

**SỞ GIÁO DỤC & ĐÀO TẠO HÒA BÌNH
TRƯỜNG TRUNG HỌC KINH TẾ KỸ THUẬT HÒA BÌNH**

Giáo viên biên soạn: Nguyễn Trần Phương

**PHÂN TÍCH THIẾT KẾ HỆ THỐNG THÔNG TIN
Lưu hành nội bộ**

HÒA BÌNH – NĂM 2009

PHÂN TÍCH & THIẾT KẾ HỆ THỐNG THÔNG TIN

MỤC LỤC

Mục lục	1
Lời nói đầu	2
Chương 1: Những vấn đề chung	4
1.1 Hệ thống thông tin	4
1.1.1 Hệ thống thông tin	4
1.1.2 Các tính năng của hệ thống thông tin	6
1.2 Phân tích và thiết kế hệ thống thông tin	6
1.2.1 Các quan điểm phân tích	7
1.2.2 Các phương pháp phân tích	7
1.3. Các giai đoạn trong quá trình xây dựng hệ thống thông tin	12
1.3.1 Lập kế hoạch, xác định yêu cầu	12
1.3.2 Phân tích	13
1.3.3 Thiết kế	17
1.3.4 Thi công	18
1.3.5 Thử nghiệm	19
1.3.6 Bảo trì	19
1.4. Các mức trừu tượng hóa	23
1.4.1 Mức quan niệm	24
1.4.2 Mức tổ chức	24
1.4.3 Mức tác nghiệp	25
Chương 2: Nghiên cứu & xử lý kết quả nghiên cứu	26
2.1 Nghiên cứu hiện trạng	26
2.1.1 Một số khái niệm	26
2.1.2 Các phương pháp nghiên cứu	27
2.1.2.1 Phương pháp quan sát	29
2.1.2.2 Phương pháp điều tra	31
2.1.2.3 Phương pháp phỏng vấn	36
2.2 Xử lý kết quả nghiên cứu	40
2.2.1 Tổng hợp các kết quả điều tra	40
2.2.2 Tổng hợp dữ liệu	41
2.2.3 Rà soát lại tài liệu, rút ra yếu tố cần thiết	42
2.2.3.1 Mục đích của hệ thống sẽ được tin học hóa	42
2.2.3.2 Các quy tắc ràng buộc	42
2.2.3.3 Làm rõ các nhiệm vụ	42
2.2.4 Tổng hợp các yêu cầu	42
2.3 Tài liệu và sơ đồ hóa kết quả nghiên cứu	50
Chương 3: Mô hình hoá quan niệm	56
3.1 Mô hình thực thể	56
3.1.1 Khái niệm	56
3.1.2 Thuộc tính	56
3.1.3 Ràng buộc toàn vẹn trên thực thể	57
3.2 Mô hình quan niệm về dữ liệu	57
3.3. Mối quan hệ giữa các thực thể	63
3.3.1 Xây dựng mô hình ER	65
3.3.2 Bản số	69

3.3.3 Các bài tập về xác định bản số	73
Chương 4: Mức tổ chức và vật lý	76
4.1 Mô hình tổ chức về xử lý	76
4.1.1 Các yếu tố tổ chức	76
4.1.2 Các quy tắc tổ chức	78
4.1.3 Khái niệm pha	78
4.2 Mô hình cơ sở dữ liệu	79
4.2.1 Mô hình phân cấp	79
4.2.2 Mô hình mạng	80
4.2.3 Mô hình quan hệ	80
4.2.4 Mô hình CSDL hướng đối tượng	85
4.3 Chuẩn hóa trên mô hình CSDL	88
4.4 Mô hình lô gic	93
4.5 Chuyển từ mô hình quan niệm sang mô hình lô gic	100
4.5.1 Chuyển sang mô hình CSDL mạng	100
4.5.2 Chuyển sang mô hình CSDL quan hệ	100
4.6 Mức vật lý	101
4.6.1 Mô hình vật lý về dữ liệu	101
4.6.2 Các mô đun xử lý	105
4.6.3 Phân dã mô đun	106
4.6.4 Phân dã chức năng	106
4.7 Chuyển từ mức lô gic sang mức vật lý	108
4.7.1 Sử dụng một ngôn ngữ lập trình	108
4.7.2 Sử dụng một hệ quản trị cơ sở dữ liệu	109
4.7.3 Giới thiệu rational rose 2003	109
Chương 5: Quản lý các giai đoạn xây dựng HTTT	117
5.1 Quản lý giai đoạn lập kế hoạch, xác định yêu cầu	117
5.1.1 Xây dựng nhóm	119
5.1.2 Bố trí nhân sự, phân công nhiệm vụ	122
5.1.3 Lập kế hoạch tiếp cận vấn đề	126
5.1.4 Xây dựng các mối quan hệ	128
5.1.5 Ký kết các thỏa thuận tài chính, kỹ thuật	129
5.1.6 Xây dựng dự toán cho các giai đoạn	131
5.2 Quản lý giai đoạn phân tích	133
5.2.1 Tiếp cận hiện trạng	133
5.2.2 Thu thập các tài liệu có liên quan đến hệ thống sẽ tin học hóa	135
5.2.3 Triển khai theo kế hoạch đã xác lập ở giai đoạn trước	136
5.2.4 Tổng hợp các yêu cầu, phân loại theo nhóm, loại bỏ các yêu cầu trùng lặp, lập hồ sơ cho từng nhóm việc.	140
5.2.5 Lập kế hoạch phân công, bố trí nhân sự tiếp cận đối tác làm rõ các yêu cầu, khái niệm chuyên môn chưa rõ	141
5.2.6 Hiệp thương với đối tác để giải quyết các vấn đề mới nảy sinh về mặt kỹ thuật và tài chính.	142
5.3. Quản lý giai đoạn thiết kế	142
5.3.1 Qhân công nhóm thiết kế tổng thể, nhóm thiết kế chi tiết	142
5.3.2 Lập văn bản đặc tả thiết kế mức tổng thể, mức chi tiết	144

5.3.3 Kế hoạch nhất thể hóa	152
5.3.4 Đặc tả các thử nghiệm cho từng cá thể	152
5.3.5 Phân tích các rủi ro	153
5.4 Quản lý giai đoạn thực hiện	154
5.4.1 Phát triển mã nguồn	154
5.4.2 Xây dựng kế hoạch bảo trì	155
5.4.3 Lập sổ ghi các giải pháp cho các mô đun thử nghiệm	156
5.4.4 Xây dựng trợ giúp cho người dùng cuối	156
5.4.5 Nhất thể hóa từng phần, tất cả, rà soát lỗi mã khi nhất thể hóa	157
5.4.6 Xây dựng kế hoạch dự phòng cho các rủi ro	158
5.4.7 Quyết toán các khâu đã hoàn thiện.	158
5.5 Quản lý giai đoạn thử nghiệm	159
5.5.1 Thực hiện việc nhất thể hóa	159
5.5.2 Thực hiện các thử nghiệm an pha, bê ta, thử nghiệm thu	159
5.5.3 Lập các báo cáo thử nghiệm	160
5.5.4 Đánh giá mức độ do các các lỗi sinh ra. giải pháp khắc phục	161
5.5.5 Lập ngân sách cuối cùng của dự án	162
5.5.6 Lắp đặt, tích hợp hệ thống và đào tạo người vận hành	162
5.6 Quản lý giai đoạn bảo trì	163
Gợi ý trả lời các bài tập	165
Tài liệu tham khảo.	166
Phần phụ lục	167

LỜI NÓI ĐẦU

Ngày nay với việc ứng dụng của công nghệ thông tin vào mọi lĩnh vực của đời sống xã hội, tất nhiên là sẽ có rất nhiều nhu cầu xây dựng các hệ thống thông tin tin học hóa từ quy mô nhỏ tới lớn, từ đơn giản đến phức tạp đòi hỏi có phương pháp luận để phát triển các hệ thống thông tin.

