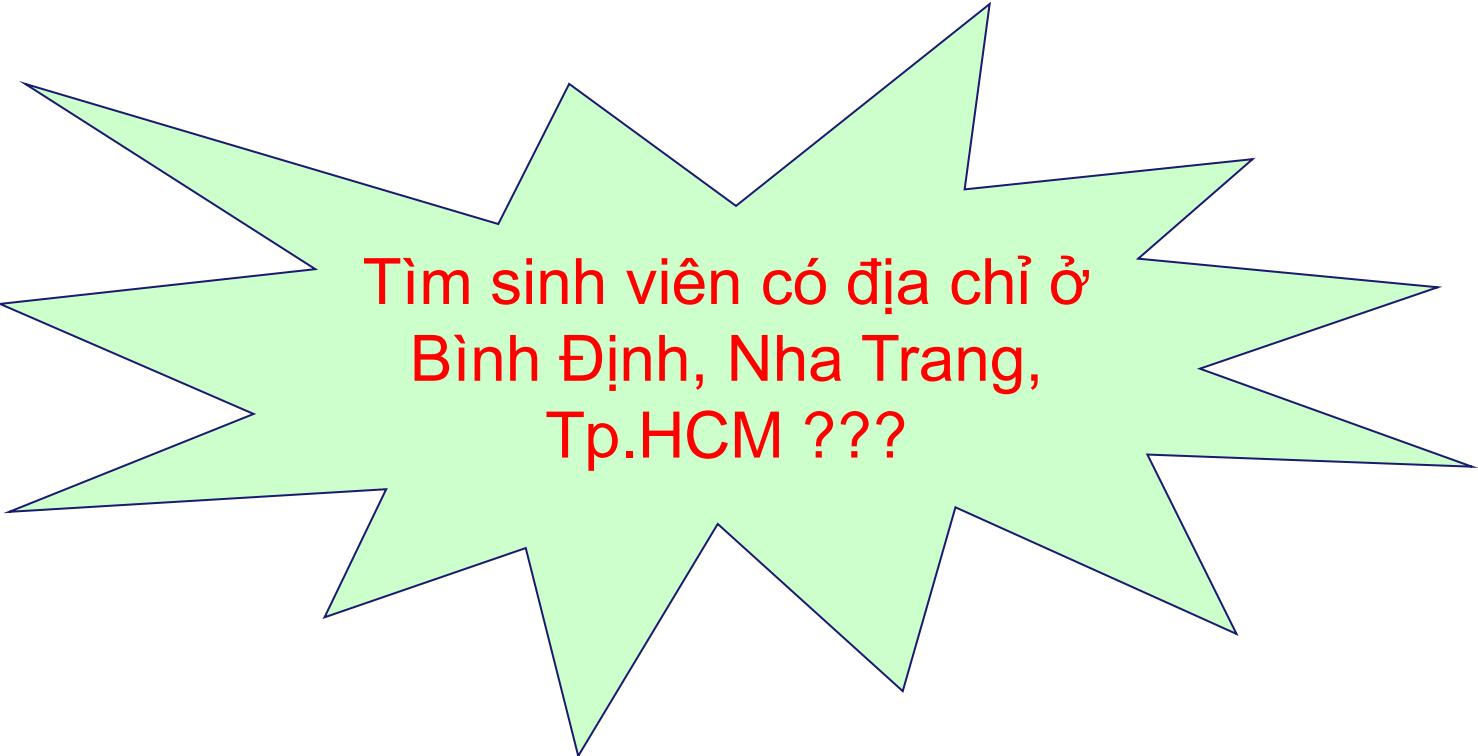


Phân loại thuộc tính

- ❖ Thuộc tính **đa trị**:
 - **Thuộc tính điện thoại của nhân viên** là thuộc tính đa trị:
 - 081234567
 - 09081234567,...
 - **Thuộc tính email của sinh viên** cũng là một thuộc tính đa trị:
 - xyz@yahoo.com,
 - xyz@gmail.com,...

Phân loại thuộc tính

- ❖ Thuộc tính là đối tượng phụ:



Tìm sinh viên có địa chỉ ở
Bình Định, Nha Trang,
Tp.HCM ???

Phân loại thuộc tính

- ❖ Thuộc tính là đối tượng phụ:
 - **Thuộc tính địa chỉ** của sinh viên, nhân viên là một đối tượng phụ, bao gồm các thông tin:
 - Số nhà
 - Đường
 - Phường/xã
 - Quận/huyện
 - Tỉnh thành
 - **Thuộc tính ngày sinh** của sinh viên, nhân viên cũng là một đối tượng phụ gồm các thông tin: ngày, tháng, năm

Phân loại thuộc tính

❖ Thuộc tính **tính toán**:

- Thuộc tính **thành tiền** trong hóa đơn là một thuộc tính tính toán, được tính bằng **tổng số lượng * đơn giá** của các mặt hàng
- Thuộc tính **điểm trung bình** của học sinh, sinh viên cũng là một thuộc tính tính toán

Phân loại thuộc tính

- ❖ **Quan hệ giữa các thực thể:** Phân loại quan hệ dựa trên bản số
 - Quan hệ “1-1”
 - Quan hệ “1-nhiều”
 - Quan hệ “nhiều-nhiều”

Sơ đồ Logic

- Bao gồm các bảng dữ liệu và mối quan hệ giữa chúng.
- Các ký hiệu:

Tên Table

Bảng dữ liệu



Liên kết

■ Các ký hiệu:



- ⇒ Một phần tử của bảng A xác định duy nhất một phần tử của bảng B
- ⇒ Ngược lại, một phần tử của bảng B có thể tương ứng với nhiều phần tử của bảng A
- ⇒ Thông thường bảng A sẽ chứa thuộc tính khóa của bảng B (là khóa ngoại của bảng A).

Các lưu ý

- Tên Table được viết bằng chữ IN HOA, không dấu, không có khoảng cách giữa các từ.

Ví dụ: NHANVIEN, KHACHHANG...

- Tên thuộc tính: viết hoa các ký tự đầu của mỗi từ, không dấu, không có khoảng cách giữa các từ.

Ví dụ: HoTen, NgaySinh, DiaChi...

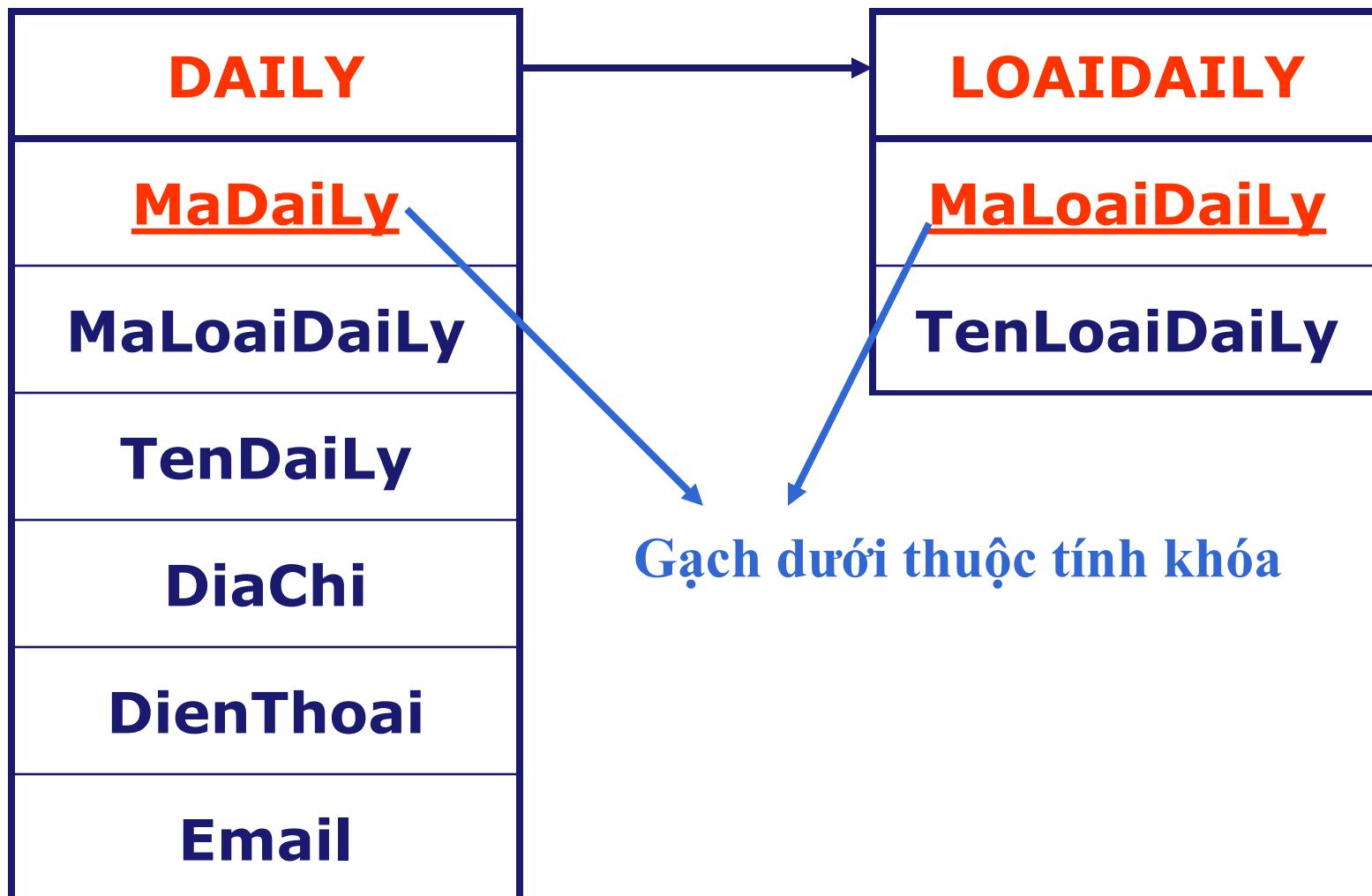
- Đặt tên table, tên thuộc tính của table súc tích, cô đọng:

Ví dụ: ~~HOSODAILY~~ -> DAILY

Các lưu ý (tt)

- **Thuộc tính khóa là thuộc tính có giá trị:**
 - ✓ Không trùng lắp;
 - ✓ Không thay đổi theo thời gian.
- **Xác định thuộc tính khóa:** từ danh sách các thuộc tính đã có, nếu không chọn được thuộc tính làm khóa thì sử dụng thêm **thuộc tính trừu tượng** để làm thuộc tính khóa cho table.
- **Thuộc tính trừu tượng** là thuộc tính không có trong thế giới thực, chỉ có trong phần mềm.
Ví dụ: MaDaiLy, MaLoaiDaiLy...

Các lưu ý (tt)





Các lưu ý (tt)

■ Xác định kiểu dữ liệu cho thuộc tính khóa:

- ✓ Cân nhắc chọn lựa giữa kiểu số và kiểu chuỗi;
- ✓ Sử dụng tối ưu chiều dài của mã đồng thời phải cân nhắc khả năng mở rộng.

■ Đặt mã có tính gợi ý:

Ví dụ: Hà Nội -> HAN, Hồ Chí Minh -> HCM

■ Không nên dùng lại một mã đã dùng cho dù mã đó đã bị xóa.

Các kiểu mã hóa:

- **Mã hóa liên tiếp:** 1,2,3...
- **Mã hóa theo lát:** dùng từng lát cho từng nhóm đối tượng, trong mỗi lát thường dùng kiểu mã hóa liên tiếp.
- **Mã hóa phân đoạn:** mã được phân thành nhiều đoạn, mỗi đoạn mang một ý nghĩa riêng.
- **Mã hóa phân cấp:** là mã hóa phân đoạn, mỗi đoạn trả đến một tập hợp các đối tượng và các đối tượng này được phân cấp theo thứ tự từ trái qua phải.
- **Mã hóa diễn nghĩa:** gán một tên ngắn gọn nhưng hiểu được cho từng đối tượng.

Bảng THAMSO

- Là một table đặc biệt, **không có thuộc tính khóa**.
- Chỉ có các thao tác **select và update**, không có **insert và delete** sau khi đã hoàn tất việc thiết kế.
- **Chức năng**: dùng để lưu các giá trị trong các qui định mà **không liên quan** đến các đối tượng dữ liệu khác.
- Các giá trị này dùng trong các biểu thức tính toán hoặc kiểm tra.
- Kiểu dữ liệu của các giá trị này thường là number.
(đối với kiểu boolean ta qui định true ⇔ 1 và false ⇔ 0)

- **Thiết kế dữ liệu dựa vào sơ đồ luồng dữ liệu của yêu cầu phần mềm đang xét.**
- **Các bước thực hiện:**
 - ❖ **Bước 1: Xét yêu cầu phần mềm thứ I**
 - Thiết kế dữ liệu với tính đúng đắn
 - Thiết kế dữ liệu với tính tiến hóa
 - ❖ **Bước 2: Xét yêu cầu phần mềm thứ II**
...
 - ❖ **Bước n: Xét yêu cầu phần mềm cuối cùng**

➤ **Thiết kế dữ liệu với tính đúng đắn:**

- **Biểu mẫu liên quan:** BMx
- **Sơ đồ luồng dữ liệu:** SĐx
- **Các thuộc tính mới:**
 - **Thiết kế dữ liệu:** bố trí các thuộc tính mới vào các bảng đã có, trong trường hợp không bố trí được thì phải tạo ra bảng mới để chứa các thuộc tính mới này.
 - **Các thuộc tính trừu tượng:**
 - **Sơ đồ Logic**

- **Thiết kế dữ liệu với tính tiến hóa:**
 - **Qui định liên quan:** QĐx
 - **Sơ đồ luồng dữ liệu về việc thay đổi qui định:** SĐx
 - **Các thuộc tính mới:**
 - **Thiết kế dữ liệu:** bố trí các thuộc tính mới vào các bảng đã có, trong trường hợp không bố trí được thì phải tạo ra bảng mới để chứa các thuộc tính mới này.
 - **Các thuộc tính trừu tượng:**
 - **Sơ đồ Logic**

Lưu ý khi thiết kế dữ liệu

Thiết kế dữ liệu với yêu cầu hiệu quả về mặt truy suất và lưu trữ:

- **Hiệu quả về mặt truy suất:**
 - + Thêm vào các **thuộc tính tính toán**.
 - + Lưu ý: giá trị này phải được tự động cập nhật khi có những thay đổi liên quan.
- **Hiệu quả về mặt lưu trữ:**
 - + Tách bảng có các giá trị cố định được lặp lại nhiều lần thành 2 bảng, một bảng chứa thông tin tổng quát và một bảng chứa thông tin chi tiết.
 - + Thêm đối tượng mới và sử dụng mã của đối tượng thay cho việc lưu trữ thông tin chi tiết về đối tượng.

Lưu ý khi thiết kế dữ liệu (tt)

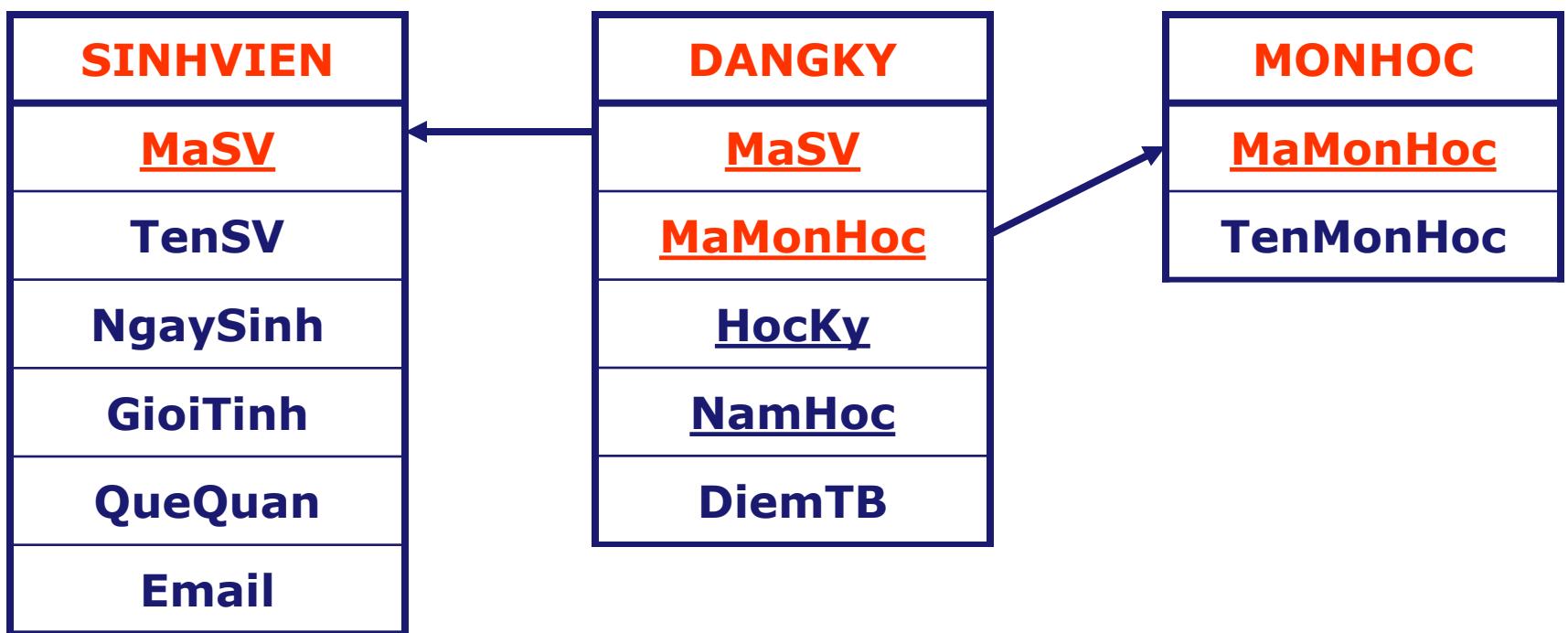
Chuyển quan hệ “nhiều – nhiều” thành 2 quan hệ “1 – nhiều”:

Ví dụ:



Chuyển thành:





Ví dụ

- Thiết kế dữ liệu cho đề tài Quản lý các đại lý.
- Các bước thực hiện:
 - ❖ Bước 1: Xét yêu cầu Tiếp nhận đại lý
 - Thiết kế dữ liệu với tính đúng đắn
 - Thiết kế dữ liệu với tính tiến hóa
 - ❖ Bước 2: Xét yêu cầu Lập Phiếu xuất hàng
 - ...
 - ❖ Bước n: Xét yêu cầu phần mềm cuối cùng

Ví dụ (tt)

➤ *Thiết kế dữ liệu với tính đúng đắn:*

- **Biểu mẫu liên quan:** BM1
- **Sơ đồ luồng dữ liệu:** SĐ1
- **Các thuộc tính mới:** TenDaiLy, LoaiDaiLy, DienThoai, DiaChi, Quan, NgayTiepNhan.
- **Thiết kế dữ liệu:** table DAILY
- **Các thuộc tính trừu tượng:** MaDaiLy.
- **Sơ đồ Logic**

