

THIẾT KẾ GIAO DIỆN

I. Những tính chất cơ bản mà giao diện cần có:

1. **Tính quen thuộc:** Chúng ta phải đứng trên lập trường của người dùng để thiết kế giao diện sao cho gần gũi nhất đối với người sử dụng. Ta cần xác định người dùng làm việc trong môi trường nào, chuyên môn nào để sử dụng trên giao diện các thuật ngữ, các khái niệm phù hợp với môi trường làm việc cũng như chuyên môn ấy.
2. **Tính nhất quán:** Tính nhất quán nghĩa là các lệnh, các menu, các thông số hệ thống... thuộc những chức năng, những phân hệ khác nhau phải được thiết kế theo một dạng thức thống nhất với nhau. Nhờ vậy, người dùng có thể tiết kiệm thời gian và sức lực trong việc làm quen với hệ thống (vì những gì họ học được ở chức năng này có thể được ứng dụng ở cả những chức năng khác). Xa hơn nữa, hiện nay đã có rất nhiều phần mềm được dùng phổ biến đến mức cách thức chúng tiếp xúc với người dùng đã trở nên tiêu chuẩn, quen thuộc với cộng đồng. Chúng ta nên tham khảo những phần mềm này để hệ thống của chúng ta nhanh chóng trở nên thân thiết với người dùng.
3. **Tính hạn chế gây ngạc nhiên:** Chúng ta phải đặt mình vào vị trí người dùng để xem xét người dùng mong đợi hệ thống ứng xử với họ ra sao. Từ đó chúng ta thiết kế giao diện đáp ứng với người dùng theo đúng cách thức ấy, nhờ vậy người dùng không bị bất ngờ (khi hệ thống ứng xử khác hẳn với những gì họ mong đợi).
4. **Tính an toàn:** Người dùng có thể gây sai sót trong khi sử dụng hệ thống. Giao diện có thể được thiết kế sao cho các thao tác không hợp lệ bị từ chối (Ví dụ: kiểm tra ngày sinh của sinh viên được nhập vào và báo lỗi khi ngày không hợp lệ). Ngoài ra giao diện cũng nên yêu cầu người dùng xác nhận lại một lần trước khi đáp ứng các lệnh nguy hiểm của người dùng (Ví dụ: Xác nhận việc xóa vĩnh viễn một dòng dữ liệu). Sau cùng, giao diện phải cung cấp chức năng Undo với số lần Undo chấp nhận được.
5. **Tính tận tụy:** Giao diện phải “năng nổ” trong việc đưa ra các trợ giúp ngữ cảnh (tooltip, help facility, brief discription...). Ngoài ra, các thông báo, đặc biệt là các câu báo lỗi phải rõ ràng, đầy đủ ngữ nghĩa và kèm theo một số tùy chọn cho người dùng khắc phục vấn đề (Ví dụ: các nút Retry, More information, hay các link đến những trang trợ giúp liên quan).
6. **Tính đa thích ứng:** Có những người dùng đã đạt trình độ lão luyện, đối với họ các nhắc nhở thường xuyên và những tooltip dài dòng là những điều phiền toái. Lại có những người dùng rất non tay và rất cần có những nhắc nhở đầy đủ ngữ nghĩa và những tooltip. Lại có những người dùng thích dùng chuột để điều khiển giao diện, ngược lại có những người thích dùng bàn phím. Giao diện phải có khả năng chuyển đổi qua lại giữa các chế độ (mode) và cung cấp nhiều phím tắt để thích ứng với hầu hết phong cách người dùng.

II. Giao tiếp với người dùng:

Hai nhiệm vụ của giao diện là nhận thông tin từ người dùng và xuất thông tin ra cho người dùng.

Nhận thông tin từ người dùng: Có 5 kiểu thiết kế cách thức mà giao diện nhận thông tin từ người dùng, đó là:

1. **Dùng thao tác trực tiếp:** Gọi nôm na là kéo và thả (drag and drop). Ví dụ: kéo một record cần xóa vào biểu tượng thùng rác. Cách này có ưu điểm là dễ dàng cho người dùng nhưng có khuyết điểm là khó cài đặt.
2. **Dùng menu:** Là cách rất phổ biến, hiện diện trong hầu hết mọi phần mềm, giúp hệ thống biết người dùng muốn làm gì. Cách này có ưu điểm là dễ dùng và hạn chế thao tác không hợp lệ nhưng có khuyết điểm là thao tác chậm và gây bối rối nếu menu lồng nhiều cấp.

3. **Dùng form nhập liệu:** Là cách rất thông dụng đối với các ứng dụng quản lý. Cách này có ưu điểm là dễ dùng và từ chối được dữ liệu không hợp lệ nhưng có khuyết điểm là cần nhiều không gian màn hình.
4. **Dùng dòng lệnh:** Chỉ thích hợp với những người dùng thành thạo. Cách này có ưu điểm là linh hoạt nhưng có khuyết điểm là khó học và khó hạn chế lỗi người dùng.
5. **Dùng ngôn ngữ tự nhiên:** Cách lý tưởng được mọi nhà phát triển muốn hướng đến.