Phân tích thiết kế hệ thống thông tin tin học hóa là một trong các môn cung cấp phương pháp luận để thực hiện một dự án tin học dành cho tất cả học sinh, sinh viên của bậc học chuyên nghiệp trong ngành Công nghệ thông tin. Tài liệu này với những kiến thức chính, cơ bản từ giáo trình phân tích và thiết kế hệ thống thông tin của trường đại học Quốc gia và một số bộ môn có liên quan nhằm đề cập một số phương pháp tiếp cận vấn đề của một dự án tin học hóa, song song với các kiến thức trên còn có cập nhật bổ sung và giới thiệu một số cách tiếp cận hiện đại đang thịnh hành mà tác giả sưu tầm và tự nghiên cứu qua sách, báo và Internet nhằm giúp học sinh, sinh viên có cái nhìn toàn diện. Đồng thời giúp học sinh, sinh viên tốt nghiệp ra trường có khả năng lựa chọn giải pháp khả thi nhất khi tham gia, đảm nhiệm dự án tin học hóa.

Tài liệu trình bày khá chi tiết các kỹ năng trong phân tích thiết kế một hệ thống thông tin, từ việc trình bày các nguyên tắc cần thiết để phân biệt một số khái niệm liên quan đến việc trình bày hình ảnh dữ liệu bằng mô hình. Tài liệu có nhiều ví dụ cụ thể dễ hiểu từ đơn giản đến phức tạp.

Phương pháp luận phát triển các hệ thống thông tin luôn là một trong những chủ đề quan trọng nhất của công nghệ thông tin. Trải qua một giai đoạn phát triển, tiếp cận hướng đối tượng đã dần dần chiếm ưu thế và ngày càng trở nên phổ biến và đã được chuẩn hoá trong công nghiệp phần mềm. Tuy nhiên cũng cần nhìn nhận khách quan thì ngay trong cách tiếp cận hướng đối tượng vẫn còn đâu đó đường nét của tiếp cận hướng chức năng. Đặc biệt quan điểm “*Chia để trị*” thì vẫn không thay đổi cho dù bất cứ cách tiếp cận nào tính cho tới thời điểm tác giả biên soạn tài liệu này.

Bên cạnh đó trong tài liệu này tác giả cũng sưu tầm và giới thiệu một vài nét chính khi sử dụng công cụ hỗ trợ phân tích thiết kế để học sinh, sinh viên có thể sử dụng trong quá trình học môn phân tích thiết kế hệ thống thông tin như công cụ hỗ trợ như Rational Rose 2003. Ngoài ra cũng nêu tên một số công cụ khác hiện đang được giới công nghệ thông tin tại Việt Nam quan tâm để học sinh, sinh viên tham khảo và tự học khi có điều kiện.

Nội dung của tài liệu gồm 5 chương và phần Phụ lục:

Chương 1: Những vấn đề chung.

Các khái niệm cơ bản, các định hướng mới trong cách tiếp cận để thực hiện một dự án tin học; Giới thiệu các dạng hệ thống thông tin; Vòng đời phát triển hệ thống và so sánh các cách tiếp cận phát triển hệ thống; Các giai đoạn trong quá trình xây dựng hệ thống thông tin.

Chương 2: Nghiên cứu & xử lý kết quả nghiên cứu.

Phương pháp chung để nghiên cứu và xử lý các thông tin thu nhận được. Các mặt mạnh, yếu của các phương nghiên cứu; Các xu hướng hiện nay. Một vài

kinh nghiệm khi thực hiện dự án tin học mà tác giả đã tiếp cận theo các phương pháp này để cùng tham gia xây dựng phần mềm tại đơn vị đang công tác.

Chương 3: Mô hình quan niệm.

Cung cấp phương pháp để xây dựng và mô hình hoá các quan niệm. Nắm được những khái niệm căn bản của CSDL quan hệ; Các kỹ thuật cơ bản để xây dựng mô hình dữ liệu. Phương pháp đơn giản có thể để giải quyết được mọi kết hợp nhiều.

Chương 4: Mức tổ chức và vật lý.

Cung cấp phương pháp để xây dựng và mô hình hoá các vật lý. Giới thiệu một số công cụ hiện có và đang được sử dụng rộng rãi trên thị trường. Từ đó học sinh, sinh viên biết lựa chọn kết hợp các giải pháp để giải quyết bài toán bằng một công cụ thích hợp cho phát triển phần mềm.

Chương 5: Quản lý các giai đoạn xây dựng hệ thống thông tin

Với mục tiêu tham khảo, chương này cung cấp một số kiến thức được chất lọc từ bộ môn quản lý dự án phần mềm, công nghệ phần mềm nhằm hỗ trợ tốt hơn khi học sinh, sinh viên tốt nghiệp và công tác tại đơn vị phát triển phần mềm.

Phần Phụ lục.

Trích quy trình thực hiện việc nghiên cứu hiện trạng website của trường *(mẫu về nghiên cứu hiện trạng)*.

Trình bày giải pháp quản lý thư viện và phát sinh mã cho hệ thống này của trường *(mẫu về giải pháp lựa chọn, quản lý và thực hiện một dự án tin học)*.

Trình bày hồ sơ phân tích & thiết kế phần mềm quản lý đào tạo của trường *(trích một phần làm mẫu cho học sinh, sinh viên tham khảo)*.

Hồ sơ phân tích & thiết kế Hệ thống thông tin Quản lý đối tượng BHXH Thành phố HCM *(mẫu về cách lập hồ sơ phân tích thiết kế hệ thống thông tin để học sinh, sinh viên tham khảo)*.

Mỗi chương đều có phần câu hỏi, bài tập để giúp sinh viên hiểu rõ hơn kiến thức được học và kiểm tra khả năng áp dụng kiến thức của sinh viên vào các bài toán thực tế.

Tài liệu này được xây dựng nhằm đáp ứng nhu cầu học tập của học sinh, sinh viên Trung cấp Kinh tế Kỹ thuật Hòa Bình. Dù đã cố gắng hết sức, nhưng tài liệu đều mới được cập nhật và bổ sung với nhiều kiến thức mới nên việc tự nghiên cứu và biên tập chắc chắn còn nhiều hạn chế và không thể tránh khỏi nhiều thiếu sót, tác giả rất mong nhận được những ý kiến đóng góp của các thầy cô, chuyên gia, đồng nghiệp và các bạn sinh viên, học sinh.

Qua đây cá nhân cũng gửi lời tri ân tới các thầy cô, chuyên gia và đồng nghiệp chưa có điều kiện gặp mặt, mà trong tài liệu tác giả đã trích tư liệu, hình ảnh để biên tập cuốn sách này.

Tác giả chân thành cảm ơn sự động viên và đóng góp ý kiến của các thầy cô, chuyên gia và bạn đồng nghiệp để tài liệu đã biên soạn được hiệu chỉnh tốt hơn.

Tác giả

CHƯƠNG 1: NHỮNG VẤN ĐỀ CHUNG

Chương này trình bày các nội dung trọng tâm:

- Các khái niệm cơ bản.
- Các hệ thống thông tin và vấn đề phát triển hệ thống thông tin
- Các giai đoạn trong quá trình xây dựng HTTT
- Các định hướng trong cách tiếp cận để thực hiện một dự án tin học.

1.1 HỆ THỐNG THÔNG TIN

1.1.1 Hệ thống thông tin

Hệ thống là tập hợp các phần tử tương tác được tổ chức nhằm thực hiện một mục đích xác định. Áp dụng định nghĩa này vào các tổ chức ta cần chính xác hoá một số khái niệm:

- Các phần tử ở đây tập hợp các phương tiện vật chất và nhân lực.
- Tổ chức tạo thành một hệ thống mở, nghĩa là liên hệ với một môi trường. Một số phần tử của hệ thống có sự tương tác với bên ngoài (cung ứng, thương mại, v.v...).

Hệ thống thông tin là một hệ thống mà mối liên hệ giữa các thành phần của nó cũng như những liên hệ của nó với các hệ thống khác là liên hệ thông tin.

Hệ thống thông tin của một tổ chức là bao gồm các nguồn lực và quy trình xử lý thông tin trong tổ chức đương nhiên không thể bỏ qua các thông tin. Khi nói đến thông tin thì nhất thiết phải kể đến:

- Kiểu thông tin: Ví như văn bản, số liệu, âm thanh, hình ảnh, tri thức.
- Phương tiện lưu trữ thông tin: Giấy, phim, ảnh, đĩa vv....
- Các quy tắc và phương pháp biến đổi thông tin.

Như vậy hệ thống thông tin tin học hóa chính là phần mềm tin học được cấu thành bởi các yếu tố con người, thông tin, phương tiện và phương pháp xử lý tin.