DAILY

Ví dụ (tt)

DAILY

MaDaiLy

TenDaiLy

LoaiDaiLy

DienThoai

DiaChi

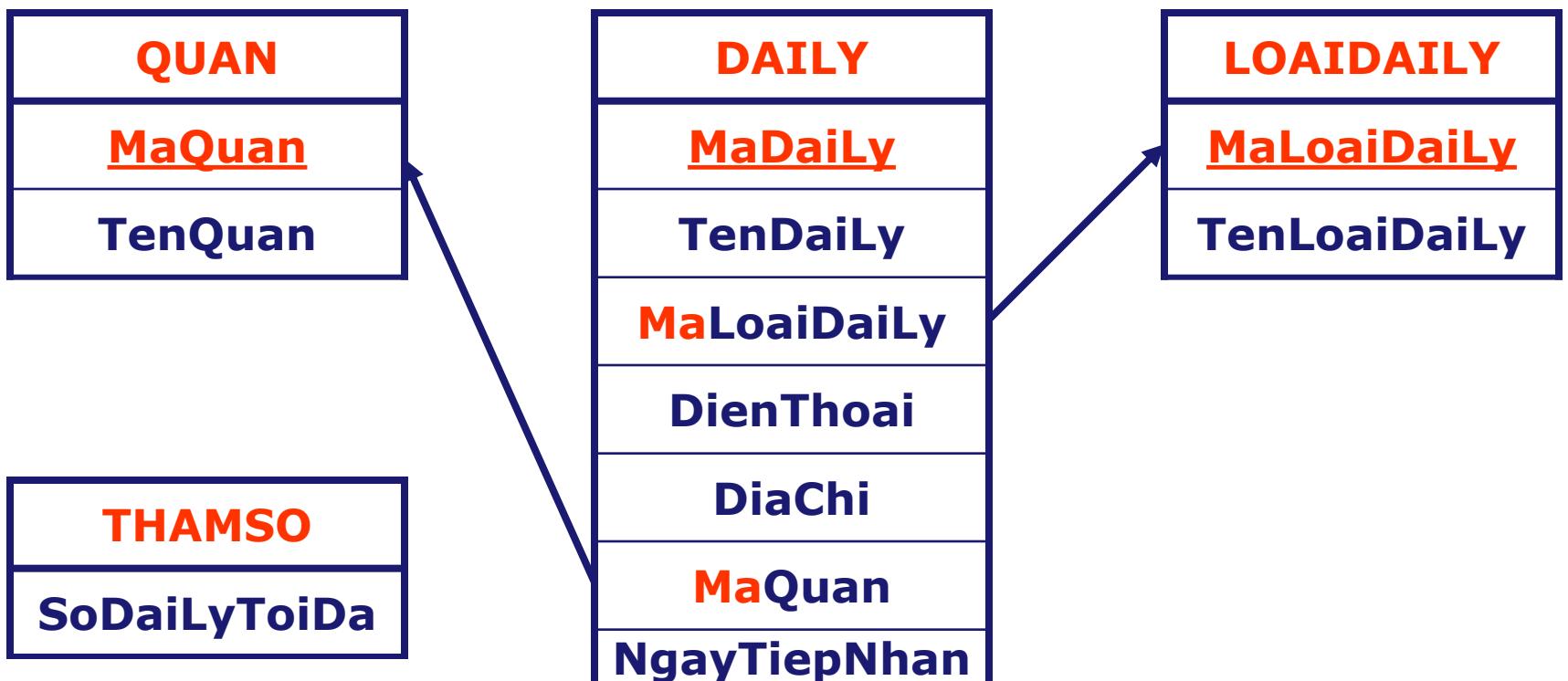
Quan

NgayTiepNhan

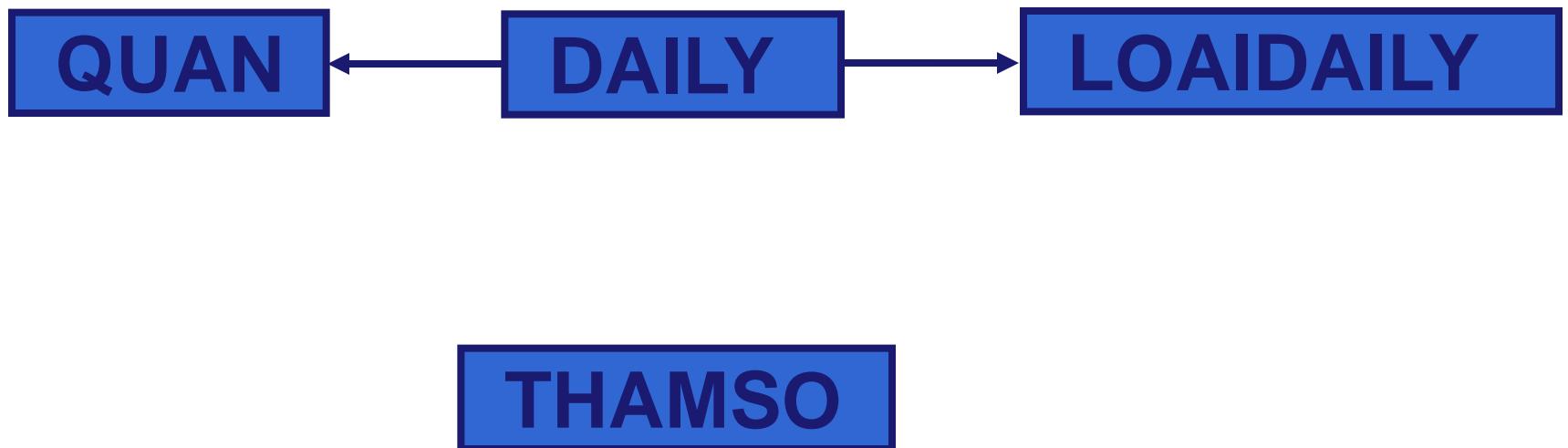
Ví dụ (tt)

- **Thiết kế dữ liệu với tính tiến hóa:**
 - **Qui định liên quan:** QĐ1
 - **Sơ đồ luồng dữ liệu về việc thay đổi qui định:** SĐ6
 - **Các thuộc tính mới:** TenLoaiDaiLy, TenQuan, SoDaiLyToiDa.
 - **Thiết kế dữ liệu:** table DAILY, table LOAIDAILY, table QUAN, table THAMSO
 - **Các thuộc tính trừu tượng:** MaLoaiDaiLy, MaQuan.
 - **Sơ đồ Logic**

Ví dụ (tt)



Ví dụ (tt)



Ví dụ 1

- Xét phần mềm quản lý bán hàng với nghiệp vụ lập phiếu thu tiền của khách hàng

Biểu mẫu	Phiếu thu tiền
Họ tên khách hàng:.....	CMND:.....
Địa chỉ:.....	Ngày thu:.....
Số tiền:.....	Lý do thu:.....

Qui định: Tiền thu tối thiểu là 100.000đ, Có 3 lý do thu là A, B, C

Hãy lập sơ đồ logic dữ liệu



Ví dụ 1 (tt)

- ❖ Cách 0: dùng 1 bảng
- ❖ Cách 1: dùng 2 bảng (PHIEU_THU, LYDO_THU)
- ❖ Cách 2: dùng 2 bảng (PHIEU_THU, KHACH_HANG)
- ❖ Cách 3: dùng 4 bảng

Ví dụ 1 (tt)

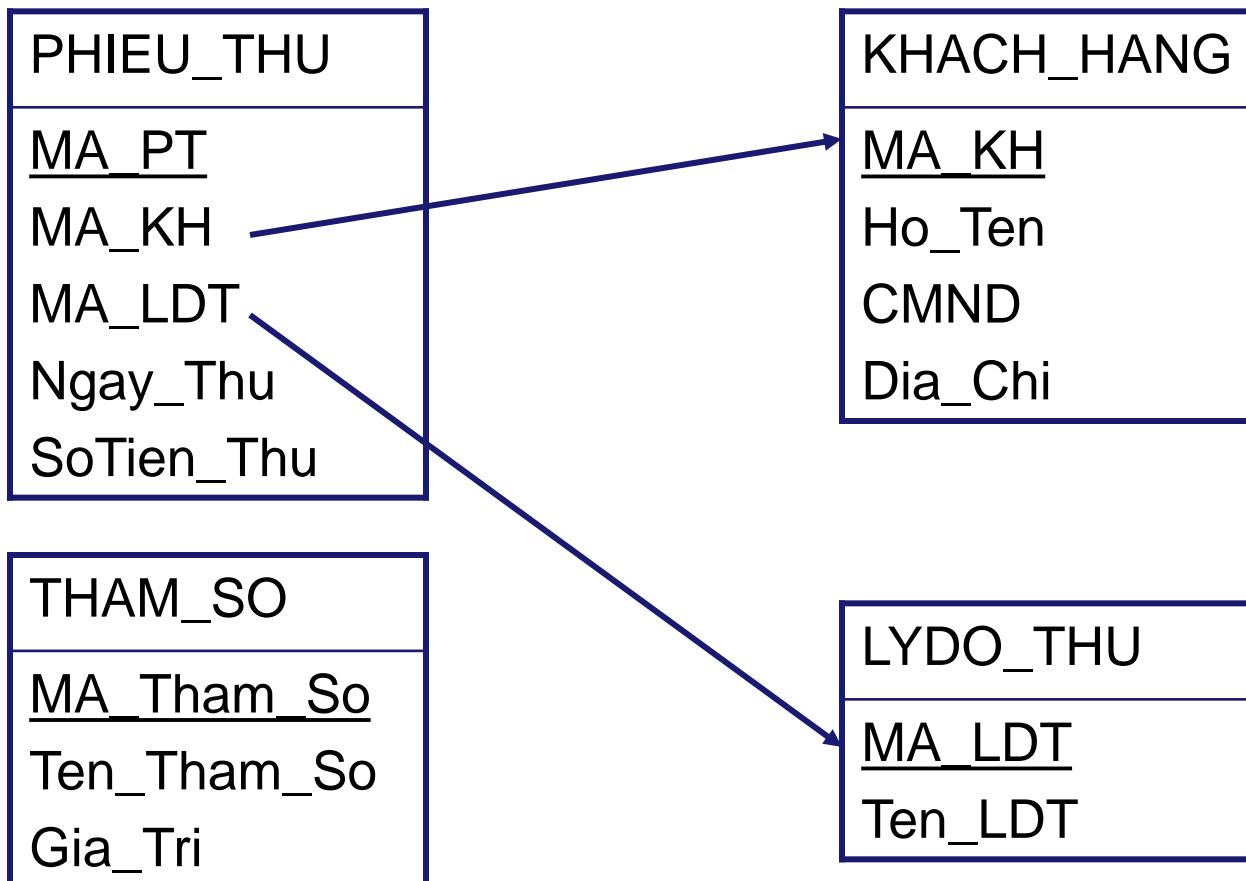


Table THAM_SO dùng để lưu những qui định

Ví dụ 2

- Xét phần mềm quản lý nhân sự với nghiệp vụ tiếp nhận nhân viên

Biểu mẫu	Hồ sơ nhân viên
Họ tên :	Giới tính:
Ngày sinh:	Địa chỉ:
Điện thoại:	Trình độ:
Đơn vị:	

Qui định: Công ty có 4 đơn vị X, Y, Z, T. Tuổi nhân viên nam từ 20 đến 60. Tuổi nhân viên nữ từ 18 đến 55. Có 4 trình độ: Sau đại học, đại học, cao đẳng, trung cấp.

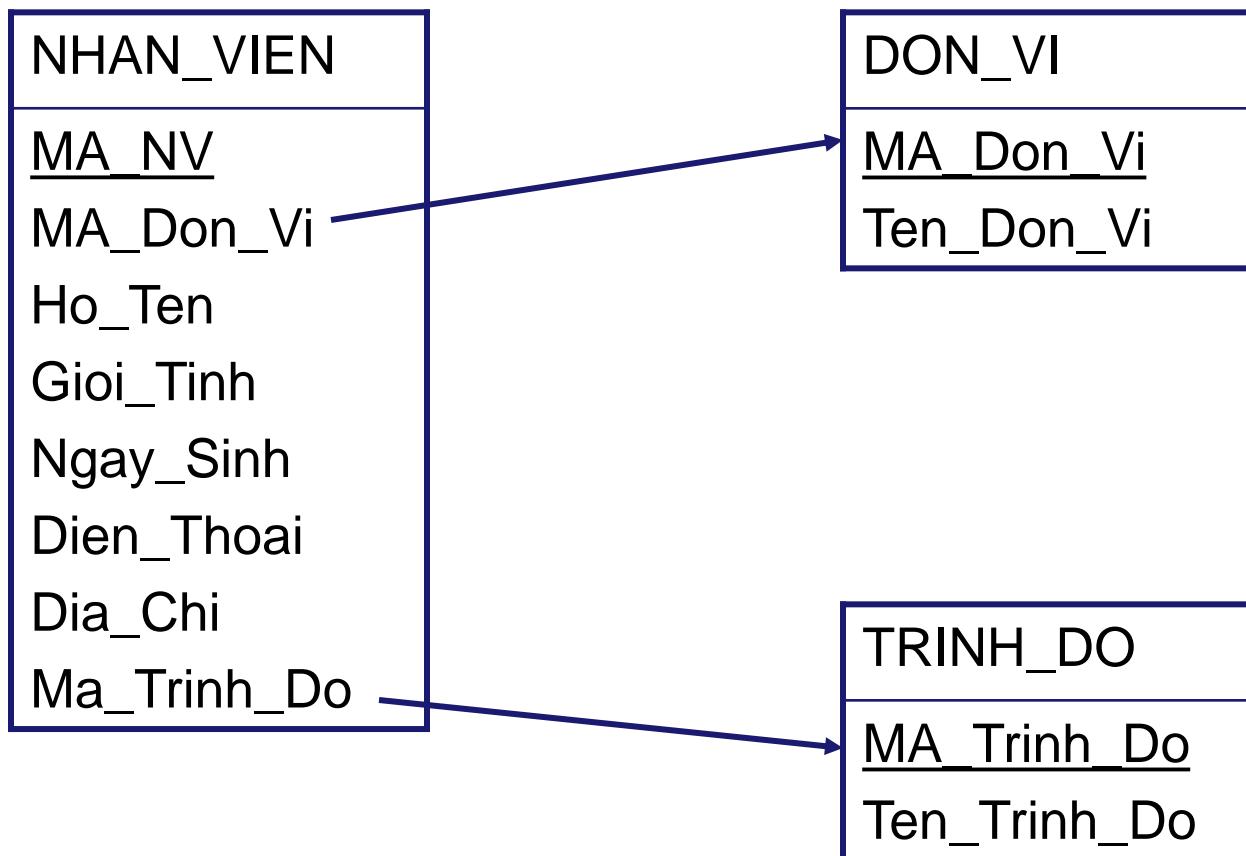
Ví dụ 2 (tt)

Hãy lập sơ đồ logic dữ liệu cho yêu cầu nghiệp vụ trên trong các trường hợp sau:

- a. Không yêu cầu theo dõi quá trình chuyển nhân viên từ đơn vị này sang đơn vị khác
- b. Có yêu cầu theo dõi quá trình chuyển nhân viên
- c. Bổ sung qui định: Đơn vị X chỉ nhận các nhân viên có trình độ đại học và sau đại học

Ví dụ 2 (tt)

a.



Ví dụ 2 (tt)

❖ Table THAM_SO có thể thiết kế theo nhiều cách

- Cách 1

Khóa	ThamSố#1	ThamSố#2	...	ThamSố#N
...

Mỗi tham số tương ứng với một cột trong bảng tham số
Bảng tham số thường chỉ gồm 1 dòng (chứa giá trị các tham số hiện hành)

- ?
- Cần bổ sung tham số mới?
- Cần vô hiệu hóa tác dụng của một tham số?

Ví dụ 2 (tt)

- Cách 2

Được lưu
dạng chuỗi

MãThamSố	TênThamSố	Kiểu	GiáTrị	TìnhTrạng
...

Mỗi tham số tương ứng với một dòng trong bảng tham số
Giá trị hiện tại của tham số được lưu bằng dạng chuỗi
Mỗi tham số cần lưu trữ kiểu giá trị để phần mềm “hiểu”
đúng nội dung giá trị hiện tại của tham số

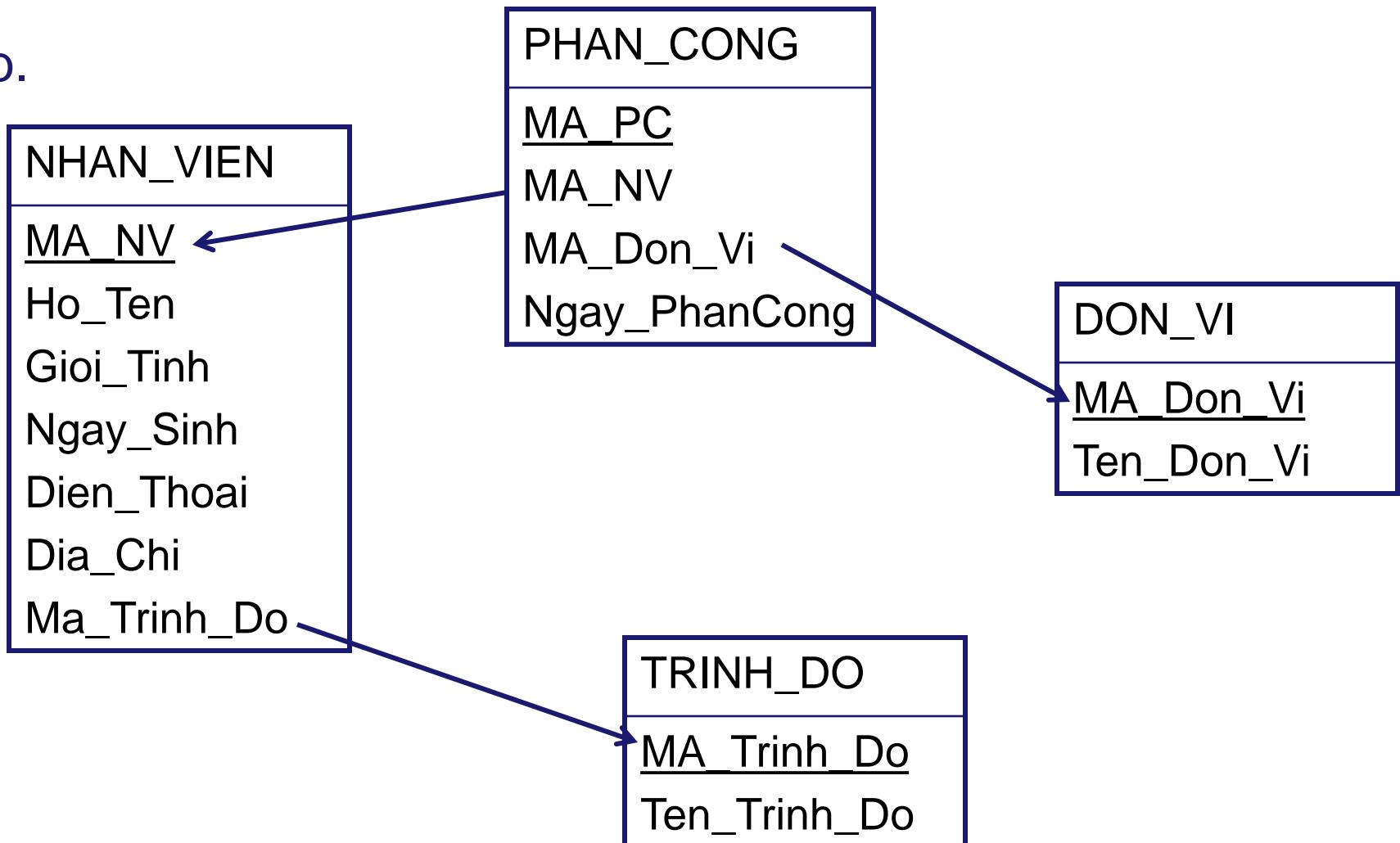


Cần bổ sung tham số mới?