Xuất thông tin ra cho người dùng : Khi chọn lựa hình thức xuất thông tin ra giao diện cho người dùng, chúng ta cần cân nhắc một số vấn đề sau:

1. Nếu thông tin là dạng chuỗi hoặc dạng số đòi hỏi tính chính xác cao thì phải xuất ra dạng văn bản.
2. Nếu thông tin thay đổi thường xuyên theo thời gian thì nên xuất ra dạng đồ họa. Ví dụ: Lượng hàng trong kho có thể biểu hiện bằng dạng bar: hàng vừa nhập vào thì bar dài ra, nhưng không vượt quá sức chứa của kho biểu hiện bằng vạch Max, hàng được xuất bán bao nhiêu thì bar ngắn đi tương ứng nhưng không thấp hơn lượng tồn tối thiểu biểu hiện bằng vạch Min.
3. Nếu thông tin vừa đòi hỏi tính chính xác, vừa thường xuyên thay đổi thì nên phối hợp cả hai cách thể hiện trên đây.
4. Khi người dùng cần tương tác với thông tin thì thông tin buộc phải biểu diễn bằng dạng đồ họa. Ví dụ: Lớp học biểu diễn bằng các box và sinh viên biểu diễn bằng các item trong từng box (ứng với từng lớp), thể hiện này đáp ứng cho thao tác xếp lớp của người dùng (kéo item-sinh viên từ box-lớp này thả vào box-lớp khác).
5. Khi cần so sánh thông tin thì nên hiển thị thông tin dạng đồ họa. Ví dụ: Biểu hiện mối tương quan giữa chi phí tiếp thị và doanh thu bán hàng bằng hai đồ thị trong cùng một hệ tọa độ để người dùng tiện so sánh một cách trực quan.

III. Thiết kế thông báo trên giao diện:

Khi thiết kế các thông báo, ta cần chú ý một số điểm sau:

1. Người dùng thành thạo chỉ cần các thông báo ngắn gọn, mang tính nhắc nhở. Trong khi đó, người dùng không thành thạo cần các thông báo cụ thể tường tận và (nếu được) kèm thêm một hướng dẫn ngắn gọn về những gì họ phải làm. Hệ thống nên cho phép chuyển đổi qua lại giữa hai chế độ thông báo này.
2. Thông báo cho các đối tượng người dùng khác nhau thì sử dụng ngôn ngữ khác nhau. Ví dụ: Những thông báo thuộc phân hệ dành cho ban lãnh đạo doanh nghiệp thì sử dụng ngôn ngữ mang tính vĩ mô, quản trị. Những thông báo thuộc phân hệ dành cho nhân viên thì sử dụng ngôn ngữ mang tính tác nghiệp.
Nội dung thông báo phải trung lập, khách quan hoàn toàn. Trong bối cảnh Việt Nam chúng ta, đại từ nhân xưng rất đa dạng, do đó thông báo tránh sử dụng đại từ nhân xưng. Ví dụ: Nên xuất thông báo là “Ngày sinh nhân viên không hợp lệ, xin vui lòng nhập lại!”. Không nên xuất thông báo là “**Bạn** nhập ngày sinh nhân viên không hợp lệ, xin vui lòng nhập lại!”
4. Hoạt động bên trong hệ thống là một blackbox với người dùng, các thông báo không được chứa các thuật ngữ chuyên môn của người lập trình. Ví dụ: Các thông báo sau đây là không nên dùng “**ID** sinh viên không hợp lệ!”, “**Lỗi hệ thống, mã lỗi #27**”,...
5. Nội dung thông báo phải cụ thể, không được chung chung. Ví dụ: Các thông báo sau đây là quá chung chung “Nhập liệu sai!”, “Dữ liệu không hợp lệ!”,...

IV. Sơ đồ liên kết các màn hình:

Mỗi ứng dụng đều phải có một giao diện xuất hiện đầu tiên khi hệ thống được khởi động, chúng ta gọi đó là màn hình chính (hoặc home page với ứng dụng web). Từ giao diện chính, người dùng sẽ được dẫn dắt đến những chức năng khác nhau của hệ thống.

Trước tiên, chúng ta phải có danh sách tất cả các chức năng mà hệ thống cần có.