1.1.1.1 Các hệ thống thông tin

Ngày nay, các ứng dụng công nghệ thông tin đã được áp dụng trong hầu hết các lĩnh vực khác nhau của đời sống xã hội. Tùy theo quan điểm mà có thể phân loại các hệ thống thông tin theo các tiêu chí khác nhau. Tuy nhiên xét về mặt ứng dụng, hệ thống thông tin có thể được phân chia thành một số nhóm như sau:

- ***Hệ thống thông tin quản lý:***

Bao gồm các hệ thống thông tin hỗ trợ các hoạt động nghiệp vụ và quản lý của các doanh nghiệp, các tổ chức. Ví dụ các hệ thống quản lý nhân sự, hệ thống kế toán, hệ thống tính cước và chăm sóc khách hàng, hệ thống quản lý thư viện, hệ thống đào tạo trực tuyến, hệ thống quản lý các đại lý bán hàng từ xa ...

- ***Các hệ thống Website:***

Gồm các hệ thống có nhiệm vụ cung cấp thông tin cho người dùng trên môi trường mạng Internet. Các hệ thống Website có đặc điểm là thông tin cung cấp cho người dùng có tính đa dạng (có thể là tin tức hoặc các dạng file đa phương tiện) và được cập nhật thường xuyên.

Gồm cả các hệ thống website đặc biệt cung cấp các dịch vụ chủ yếu là việc trao đổi mua bán hàng hoá trên môi trường Internet. Hệ thống thương mại điện tử bao gồm cả các nền tảng hỗ trợ các giao thức mua bán, các hình thức thanh toán, chuyển giao hàng hoá ...

- **Hệ thống điều khiển:**

Gồm các hệ thống phần mềm gắn với các thiết bị phần cứng hoặc các hệ thống khác nhằm mục đích điều khiển và giám sát hoạt động của thiết bị hay hệ thống đó. Ví dụ như hệ thống máy tính nhúng trong một dây truyền sản xuất, hệ thống điều khiển lò nung, hệ thống điều khiển cửa phai thủy điện, v.v...

Mỗi loại hệ thống thông tin có những đặc trưng riêng và cũng đặt ra những yêu cầu riêng cho việc phát triển hệ thống. Ví dụ, các hệ thống điều khiển đòi hỏi những yêu cầu về môi trường phát triển, hệ điều hành và ngôn ngữ lập trình riêng; các hệ website thực thi các chức năng trên môi trường mạng phân tán đòi hỏi cách phát triển riêng...Do vậy, không có một phương pháp luận chung cho tất cả các dạng hệ thống thông tin.

Có một cách phân chia khác:

- **Hệ thống thông tin** (Information System):

Cất giữ, lấy, biến đổi biểu diễn thông tin cho người sử dụng. Xử lý những khoảng dữ liệu lớn có các quan hệ phức tạp, mà chúng được lưu trữ trong các cơ sở dữ liệu quan hệ hay hướng đối tượng.

- **Hệ thống kỹ thuật** (Technical System):

Xử lý và điều khiển các thiết bị kỹ thuật như viễn thông, hệ thống quân sự, hay các quá trình công nghiệp. Đây là loại thiết bị phải xử lý các giao tiếp đặc biệt, không có phần mềm chuẩn và thường là các hệ thống thời gian thực (real time).

- **Hệ thống nhúng** (Embedded System):

Thực hiện trên phần cứng gắn vào các thiết bị như điện thoại di động, điều khiển xe hơi, ... Điều này được thực hiện bằng việc lập trình mức thấp với hỗ trợ thời gian thực. Những hệ thống này thường không có các thiết bị như màn hình đĩa cứng, ...

- **Hệ thống phân bố** (Distributed System):

Được phân bố trên một số máy cho phép truyền dữ liệu từ nơi này đến nơi khác một cách dễ dàng. Chúng đòi hỏi các cơ chế liên lạc đồng bộ để đảm bảo toàn vẹn dữ liệu và thường được xây dựng trên một số các kỹ thuật đối tượng như CORBA, COM/DCOM, hay Java Beans/RMI.

- **Hệ thống Giao dịch** (Business System):

Mô tả mục đích, tài nguyên (con người, máy tính, ...), các quy tắc (luật pháp, chiến thuật kinh doanh, cơ chế, ...), và công việc hoạt động kinh doanh.

- **Phần mềm hệ thống** (System Software):

Định nghĩa cơ sở hạ tầng kỹ thuật cho phần mềm khác sử dụng, chẳng hạn như hệ điều hành, cơ sở dữ liệu, giao diện người sử dụng.

1.1.1.2 Các tính năng của hệ thống thông tin

Các tính năng của một hệ thống thông tin gồm:

- *Thời gian trả kết quả:* Tức là thời gian kể từ khi hệ thống nhận được thông tin yêu cầu cho đến khi có quyết định tương ứng với thông tin yêu cầu đó.
- *Khối lượng của thông tin* được xử lý.
- *Độ phức tạp của dữ liệu* cần xử lý và cả *độ phức tạp của xử lý*.
- *Độ phức tạp của cấu trúc hệ thống* cho dù quy mô của hệ thống không lớn.

○ *Độ tin cậy của hệ thống* trong đó phải tính đến cơ chế an toàn, dự phòng các sự cố ngoài mong đợi. Bên cạnh đó là sự chính xác của quá trình xử lý thông tin cũng như độ ổn của hệ thống mà nguyên nhân chủ yếu đề phát sinh từ con người.

1.2 PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG THÔNG TIN

Phân tích và thiết kế hệ thống thông tin là một phương pháp được sử dụng để tạo và duy trì hệ thống thông tin nhằm thực hiện các chức năng cơ bản như lưu trữ, xử lý thông tin và truyền thông tin. Mục tiêu chính của phân tích và thiết kế hệ thống là cải tiến hệ thống cấu trúc theo một hay một nhóm các quy trình, điển hình là qua ứng dụng phần mềm, phần mềm có thể giúp đỡ những người sử dụng hoàn tất các công việc chính của đơn vị được dễ dàng và hiệu quả hơn. Là một người phân tích hệ thống, bạn sẽ phải là trung tâm của sự phát triển phần mềm đó. Phân tích và thiết kế hệ thống thông tin được dựa trên:

- *Sự hiểu biết của bạn về các mục tiêu, các cấu trúc và các qui trình của tổ chức.*
- *Kiến thức về tin học và am hiểu chuyên môn của bạn đối với lĩnh vực sẽ triển khai công nghệ thông tin nhằm mang lại lợi ích cho tổ chức theo yêu cầu.*

Trên thực tế chưa có một phần mềm nào ngay khi đưa vào sử dụng cũng như quá trình sử dụng mà không có những phiếm khuyết. Vậy cho nên việc nghiên cứu nhu cầu đánh giá hiện trạng và phân tích trước khi thiết kế là điều không thể thiếu với dự án tin học hóa hệ thống, điều đó sẽ giảm chi phí cho việc phát triển hệ thống.

Thông thường phía đầu tư thuê nhà phát triển với yêu cầu hết sức mờ, mơ hồ có dạng ***“Phần mềm quản lý...”*** hoặc ***“Trang tin điện tử của đơn vị ...”*** chứ không nói rõ là sẽ tin học hóa một vấn đề cụ thể nào. Thế nhưng khi vận hành nhà đầu tư chỉ quan tâm tới những nhu cầu trực tiếp và bỏ qua phần tổng thể của hệ thống tất nhiên là không bao giờ nói đến các mối liên hệ công tác giữa các nhóm trong đơn vị cũng như các quy định, quy trình riêng hiện đang có và đang vận hành tại đơn vị của mình. Vì thế nhất thiết phải có một phương pháp khoa học hướng dẫn việc thực hiện dự án tin học hóa tùy theo mô hình sẽ phát triển.

Có rất nhiều quan điểm về cách tiếp cận vấn đề cũng như việc phân chia các giai đoạn và cách gọi tên (có tài liệu, giáo trình chỉ có bốn giai đoạn, có tài liệu giáo trình phân chia thành tám giai đoạn. Tuy nhiên trong phạm vi khuôn khổ của tài liệu này và điều kiện cụ thể tại nhà trường, tôi xin phân chia các giai đoạn như sau: ***Lập kế hoạch, xác định yêu cầu; Phân tích; Thiết kế; Thi công; Thử***

nghiệm; Bảo trì và nội dung chủ yếu của từng giai đoạn được tóm tắt ở các mục tương ứng.