Cần vô hiệu hóa tác dụng của một tham số?

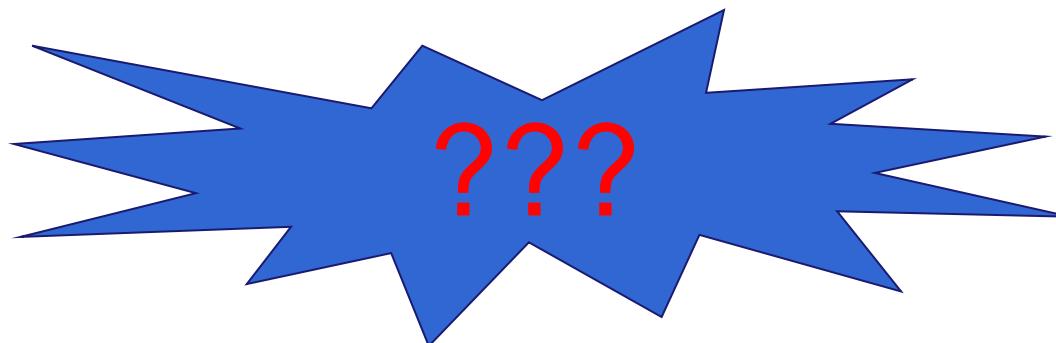
Ví dụ 2 (tt)

b.



Ví dụ 2 (tt)

- c. Bổ sung qui định: Đơn vị X chỉ nhận các nhân viên có trình độ đại học và sau đại học



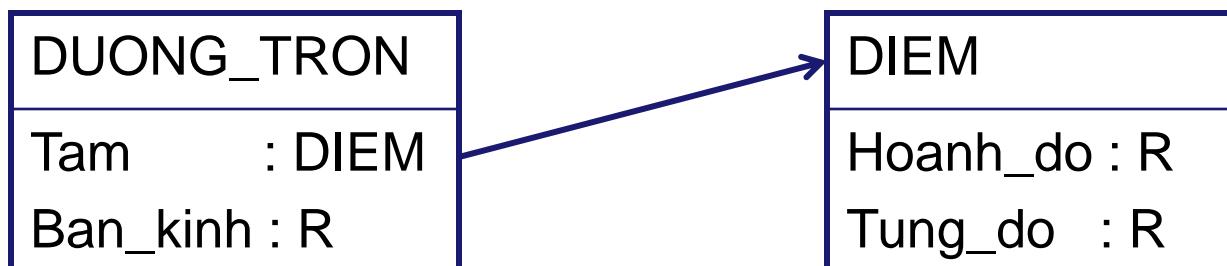


Thiết kế dữ liệu

- ❖ Khi thiết kế dữ liệu phải **chú ý** đến các vấn đề sau:
 - Không gian
 - Thời gian
 - Khối lượng dữ liệu phát sinh rất nhanh theo thời gian
 - Đáp ứng yêu cầu truy xuất nhanh
 -

Sơ đồ kiểu dữ liệu

- ❖ Cho phép mô tả cách thức tổ chức lưu trữ dữ liệu trong bộ nhớ chính
- ❖ Ký hiệu: Tương tự như sơ đồ logic
- ❖ Ví dụ: Xét chức năng tính diện tích đường tròn





Một phương pháp thiết kế

❖ **Bước 01:** Xét yêu cầu phần mềm thứ nhất (Thông thường ta xét ycpm LưuTrữ)

- Thiết kế dữ liệu với tính đúng đắn

- Biểu mẫu liên quan: **BMx**
- Sơ đồ luồng dữ liệu:
- Các thuộc tính mới
- Thiết kế dữ liệu
- Sơ đồ logic



Một phương pháp thiết kế

❖ **Bước 01:** Xét yêu cầu phần mềm thứ nhất (Thông thường ta xét ycpm LưuTrữ)

- Thiết kế dữ liệu với tính tiến hóa

- Qui định liên quan: **QĐx**
- Sơ đồ luồng dữ liệu về việc thay đổi qui định
- Các thuộc tính mới
- Thiết kế dữ liệu
- Sơ đồ logic



Một phương pháp thiết kế

- ❖ **Bước 02:** Xét yêu cầu phần mềm thứ hai.
- ❖ **Bước 03:** Xét yêu cầu phần mềm thứ ba.
- ❖
- ❖ **Bước n:** Xét yêu cầu phần mềm cuối cùng.



Một phương pháp thiết kế

❖ Lưu ý:

- Khi thiết kế dữ liệu ở bước đúng đắn và tiến hóa ta lần lượt bố trí các thuộc tính mới vào các bảng đã có sẵn (nếu được). Trong trường hợp không bố trí được thì khi đó ta buộc phải tạo ra bảng mới để chứa thuộc tính mới này

Bài tập

- Xét phần mềm quản lý học sinh với chức năng ghi nhận bảng điểm danh

Bảng điểm danh

Lớp:

Tháng:

STT	Họ tên	Vắng có phép	Vắng không phép

Hãy lập: sơ đồ luồng dữ liệu, sơ đồ logic dữ liệu

Bài tập

2. Xét phần mềm quản lý bán hàng với chức năng lập hóa đơn bán hàng. Hãy lập sơ đồ logic dữ liệu

Hóa đơn bán hàng

Họ tên khách hàng: CMND:

Địa chỉ: Ngày lập hóa đơn: 

STT	Mặt hàng	Số lượng	Đơn giá	Thành tiền

Qui định: Số lượng bán phải nhỏ hơn số lượng tồn của mặt hàng tương ứng ít nhất 100.

Bài tập

3. Xét phần mềm quản lý tiết kiệm với chức năng mở sổ tiết kiệm. Hãy lập sơ đồ logic dữ liệu

Sổ tiết kiệm

Họ tên khách:.....

CMND:.....

Địa chỉ:.....

Loại tiết kiệm:.....

Ngày mở sổ:.....

Số tiền gửi:.....

Loại ngoại tệ:.....

Qui định: Có 5 loại tiết kiệm: không kỳ hạn, kỳ hạn 3 tháng, 6 tháng, 9 tháng, 12 tháng. Số tiền gửi ít nhất là 200.000đ

Có 2 loại ngoại tệ: tiền VN và đồng USD.

Mỗi loại kỳ hạn có một lãi suất riêng ứng với từng ngoại tệ.

Thiết kế giao diện



Tầm quan trọng của giao diện

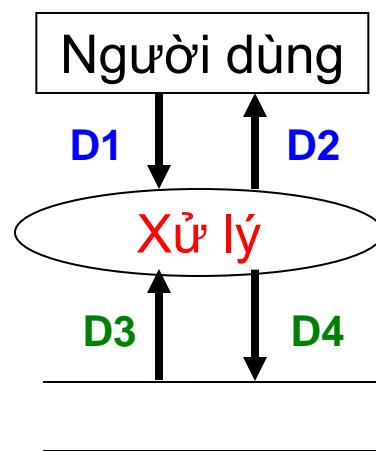


Các thành phần của giao diện



Mục tiêu

- ❖ Mục tiêu: mô tả chi tiết cách thức **giao tiếp** giữa người dùng và phần mềm trong quá trình thực hiện các nghiệp vụ liên quan



- Nội dung trình bày của D1, D2
- Hình thức trình bày của D1, D2
- Biến cố phải xử lý



THIẾT KẾ GD – Thiết bị

❖ Thiết bị vào và ra

- Màn hình
- Bàn phím
- Mouse, bút từ, ...
- Màn hình cảm biến
- Mic/Speaker
- Smart cards,...
- Máy in

Cả thiết bị lẫn phương thức đều đang tiến hóa

- nhận dạng tiếng nói, chữ viết...

THIẾT KẾ GD - Loại giao diện

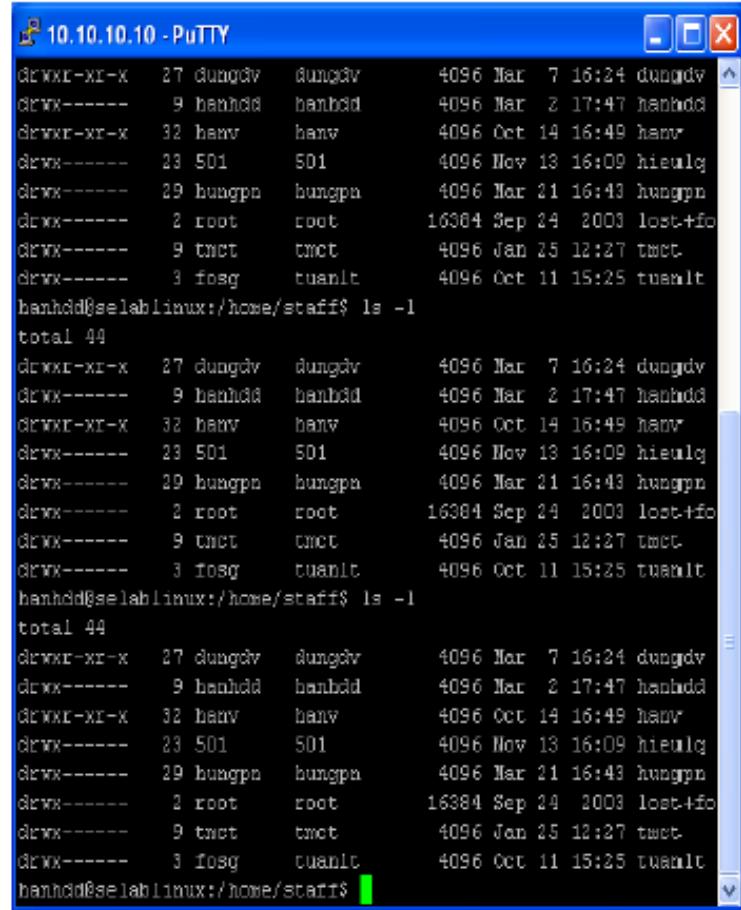
Giao diện dòng lệnh

- ❑ Là phương thức tương tác đầu tiên
- ❑ Nhập lệnh/dữ liệu từ bàn phím
- ❑ Dễ cài đặt so với GUI
 - thực hiện thông qua hàm chuẩn của ngôn ngữ
 - không tồn tại nguyên hệ thống
- ❑ Có khả năng tổ hợp lệnh để tạo các lệnh phức tạp
 - phối hợp các filter, tạo các lô xử lý (batch)
 - có thể lập trình bằng (Unix) shell
 - *có thể tự động hóa*

```
drwxr-xr-x  27 dungdv  dungdv   4096 Mar  7 16:24 dungdv
drwx-----  9 hanhdd  hanhdd   4096 Mar  2 17:47 hanhdd
drwxr-xr-x  32 hanv   hanv    4096 Oct 14 16:49 hanv
drwx----- 23 501   501    4096 Nov 13 16:09 hieulq
drwx----- 29 hungpn hungpn   4096 Mar 21 16:43 hungpn
drwx-----  2 root   root    16384 Sep 24  2003 lost+fo
drwx-----  9 tmct   tmct    4096 Jan 25 12:27 tmct
drwx-----  3 fosg   tuanlt  4096 Oct 11 15:25 tuanlt
hanhdd@selablinux:/home/staff$ ls -l
total 44
drwxr-xr-x  27 dungdv  dungdv   4096 Mar  7 16:24 dungdv
drwx-----  9 hanhdd  hanhdd   4096 Mar  2 17:47 hanhdd
drwxr-xr-x  32 hanv   hanv    4096 Oct 14 16:49 hanv
drwx----- 23 501   501    4096 Nov 13 16:09 hieulq
drwx----- 29 hungpn hungpn   4096 Mar 21 16:43 hungpn
drwx-----  2 root   root    16384 Sep 24  2003 lost+fo
drwx-----  9 tmct   tmct    4096 Jan 25 12:27 tmct
drwx-----  3 fosg   tuanlt  4096 Oct 11 15:25 tuanlt
hanhdd@selablinux:/home/staff$ ls -l
total 44
drwxr-xr-x  27 dungdv  dungdv   4096 Mar  7 16:24 dungdv
drwx-----  9 hanhdd  hanhdd   4096 Mar  2 17:47 hanhdd
drwxr-xr-x  32 hanv   hanv    4096 Oct 14 16:49 hanv
drwx----- 23 501   501    4096 Nov 13 16:09 hieulq
drwx----- 29 hungpn hungpn   4096 Mar 21 16:43 hungpn
drwx-----  2 root   root    16384 Sep 24  2003 lost+fo
drwx-----  9 tmct   tmct    4096 Jan 25 12:27 tmct
drwx-----  3 fosg   tuanlt  4096 Oct 11 15:25 tuanlt
hanhdd@selablinux:/home/staff$ ls -l
```

Giao diện dòng lệnh (2)

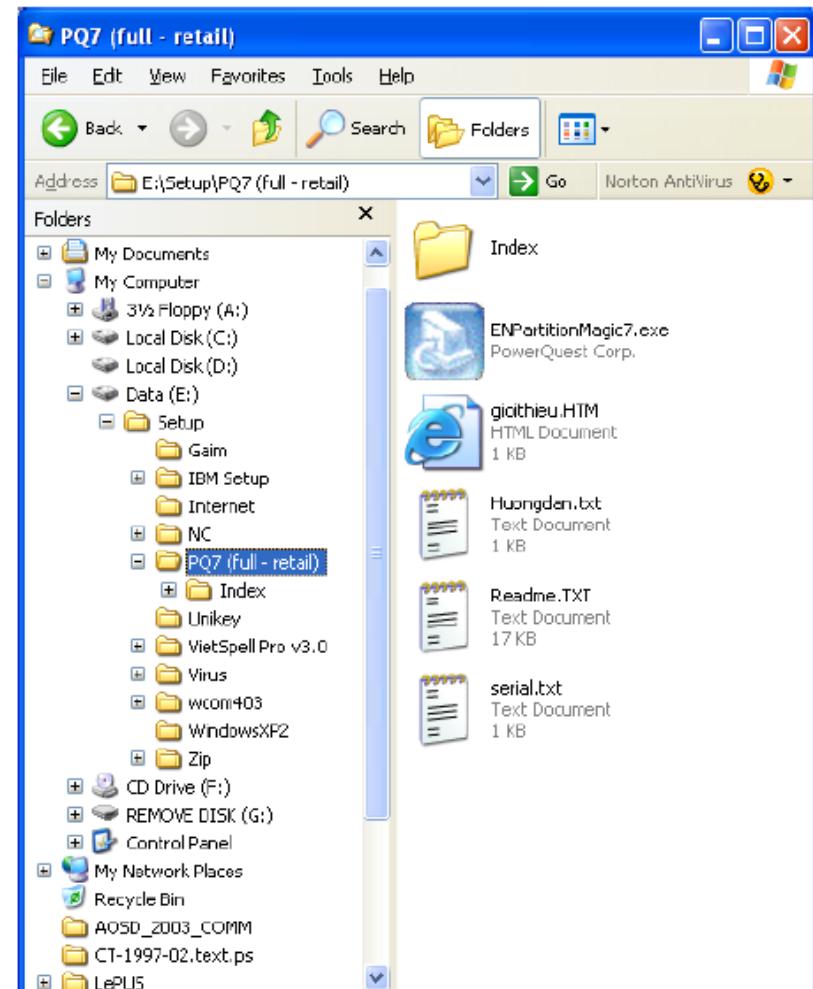
- ❑ Thao tác thực hiện tuần tự
 - khó sửa lỗi thao tác trước đó
- ❑ Không phù hợp với người dùng ít kinh nghiệm
 - khó học, khó nhớ
 - dễ nhầm
 - đòi hỏi kỹ năng sử dụng bàn phím



```
10.10.10.10 - PuTTY
drwxr-xr-x 27 dungdvn dungdvn 4096 Mar  7 16:24 dungdvn
drwx----- 9 banhddd banhddd 4096 Mar  2 17:47 banhddd
drwxr-xr-x 32 hanv hanv 4096 Oct 14 16:49 hanv
drwx----- 23 501 501 4096 Nov 13 16:09 hieuqlq
drwx----- 29 hungpn hungpn 4096 Mar 21 16:43 hungpn
drwx----- 2 root root 16384 Sep 24 2003 lost+fo
drwx----- 9 tmct tmct 4096 Jan 25 12:27 tmct
drwx----- 3 fosg tuanlt 4096 Oct 11 15:25 tuanlt
banhddd@selahlinux:/home/staff$ ls -l
total 44
drwxr-xr-x 27 dungdvn dungdvn 4096 Mar  7 16:24 dungdvn
drwx----- 9 banhddd banhddd 4096 Mar  2 17:47 banhddd
drwxr-xr-x 32 hanv hanv 4096 Oct 14 16:49 hanv
drwx----- 23 501 501 4096 Nov 13 16:09 hieuqlq
drwx----- 29 hungpn hungpn 4096 Mar 21 16:43 hungpn
drwx----- 2 root root 16384 Sep 24 2003 lost+fo
drwx----- 9 tmct tmct 4096 Jan 25 12:27 tmct
drwx----- 3 fosg tuanlt 4096 Oct 11 15:25 tuanlt
banhddd@selahlinux:/home/staff$ ls -l
total 44
drwxr-xr-x 27 dungdvn dungdvn 4096 Mar  7 16:24 dungdvn
drwx----- 9 banhddd banhddd 4096 Mar  2 17:47 banhddd
drwxr-xr-x 32 hanv hanv 4096 Oct 14 16:49 hanv
drwx----- 23 501 501 4096 Nov 13 16:09 hieuqlq
drwx----- 29 hungpn hungpn 4096 Mar 21 16:43 hungpn
drwx----- 2 root root 16384 Sep 24 2003 lost+fo
drwx----- 9 tmct tmct 4096 Jan 25 12:27 tmct
drwx----- 3 fosg tuanlt 4096 Oct 11 15:25 tuanlt
banhddd@selahlinux:/home/staff$
```