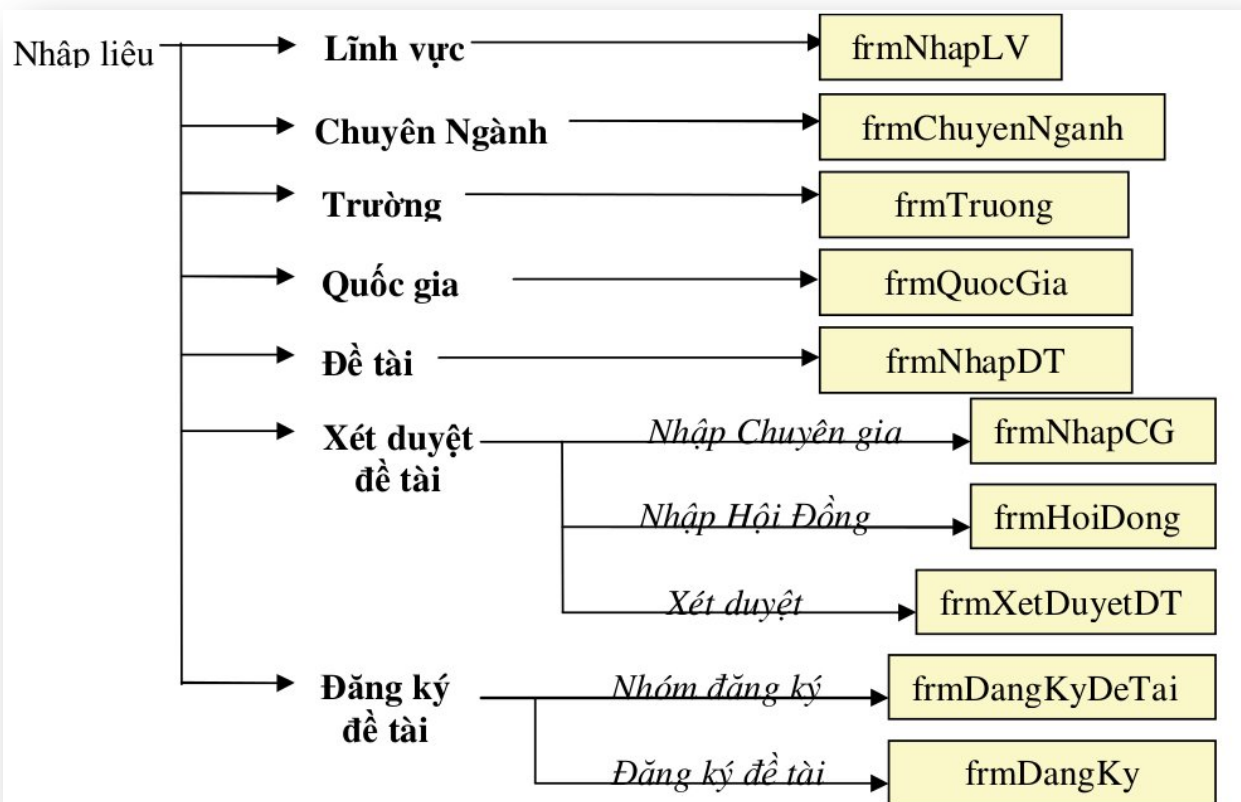
Kế tiếp, với mỗi chức năng ta xác định tất cả những giao diện cần có cho nó.