1.2.1 Các quan điểm phân tích

Quan điểm tiếp cận tổng thể theo đó mọi thành phần, mọi dữ liệu, mọi chức năng đều là hợp phần trong tổng thể và sự hiểu biết tất cả những điều đó là cần thiết cho việc phát triển hệ thống. Thế nhưng sự hiểu biết của mỗi cá thể là hữu hạn, hơn nữa do điều kiện sống cũng như môi trường đào tạo, v.v.. là khác nhau vậy đòi hỏi sự hiểu biết tất cả có lẽ là ít khả thi.

Quan điểm cấu trúc xem toàn bộ tổ chức, lĩnh vực, điểm công tác là các phần tử có cấu trúc tức là có các quy tắc hoạt động, quy tắc kiểm tra và có hệ thống thông tin riêng. Bên cạnh đó cần phải nói thêm rằng có những quy tắc hoạt động sẽ là những ràng buộc của hệ thống được tin học hóa sau này nhưng điều đó không phải lúc nào cũng có thể mô tả trọn vẹn được vì đôi khi các quy tắc đó có mâu thuẫn không thể dung hòa. Để làm được điều đó đòi hỏi phải tất cả các quy tắc hoạt động phải được đồng bộ hóa.

Trong những năm 70 - 80, những trên quan điểm được coi là chuẩn mực để phát triển phần mềm. Tuy nhiên, ngày nay cùng với thay đổi, phát triển của công nghệ những chuẩn trên tỏ ra không phù hợp trong phát triển các hệ phần mềm lớn và đặc biệt là kém hiệu quả trong sử dụng và tái sử dụng lại - Một yêu cầu quan trọng trong công nghiệp phần mềm.

Quan điểm từ trên - xuống (top - down) là quan điểm tiếp cận từ trên xuống, từ tổng thể đến riêng biệt trong đó định ra các mức trừu tượng hóa tương ứng với chu trình phát triển của hệ thống (*Các giai đoạn trong quá trình xây dựng hệ thống thông tin sẽ được nói ở phần 1.3 của tài liệu này*) và đồng thời cũng định ra kết quả mong đợi cho từng giai đoạn. Tuy nhiên với quan điểm này thì cần có những phương pháp phân tích khoa học và đội ngũ phân tích viên chuyên nghiệp.

Tóm lại: Cho dù có nhiều quan điểm và cách nhìn nhận khác nhau thế nhưng cho đến thời điểm hiện nay thì vẫn nổi bật là quan điểm **“Chia để trị”** đó là kim chỉ nam cho việc phân tích và thiết kế bất cứ hệ thống thông tin cần phát triển.

1.2.2 Các phương pháp phân tích

Có nhiều phương pháp phân tích, Thập niên 90 chứng kiến sự nở rộ trong nghiên cứu và xây dựng phương pháp luận phát triển phần mềm hướng đối tượng và nhanh chóng trở thành phổ biến trong công nghiệp phần mềm ngày nay. Tuy nhiên trong phạm vi tài liệu này chỉ nêu tên và thừa nhận kết quả còn nghiên cứu, tìm hiểu chuyên sâu sẽ dành cho các bậc học cao hơn.

Phương pháp SADT (*Structured Analysis and Design Technique*): Kỹ thuật phân tích cấu trúc và thiết kế, phương pháp này xuất phát từ Mỹ. Phương pháp phân tích này bao gồm sơ đồ luồng dữ liệu (*Data Flow Diagram*) và từ điển dữ liệu (*Data Dictionary*) ;

Phương pháp phân tích MERISE (*Method Pour Rassembler les Idées Sans Effort*) theo [GTPPTK.ĐHQG] dựa vào các mức bất biến kết hợp với mô hình. Tạm dịch là "Các phương pháp tập hợp ý tưởng không cần cố gắng", ra đời tại Pháp cuối thập niên 70.

Phương pháp MXC (*Méthode de Xavier Castellani*): Nguồn gốc từ Pháp.

Phương pháp GALACSI (*Groupe d' Animation et de Liaison pour d' Analyse et la Conception de Systeme d' Information*): Tạm dịch nguyên văn: "Nhóm cọ vẽ và liên lạc để phân tích và quan niệm hoá hệ thống tin" ra đời tại Pháp vào tháng 4 năm 1982.

1.2.2.1 Phương pháp hướng cấu trúc

Đặc trưng của phương pháp hướng cấu trúc là phân chia chương trình chính thành nhiều chương trình con, mỗi chương trình con nhằm đến thực hiện một công việc xác định.

Trong phương pháp hướng cấu trúc, phần mềm được thiết kế dựa trên một trong hai hướng: *Hướng dữ liệu* và *hướng hành động*.

- Cách tiếp cận hướng dữ liệu xây dựng phần mềm dựa trên việc phân rã phần mềm theo các chức năng cần đáp ứng và dữ liệu cho các chức năng đó. Cách tiếp cận hướng dữ liệu sẽ giúp cho những người phát triển hệ thống dễ dàng xây dựng ngân hàng dữ liệu.

- Cách tiếp cận hướng hành động lại tập trung phân tích hệ phần mềm dựa trên các hoạt động thực thi các chức năng của phần mềm đó.

Cách thức thực hiện của phương pháp hướng cấu trúc là ***phương pháp thiết kế từ trên xuống (top-down)***. Phương pháp này tiến hành phân rã bài toán thành các bài toán nhỏ hơn, rồi tiếp tục phân rã các bài toán con cho đến khi nhận được các bài toán có thể cài đặt được bằng cách sử dụng các hàm của ngôn ngữ lập trình hướng cấu trúc.

Phương pháp hướng cấu trúc có ưu điểm là tư duy phân tích thiết kế rõ ràng, chương trình sáng sủa dễ hiểu. Tuy nhiên, phương pháp này có một số nhược điểm sau:

- Không hỗ trợ việc sử dụng lại. Các chương trình hướng cấu trúc phụ thuộc chặt chẽ vào cấu trúc dữ liệu và bài toán cụ thể, do đó không thể dùng lại một modul nào đó trong phần mềm này cho phần mềm mới với các yêu cầu về dữ liệu khác.

- Tổ ra không phù hợp cho phát triển các phần mềm lớn có sở sở dữ liệu phân tán. Nếu hệ thống thông tin lớn, việc phân ra thành các bài toán con cũng như phân các bài toán con thành các modul và quản lý mối quan hệ giữa các modul đó sẽ là không phải là dễ dàng và dễ gây ra các lỗi trong phân tích và thiết kế hệ thống, cũng như khó kiểm thử và bảo trì.

1.2.2.2 Phương pháp hướng đối tượng

Khác với phương pháp hướng cấu trúc chỉ tập trung hoặc vào dữ liệu hoặc vào hành động, phương pháp hướng đối tượng tập trung vào cả hai khía cạnh của hệ thống là dữ liệu và hành động theo [TĐQ&NMS.PTTK].

Hướng đối tượng là thuật ngữ thông dụng hiện thời của ngành công nghiệp phần mềm. Các công ty đang nhanh chóng tìm cách áp dụng và tích hợp công nghệ mới này vào các ứng dụng của họ. Thật sự là đa phần các ứng dụng hiện thời đều mang tính hướng đối tượng.

Cách tiếp cận hướng đối tượng là một lối tư duy theo cách ánh xạ các thành phần trong bài toán vào các đối tượng ngoài đời thực. Với cách tiếp cận này, một hệ thống được chia tương ứng thành các thành phần nhỏ gọi là các *đối tượng*, mỗi

đối tượng bao gồm đầy đủ cả dữ liệu và hành động liên quan đến đối tượng đó. Các đối tượng trong một hệ thống tương đối độc lập với nhau và phần mềm sẽ được xây dựng bằng cách kết hợp các đối tượng đó lại với nhau thông qua các mối quan hệ và tương tác giữa chúng. Các nguyên tắc cơ bản của phương pháp hướng đối tượng bao gồm :

- **Trừu tượng hóa (abstraction):**

Trong phương pháp hướng đối tượng, các thực thể phần mềm được mô hình hóa dưới dạng các đối tượng. Các đối tượng này được trừu tượng hóa ở mức cao hơn dựa trên thuộc tính và phương thức mô tả đối tượng để tạo thành các lớp. Các lớp cũng sẽ được trừu tượng hóa ở mức cao hơn nữa để tạo thành một sơ đồ các lớp được kế thừa lẫn nhau.