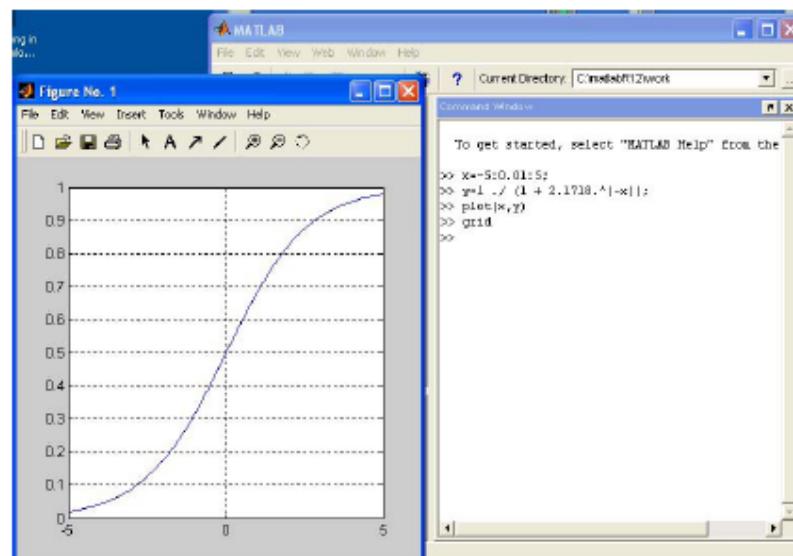
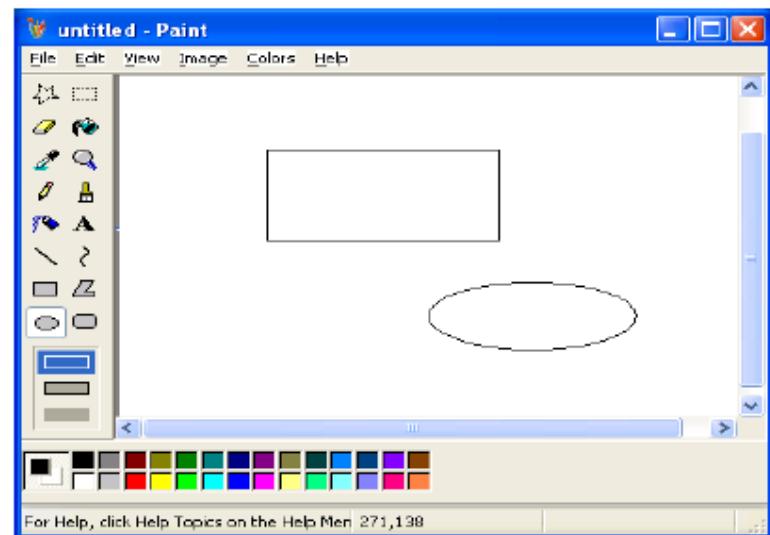
Giao diện đồ họa (GUI)

- ❑ Là giao diện thông dụng trên PC, Apple, Unix WS
- ❑ Dễ học, dễ sử dụng, thuận tiện với người ít kinh nghiệm
- ❑ Có nhiều cửa sổ, có thể tương tác song song trên nhiều cửa sổ mà không bị mất thông tin
- ❑ Có thể hiển thị, tương tác dữ liệu trên nhiều vị trí trong cửa sổ



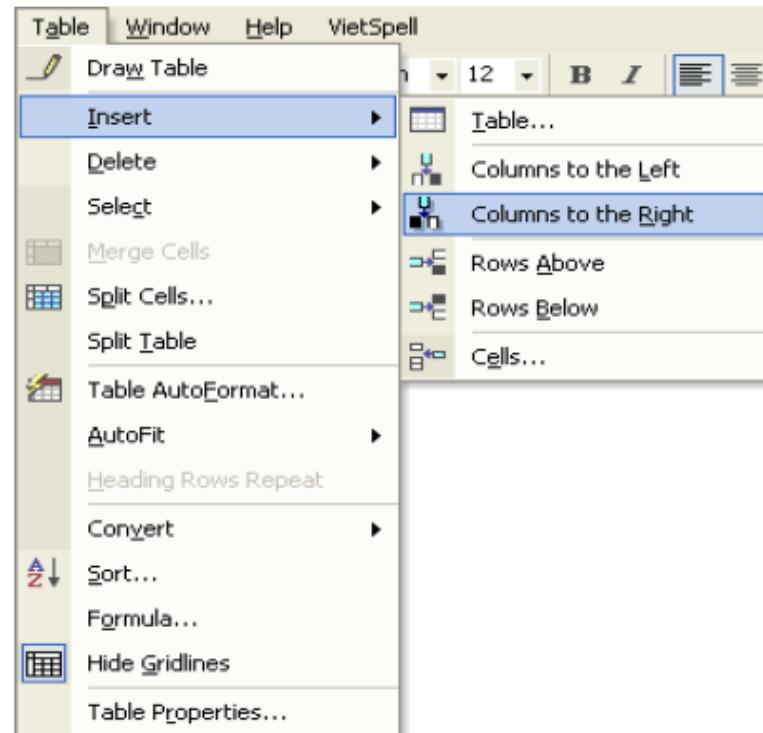
TKGD– Hình thức tương tác

- ❑ **Tương tác trực tiếp với thông tin**
 - ví dụ: soạn thảo; nhập dữ liệu vào các form...
 - dễ học, dễ sử dụng
 - nhận được tức thời kết quả thao tác
 - cài đặt phức tạp, tốn tài nguyên phần cứng
- ❑ **Tương tác gián tiếp**
 - ví dụ: chọn lệnh từ menu, giao diện dòng lệnh
 - kém trực quan
 - thuận tiện khi lặp lại thao tác phức tạp



TKGD– Hình thức tương tác

- ❑ Sử dụng thực đơn (menu)
 - Không cần nhớ lệnh
 - Tối thiểu hóa dùng bàn phím
 - Tránh các lỗi như sai lệnh, sai tham số
 - Dễ dàng tạo các trợ giúp theo ngữ cảnh
- ❑ Đối thoại (Dialog)
 - Dùng khi cần người dùng đưa ra lựa chọn quyết định



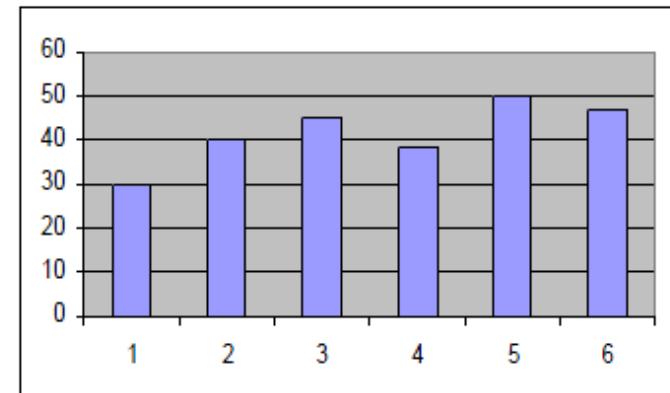
Phương pháp hiện thị thông tin

- Hiển thị bằng văn bản (text)
 - chính xác
 - dễ cài đặt
- Hiển thị bằng đồ họa (graphic)
 - trực quan
 - dễ dàng nhìn nhận các mối quan hệ

Tháng 1: 30

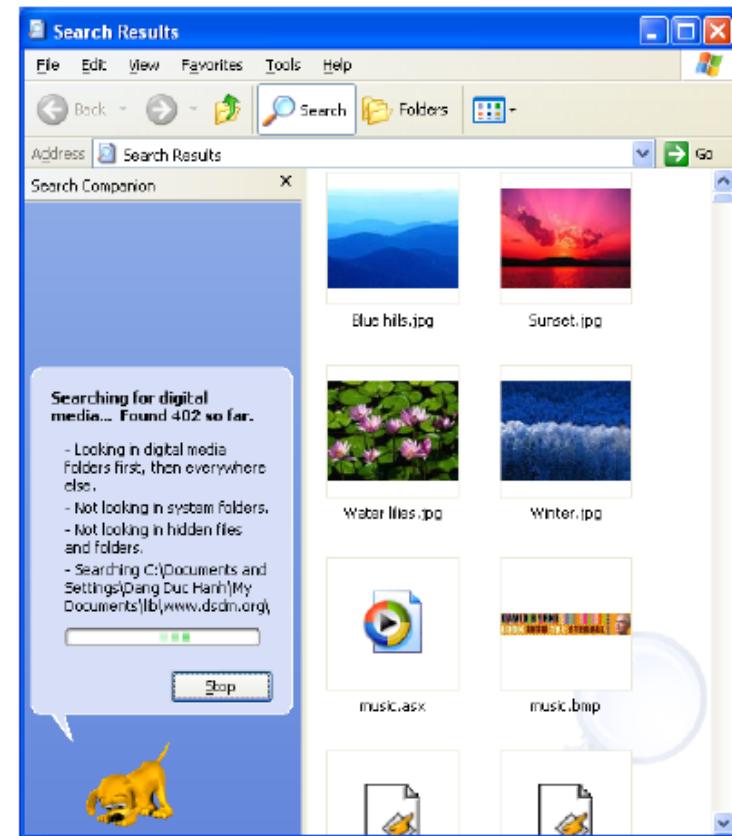
Tháng 2: 40

....



Thời gian phản hồi

- ❑ Thời gian trung bình
 - thời gian trung bình phản hồi với thao tác
 - người dùng không thể đợi quá lâu (< 3s)
 - cần chứng tỏ hệ thống đang hoạt động
- ❑ Độ biến thiên thời gian phản hồi
 - Gây cảm giác hệ thống gấp lõi



Thông báo

- ❑ Phản hồi của hệ thống đối với thao tác
- ❑ Cần có nghĩa, dễ hiểu, đưa ra các thông tin hữu ích với người dùng
 - tránh đưa ra các số hiệu
 - định dạng thông báo phải nhất quán
- ❑ Thông báo lỗi
 - chính xác
 - có tính xây dựng (nguyên nhân, cách khắc phục,...)

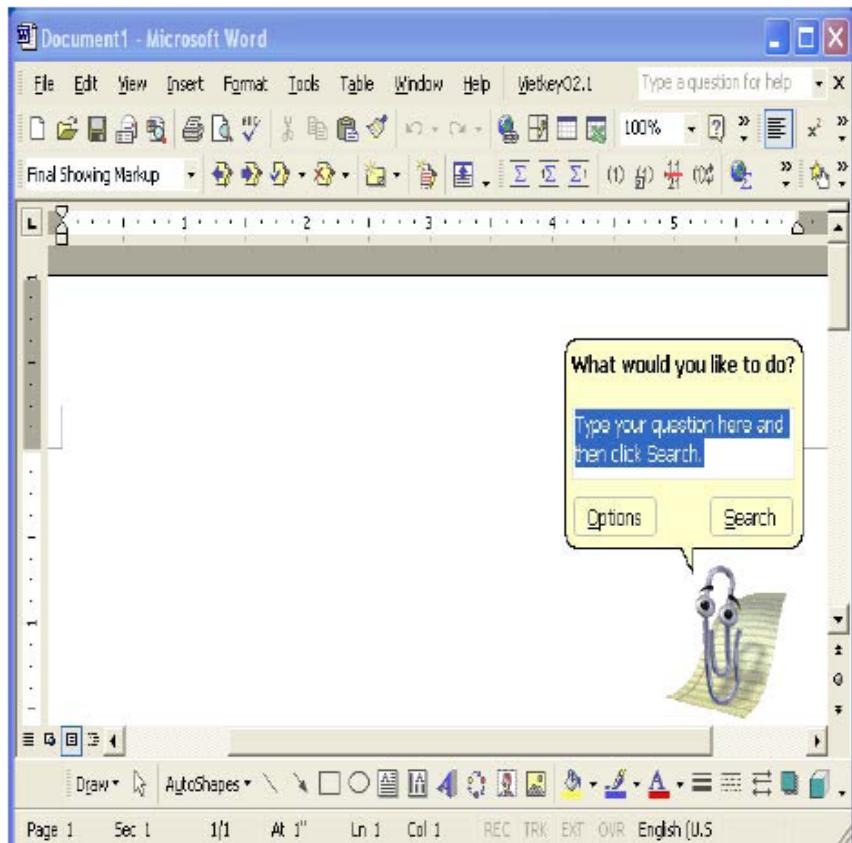


Thông báo (2)

- Số lượng thông báo:
 - Đưa ra càng nhiều càng tốt = càng thân thiện
 - Đưa ra một lượng tối thiểu = im lặng là vàng
- Thời điểm và thứ tự đưa ra thông báo
- Yêu cầu phản hồi đối với thông báo

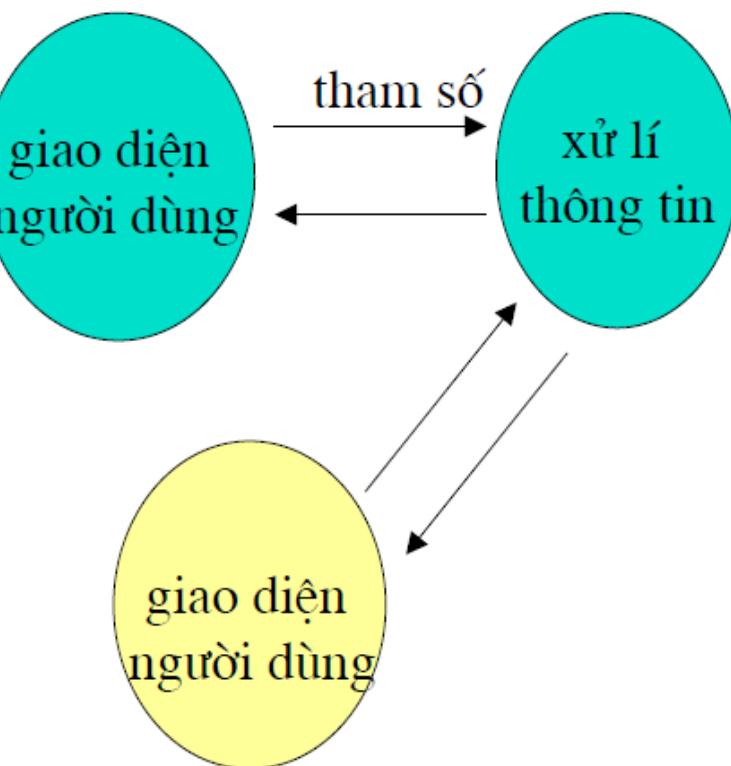
Tiện ích (trợ giúp)

- ❑ Cần có các tiện ích trợ giúp người sử dụng
- ❑ Tiện ích tích hợp: trợ giúp trực tuyến, theo ngữ cảnh
 - chú giải thao tác, giao diện
- ❑ Các tài liệu trực tuyến
 - tra cứu chức năng hệ thống
- ❑ Các macro: tự động hóa thao tác
 - ví dụ: MS Word macro



TKGD - Tính kỹ nghệ

- Giao diện là phần tử dễ thay đổi
 - thay đổi quy trình, phương thức thao tác
 - thay đổi môi trường (phần cứng, hệ điều hành)
 - nâng cấp (đẹp hơn, dễ sử dụng hơn...)
- Giao diện phải dễ sửa đổi
- Giao diện phải có tính khả chuyển



→ *Giao diện nên độc lập với xử lý*



Các yêu cầu chất lượng

❖ Tính tiện dụng

- Dễ học
- Thứ tự nhập trực quan, dễ sử dụng

❖ Tính hiệu quả

- Quy trình nhập **đơn giản nhất, tự nhiên nhất**
- **Tận dụng** những bước xử lý trên màn hình
- Tránh thêm những công đoạn (thao tác) thừa (không cần thiết)
- Tốc độ thực hiện nhanh

❖ Tính tiến hóa

- Các tùy chọn về nội dung
- Hình thức trình bày
- Biến cố phải xử lý

Tính dễ dùng

- **Tính thân thiện**
 - Chức năng dễ hiểu
 - Phát hiện ngay sai sót
 - Dự trù sẵn phản ứng khi NSD phạm sai sót
 - Đủ uyển chuyển
 - Trình tự khai thác có tự nhiên hay không?
 - Câu hỏi đ/v NSD:
 - Tôi đang ở đâu?
 - Tôi đã đến đây như thế nào?
 - Tôi có thể làm gì tại đây?
 - Sau giao diện này, tôi có thể đi đến đâu?
- **Tính “ergonomic”:**
 - Màu sắc
 - Vị trí
 - Cách giao tiếp hệ thống



Tính nhất quán: “chuẩn”

- Chuẩn về dữ liệu: tên gọi, chiều dài, kiểu, cách trình bày
- Chuẩn về mã
 - Độc lập với việc Tin học hóa, có ý nghĩa về mặt nghiệp vụ
=> xem như dữ liệu bình thường
 - Do yêu cầu của việc Tin học hóa => đừng làm rối cho NSD
- Chuẩn về cấu trúc hệ thống
 - Cách trình bày thực đơn
 - Cách trình bày các thành phần trên màn hình
 - Cách xử lý trên màn hình
- Chuẩn về sưu liệu:
 - Hướng dẫn trực tuyến
 - Hướng dẫn trong tài liệu



Các yêu cầu chất lượng

- ❖ Phải quan sát **thói quen** của người sử dụng
- ❖ Thói quen phải tôn trọng => **bắt buộc phải tôn trọng**
 - Quen với **phím enter** khi nhập liệu
- ❖ Thói quen **chưa hợp lý** hoặc **mâu thuẫn** với kỹ thuật=> Phải trao đổi, thuyết phục và thống nhất với người sử dụng



Kỹ thuật nâng cao chất lượng

1. Bổ sung nội dung

- Hướng dẫn sử dụng
- Thuộc tính tính toán
- Thông tin chi tiết liên quan

2. Tăng tốc thao tác

- Dùng giá trị định sẵn
- Chuyển ô nhập liệu thành cột nhập liệu
- Sử dụng giá trị thay thế
- ...



Kỹ thuật nâng cao chất lượng

3. Xử lý lỗi

- Thông báo lỗi chính xác, cơ hội sửa lỗi
- Hạn chế lỗi
- Cấm tuyệt đối lỗi

4. Bổ sung, thay thế hình thức trình bày

- Dùng biểu tượng
- Dùng thực đơn động
- Dùng cây
- Dùng sơ đồ
- Thao tác trực tiếp



Hệ thống giao diện

- Nội dung và hình thức trình bày các màn hình giao tiếp của phần mềm.
- Hệ thống các thao tác mà người dùng thực hiện trên màn hình giao tiếp và xử lý tương ứng của phần mềm.