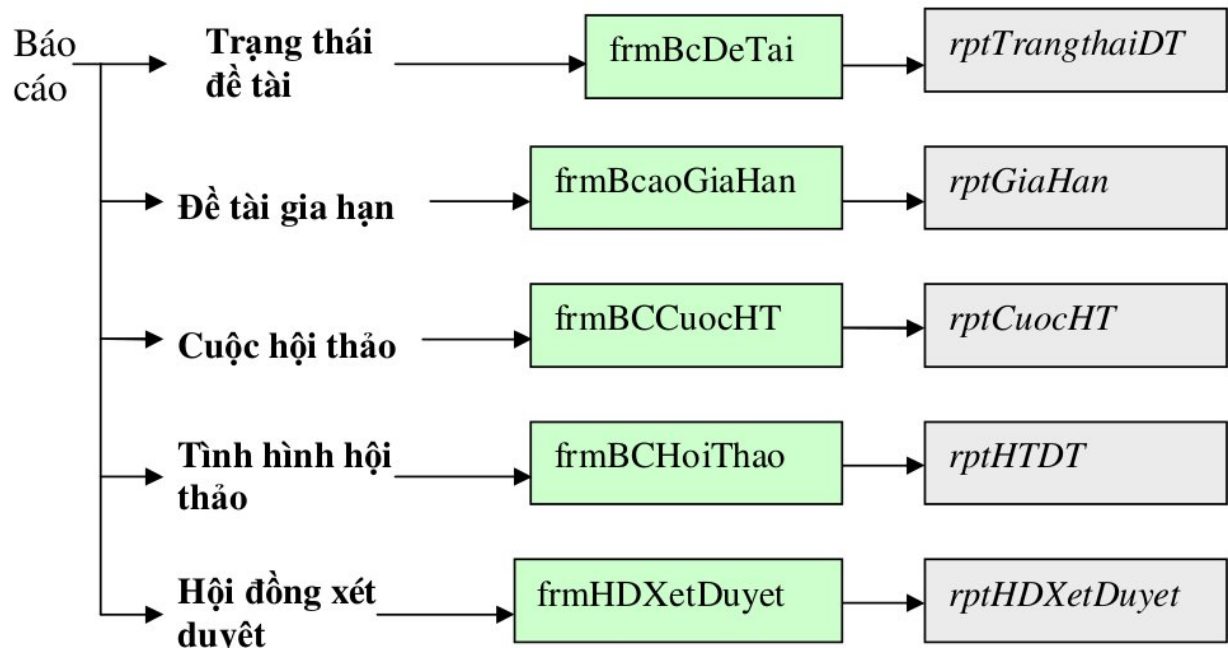
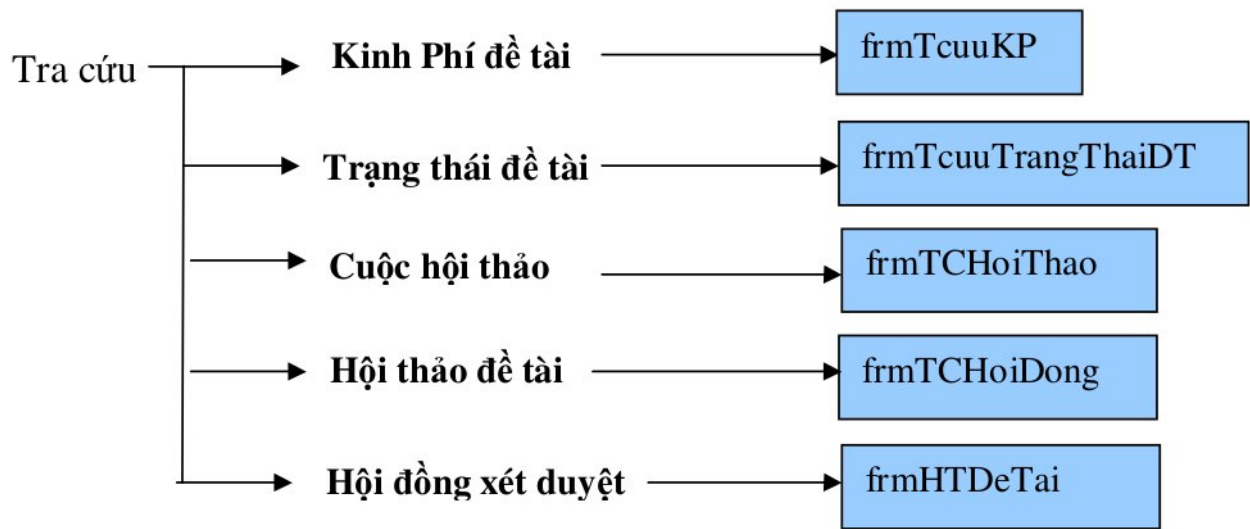
Trong số các giao diện thuộc phạm vi của một chức năng, phải chọn ra 1 giao diện chính sẽ được load đầu tiên khi người dùng chọn chức năng ấy. Bắt đầu từ giao diện này ta xác định những liên kết giữa các giao diện còn lại của chức năng đang xét.

Các chức năng được tổ chức thành từng nhóm. Vì vậy các giao diện theo đó cũng được tổ chức thành từng nhóm. Việc gom nhóm này phải theo một tiêu chí nhất định. Trong phạm vi tài liệu này xin giới thiệu 3 tiêu chí gom nhóm thường được dùng:

1. Gom nhóm theo mục đích của giao diện:

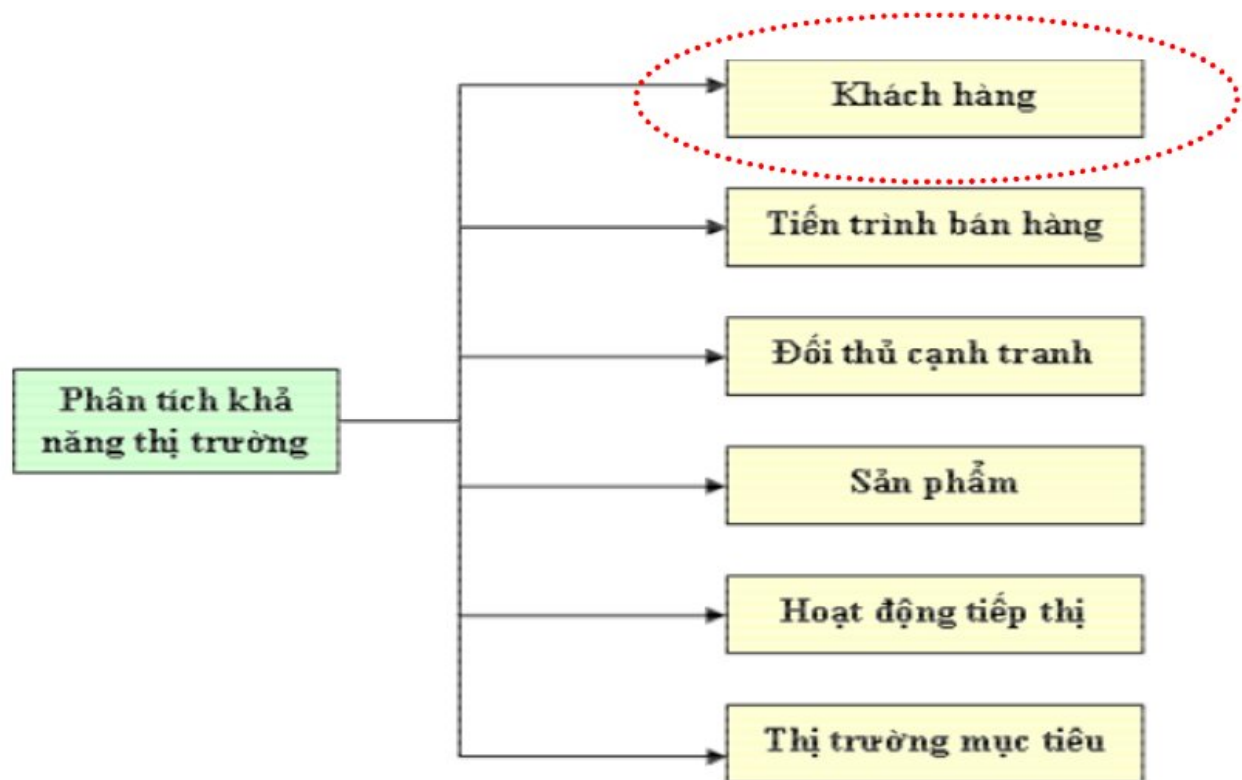
Theo tiêu chí này, giao diện được phân thành các nhóm theo mục đích của giao diện. Ví dụ: Nhóm giao diện nhập liệu, nhóm giao diện tra cứu, nhóm giao diện báo cáo,... Hình dưới đây là một ví dụ trong bối cảnh ứng dụng quản lý đề tài tốt nghiệp, chúng ta thấy các giao diện được gom nhóm rất rõ rệt theo mục đích của chúng.