Trong phương pháp hướng đối tượng có thể tồn tại những lớp không có đối tượng tương ứng, gọi là *lớp trừu tượng*. Như vậy, nguyên tắc cơ bản để xây dựng các khái niệm trong hướng đối tượng là sự trừu tượng hóa theo các mức độ khác nhau.

Quá trình phân tích phạm vi bài toán thường được bắt đầu với các khái niệm then chốt (*Key Abstraction*).

Khái niệm then chốt rất quan trọng vì dựa vào nó chúng ta có thể tìm và chỉ ra các lớp (class) :

- *Định nghĩa ranh giới của vấn đề.*
- *Nhấn mạnh đến các thực thể có liên quan đến thiết kế của hệ thống.*
- *Loại bỏ thực thể nằm ngoài phạm vi hệ thống.*

Các khái niệm then chốt thường sẽ trở thành các lớp trong mô hình phân tích. Mỗi lần tìm thấy một khái niệm then chốt mới, cần xem xét nó theo cách nhìn của vấn đề, có thể hỏi các câu hỏi sau :

- *Những chức năng nào có thể được thực hiện đối với thực thể này ?*
- *Điều gì khiến những thực thể loại này được tạo ra ?*

Nếu không có câu trả lời thích hợp, cần phải suy nghĩ lại về thực thể đó. Mỗi khái niệm then chốt mới cần phải được đặt tên cho thích hợp, miêu tả đúng chức năng của khái niệm.

Một khái niệm then chốt tóm lại là một lớp hay đối tượng thuộc chuyên ngành của phạm vi bài toán. Khi trình bày với người sử dụng, chúng có một ánh xạ 1-1 giữa với những thực thể liên quan tới người sử dụng ví như hóa đơn, séc, giấy đề nghị rút tiền, sổ tiết kiệm, thẻ rút tiền tự động, nhân viên thu ngân, nhân viên nhà băng, các phòng ban,....

Khi phân tích phạm vi bài toán, cần chú ý rằng mức độ trừu tượng của các khái niệm then chốt là rất quan trọng, bởi mức độ trừu tượng quá cao hay quá thấp đều rất dễ gây nhầm lẫn.

Mức trừu tượng quá cao dẫn tới những định nghĩa quá khái quát về một thực thể, tạo nên một cái nhìn vĩ mô và thường không nhằm vào một mục tiêu cụ thể. Ví dụ trong một nhà băng, ta không thể chọn khái niệm then chốt là "*người*", bởi nó sẽ dẫn đến lời miêu tả: "*Một người đến nhà băng để gửi tiền vào, và số tiền đó được một người khác tiếp nhận*". Trong khi một yêu cầu quan trọng ở đây là phải phân biệt giữa nhân viên với khách hàng vì chức năng của họ là khác hẳn nhau.

Tương tự như vậy, mức trừu tượng quá thấp cũng dễ gây hiểu lầm, bởi những thông tin quá vụn vặt chưa thích hợp với thời điểm đang nghiên cứu.

Các nguyên tắc cơ bản của phương pháp hướng đối tượng bao gồm:

- **Tính đóng gói (encapsulation) và ẩn dấu thông tin:** Các đối tượng có thể có những phương thức hoặc thuộc tính riêng (*kiểu private*) mà các đối tượng khác không thể sử dụng được. Dựa trên nguyên tắc ẩn giấu thông tin này, cài đặt của các đối tượng sẽ hoàn toàn độc lập với các đối tượng khác, các lớp độc lập với nhau và cao hơn nữa là cài đặt của hệ thống hoàn toàn độc lập với người sử dụng cũng như các hệ thống khác sử dụng kết quả của nó.

- **Tính modul hóa (modularity):** Các bài toán sẽ được phân chia thành những vấn đề nhỏ hơn, đơn giản và quản lý được.

- **Tính phân cấp (hierarchy):** Cấu trúc chung của một hệ thống hướng đối tượng là dạng phân cấp theo các mức độ trừu tượng từ cao đến thấp.

Ưu điểm nổi bật của phương pháp hướng đối tượng là đã giải quyết được các vấn đề nảy sinh với phương pháp hướng cấu trúc:

- **Hỗ trợ sử dụng lại mã nguồn :** Chương trình lập trình theo phương pháp hướng đối tượng thường được chia thành các gói là các nhóm của các lớp đối tượng khác nhau. Các gói này hoạt động tương đối độc lập và hoàn toàn có thể sử dụng lại trong các hệ thống thông tin tương tự.

- **Phù hợp với các hệ thống lớn:** Phương pháp hướng đối tượng không chỉ chia bài toán thành các bài toán nhỏ mà tập trung vào việc xác định các đối tượng, dữ liệu và hành động gắn với đối tượng và mối quan hệ giữa các đối tượng. Các đối tượng hoạt động độc lập và chỉ thực hiện hành động khi nhận được yêu cầu từ các đối tượng khác. Vì vậy, phương pháp này hỗ trợ phân tích, thiết kế và quản lý một hệ thống lớn, có thể mô tả các hoạt động nghiệp vụ phức tạp bởi quá trình phân tích thiết kế không phụ thuộc vào số biến dữ liệu hay số lượng thao tác cần thực hiện mà chỉ quan tâm đến các đối tượng tồn tại trong hệ thống đó.

1.2.2.3 Các khái niệm cơ bản của hướng đối tượng

Một số khái niệm cơ bản trong hướng đối tượng bao gồm:

- **Đối tượng (object):** Một đối tượng biểu diễn một thực thể vật lý, một thực thể khái niệm hoặc một thực thể phần mềm. Có thể định nghĩa một đối tượng là một khái niệm, sự trừu tượng hoặc một vật với giới hạn rõ ràng và có ý nghĩa với một ứng dụng cụ thể.

- **Lớp (Class):** Là mô tả của một nhóm đối tượng có chung các thuộc tính, hành vi và các mối quan hệ. Như vậy, một đối tượng là thể hiện của một lớp và một lớp là một định nghĩa trừu tượng của đối tượng.

- **Thành phần (component):** Là một phần của hệ thống hoạt động độc lập và giữ một chức năng nhất định trong hệ thống.

- **Gói (package):** Là một cách tổ chức các thành phần, phần tử trong hệ thống thành các nhóm. Nhiều gói có thể được kết hợp với nhau để trở thành một hệ thống con (*subsystem*).

- **Kế thừa:** Trong phương pháp hướng đối tượng, một lớp có thể có sử dụng lại các thuộc tính và phương thức của một hoặc nhiều lớp khác. Kiểu quan hệ này gọi là quan hệ kế thừa, được xây dựng dựa trên mối quan hệ kế thừa trong bài toán thực tế. Ví dụ, giả sử ta có lớp *Người* gồm các thuộc tính : *tên, ngày sinh, quê quán, giới tính* ; Lớp *Nhân Viên* có quan hệ kế thừa từ lớp *Người* sẽ có tất cả các thuộc tính trên và bổ sung thêm các thuộc tính mới gồm : *chức vụ, lương*.

Vòng đời phát triển phần mềm hướng đối tượng cũng có các pha tương tự như các vòng đời phát triển phần mềm nói chung. Các pha cơ bản đặc trưng trong phát triển phần mềm hướng đối tượng bao gồm:

- **Phân tích hướng đối tượng (Object Oriented Analysis - OOA):** xây dựng một mô hình chính xác để mô tả hệ thống cần xây dựng là gì. Thành phần của mô hình này là các đối tượng gắn với hệ thống thực.

Là giai đoạn phát triển một mô hình chính xác và súc tích của vấn đề, có thành phần là các đối tượng và khái niệm đời thực, dễ hiểu đối với người sử dụng.

Trong giai đoạn OOA, vấn đề được trình bày bằng các thuật ngữ tương ứng với các đối tượng có thực. Thêm vào đó, hệ thống cần phải được định nghĩa sao cho người không chuyên Tin học có thể dễ dàng hiểu được.