Kết quả đạt được

■ Danh sách các màn hình

■ Sơ đồ màn hình

■ Mô tả từng màn hình:

- Mô tả các đối tượng trên màn hình
- Danh sách biến cố và xử lý tương ứng trên màn hình

Danh sách các màn hình

STT	Màn hình	Loại màn hình	Chức năng
1	Màn hình Tiếp nhận bảo trì xe	Màn hình nhập liệu	Cho phép nhập và lưu trữ thông tin về các xe sửa chữa
2	Màn hình Lập phiếu sửa chữa	Màn hình nhập liệu	Cho phép nhập và lưu trữ thông tin về các phiếu sửa chữa
3	Màn hình Tra cứu xe	Màn hình tra cứu	Cho phép nhập các tiêu chuẩn tra cứu và trình bày các kết quả tra cứu được
4	Màn hình Lập phiếu thu tiền	Màn hình nhập liệu	Cho phép nhập và lưu trữ thông tin về các phiếu thu tiền
5	Màn hình Báo cáo tháng	Báo biểu	Trình bày kết quả báo cáo



Sơ đồ màn hình

- Hệ thống các màn hình cùng với quan hệ về việc chuyển điều khiển giữa chúng.
- Hệ thống các màn hình = Màn hình chính + Các màn hình thực hiện các công việc của phần mềm.
- Ký hiệu:

Tên màn hình



Mô tả chi tiết từng màn hình

■ Mô tả các đối tượng trên màn hình

STT	Tên	Kiểu	Chức năng
1	txtBienSoXe	TextBox	Nhập biển số xe

■ Danh sách biến cố và xử lý tương ứng trên màn hình

STT	Biến cố	Xử lý	Mã số xử lý
1	Chọn button Cập nhật	Lưu thông tin về xe sẽ sửa chữa xuống CSDL	XLx



Phân loại màn hình

- **Màn hình chính:** cho phép người sử dụng chọn các công việc sẽ thực hiện với phần mềm.
- **Màn hình nhập liệu:** cho phép người sử dụng nhập vào các thông tin để lưu trữ hoặc tính toán.
- **Màn hình tra cứu:** cho phép tìm kiếm thông tin đã được lưu trữ với các tiêu chuẩn tìm kiếm.
- **Màn hình thông báo:** hiển thị các thông báo, nhắc nhở.
- **Báo biểu:** các báo cáo thống kê theo một mốc thời gian định sẵn.

Một số nguyên tắc trong thiết kế giao diện

- Tất cả màn hình phải có tên.
- Thiết kế phù hợp với đối tượng sử dụng.
- Dễ học, dễ nhớ, phù hợp với người mới sử dụng, đồng thời hỗ trợ các cách làm nhanh, làm tắt cho người sử dụng có kinh nghiệm.
- Chú ý môi trường triển khai ứng dụng cũng đòi hỏi những nguyên tắc khác nhau do có sự khác nhau về tốc độ xử lý (web form, window form...).
- Thứ tự trình bày trên màn hình phải phù hợp với văn hóa, thói quen của người sử dụng.

Một số nguyên tắc trong thiết kế giao diện (tt)

- Chỉ trình bày những nội dung thật sự cần thiết, không trình bày quá nhiều thông tin trên một màn hình.
- Chọn font chữ rõ ràng, cỡ chữ phù hợp, dùng chữ in hoa đúng trường hợp.
- Màu sắc hài hòa, nên dùng màu lạnh, chỉ dùng màu nóng khi cần gây chú ý (hoặc có thể dùng chớp, nháy).
- Kết hợp màu nền và màu chữ hợp lý.
- Không dùng quá nhiều màu sắc trên một màn hình.

Một số nguyên tắc trong thiết kế giao diện (tt)

■ **Nhất quán trong toàn bộ hệ thống về:**

- **Cách trình bày**
- **Tên các nút điều khiển**
- **Ý nghĩa biểu tượng**
- **Vị trí các mục**

■ **Số bước để đi đến một màn hình công việc chính phải ≤ 3**

Kiến trúc màn hình

Thành phần nhập liệu



Các nút xử lý

Thông tin kết quả



Thành phần nhập liệu

■ Cho phép người sử dụng nhập dữ liệu dưới nhiều hình thức khác nhau:

- Text Box
- Combo Box
- Radio Button/Option Button
- Check Box/Tick Box



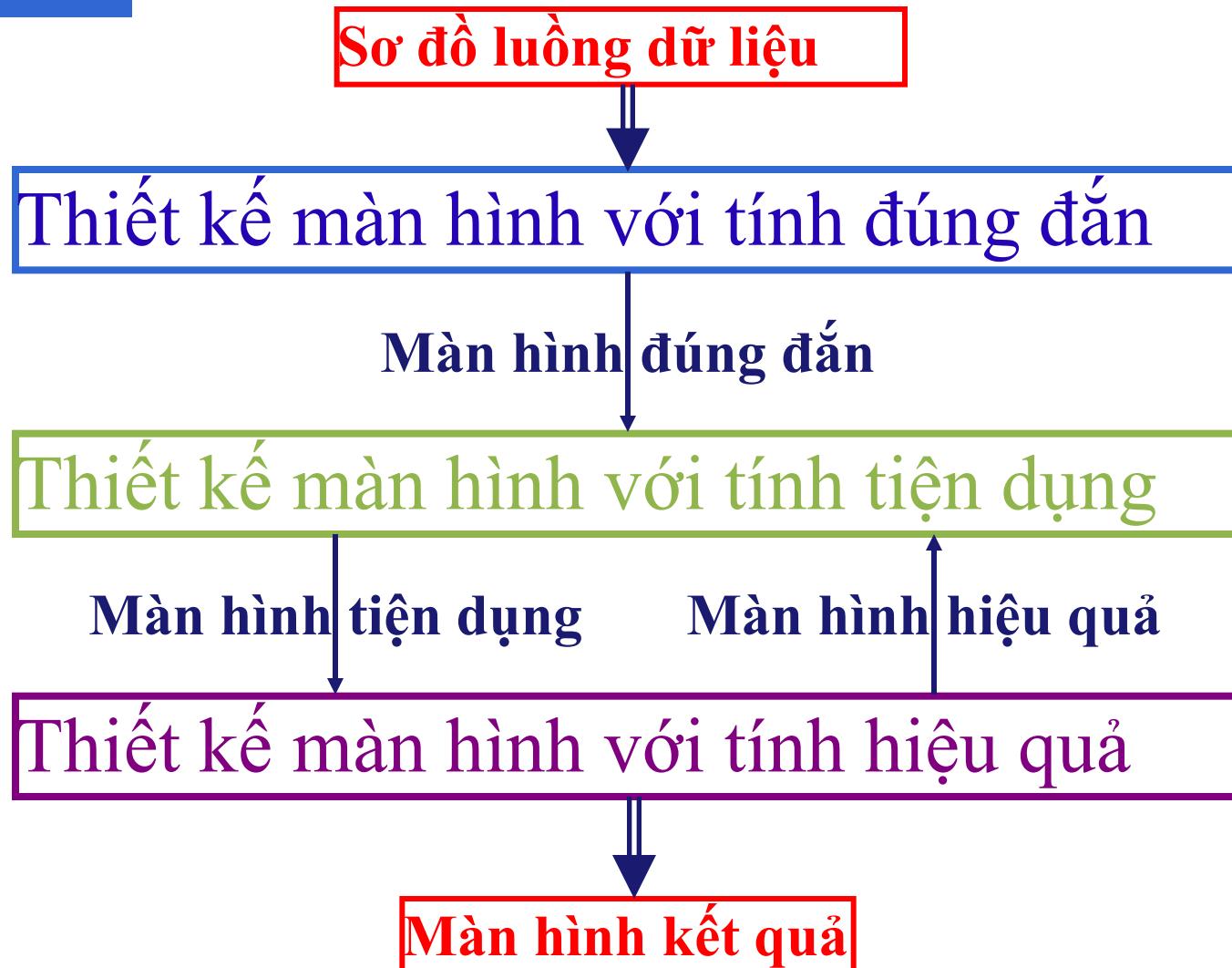
Các nút xử lý

- Các nút xử lý cho phép người sử dụng yêu cầu phần mềm thực hiện một xử lý nào đó.
- Tên các nút xử lý:
 - Không quá dài
 - Gợi nhớ
 - Nhất quán trong toàn hệ thống

■ Cho phép người sử dụng xem thông tin kết quả dưới nhiều hình thức khác nhau:

- Label
- Text Box
- List Box

Các bước thực hiện



Thiết kế màn hình với tính Tiện dụng

- Giao diện quen thuộc (dựa trên biểu mẫu tương ứng).
- Bố trí hợp lý.
- Cung cấp thêm thông tin cho người dùng.
- Cho phép nhập nhiều giá trị đồng thời (trên List Box).

Thiết kế màn hình với tính Hiệu quả

- **Chọn control thích hợp** (Text Box/Combo Box/Radio Button/Check Box...).
- **Cung cấp giá trị định sẵn, mặc định cho ô nhập liệu.**
- **Hiển thị kết quả một cách trực quan** (dùng màu sắc, hình vẽ, chú thích...).



Thiết kế màn hình chính

■ Từ danh sách các yêu cầu phần mềm (nghiệp vụ, chất lượng và hệ thống), tiến hành phân nhóm các công việc:

➤ **Theo chức năng Tin học:**

- Hệ thống
- Lưu trữ
- Tra cứu
- Tính toán
- Báo biểu



Thiết kế màn hình chính (tt)

- **Theo đối tượng:** danh sách các đối tượng kèm các công việc sẽ thực hiện trên từng đối tượng.
- **Theo nghiệp vụ, qui trình:** các giai đoạn thực hiện kèm các công việc tương ứng trong từng giai đoạn.

■ **Lựa chọn hình thức trình bày:**

- **Thực đơn (kết hợp sử dụng phím nóng)**
- **Biểu tượng**
- **Sơ đồ**



Thiết kế màn hình Tra cứu

- Cho phép tra cứu theo nhiều tiêu chuẩn khác nhau, ưu tiên chọn tiêu chuẩn tra cứu đơn giản nhất.
- Cho phép *tìm và cập nhật lại thông tin đã lưu trữ*.
- Kết quả tra cứu = Thông tin của đối tượng và quá trình hoạt động của đối tượng.
- Hỗ trợ xem kết quả tra cứu dưới nhiều hình thức khác nhau (danh sách, biểu đồ, hình ảnh...).
- Bổ sung các nút điều khiển cho các nhu cầu kết xuất (máy in, tập tin...).



Thiết kế màn hình Nhập liệu

- **Thành phần nhập liệu:** kiểm tra tính hợp lệ của các giá trị nhập vào dựa vào qui định liên quan.

Thông thường mã số được phát sinh tự động.

- **Các nút xử lý:**

- Thêm mới, cập nhật, lưu thông tin.
- Bổ sung các nút chuyển điều khiển đến màn hình có liên quan (thêm loại đại lý, quận...).

- **Thông tin kết quả:** trình bày kết quả nhập liệu, nên bổ sung thông tin tính toán (số sách mượn hiện tại, tổng số tiền hiện tại của hóa đơn...).



Thiết kế màn hình thông báo

- Thông báo đúng lúc.
- Thông báo về kết quả thực hiện yêu cầu (thành công/thất bại, nguyên nhân).
- Cung cấp phản hồi của hệ thống khi cần thiết (progress bar, thông báo chờ khi hệ thống đang thực hiện một xử lý mất nhiều thời gian...).
- Thiết kế đơn giản, gồm thông tin cần thông báo và các nút chọn, chú ý không quá nhiều nút chọn.

- **Giữ lại tiêu đề báo cáo khi qua trang khác hoặc khi kéo thanh trượt lên xuống.**
- **Lưu ý kích thước khác nhau về không gian hiển thị giữa báo biểu in ra giấy và báo biểu xuất ra màn hình.**
- **Chỉ hiển thị những thông tin thật sự cần thiết, tránh làm rối báo biểu.**

Ví dụ

- ❖ Xét phần mềm Quản lý học sinh với chức năng tiếp nhận học sinh mới

Tiếp nhận học sinh

Họ tên:.....

Giới tính:.....

Ngay sinh:.....

Địa chỉ:.....

Lớp:.....

Qui định: Họ tên phải có. Tuổi từ 15-20. Trường có 20 lớp và 3 khối. Khối 10 có 8 lớp, Khối 11 có 7 lớp. Khối 12 có 5 lớp

- ❖ Hãy thiết kế dữ liệu và giao diện

Ví dụ

Tiếp nhận học sinh

Họ tên	<input type="text"/>	Nam <input checked="" type="checkbox"/>
Ngày sinh	<input type="text"/>	Lớp <input type="text"/> 
Địa chỉ	<input type="text"/>	
Ghi		

Danh sách học sinh đã tiếp nhận

STT	Mã HS	Tên HS	Giới tính	Ngày sinh
...

Ví dụ

❖ Mô tả các thành phần của giao diện

STT	Tên	Kiểu	Ý nghĩa	Miền giá trị	Giá trị mặc định	Ghi chú
1	Lb_Tieu_de	A_Label	Tiêu đề màn hình			
2	Lb_Hoten	A_Label	Tiêu đề họ tên			
3	Txt_Hoten	A_Textbox	Text box nhập họ tên			
4	Ch_Phai	A_Checkbox				
5	Lb_Ngaysinh	A_Datetime				
...

Bài tập

- ❖ Xét phần mềm quản lý giải bóng đá với các yêu cầu sau:
 - Tiếp nhận đăng ký tham gia
 - Xếp lịch thi đấu
 - Ghi nhận kết quả thi đấu
 - Lập bảng xếp hạng
- ❖ Giải gồm có 4 đội tham gia.
 - Thông tin về mỗi đội bao gồm: Tên đội, danh sách các cầu thủ của đội
 - Thông tin về mỗi cầu thủ bao gồm: Tên cầu thủ, vị trí sở trường
- ❖ Xếp lịch thi đấu
 - Các đội thi đấu vòng tròn hai lượt. Thông tin về trận đấu được xếp lịch: Hai đội bóng tham dự, ngày giờ thi đấu, sân thi đấu (giải diễn ra trên hai sân: A, B)

Bài tập

- ❖ Ghi nhận kết quả thi đấu, chỉ yêu cầu ghi nhận:
 - Tỉ số
 - ❖ Lập bảng xếp hạng:
 - Hạng được dựa trên các tiêu chí sau:
 - Điểm số
 - Hiệu số
 - Số bàn thắng
 - Điểm cho mỗi trận thua là 0, hòa là 1, thắng là 3
1. Hãy lập sơ đồ luồng dữ liệu cho mỗi yêu cầu
 2. Lập sơ đồ logic dữ liệu
 3. Thiết kế màn hình giao diện cho từng yêu cầu.

Giao diện ghi nhận KQTĐ

❖ Cách 1:

GHI NHẬN KẾT QUẢ TRẬN ĐẤU

Trận đấu

 -

Tỷ số

 -

Ghi

Giao diện ghi nhận KQTĐ

❖ Cách 2:

GHI NHẬN KẾT QUẢ TRẬN ĐẤU

Đội bóng 1

	▽
--	---

Đội bóng 2

	▽
--	---

Tỷ số

--

--

Ghi

Giao diện ghi nhận KQTĐ

❖ Cách 3:

GHI NHẬN KẾT QUẢ TRẬN ĐẤU

Đội bóng 1	Đội bóng 2	Tỷ số

Ghi

Giao diện ghi nhận KQTĐ

❖ Cách 4:

GHI NHẬN KẾT QUẢ TRẬN ĐẤU

STT	Trận đấu	Tỷ số

Ghi

Giao diện ghi nhận KQTĐ

❖ Cách 5:

GHI NHẬN KẾT QUẢ TRẬN ĐẤU

	Đội bóng 1	Đội bóng 2	...	Đội bóng k
Đội bóng 1	X			
Đội bóng 2		X		
.....			X	
Đội bóng k				X

Ghi



Thiết kế giao diện

- ❖ Quản lý thiết bị nhập (bàn phím, chuột)
- ❖ Hiệu chỉnh thông tin input
- ❖ Kiểm soát lỗi và hiển thị thông báo lỗi
- ❖ Cung cấp trợ giúp và hiển thị thông báo nhắc nhở
- ❖ Cung cấp feedback (ví dụ như tự động hiển thị ký tự đánh vào)
- ❖ Kiểm soát cửa sổ và vùng, khả năng cuộn
- ❖ Thiết lập giao tiếp giữa chương trình với giao diện (vd: hàm đáp ứng)
- ❖ Cách ly chương trình với các hàm quản lý giao diện
- ❖ Cho phép tùy biến giao diện: màu sắc, kích thước,..



Một số định hướng về thiết kế giao diện

Một số hướng dẫn chung

- ❖ Nên đồng nhất (menu, lệnh, hiển thị...)
- ❖ Nên cung cấp feedback cho người dùng
- ❖ Yêu cầu xác nhận những tác vụ mang tính phá hoại (xoá file, account)
- ❖ Nên hỗ trợ UNDO, REDO
- ❖ Hạn chế lượng thông tin phải ghi nhớ giữa 2 tác vụ liên tiếp
- ❖ Tối ưu trong trình bày hộp thoại và di chuyển mouse
- ❖ Chấp nhận lỗi từ phía người sử dụng
- ❖ Cung cấp trợ giúp trực tuyến
- ❖ Dùng động từ đơn giản và ngắn gọn để đặt tên các lệnh

Đối với thông tin hiển thị

- ❖ Chỉ hiển thị những thông tin phù hợp với ngữ cảnh hiện tại
- ❖ Dùng tên, từ viết tắt và màu gợi nhớ
- ❖ Cho phép tương tác trực quan
- ❖ Tạo thông báo lỗi có ý nghĩa
- ❖ Hiển thị dữ liệu ở nhiều dạng khác nhau trong cửa sổ
- ❖ Thiết lập biểu diễn tương tự
- ❖ Sử dụng không gian màn hình một cách tối ưu

Đối với thông tin input

- ❖ Hạn chế input trực tiếp (có thể chọn lựa từ một số dữ liệu có sẵn)
- ❖ Nên đồng nhất giữa thông tin input và hiển thị
- ❖ Nên cho phép tuỳ biến input
- ❖ Cấm các chức năng không thích hợp trong ngữ cảnh hiện tại
- ❖ Cho phép input ở nhiều dạng khác nhau
- ❖ Để cho người sử dụng kiểm soát dòng sự kiện tương tác
- ❖ Tự động tính các giá trị input cho người sử dụng nếu có thể

Ví dụ 1

Nhập Luận Văn

NHẬP LUẬN VĂN

Mã LV	11	Ngày nhập	10/10/2006	Tựa	
Ngày BV	12/12/2005	Nơi BV		Tựa Nước ngoài	
Ngôn Ngữ	Anh	+ Loại	Cử Nhân	Tóm Tắt	
Số Bản		PL Dewey		GV Hướng Dẫn	Duong Anh Duc
Mã Số		Chuyên Ngành	Công Nghệ Phần	GV Phản Biện	Nguyen Thi Bich
Lĩnh Vực AD				Điểm	
Thuật Toán					
Tác giả 1					
Tác giả 2					

Lưu Bỏ Thoát

Ví dụ 1

Phiếu nhập luận văn

Số đăng ký	ILA395	Học vị	Thạc sĩ	KHXG	V5(2)5.4
Ngôn ngữ	Việt	Mã số CN	5.04.33	Mã hóa	S550T
Môn loại	V4/5	Lý luận nghiên cứu văn	Năm TH	1999	Ngày duyệt
PL BBK	V5(2)5.4	PL 19 dãy	8(V)1	PL Dewey	
Tên LV	Sự tiếp biến ba hệ tư tưởng Nho - Phật - Lão	Ngày BV	1999	Bản LV	1
Tên dịch		Kích thước	30	Số trang	162
Người TH	Trần Trọng Khiêm	Người HD	GS. Mai Cao Chương		
Nơi viết	Tp. HCM.	Chọn đề mục			
Nơi BV	Trường ĐH KHXH & NV Tp. Hồ Chí	X _✓	Thơ	▼	▲
Minh họa		X _✓	Nguyễn Trãi	▼	▲
TLĐK		X _✓	Nho giáo	▼	▲
Thêm Sao chép Hủy Ghi Không ghi Tra cứu Thoát					