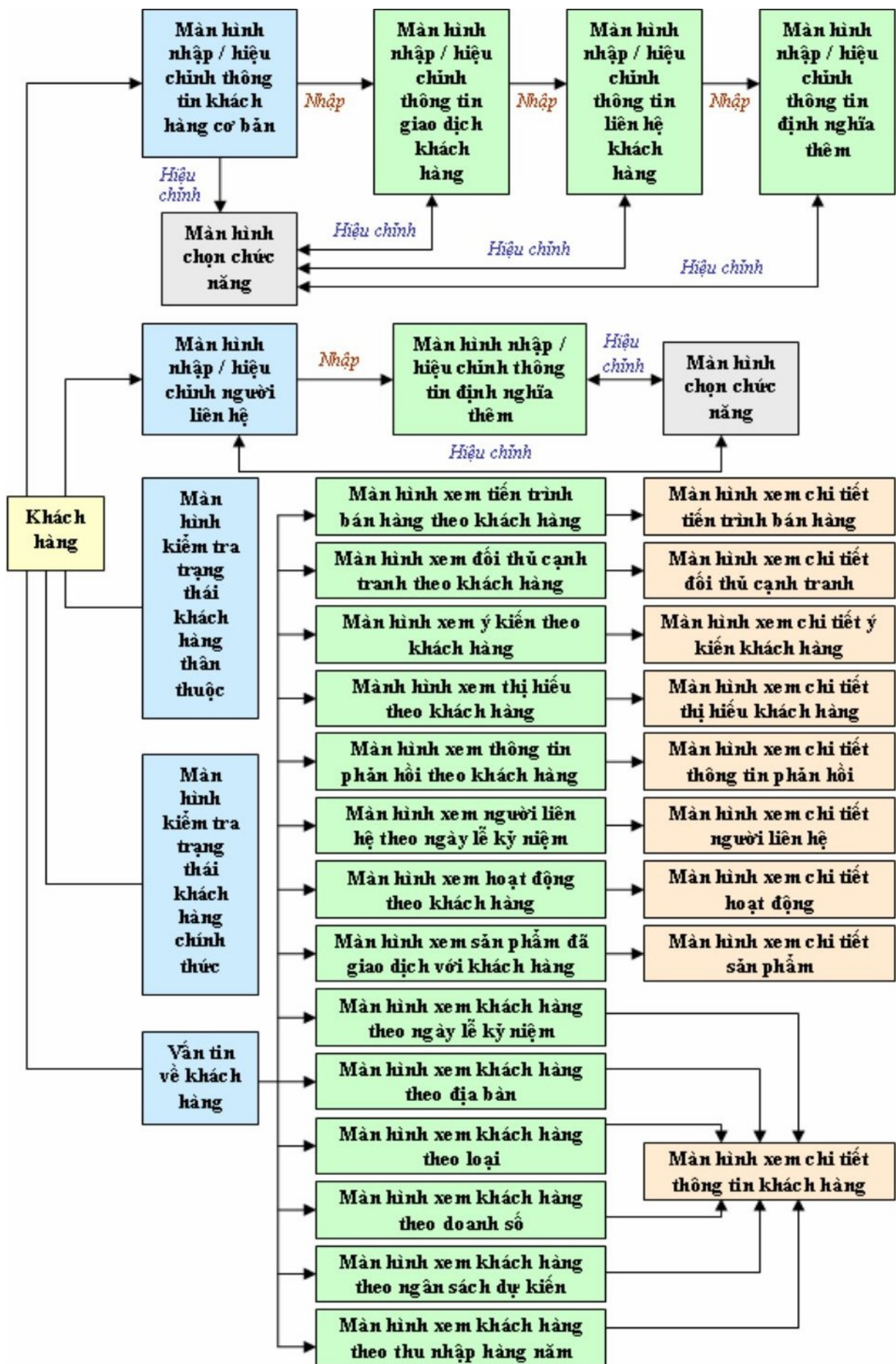


2. Gom nhóm theo loại thực thể được quản lý:

Theo tiêu chí này, các giao diện xoay quanh việc quản lý cùng một loại thực thể sẽ được gom vào cùng một nhóm. Hình dưới đây là một ví dụ trong bối cảnh ứng dụng phân tích khả năng thị trường:

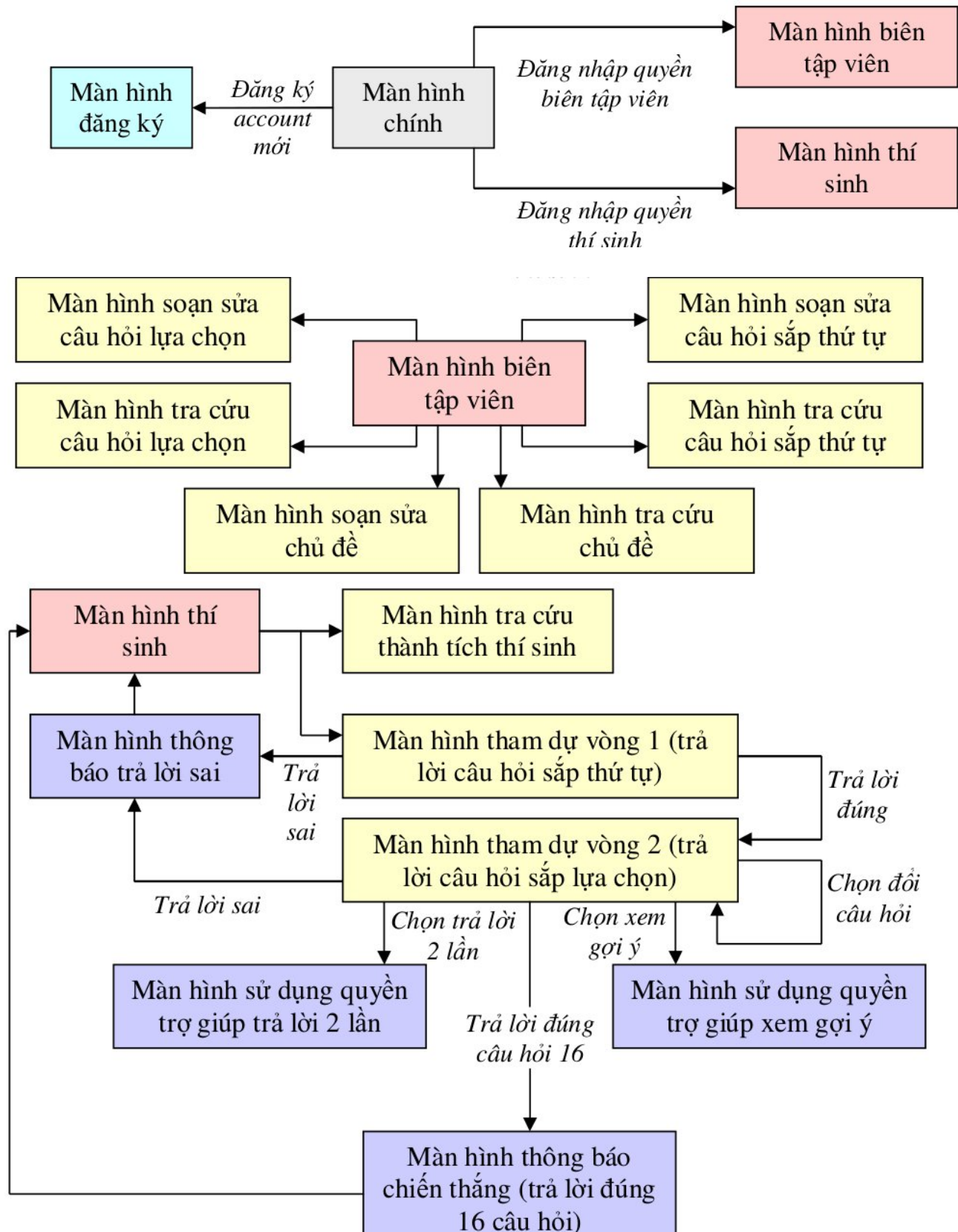


Chúng ta thấy rằng ứng dụng này quan tâm 6 loại thực thể: Khách hàng, tiến trình bán hàng, đối thủ cạnh tranh, sản phẩm, hoạt động tiếp thị, thị trường mục tiêu. Do đó sẽ có 6 nhóm giao diện tương ứng. Sau đây là ví dụ về sơ đồ liên kết nhóm giao diện quản lý loại thực thể khách hàng (các nhóm kia cũng tương tự như vậy).