Dựa trên một vấn đề có sẵn, nhà phân tích cần ánh xạ các đối tượng hay thực thể có thực vào thiết kế để tạo ra được bản thiết kế gần cận với tình huống thực. Mô hình thiết kế sẽ chứa các thực thể trong một vấn đề có thực và giữ nguyên các mẫu hình về cấu trúc, quan hệ cũng như hành vi của chúng. Nói một cách khác, sử dụng phương pháp hướng đối tượng chúng ta có thể mô hình hóa các thực thể thuộc một vấn đề có thực mà vẫn giữ được cấu trúc, quan hệ cũng như hành vi của chúng.

Kiểu phân tích bằng kiểu ánh xạ "đời thực" vào máy tính như thế thật sự là ưu điểm lớn của phương pháp hướng đối tượng. Thế nhưng cũng có một yếu điểm là đôi khi phân tích viên cũng như nhóm phát triển có thể không nhìn thấy tính tổng thể của dự án tin học hóa sẽ phát triển trong tương lai. Do vậy kết quả của toàn bộ dự án không như mong đợi của cả các bên. Điều này một trong những nguyên nhân cơ bản dẫn tới sự thất bại của một dự án tin học hóa.

- **Thiết kế hướng đối tượng (Object Oriented Design - OOD):** Là giai đoạn tổ chức chương trình thành các tập hợp đối tượng cộng tác, mỗi đối tượng trong đó là thực thể của một lớp. Kết quả của pha thiết kế cho biết hệ thống sẽ được xây dựng như thế nào qua các bản thiết kế kiến trúc và thiết kế chi tiết.

Mục đích của giai đoạn OOD là tạo thiết kế dựa trên kết quả của giai đoạn OOA, dựa trên những quy định phi chức năng, những yêu cầu về môi trường, những yêu cầu về khả năng thực thi, OOD tập trung vào việc cải thiện kết quả của OOA, tối ưu hóa giải pháp đã được cung cấp trong khi vẫn đảm bảo thỏa mãn tất cả các yêu cầu đã được xác lập.

Trong giai đoạn OOD, nhà thiết kế định nghĩa các chức năng, thủ tục (*operations*), thuộc tính (*attributes*) cũng như mối quan hệ của một hay nhiều lớp (*class*) và quyết định chúng cần phải được điều chỉnh sao cho phù hợp với môi trường phát triển. Đây cũng là giai đoạn để thiết kế ngân hàng dữ liệu và áp dụng các kỹ thuật tiêu chuẩn hóa.

Về cuối giai đoạn OOD, nhà thiết kế đưa ra một loạt các biểu đồ (*diagram*) khác nhau. Các biểu đồ này có thể được chia thành hai nhóm chính là *tĩnh* và *động*. Các biểu đồ tĩnh biểu thị các lớp và đối tượng, trong khi biểu đồ động biểu thị tương tác giữa các lớp và phương thức hoạt động chính xác của chúng. Các lớp đó sau này có thể được nhóm thành các gói (*Packages*) tức là các đơn vị thành phần nhỏ hơn của ứng dụng.

• **Lập trình hướng đối tượng (Object Oriented Programming - OOP):** Thực hiện bản thiết kế hướng đối tượng bằng cách sử dụng các ngôn ngữ lập trình hướng đối tượng (*C++*, *Java*, ...). Kết quả chung cuộc của giai đoạn này là một loạt các code chạy được, nó chỉ được đưa vào sử dụng sau khi đã trải qua nhiều vòng quay của nhiều bước thử nghiệm khác nhau. Tuy nhiên trên thực tế các ngôn ngữ lập trình hướng đối tượng như đã nêu đều được nhúng một hệ quản trị cơ sở dữ liệu (HQTCSĐL) nào đó. Hiện nay có các loại hệ quản trị cơ sở dữ liệu phổ dụng:

- ✧ Loại phân cấp như hệ IMS của IBM
- ✧ Loại mạng IDMS của Cullinet Software
- ✧ Loại tập tin đảo như ADABAS của Software AG
- ✧ Loại quan hệ như ORACLE của Oracle, DB2 của IBM, ACCESS của Microsoft Access
- ✧ Loại đối tượng. Loại này là một tiếp cận khá mới trong thiết kế HQTCSĐL và việc sử dụng hệ loại này sớm trở nên phổ biến.

Hiện tại, loại HQTCSĐL chính được sử dụng trong công nghệ là loại HQTCSĐL quan hệ (RDBMS). Loại này đã chiếm lĩnh trong công nghệ trên 10-15 năm cuối cùng khi đánh bật loại HQTCSĐL phân cấp và gần đây là HQTCSĐL mạng.

1.3. CÁC GIAI ĐOẠN TRONG QUÁ TRÌNH XÂY DỰNG HTTT

Việc phát triển các hệ thống thông tin không chỉ đơn giản là lập trình mà luôn được xem như một tiến trình hoàn chỉnh.

Tiến trình phần mềm là phương cách sản xuất ra phần mềm với các thành phần chủ yếu bao gồm: Mô hình vòng đời phát triển phần mềm, các công cụ hỗ trợ cho phát triển phần mềm và những người trong nhóm phát triển phần mềm.

Như vậy, tiến trình phát triển phần mềm nói chung là sự kết hợp cả hai khía cạnh kỹ thuật (*vòng đời phát triển, phương pháp phát triển, các công cụ và ngôn ngữ sử dụng, ...*) và khía cạnh quản lý (*quản lý dự án phần mềm*). Xin lưu ý về cách dùng ngôn ngữ của một số tài liệu thì các giai đoạn trong quá trình xây dựng hệ thống thông tin cũng tương ứng với các vòng đời các bước phát triển một sản phẩm phần mềm. Các giai đoạn trong quá trình xây dựng hệ thống thông tin gồm các bước cơ bản sau:

1.3.1 Lập kế hoạch, xác định yêu cầu

Đây là giai đoạn đầu tiên nhằm nghiên cứu thực trạng của tổ chức (*nhà đầu tư*) có nhu cầu tin học hóa một quá trình nào đó được nhà phát triển gọi là dự án tin học hóa (*bản chất là xây dựng một phần mềm phục vụ mục đích chuyên dụng của một tổ chức đó, không loại trừ phần mềm này đã được thương mại hóa và có bán trên thị trường nhưng có một số điểm không phù hợp với yêu cầu cụ thể của đơn vị đó*). Nếu là dự án tin học hóa lớn người ta cũng có thể tổ chức đấu thầu như việc

đầu thầu trong xây dựng vậy. Điều đặc biệt ở chỗ trước khi bỏ thầu nhà phát triển cũng phải khảo sát hiện trạng thực tế sơ bộ, còn nếu không thì khôn ngoan hơn cả là có lẽ chỉ gia công một số công đoạn nào đó trong dự án tin học đó.

Trong giai đoạn này nhà phát triển phải nhất thiết xác định được các yêu cầu cụ thể của nhà đầu tư trước khi tiến hành nghiên cứu và phát triển dự án.

Nhà phát triển phải có kế hoạch và thống nhất với nhà đầu tư thời gian khảo sát thực tiễn tại đơn vị đó bao gồm:

- Thời gian và địa điểm sẽ khảo sát.
- Số lượng các thành viên, cơ sở vật chất có liên quan, ảnh hưởng đến việc triển khai dự án đó.
- Khám phá các khái niệm liên quan đến việc phát triển phần mềm.
- Trình độ và tính sẵn sàng cho việc tin học hóa của thành viên tại đơn vị đó.
- Xác định chính xác yêu cầu và các ràng buộc của khách hàng với sản phẩm phần mềm đó.
- Bên cạnh đó cũng thống nhất thời gian tiến hành và kết thúc dự án, các giai đoạn chính trong quá trình thực hiện dự án.
- Các vấn đề về tài chính, nhân sự và trách nhiệm các bên trong từng giai đoạn.
- Các hợp đồng và các thủ tục hành chính phù hợp với pháp luật của quốc gia nơi nhà đầu tư sẽ phải được ký kết trước khi tiến hành dự án

1.3.2 Phân tích

1.3.2.1 Phân tích hướng chức năng theo [GTPPTK.ĐHQG]

Đây là công việc trọng tâm trong cả quá trình xây dựng hệ thống thông tin. Bản thân giai đoạn này cũng có khởi điểm nằm trong các giai đoạn khác. Đặc biệt là nằm quá trình nghiên cứu hiện trạng của giai đoạn lập kế hoạch trước nhưng ở mức độ chi tiết hơn.