Người dùng Lê Trọng Vinh
Công tác Quản lý thư viện

Phát sinh ngày 10-04-2002
Bởi Lê Trọng Vinh

Cập nhập ngày 10-04-2002
Bởi Lê Trọng Vinh



TK TĐ THEO HƯỚNG TIN HỌC

- ❖ Các chức năng được phân nhóm theo ý nghĩa tin học
- ❖ Ví dụ: PM quản lý giáo vụ

Hệ thống	Danh mục	Cập nhật	Xử lý	Tìm kiếm	Báo biểu
Sao chép	Môn học	Học phí	Tính thù lao	Giáo viên	Danh sách lớp
Phân quyền	Giáo viên	Sinh viên	Phân công	Sinh viên	Danh sách thi
Tham số	Phòng	Đăng ký	Xếp TKB	Khoa	DS tốt nghiệp
		Điểm			

TK TD THEO HƯỚNG ĐỐI TƯỢNG

Các chức năng được phân nhóm theo đối tượng

Sinh viên	Giáo viên	Học phần	Phòng	Trường
Cập nhật	Cập nhật	Cập nhật	Cập nhật	Sao chép
Tìm kiếm	Tìm kiếm	Danh sách thi	Xếp TKB	Phân quyền
Đăng ký	Phân công	Nhập điểm		Quy định
Xem điểm	TH thao tác			

TK TD THEO HƯỚNG NGHIỆP VỤ

❖ Các chức năng được phân nhóm qui trình nghiệp vụ

Tổ chức	Lập kế hoạch	Ghi danh	Theo dõi	Thi	Tổng kết
Môn học	Mở học phần	Đăng ký	Tính thù lao	DS thi	DS tốt nghiệp
Phòng	Phân công	DS lớp		Nhập điểm	DS rót
Giáo viên	Xếp TKB				
Sinh viên					
Quy định					
Sao Chép					
Phân Quyền					



Thiết kế giao diện nhập

- Tránh tình trạng cổ chai nhưng vẫn cho phép nhiều người cùng nhập
 - Tận dụng những bước xử lý trên màn hình
 - Tránh thường xuyên truy xuất dữ liệu
- Tránh tạo cơ hội cho NSD phạm lỗi/sơ xuất
- Tránh thêm những công đoạn thừa
- Quy trình nhập đơn giản nhất, tự nhiên nhất
 - Phải đảm bảo thói quen xử lý chứng từ gốc



Thiết kế giao diện nhập

- Phát hiện sai sót và kiểm tra ràng buộc toàn vẹn
 - Kiểm tra ngay?
 - Kiểm tra cuối màn hình?
 - Kiểm tra cuối kỳ khai thác?
- => Phải quan sát thói quen của người sử dụng
 - Thói quen phải tôn trọng => bắt buộc phải tôn trọng
 - Thói quen chưa hợp lý hoặc mâu thuẫn với kỹ thuật
 - => Phải trao đổi, thuyết phục và thống nhất với NSD



Thiết kế giao diện nhập

- ❖ Có bao nhiêu chức năng nhập?
 - Danh mục: mỗi bảng là 1 chức năng nhập (Thêm, Xoá, Sửa)
 - Đối tượng: mỗi bảng là 1 chức năng nhập (Thêm, Xoá, Sửa)
- ❖ Quan hệ m – n: tùy các mối quan hệ (1-n, n-m) chung quanh đối tượng này và tùy ngữ cảnh trong thế giới thực se có thêm các chức năng nhập cho các quan hệ đó..

Thiết kế giao diện nhập

❖ Danh mục

1. Dạng chi tiết:

Nhập Tác Giả Mới

Mã	<input type="text"/>
Tên	<input type="text"/>
<input type="button" value="Lưu"/> <input type="button" value="Bỏ"/> <input type="button" value="Thoát"/>	

2. Dạng danh sách:

Nhập Tác Giả Mới

Mã	<input type="text"/>
Tên	<input type="text"/>
<input type="button" value="Lưu"/> <input type="button" value="Bỏ"/> <input type="button" value="Thoát"/>	

Nhắc nếu quên
lưu



Thiết kế giao diện xuất

❖ Nội dung:

- Dữ liệu trong HTTT,
- Dữ liệu tính toán,
- Dữ liệu vừa được nhập vào

❖ Ai sẽ sử dụng nội dung kết xuất:

- Đối tượng trong HT: dùng kết xuất cho mục tiêu gì?
 - Đối tượng ngoài môi trường: muốn đưa nội dung gì bên ngoài?
- ❖ Khối lượng dữ liệu xuất hiện trên kết xuất
- ❖ Khi nào thực hiện kết xuất đó?
- ❖ Kết xuất ra dạng gì? Màn hình, giấy in, file theo
- ❖ định dạng nào?

❖ Trình bày bảng biểu

- Nếu dữ liệu nhiều thì lấy những nội dung gì
 - Cách bố trí nội dung theo cột/dòng/tiểu cột/ tiểu dòng..., tổng các tiểu dòng/tiểu cột, thứ tự trình bày các tổng trước hay sau thông tin chi tiết, chữ hoa/thường, thứ tự trình bày dữ liệu, chấm/phẩy đối với số lẻ, canh lề, số lượng số lẻ
- ⇒ **Phải mô phỏng nội dung dữ liệu trong phần thiết kế giao diện, không nên chỉ đưa cho NSD xem những template rỗng**

❖ Trình bày biểu đồ

- Chỉ chứa dữ liệu tổng hợp, không có dữ liệu chi tiết
⇒ Thể hiện khuynh hướng, so sánh. Không hiệu quả khi cần xác định số liệu cụ thể
- Trình bày biểu đồ dạng nào?



Thiết kế đối thoại

❖ Đặc điểm:

- Giao diện tương tác
- Giao diện đồ họa
- Thao tác “trực tiếp”
- Nguyên tắc: LOOK and FEEL
- Mô hình WIMP: Window – Icon – Menu – Pointer

Các mức thiết kế

- Mức ngữ nghĩa: nội dung dữ liệu, nội dung chức năng
- Mức cú pháp: quá trình tương tác đối thoại, kịch bản khai thác diễn ra như thế nào?
- Mức từ vựng: từ ngữ và cách trình bày nội dung ngữ nghĩa



Thiết kế đối thoại – mức ngũ nghĩa

❖ Mức ngũ nghĩa

- Hệ thống hướng dẫn trực tuyến đi song song với từng chức năng trong hệ thống thực đơn

- Các hộp thoại cần thiết thông báo lỗi/hướng dẫn

- Dựa vào hệ thống thực đơn để xác định các giao diện đối thoại

❖ Cần nhớ: tính “ergonomic” và tính nhất quán

- Gây chú ý nhưng không được làm NSD mệt mỏi

- Nhất quán

- Giá trị mặc định (câu trả lời thường hay xảy ra nhất)

- Nội dung thông báo phải thể hiện đúng nội dung phản ứng của hệ thống

❖ Mức ngũ nghĩa tốt

=> Có thể dễ dàng thay đổi ngôn ngữ



Thiết kế đối thoại – mức cú pháp

- ❖ Chọn kiểu đối thoại và kiểu điều khiển đối thoại

Kiểu đối thoại:

- Hỏi – Trả lời
- Thực đơn
- Mẫu biểu để điền vào
- Ngôn ngữ lệnh nhập qua bàn phím
- Dùng các phím chức năng
- Ngôn ngữ tự nhiên
-



Thiết kế đối thoại - Mức từ vựng

Nên dùng biểu tượng => chọn lựa biểu tượng thể hiện đúng nội dung mong muốn

- Biểu tượng đi kèm tên gọi ngắn gọn, súc tích và nhất quán
- Cách sử dụng các vật thể: ý nghĩa khi nhấn bên trái/ phải, nhấn đúp trên mouse...
- Cách di chuyển con trỏ
- Cách dùng màu:
 - + Phân biệt các cửa sổ khác nhau lần lượt được mở ra
 - + Tạo mối liên hệ giữa các đối tượng thông tin
 - + Tình trạng khác nhau của đối tượng
 - + Gây chú ý đặc biệt
- Các màu phải hòa hợp



Hệ thống thực đơn (Menu)

❖ Hệ thống thực đơn chính

- Hệ thống thực đơn trong màn hình chính của mỗi phân hệ (module)

❖ Hệ thống thực đơn ngũ cảnh

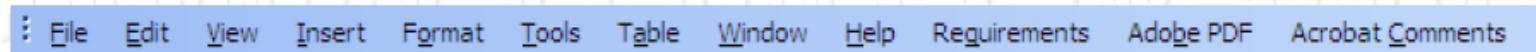
- Thực đơn sử dụng khi click chuột phải lên một hoặc một nhóm đối tượng trên màn hình
- Thực đơn sử dụng khi click chuột phải lên một dòng trong lưới (Grid)



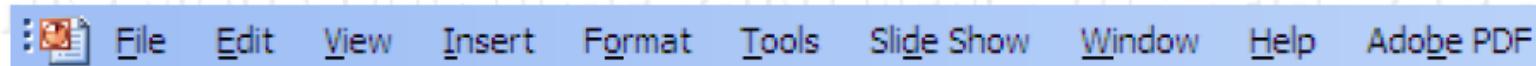
Tổ chức hệ thống thực đơn chính

- ❖ Hệ thống
- ❖ Danh mục
- ❖ Nhóm chức năng 1 (nghiệp vụ 1)
- ❖ Nhóm chức năng 2 (nghiệp vụ 2)
- ❖ Tra cứu
- ❖ Báo cáo (Report)
- ❖ Công cụ/tiện ích (Tools/Utilities)
- ❖ Trợ giúp (Help)

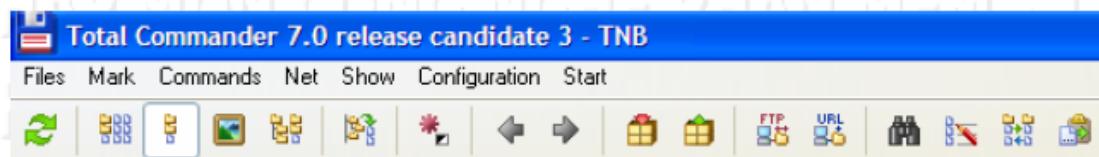
- **Hệ thống thực đơn Microsoft Word 2003**



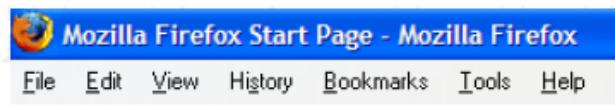
- **Hệ thống thực đơn Microsoft PowerPoint 2003**



- **Hệ thống thực đơn Total Commander**



- **Hệ thống thực đơn Mozilla FireFox**



- **Hệ thống thực đơn Adobe Acrobat Reader**



- **Hệ thống thực đơn ứng dụng quản lý học sinh**

Chương trình quản lý học sinh

[Hệ thống](#) [Danh mục](#) [Quản lý học sinh](#) [Tra cứu](#) [Báo cáo thống kê](#) [Tiện ích](#) [Giúp đỡ](#)

- **Hệ thống thực đơn ứng dụng quản lý kho**



[Dashboard](#) [Admin](#) [Receiving](#) [Shipping](#) [Tools](#) [Reports](#) [Logout](#) [Languages](#) [Help](#)

- **Hệ thống thực đơn ứng dụng quản lý sinh viên**

School On Demand System



TRƯỜNG ĐẠI HỌC SƯ PHẠM - TP.HCM

280 An Dương Vương St. - District 05

[Home](#) | [Admin](#) | [Student Management](#) | [Lead Management](#) | [Report](#) | [Logout](#) | [Help](#)



Hệ thống thực đơn chính

❖ Hệ thống

- Cấu hình hệ thống
- Đăng nhập
- Đổi mật khẩu
- Thoát

❖ Danh mục

- Danh mục nhân viên
- Danh mục quận huyện
- Danh mục tỉnh thành
- Danh mục hàng hóa
-

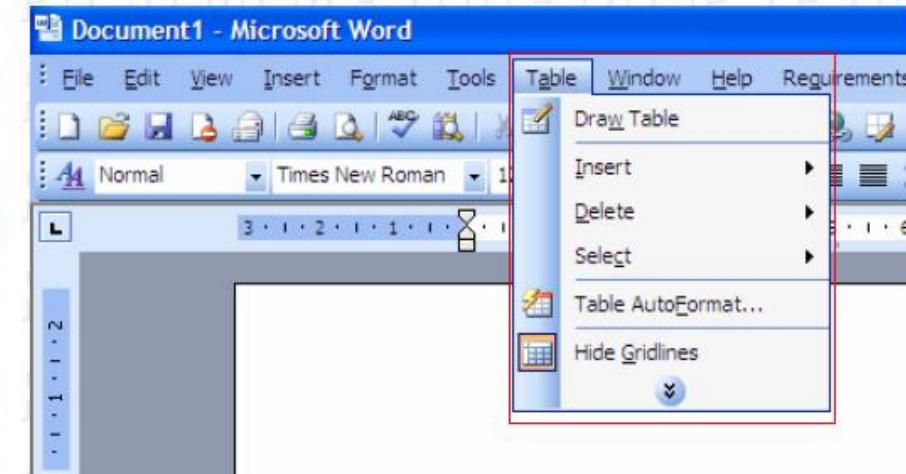
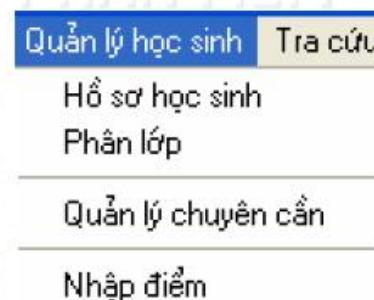
Hệ thống thực đơn chính

- Nhóm chức năng 1

- Chức năng 1.1
- Chức năng 1.2
- Chức năng 1.3
- Chức năng 1.4

- Nhóm chức năng 2

- Chức năng 2.1
- Nhóm chức năng 2.2
- Chức năng 2.3
- Chức năng 2.4
-





Hệ thống thực đơn chính

❖ Tra cứu

- Tra cứu điểm thi
- Tra cứu thông tin sinh viên
- Tra cứu lịch thi
- Tra cứu thời khóa biểu
- ...

❖ Báo cáo/thống kê

- Bảng điểm sinh viên
- Hồ sơ sinh viên/nhân viên/học sinh
- Thống kê doanh số bán trong ngày/tháng/năm
- Bảng lương tháng nhân viên
- Thống kê số lượng hàng tồn kho



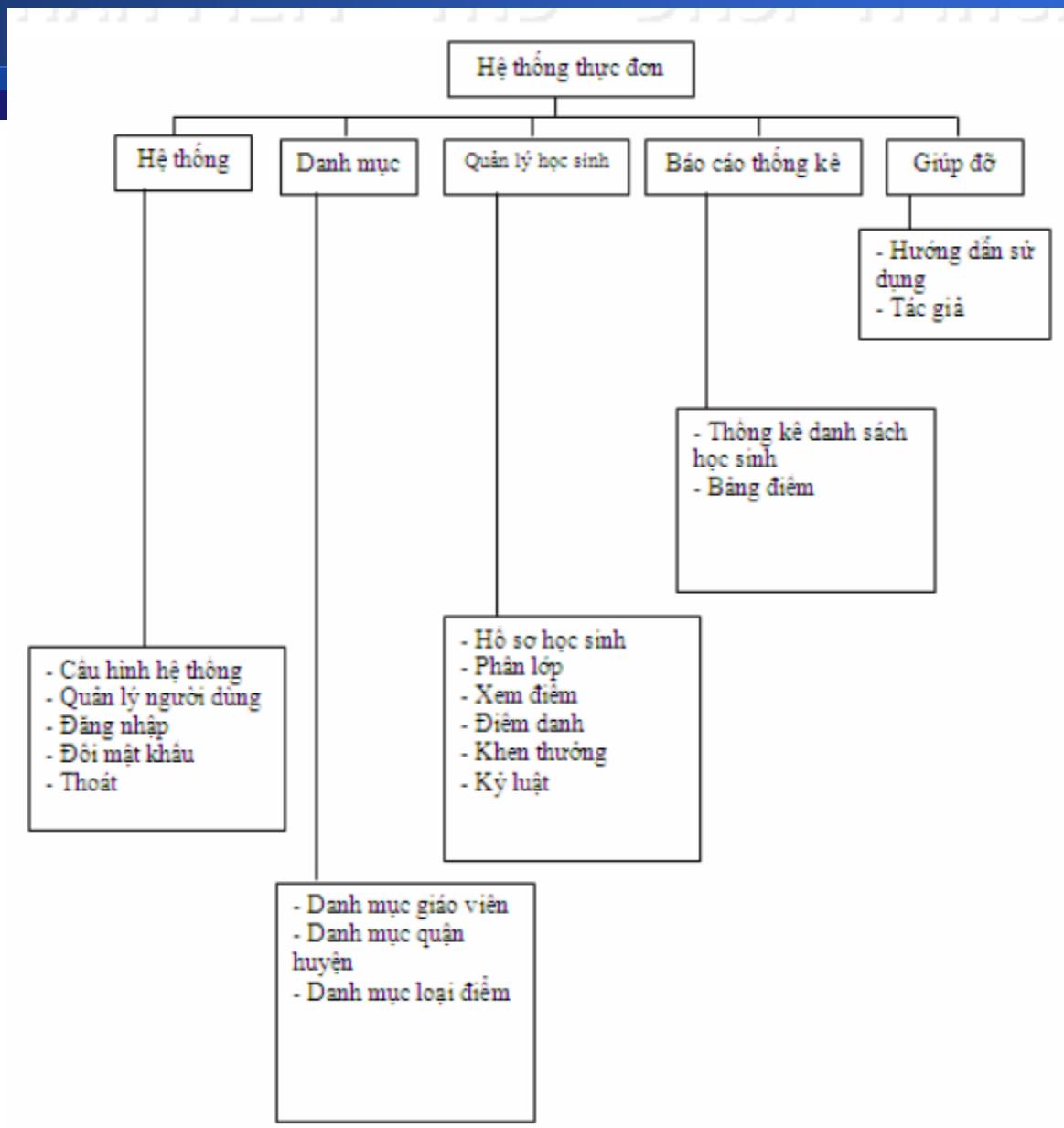
Hệ thống thực đơn chính

❖ Công cụ/tiện ích

- Sao lưu dữ liệu (Backup Data)
- Phục hồi dữ liệu (Restore Data)
- Archive Data
- Tùy chọn (Options/Customize)
- ...

❖ Giúp đỡ

- Hướng dẫn sử dụng (Online Help)
- Thông tin tác giả (About)
- Đăng ký sử dụng (Register)
- Nâng cấp (Upgrade/Online Update)
- ...