3. Gom nhóm theo đối tượng người dùng:

Theo tiêu chí này, các giao diện phục vụ cho cùng một đối tượng người dùng sẽ được gom vào cùng một nhóm. Hình dưới đây là một ví dụ về ứng dụng quản lý trò chơi truyền hình Rồng Vàng. Ứng dụng này phục vụ hai đối tượng người dùng là Biên tập viên chương trình và Thí sinh. Do đó các giao diện cũng được gom thành hai nhóm rõ rệt.



V. Đặc tả giao diện:

Mỗi giao diện chúng ta đặc tả 2 nội dung:

1. Tên màn hình, chức năng, giao diện màn hình:

Chúng ta có thể sử dụng tất cả các công cụ từ đơn giản như Winword, Paint... đến phức tạp như Photoshop, Corel... để hỗ trợ việc trình bày từng giao diện.

Giảng viên

TRƯỜNG ANH NGỮ A & Z

GIẢNG VIÊN

THÔNG TIN BẮT BUỘC

Họ và tên Giảng viên : CAO ĐĂNG QUỐC TIẾN

Ngày sinh : Ngày 11 Tháng 11 Năm 1970 Phái : NAM

Địa chỉ : 643 TRẦN HƯNG ĐẠO - P.1 - Q.5 - TP. HỒ CHÍ MINH

THÔNG TIN KHÔNG BẮT BUỘC

Điện thoại : 8765295 E-mail : cdquoctien@hcm.vnn.vn

THÔNG TIN BẰNG CẤP (BẮT BUỘC)

Các loại bằng cấp :

Mã	Tên bằng cấp	Nơi cấp	Thời hạn
TP	ToeFl	ETS	2 năm
IL	ILTS	Arco	2 năm
C4	CAF	Cambridge	4ever

Bảng cấp của Giảng viên :

Tên bằng cấp	Ngày cấp	Chú thích
ToeFl	04/10/2003	630đ
CPE	06/08/2002	Flyer

THÊM MỚI **XÓA** **LƯU MỚI** **CẬP NHẬT** **LÀM LẠI** **THOÁT**

Ngày cấp : N 04 / 10 / Nm 2003

Chú thích : 630đ

ĐỒNG Ý **HỦY**

2. Mô tả các đối tượng trên màn hình:

Mỗi đối tượng trên màn hình ta đặc tả *tên đối tượng*, *chức năng*, *loại đối tượng*, *ràng buộc*.

VI. Một số lời khuyên khi thiết kế giao diện:

Khi thiết kế giao diện, ta thiết kế các control như label, button, textbox,... sao cho khoảng trống trong form được giảm tối đa; các control phải cân đối về độ rộng, hẹp, font chữ... Ví dụ màn hình sau đây tận dụng triệt để không gian quý báu của màn hình.

Nhập / Hiệu chỉnh thông tin doanh nghiệp

Thông tin liên lạc:

Tên doanh nghiệp: Công ty TNHH May mặc và Thêu trang Trâm Anh

Địa chỉ 1: 35 Mạc Đình Chi - P. 3 - Q.1 Tỉnh/Quê: 130 Hùng Vương - P. 4 - Q. 5

Địa chỉ 3: 41 Mạc Lương - P. 4 - Q. 5 Fax: 12345678

Điện thoại 1: 87654321 Điện thoại 2: 87654321 Điện thoại 3: 87654321

Email: zxcvbnm@yashco.com Số mã vùng: 500 00

Website: zxcvbnm_yashco.com.vn Doanh thu: 50.000.000.000 00

Tên tiếp xúc: Mr. P Chức vụ: Giám đốc

Quê gốc: VN Nơi sinh: VN

Tỉnh/Quê: TCCD Hệ Chk Minh: Hệ Chk Minh

Quản lý, kinh: D-0013 Email: T. Anh

Lĩnh vực kinh doanh: VN Máy móc: Máy móc

Loại máy kinh doanh: JHCC04 Từ năm: 2000 đến nay

Thông tin quản lý vận hành:

Sản phẩm	Từ	Đến	Chức năng	Từ	Đến
Sản phẩm 1	Tu 07:00	Đến 11:00	Chức năng 1	Tu 12:00	Đến 17:00
Sản phẩm 2	Tu 07:00	Đến 11:00	Chức năng 2	Tu 12:00	Đến 17:00
Sản phẩm 3	Tu 07:00	Đến 11:00	Chức năng 3	Tu 12:00	Đến 17:00
Sản phẩm 4	Tu 07:00	Đến 11:00	Chức năng 4	Tu 12:00	Đến 17:00
Sản phẩm 5	Tu 07:00	Đến 11:00	Chức năng 5	Tu 12:00	Đến 17:00
Sản phẩm 6	Tu 07:00	Đến 11:00	Chức năng 6	Tu 12:00	Đến 17:00