- Xây dựng một mô hình chính xác để mô tả hệ thống cần xây dựng là gì. Thành phần của mô hình này là các đối tượng gắn với hệ thống thực.
- Mô tả chức năng của sản phẩm, các input của sản phẩm và các output được yêu cầu; Khám phá các khái niệm trong miền quan tâm của sản phẩm và bước đầu đưa ra giải pháp xây dựng hệ thống. Hay nói cách khác chính là xác định phạm vi của dự án cần tin học hóa.
- *Nghiên cứu hiện trạng*: Như đã nói ở phần trên quá trình này cụ thể hóa các tình trạng hiện tại của hệ thống cũ, cũng như xem xét nhiệm vụ, vai trò của từng đơn vị và của từng thành viên có liên quan, mối quan hệ thông tin, luồng thông tin, quá trình xử lý thông tin để hoạt động trong tổ chức có nhu cầu tin học hóa.
- *Tần xuất, thời hạn, hiệu xuất* của các thông tin đó trong hệ thống cũ. Các khả năng và giải pháp cho hệ thống tin học sẽ phát triển trong tương lai.
- *Xây dựng mô hình hệ thống*: Căn cứ và kết quả nghiên cứu hiện trạng từ đó làm rõ mô hình thông tin và mô hình hoạt động của hệ thống. Đây là giai đoạn cực kỳ quan trọng quyết định sự thành bại của dự án tin học hóa bởi vì:

- ✓ Phụ thuộc hoàn toàn vào tài năng và kinh nghiệm của các phân tích viên.
- ✓ Tính thống nhất, hiểu biết chuyên môn giữa các phân tích viên và các thành viên các được tìm hiểu tại các đơn vị thuộc nhà đầu tư.
- ✓ Sự hợp tác của các thành viên tham gia không loại trừ các lý do tế nhị.
- ✓ Khả năng tài chính của nhà đầu tư khi phải thay đổi.
- ✓ Trình độ ứng dụng tin học cũng như trạng thái sẵn sàng thay đổi của các thành viên trong đơn vị sẽ sử dụng phần mềm này trong tương lai. Vì hầu hết các phần mềm được xây dựng đều có thể phải thay đổi một phần nhỏ về quy trình hoạt động của tổ chức đó cho dù không muốn điều đó xảy ra.

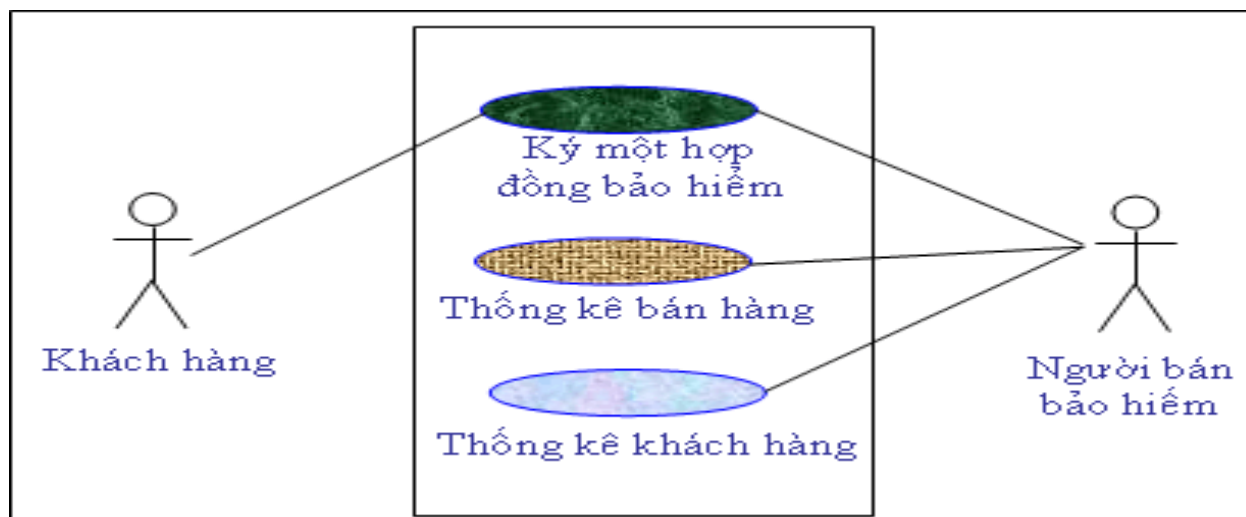
- **Lập hồ sơ nhiệm vụ:** Lựa chọn các giải pháp và định hình các chức năng có thể mà hệ thống cần đạt được; Định hình thô giao diện và các mô dul cơ bản nhất, có thể đây là một nguyên mẫu; Chuẩn các văn bản thỏa thuận các bên (*Phân tích viên, Nhà đầu tư, người sẽ sử dụng*).

1.3.2.2 Phân tích hướng đối tượng theo UML- Ngôn Ngữ Mô Hình Hoá Thống Nhất (Unifield Modeling Language) [TĐQ&NMS.PTTK].

Giai đoạn phân tích quan tâm đến quá trình trừu tượng hóa đầu tiên (*các lớp và các đối tượng*) cũng như cơ chế hiện hữu trong phạm vi vấn đề. Sau khi nhà phân tích đã nhận biết được các lớp thành phần của mô hình cũng như mối quan hệ giữa chúng với nhau, các lớp cùng các mối quan hệ đó sẽ được miêu tả bằng công cụ biểu đồ lớp (*class diagram*) của UML. Sự cộng tác giữa các lớp nhằm thực hiện các Use case cũng sẽ được miêu tả nhờ vào các mô hình động (*dynamic models*) của UML. Trong giai đoạn phân tích, chỉ duy nhất các lớp có tồn tại trong phạm vi vấn đề (*các khái niệm đời thực*) là được mô hình hóa. Các lớp kỹ thuật định nghĩa chi tiết cũng như giải pháp trong hệ thống phần mềm, ví dụ như các lớp cho giao diện người dùng, cho ngân hàng dữ liệu, cho sự giao tiếp, trùng hợp, v.v..., chưa phải là mối quan tâm của giai đoạn này.

- **Xây dựng Biểu đồ use case (Use Case Diagram):** Dựa trên tập yêu cầu ban đầu, người phân tích tiến hành xác định các tác nhân, use case và các quan hệ giữa các use case để mô tả lại các chức năng của hệ thống. Một thành phần quan trọng trong biểu đồ use case là các kịch bản mô tả hoạt động của hệ thống trong mỗi use case cụ thể.

Một biểu đồ Use case chỉ ra một số lượng các tác nhân ngoại cảnh và mối liên kết của chúng đối với Use case mà hệ thống cung cấp - **Hình 1.1**. Một Use case là một lời miêu tả của một chức năng mà hệ thống cung cấp. Lời miêu tả Use case thường là một văn bản tài liệu, nhưng kèm theo đó cũng có thể là một biểu đồ hoạt động. Các Use case được miêu tả duy nhất theo hướng nhìn từ ngoài vào của các tác nhân (*hành vi của hệ thống theo như sự mong đợi của người sử dụng*), không miêu tả chức năng được cung cấp sẽ hoạt động nội bộ bên trong hệ thống ra sao. Các Use case định nghĩa các yêu cầu về mặt chức năng đối với hệ thống.