Phân loại màn hình

- ❖ Màn hình giới thiệu
- ❖ Màn hình chính
- ❖ Màn hình tác giả
- ❖ Màn hình danh mục
- ❖ Màn hình nhập liệu
 - Nhập liệu trực tiếp
 - Nhập liệu gián tiếp (import)
- ❖ Màn hình tra cứu
- ❖ Màn hình xử lý tính toán
- ❖ Màn hình thể hiện báo biểu/thống kê
- ❖ Màn hình thể hiện đối tượng: sơ đồ, cây, biểu tượng

Tính nhất quán



Không nhất quán

Tính nhất quán

Thong tin nhan vien

Thông tin nhân viên

Mã NV	001	Họ tên	Nguyễn Công Phú
Ngày sinh	28/10/1979	Địa chỉ	153/2 Hoàng Văn Thụ
Điện thoại	0913123456	Email	Text6
Tôn giáo	Phật giáo		

Danh sách nhân viên

...

Thêm Xóa Sửa Ghi Không Thoát

Thong tin sinh vien

Thông tin sinh viên

Mã SV (*)	001	Họ tên (*)	Nguyễn Công Phú
Ngày sinh (*)	28/10/1979	Địa chỉ (*)	153/2 Hoàng Văn Thụ
Điện thoại	0913123456	Email	Text6
Tôn giáo	Phật giáo		

Danh sách sinh viên

...

Thêm Xóa Sửa Ghi Không Thoát

Tính trực quan

Select Table Number

The screenshot shows a software application for managing tables in a restaurant. On the left, a large grid displays 22 tables, each with a small icon and a label. The tables are arranged in four rows: Row 1 has Bar15, Bar16, Bar17; Row 2 has Bar14, Bar18, Bar19, Bar20, Bar21, Bar22; Row 3 has Bar4, Bar5, Bar6, Bar7, Bar8, Bar9, Bar10; Row 4 has Bar3, Bar2, Bar1. To the right of the grid is a vertical sidebar with several buttons:

- Main Dining Room
- Patio
- Banquet
- Hibachi
- Bar
- Hostess
- Assign Table
- Assign Tab
- Filter Table
- Refresh Screen
- Cancel

Tính trực quan

Select Table Number

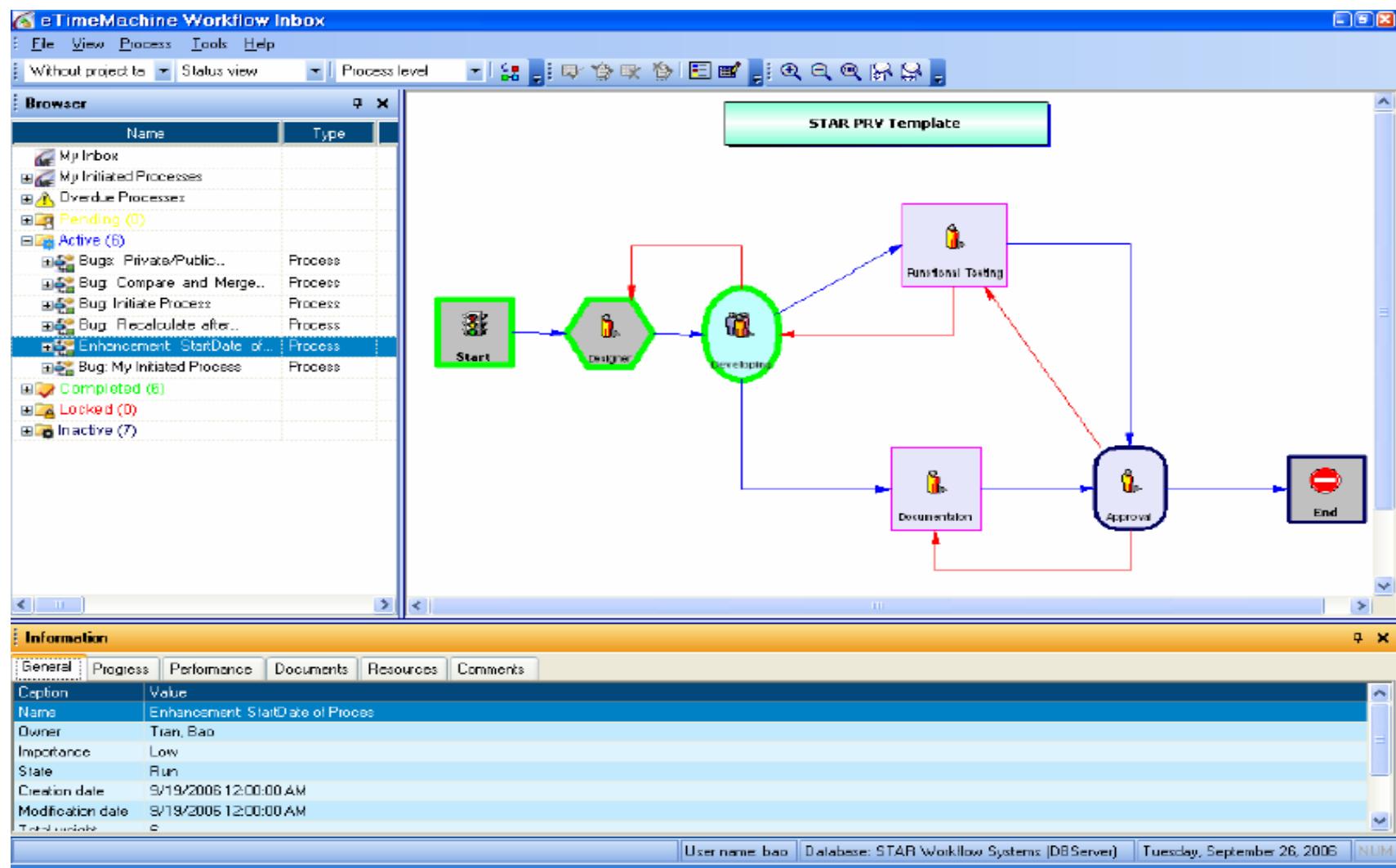
20	21	22	23	24	25	26
19						
18						
17						
16						
15						
14	13					
		8	7	6	5	
		9			4	
		10			3	
		11			2	
		12			1	

Main Dining Room Patio
Banquet Hibachi
Bar

Hostess
Assign Table
Assign Tab
Filter Table Refresh Screen
Cancel

The interface includes a sidebar on the right with the following buttons:
Hostess
Assign Table
Assign Tab
Filter Table (disabled)
Refresh Screen (disabled)
Cancel

Tính trực quan



Ví dụ

- ❖ Tính đúng đắn

Cho mượn sách

Mã Độc Giả

Mã sách

Ngày mượn

Ví dụ

❖ Tính tiện dụng

Cho mượn sách	
Mã Độc Giả	<input type="text"/>
Họ Tên	<input type="text"/>
Loại Độc Giả	<input type="text"/>
Ngày Sinh	<input type="text"/>
Mã sách	<input type="text"/>
Tên Sách	<input type="text"/>
Thể Loại	<input type="text"/>
Tác Giả	<input type="text"/>
Ngày mượn	<input type="text"/>

❖ Tính hiệu quả

Cho mượn sách	
Mã Độc Giả	
Họ Tên	
Loại Độc Giả	
Ngày Sinh	
Mã sách	
Tên Sách	
Thể Loại	
Tác Giả	
Ngày mượn	<Ngày hệ thống>

Ví dụ

❖ Tính hiệu quả

Cho mượn sách

Mã phiếu	<input type="text"/>	Ngày mượn	<input type="text" value="<Ngày hệ thống>"/>
Mã Độc Giả	<input type="text"/>	Loại Độc Giả	<input type="text"/>
Họ Tên	<input type="text"/>	Ngày Sinh	<input type="text"/>

STT	Mã sách	Tên sách	Thể loại	Tác giả
<input type="text"/>		<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>		<input type="text"/>	<input type="text"/>	<input type="text"/>

Lập Phiếu mượn sách

Mã Độc Giả

Ngày mượn

<Ngày hệ thống>

Họ Tên

Ngày hết hạn

STT	Mã sách	Tên sách	Tác giả	Tình Trạng	Ngày DK trả



Các sách mượn quá hạn



Các sách chưa trả, còn trong hạn trả

Ví dụ về menu

- ❖ Tính đúng đắn: đơn giản là xác định danh sách các công việc và các nghiệp vụ mà phần mềm hỗ trợ.

Công việc 1

Công việc 2

Công việc 3

Công việc n

Ví dụ về menu

- ❖ Tính tiện dụng.

Nhóm 1	Nhóm 2	...	Nhóm n
Công việc 1.1	Công việc 2.1	...	Công việc n.1
Công việc 1.2	Công việc 2.2	...	Công việc n.2
Công việc 1.3	Công việc 2.3	...	Công việc n.3
...
Công việc 1.k	Công việc 2.p	...	Công việc n.m



Menu quản lý thư viện

❖ Tính đúng đắn

1. *Lập thẻ độc giả*
2. *Tiếp nhận sách*
3. *Cho mượn sách*
4. *Trả sách*
5. *Tra cứu sách*
6. *Lập báo cáo thống kê tình hình
cho mượn sách.*
7. *Gia hạn thẻ*
8. *Tra cứu độc giả*
9. *Báo cáo thống kê độc giả*
10. *Thanh lý sách*
11. *Lập báo cáo thống kê sách*



Menu quản lý thư viện

❖ Tính đúng đắn

Yêu cầu chất lượng-Tính tiến hóa

12. Cập nhật bảng tham số
13. Cập nhật loại độc giả
14. Cập nhật thể loại sách
15. Cập nhật lý do thanh lý sách

Yêu cầu chất lượng-Tính hiệu quả

16. Giá trị định sẵn



❖ Tính đúng đắn

Yêu cầu chất lượng-Tính tương thích

17. Chuyển dữ liệu **độc giả** sang tập tin Excel
18. Chuyển dữ liệu **sách** sang tập tin Excel
19. Import dữ liệu **độc giả** từ tập tin Excel
20. Import dữ liệu **sách** từ tập tin Excel

Yêu cầu hệ thống-Tính bảo mật

21. Cập nhật phân quyền
22. Đăng ký sử dụng



Menu quản lý thư viện

❖ Tính đúng đắn

Yêu cầu hệ thống-Tính an toàn

23. *Sao lưu dữ liệu*
24. *Phục hồi dữ liệu*
25. *Quản lý dữ liệu được tạm xóa*

Yêu cầu hệ thống-Các công việc đặc trưng tin học (không làm trong thế giới thực)

26. *Cấu hình thiết bị*
27. *Cấu hình phần mềm*
28. *Thoát*



Menu quản lý thư viện

- ❖ Tính hiệu quả

Hệ Thống	Nghiệp Vụ	Chất lượng



Menu quản lý thư viện

- ❖ Tính hiệu quả

Hệ Thống	Lưu Trữ	Tra cứu	Báo Biểu



Menu quản lý thư viện

- ❖ Tính hiệu quả

Thư Viện	Độc Giả	Sách



Menu quản lý thư viện

- ❖ Tính hiệu quả

Tổ chức	Quản Lý Sách	Quản Lý Độc Giả	Quản Lý Mượn Trả

Màn hình tra cứu

- ❖ Tiêu chuẩn: là các thuộc tính của đối tượng
 - Nhập liệu: ô *textBox* (*NSD tự gõ*)
 - Chọn những giá trị: *comboBox* (*Mã khóa ngoại*)
 - Thuộc tính kiểu số: cho chọn 1 đoạn giá trị.
- ❖ Danh sách đối tượng: 2 cách thể hiện:
 - *Tĩnh*: Số lượng thuộc tính trong danh sách là cố định
 - *Động*: Số lượng thuộc tính trong danh sách do NSD quyết định.
- ❖ Chi tiết:
 - Xác định chi tiết trong khoảng thời gian từ ngày đến ngày
 - Có nhiều nút điều khiển khác nhau cho các chi tiết khác nhau
- ❖ Biểu thức logic: Mặc nhiên là phép AND. Mở rộng: Phủ định (NOT), Và (AND), Hay (OR)
→ Thêm *comboBox* để chọn lựa phép toán

Màn hình tra cứu

❖ Hình thức trình bày

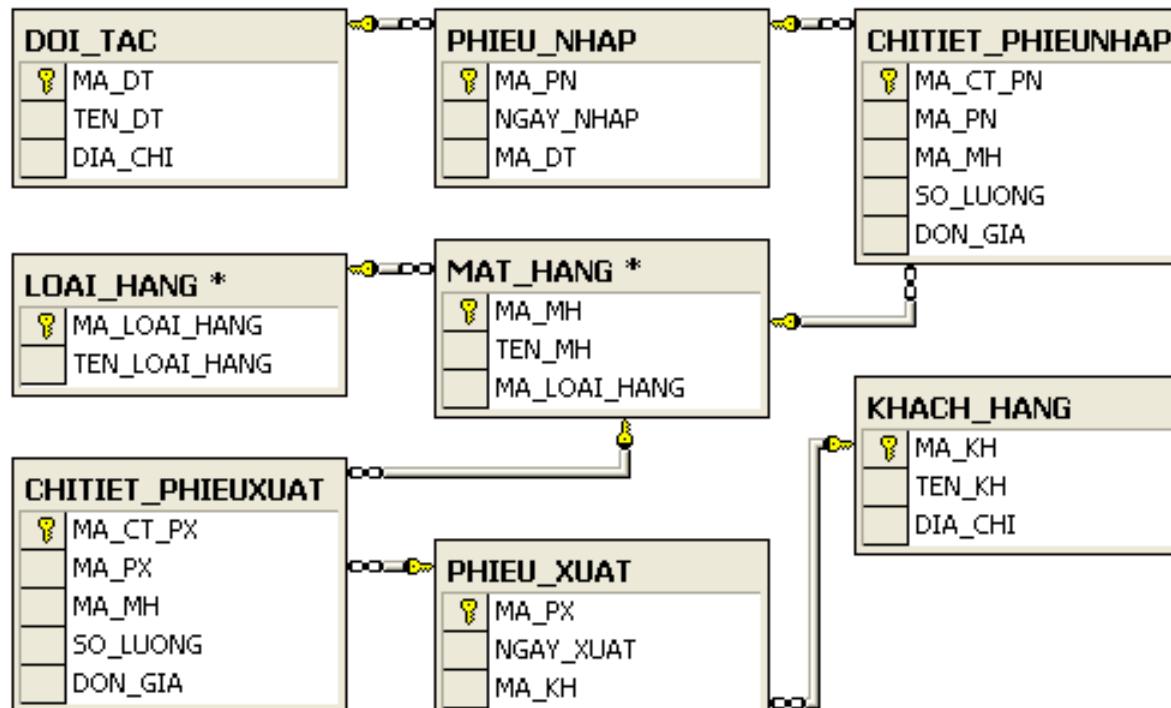
Tiêu chuẩn
Kết quả tổng quan
Chi tiết

Tiêu chuẩn	Kết quả tổng quan
Chi tiết	

Tiêu chuẩn	Chi tiết
Kết quả tổng quan	

Bài tập

- ❖ Cho CSDL của phần mềm quản lý bán hàng như sau:



- ❖ Hãy thiết kế giao diện và mô tả các thành phần của giao diện:
- Lập phiếu nhập, lập phiếu xuất
 - Tra cứu hàng hóa

Thiết kế Xử lý





Thiết kế xử lý

- ❖ Thế giới thực: xử lý là nghiệp vụ của nhà chuyên môn .
 - Ví dụ: Thư viện gồm những nghiệp vụ : mượn trả sách, thanh lý
 - Nghiệp vụ của giáo vụ : phân công giảng dạy.....
- ❖ Mức thiết kế: xử lý là thể hiện nghiệp vụ lên máy tính thành những hàm, thủ tục
- ❖ Mức lập trình : là những lệnh của một ngôn ngữ lập trình hay môi trường lập trình.
- ❖ Mức sử dụng : là những chức năng hay nút điều khiển mà người sử dụng sẽ chọn để thực hiện nghiệp vụ của mình



Thế giới thực



Thiết kế



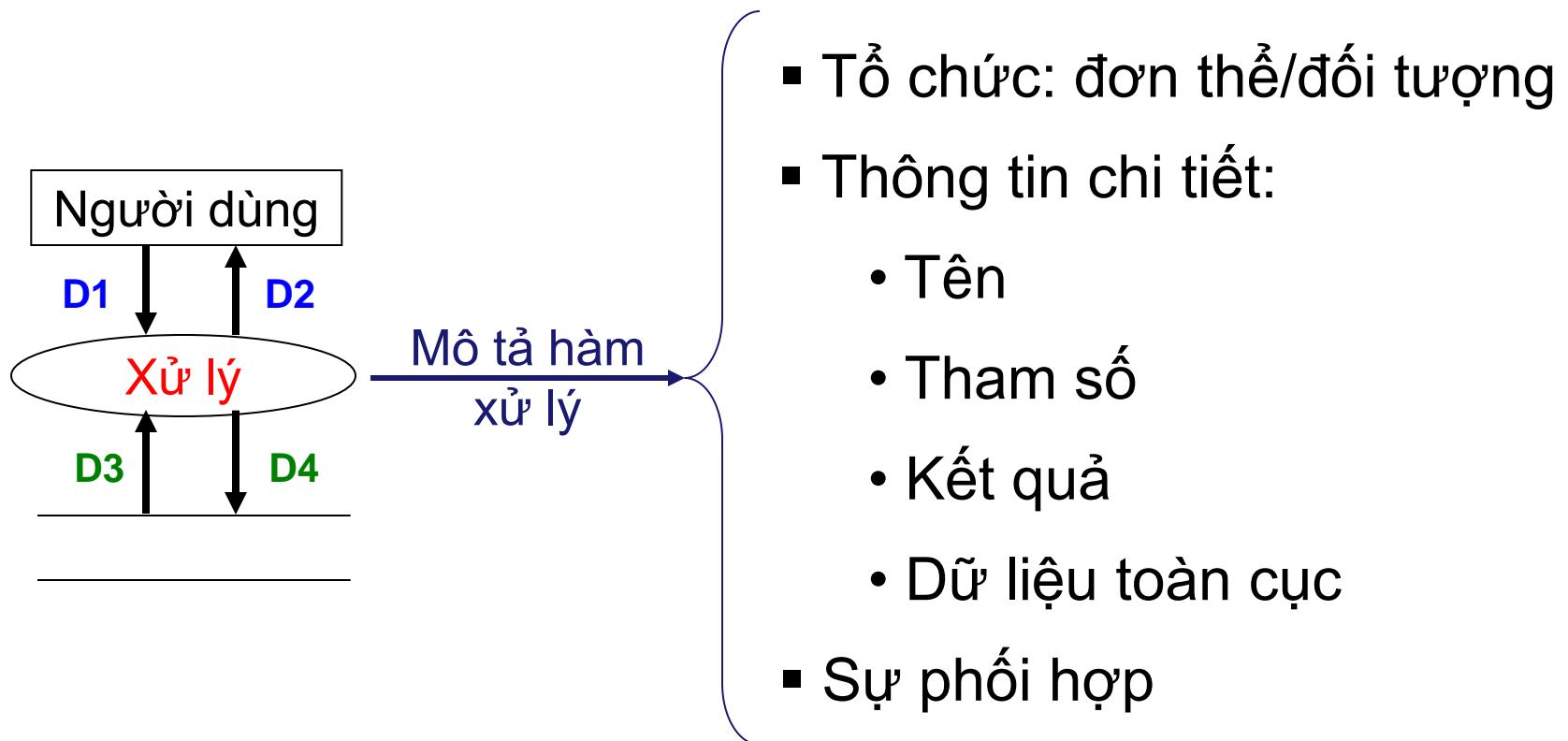
Hàm thủ tục



NN lập trình

Thiết kế xử lý

❖ Mục tiêu: Mô tả chi tiết hệ thống các hàm xử lý của phần mềm





Thiết kế xử lý

❖ Kết quả:

- Danh cách các đơn thể/đối tượng
- Danh sách các hàm của mỗi đơn thể/đối tượng
- Các sơ đồ phối hợp

❖ Yêu cầu thiết kế:

- Tính đúng đắn
- Tính dễ bảo trì
- Tính tái sử dụng
- Tính dễ mang chuyển



Thiết kế xử lý

❖ Kỹ thuật thiết kế:

- Phân rã/tích hợp
- Tham số hóa
- Đối tượng hóa

❖ Sơ đồ phối hợp:

- Mô tả cách thức phối hợp (gọi thực hiện) giữa các hàm

Thiết kế xử lý

❖ Ký hiệu:

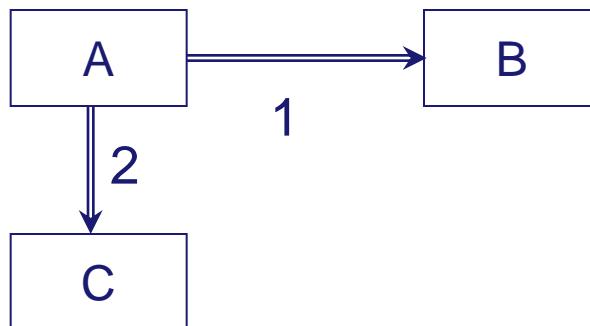
Tên hàm

Hàm xử lý

Tên đơn thể

Tên hàm

Hàm của đơn thể



A có gọi đến B, C theo thứ tự, không chuyển tham số, không nhận kết quả

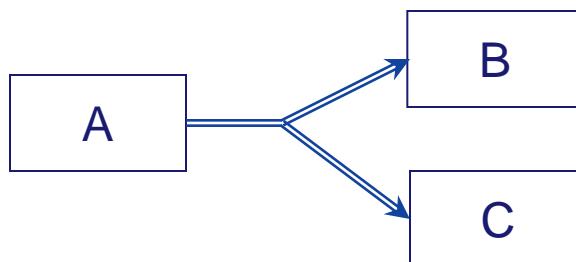


A gọi đến B có chuyển tham số, không nhận kết quả

Thiết kế xử lý



A gọi đến B không chuyển tham số, nhưng nhận kết quả



A gọi đến B hoặc C



A gọi đến B nhiều lần (ít nhất là 0 lần)



A gọi đến B nhiều lần (ít nhất là 1 lần)

Chú ý: Có n biến cố phải xử lý → Tương ứng có n sơ đồ phối hợp

Danh sách các xử lý

STT	Mã số xử lý	Tên xử lý	Diễn giải
1	XL1	Lập Phiếu sửa chữa	Lập Phiếu sửa chữa cho xe đã tiếp nhận và sửa chữa.
2	XL2	Lưu Phiếu sửa chữa	Lưu Phiếu sửa chữa vào csdl và xóa các thông tin về Phiếu sửa chữa trên màn hình.
3	XL3	Tìm Phiếu sửa chữa	Tìm Phiếu sửa chữa đã lập để xem, cập nhật thông tin...
4	XL4	Cập nhật Phiếu sửa chữa	Cập nhật thông tin cho Phiếu sửa chữa đã lập.
5	XL5	Xóa Phiếu sửa chữa	Xóa Phiếu sửa chữa đã lập khỏi csdl và cập nhật các thông tin liên quan.
6	XL6	Thoát	Kiểm tra việc lưu thông tin và thoát khỏi màn hình hiện tại.

Mô tả chi tiết từng xử lý

■ Mã số xử lý: XL1

■ Danh sách các hàm của xử lý:

STT	Hàm	Điễn giải	Ghi chú
1	getCarNumbers	Trả về danh sách các biển số xe đã tiếp nhận.	Kê cả các xe đã lập Phiếu sửa chữa rồi.
2	getFees	Trả về danh sách các loại tiền công.	Có 100 loại tiền công.
3	getDocuments	Trả về danh sách các loại vật tư phụ tùng kèm đơn giá, số lượng tồn.	Có 200 loại vật tư phụ tùng với đơn giá được niêm yết.
...			

Ví dụ

- ❖ Xét màn hình tiếp nhận học sinh mới như sau:

Tiếp nhận học sinh				
Họ tên			Nam	<input checked="" type="checkbox"/>
Ngày sinh			Lớp	<input type="button" value="▼"/>
Địa chỉ				
<input type="button" value="Ghi"/>				
Danh sách học sinh đã tiếp nhận				
STT	Mã HS	Tên HS	Giới tính	Ngày sinh
...

Ví dụ

❖ Hãy:

- Mô tả các biến cố
- Lập danh sách các hàm xử lý
- Lập các sơ đồ phối hợp (Khi có ít nhất hai hàm tham gia trong cùng một biến cố)



Ví dụ - Mô tả biến cố

❖ Biến cố 0:

- Khởi động màn hình

❖ Biến cố 1:

- Kiểm tra tuổi học sinh hợp lệ (tuổi từ 15 đến 20)

❖ Biến cố 2:

- Khi chọn một lớp học trên combobox

❖ Biến cố 3:

- Kiểm tra dữ liệu hợp lệ và ghi

Ví dụ - Danh sách các biến cố

BC	Điều kiện kích hoạt	Xử lý	Ghi chú
0	Khởi động màn hình	<ul style="list-style-type: none">- Đọc danh sách lớp, danh sách học sinh, tham số- Xuất danh sách lớp, danh sách học sinh, hồ sơ học sinh mới	
1	Kết thúc nhập ngày sinh	<ul style="list-style-type: none">- Kiểm tra ngày sinh hợp lệ và xuất thông báo lỗi nếu không hợp lệ	Tuổi theo qui định 15→20
2	Kết thúc chọn lớp	<ul style="list-style-type: none">- Ghi nhận vị trí của lớp được chọn trong danh sách lớp	Chuẩn bị khi ghi hồ sơ
3	Nút nút ghi	<ul style="list-style-type: none">- Kiểm tra hồ sơ hợp lệ- Nếu hợp lệ thì nhập hồ sơ học sinh và ghi hồ sơ học sinh. Xuất thông báo	Mã và tên phải khác rỗng

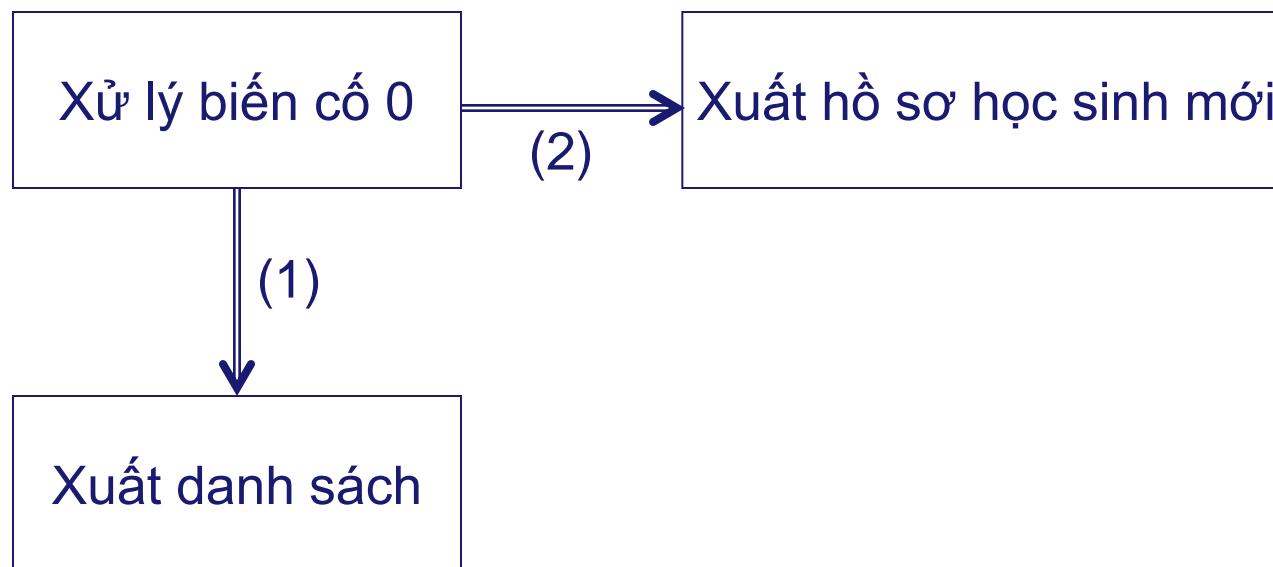


Ví dụ - Danh sách các hàm xử lý

STT	Tên hàm	Tham số	Kết quả	Ý nghĩa	Ghi chú
...					

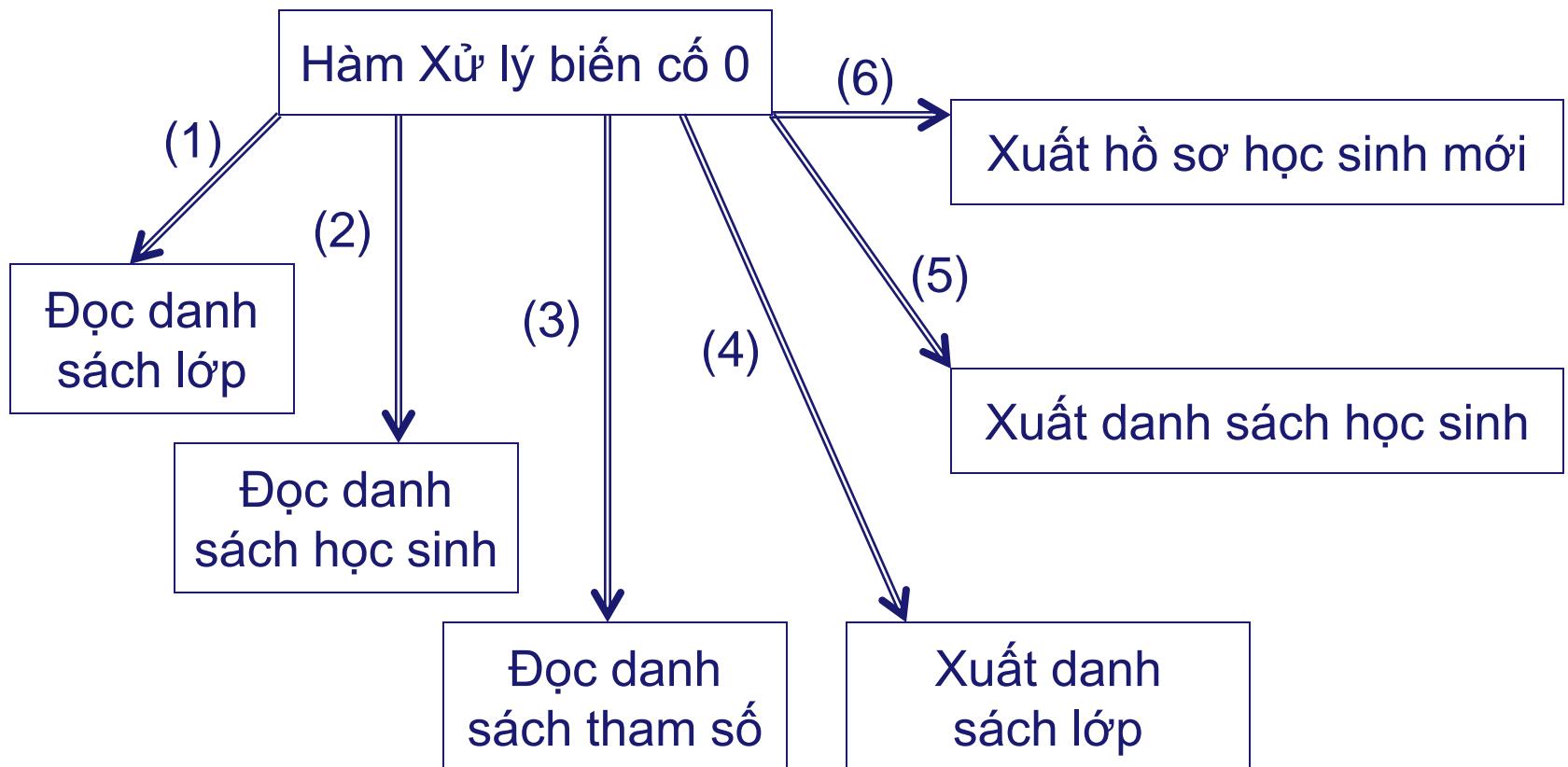
Ví dụ - Lập sơ đồ phối hợp

❖ Xử lý biến cỗ 0

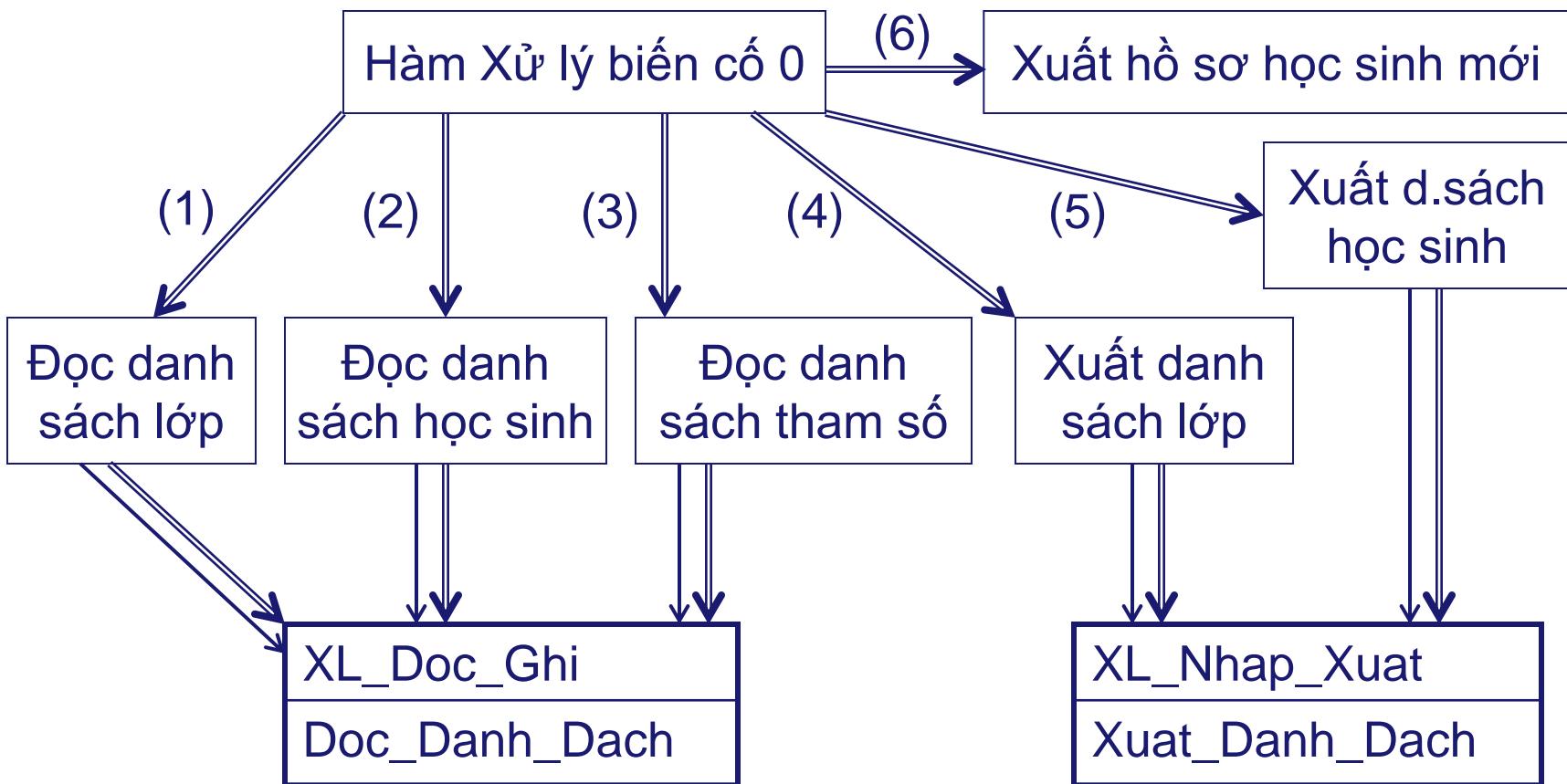


Ví dụ - Lập sơ đồ phối hợp

❖ Xử lý biến cố 0: Phân rã hàm



Ví dụ - Lập sơ đồ phối hợp





Ví dụ - Lập sơ đồ phối hợp

❖ Xử lý biến cố 1

- Dùng 1 hàm

❖ Xử lý biến cố 2

- ?

❖ Xử lý biến cố 3

- ?

Ví dụ

vđ1: Tinhkiemtientienphat (MaDG, MaS, ngayT)
{
 -Mở bảng Tham Số đọc Đơn Giá Phật: DonGP= DOCTS (2) .
 -Mở bảng Mượn Trả (MT) .
 -Tìm MT.MaDG = MaDG && MT.MaS = MaS && MT.ngayT = ngayT
 -Return DonGP * (ngayT - MT.ngayM)
}

vđ2: float Tinhkiemtientienphat (int songaytre)
{
 -int DonGP = DoctS (2)
 -return (DonGP * songaytre)
}

vđ3: float Tinhkiemtientienphat (songaytre, DonGP)
{
 -return (songaytre * DonGP)
}

STT	Tên xử lý	Điều kiện gọi thực hiện	Ý nghĩa	Ghi chú
		Xuất phát từ người dùng		
		Gọi bởi xử lý khác		



❖ Chi tiết các xử lý quan trọng:

- Tên xử lý:
- Ý nghĩa:
- Điều kiện thực hiện
- Sơ đồ luồng dữ liệu
- Thuật giải
 - Bước 1:
 - Bước 2:
 - ...
 - Bước n:

Bài tập

1. Lập sơ đồ phối hợp cho các biến cố còn lại trong ví dụ trên.
2. Đánh giá các sơ đồ phối hợp dựa trên các tính chất
 - Tính đúng đắn
 - Tính tái sử dụng
 - Tính dễ bảo trì
 - Tính dễ mang chuyền
 - ...