Người cấp phép: Nguyễn Văn Minh Ngày cấp: 15/06/2000

Đồng ý **Không**

Khi thiết kế giao diện, những thông tin có liên quan với nhau nên được gom vào cùng một frame và tốt hơn nữa là gắn nhãn cho frame nhằm tăng cường ngữ nghĩa cho giao diện. Ví dụ giao diện dưới đây cho thấy thông tin về đối thủ cạnh tranh chia làm 5 nhóm là thông tin cơ bản, thông tin liên lạc, thông tin điểm mạnh, thông tin điểm yếu và nhóm thông tin không được gắn nhãn dùng để kiểm soát (người cập nhật và ngày cập nhật).

Nhập / Hiệu chỉnh đối thủ cạnh tranh			
Thông tin cơ bản			
Mã đối thủ	DT0003	Tên đối thủ	Công ty Cổ phần may mặc YANSHI VINA
Lĩnh vực kinh doanh	LV0001	May mặc	
Loại hình kinh doanh	LH0002	Cổ phần nhà nước	
Doanh Thu	300,000,000,000.00	VND	Số nhân viên 1,000.00
Ghi chú	Công ty chuyên sản xuất các loại đồng phục văn phòng		
Thông tin liên lạc			
Địa chỉ	15 Khánh Hội -P9	Điện thoại	089326457
Quốc gia	QG0001	Việt Nam	
Thành phố	TT0001	Hồ Chí Minh	Quận huyện QH0012 Quận 4
Fax	089526342	E-mail	yanshivina@freemail.com
Thông tin điểm mạnh		Thông tin điểm yếu	
STT	Mã điểm mạnh	Tên điểm mạnh	
0001	DM0001	Số lượng nhân viên lớn	Nhiề
0002	DM0002	Số vốn kinh doanh rất	Ngu
0003			
0004			
0005			
0006			
STT	Mã điểm yếu	Tên điểm yếu	
0001	DY0004	Chậm trong việc cập nh	Nắm l
0002	DY0001	Nguồn vốn đầu tư cho	Nguồ
0003			
0004			
0005			
0006			
Người cập nhật		Ngày cập nhật	
Người dùng mặc định		13/06/2005	
Đồng ý		Xóa	
Thoát			

Có thể sử dụng màu sắc thay đổi để làm nổi bật nội dung. Như trong hình sau, màu sắc thay đổi trong từng dòng dữ liệu cho biết đề tài đó đã được xét duyệt hay chưa hay đã hoàn tất. Những màu sắc có ý nghĩa khác nhau phải thực sự tương phản nhau. Khi người dùng làm việc cường độ cao, sự căng thẳng có thể ảnh hưởng thị giác gây nhầm lẫn giữa những màu tựa nhau dẫn đến lộn lộn về ý nghĩa.

Phân công Hội đồng xét duyệt Đề tài

Mã Hội đồng: Ngày bắt đầu xét duyệt: Tên Hội Đồng: Số chuyên gia:

Mã Đề Tài	Tên Đề Tài	Lĩnh Vực
DT007	Đột biến Tiến...	Công nghệ...
DT008	Cơ sở thực...	Công nghệ...
DT010	Quá trình tiến...	Khoa học vũ trụ
DT011	Tính bất toàn...	Khoa học vũ trụ
DT013	Tính liên tục...	Khoa học vũ trụ
DT015	Du lịch và di...	Văn Hóa

Mã đề tài	Tên Đề tài	Ngày Bắt Đầu...	Ngày kết thúc...
DT006	Khai thác...	8/1/2004...	
DT014	Kỹ thuật...	7/1/2001...	7/1/2001...
DT017	Thừa kế và di...	7/1/2001...	7/31/2004...

Nhập kết quả xét duyệt

Thoát

Khi sử dụng màu sắc trên giao diện cần lưu ý: Không nên sử dụng quá năm màu trong một màn hình. Nếu giao diện sử dụng quá nhiều màu sắc, người dùng sẽ có cảm giác choáng ngợp và rối mắt. Màu sắc phải được dùng một cách nhất quán trên tất cả các giao diện của hệ thống theo một tiêu chí nhất định, tránh sử dụng màu sắc chỉ để trang trí. Ta cần tránh dùng màu chói sẽ gây cảm giác bức bối cho người dùng.

Khi thiết kế giao diện nhập liệu cho một dòng dữ liệu bảng A mà có tham chiếu (khóa ngoại) đến một dòng dữ liệu bảng B, ta nên cung cấp liên kết đến giao diện nhập liệu cho bảng B (phòng khi dòng dữ liệu tương ứng ở bảng B chưa tồn tại).