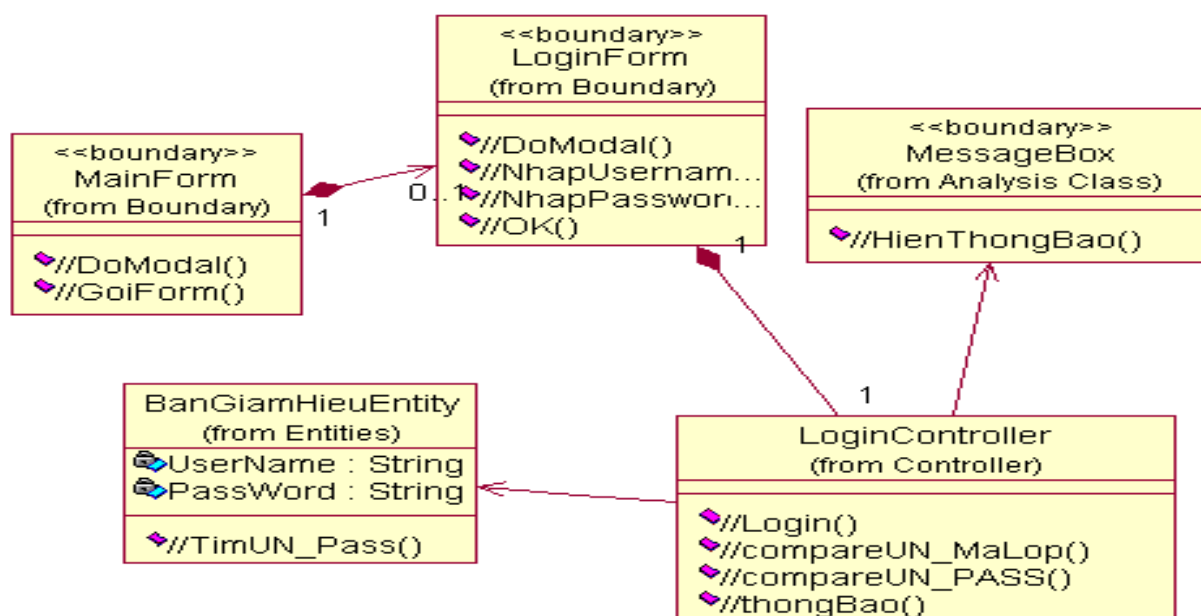


Hình 1.1: Biểu đồ use case của một công ty bảo hiểm

- **Xây dựng Biểu đồ lớp (Class Diagram):** Xác định tên các lớp, các thuộc tính của lớp, một số phương thức và mối quan hệ cơ bản trong sơ đồ lớp.

Một biểu đồ lớp chỉ ra cấu trúc tĩnh của các lớp trong hệ thống - **Hình 1.2**. Các lớp là đại diện cho các “vật” được xử lý trong hệ thống. Các lớp có thể quan hệ với nhau trong nhiều dạng thức: liên kết (*associated* - được nối kết với nhau), phụ thuộc (*dependent* - một lớp này phụ thuộc vào lớp khác), chuyên biệt hóa (*specialized* - một lớp này là một kết quả chuyên biệt hóa của lớp khác), hay đóng gói (*packaged* - hợp với nhau thành một đơn vị). Tất cả các mối quan hệ đó đều được thể hiện trong biểu đồ lớp, đi kèm với cấu trúc bên trong của các lớp theo khái niệm thuộc tính (*attribute*) và thủ tục (*operation*). Biểu đồ được coi là biểu đồ tĩnh theo phương diện cấu trúc được miêu tả ở đây có hiệu lực tại bất kỳ thời điểm nào trong toàn bộ vòng đời hệ thống.

Một hệ thống thường sẽ có một loạt các biểu đồ lớp - chẳng phải bao giờ tất cả các biểu đồ lớp này cũng được nhập vào một biểu đồ lớp tổng thể duy nhất và một lớp có thể tham gia vào nhiều biểu đồ lớp.



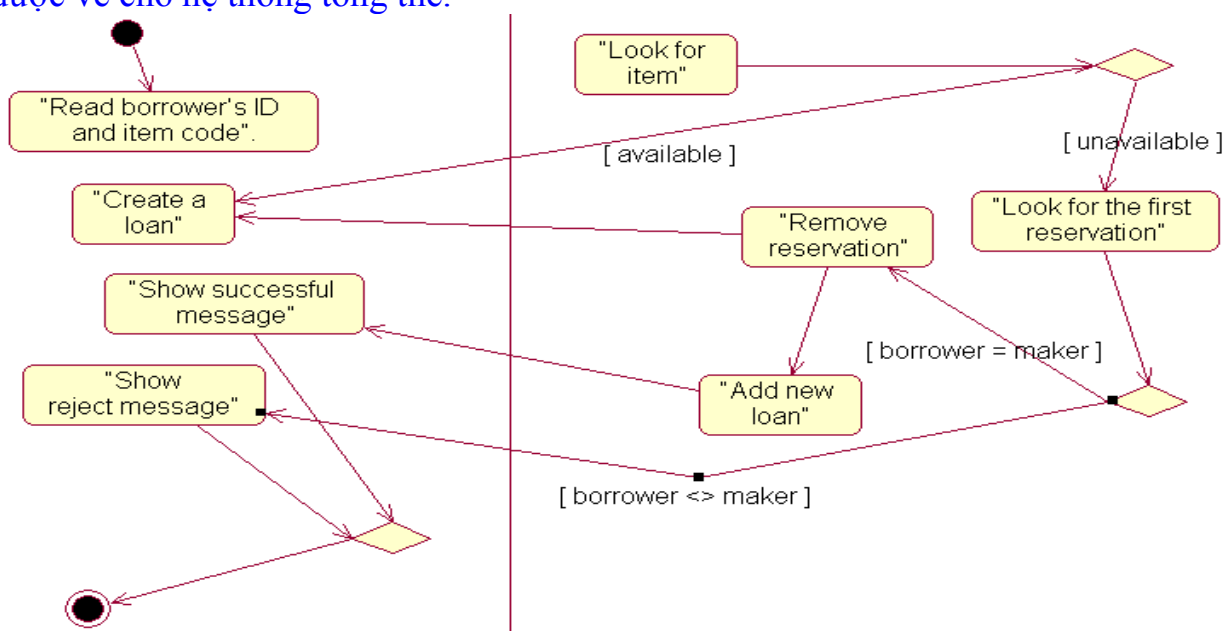
Hình 1.2 : Biểu đồ lớp cho một đăng nhập chương trình

- **Biểu đồ đối tượng (Object Diagram):** Một biểu đồ đối tượng là một phiên bản của biểu đồ lớp và thường cũng sử dụng các ký hiệu như biểu đồ lớp. Sự khác biệt giữa hai loại biểu đồ này nằm ở chỗ biểu đồ đối tượng chỉ ra một loạt các đối tượng thực thể của lớp, thay vì các lớp. Một biểu đồ đối tượng vì vậy là một ví dụ của biểu đồ lớp, chỉ ra một bức tranh thực tế có thể xảy ra khi hệ thống thực thi: bức tranh mà hệ thống có thể có tại một thời điểm nào đó. Biểu đồ đối tượng sử dụng chung các ký hiệu của biểu đồ lớp, chỉ trừ hai ngoại lệ: đối tượng được viết với tên được gạch dưới và tất cả các thực thể trong một mối quan hệ đều được chỉ ra - **Hình 1.2**. Biểu đồ đối tượng không quan trọng bằng biểu đồ lớp, chúng có thể được sử dụng để ví dụ hóa một biểu đồ lớp phức tạp, chỉ ra với những thực thể cụ thể và những mối quan hệ như thế thì bức tranh toàn cảnh sẽ ra sao. Một biểu đồ đối tượng thường thường được sử dụng làm một thành phần của một biểu đồ cộng tác (collaboration), chỉ ra lối ứng xử động giữa một loạt các đối tượng.

- **Xây dựng biểu đồ trạng thái (State Diagram):** Mô tả các trạng thái và chuyển tiếp trạng thái trong hoạt động của một đối tượng thuộc một lớp nào đó.

Một biểu đồ trạng thái thường là một sự bổ sung cho lời miêu tả một lớp. Nó chỉ ra tất cả các trạng thái mà đối tượng của lớp này có thể có, và những sự kiện (event) nào sẽ gây ra sự thay đổi trạng thái - **Hình 1.3**. Một sự kiện có thể xảy ra khi một đối tượng tự gửi thông điệp đến cho nó - ví dụ như để thông báo rằng một khoảng thời gian được xác định đã qua đi – hay là một số điều kiện nào đó đã được thỏa mãn. Một sự thay đổi trạng thái được gọi là một sự chuyển đổi trạng thái (State Transition). Một chuyển đổi trạng thái cũng có thể có một hành động liên quan, xác định điều gì phải được thực hiện khi sự chuyển đổi trạng thái này diễn ra.

Biểu đồ trạng thái không được vẽ cho tất cả các lớp, mà chỉ riêng cho những lớp có một số lượng các trạng thái được định nghĩa rõ ràng và hành vi của lớp bị ảnh hưởng và thay đổi qua các trạng thái khác nhau. Biểu đồ trạng thái cũng có thể được vẽ cho hệ thống tổng thể.



Hình 1.3: Biểu đồ trạng thái trong xây dựng hệ thống quản lý thư viện