TRƯỜNG ĐẠI HỌC ĐIỆN LỰC

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------\*\*\*----------**



**BÁO CÁO HỌC PHẦN**

**CÔNG NGHỆ PHẦN MỀM**

**Chủ đề:**

|  |  |
| --- | --- |
| **Trình bày quy trình phát triển phần mềm theo mô hình bản mẫu? Mô hình luồng dữ liệu và mô hình thực thể liên kết, cho ví dụ? Phân tích và thiết kế website bán đồ ăn quê** |  |

**GVHD: Phạm Đức Hồng**

**Nhóm sinh viên: 1) Nguyễn Huy Hoàng**

**2) Ngô Trần Mạnh**

**3) Nguyễn Văn Phương**

**Lớp: D18CNPM2**

***Hà Nội, ngày…tháng.....năm…...***

Mục lục

[**GIỚI THIỆU**](#_Toc169811892)

PHẦN 1:Trình bày quy trình phát triển phần mềm theo mô hình bản mẫu

1.1 Khái niệm quy trình phát triển phần mềm theo mô hình bản mẫu .........

1.2 Phân tích quy trình phát triển phần mềm theo mô hình bản mẫu .........

**PHẦN 2:** **Mô hình luồng dữ liệu và mô hình thực thể liên kết, cho ví dụ** ...........................................................................................................

2.1 Trình bày mô hình luồng dữ liệu. Cho ví dụ ................................................

2.2 Trình bày mô hình thực thể liên kết. Cho ví dụ ............................................

**PHẦN 3:** **Phân tích và thiết kế website bán đồ ăn quê** .....................

3.1 **Xác định các chức năng của hệ thống website bán đồ ăn quê** ......................

3.2 Mô hình dữ liệu mức khung cảnh.................................................................

**DANH MỤC HÌNH VẼ,BẢNG BIỂU**

Hình a: Mô hình bản mẫu

Hình b: Sơ đồ dòng dữ liệu hoạt động bán hàng

Hình c:Mô hình dữ liệu mức khung cảnh của hệ cung ứng vật tư

Hình d:Sơ đồ luồng dữ liệu mức đỉnh của hệ cung ứng vật tư

Hình e:Mô hình luồng dữ liệu mức dưới đỉnh của chức năng đặthàng

Hình 1:Một số quy ước về lược đồ của mô hình dữ liệu thực thể - liên kết

Hình 2: Lược đồ cơ sở dữ liệu theo mô hình thực thể - liên kết

Hình 3: Lược đồ cơ sở dữ liệu theo mô hình thực thể - liên kết ở dạng rút gọn

# GIỚI THIỆU

**Quy trình phát triển phần mềm** là một cấu trúc bao gồm tập hợp các thao tác và các kết quả tương quan sử dụng trong việc phát triển để sản xuất ra một sản phẩm phần mềm.

**Các hoạt động cơ bản của quy trình phát triển phần mềm:**

* Đặc tả phần mềm: Định nghĩa được các chức năng, điều kiện hoạt động của phần mềm.
* Phát triển phần mềm: Là quá trình xây dựng các đặc tả.
* Đánh giá phần mềm: Phầm mềm phải được đánh giá để chắc chắn rằng ít nhất có thể thực hiện những gì mà tài liệu đặc tả yêu cầu.
* Tiến hóa phần mềm: Đây là quá trình hoàn thiện các chức năng cũng như giao diện để ngày càng hoàn thiện phần mềm cũng như các yêu cầu đưa ra từ phía khách hàng.

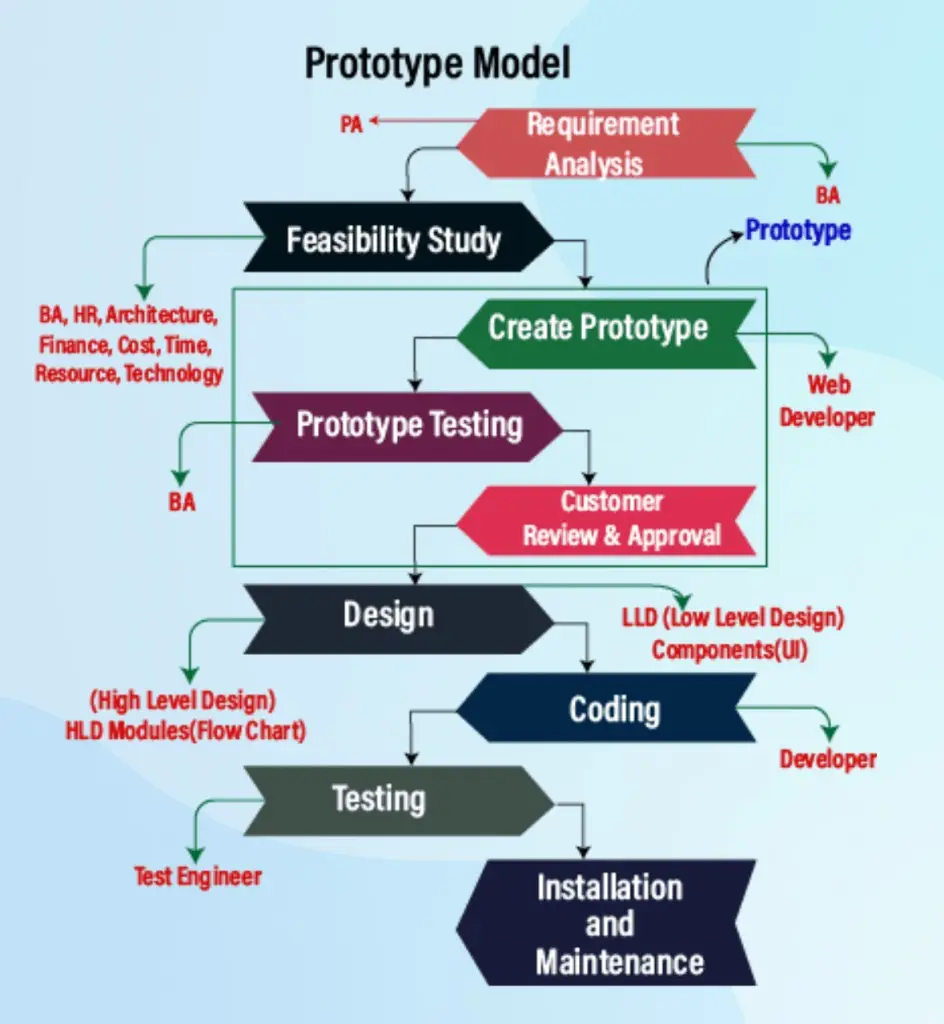
Do đó chúng em đã chọn đề tài này làm đề tài thuyết trình.

|  |  |
| --- | --- |
| **Chủ đề: Trình bày quy trình phát triển phần mềm theo mô hình bản mẫu? Mô hình luồng dữ liệu và mô hình thực thể liên kết, cho ví dụ? Phân tích và thiết kế website bán đồ ăn quê.** |  |

PHẦN 1:Trình bày quy trình phát triển phần mềm theo mô hình bản mẫu

## Khái niệm quy trình phát triển phần mềm theo mô hình bản mẫu.

Mô hình bản mẫu (Prototyping Model) là một phương pháp phát triển phần mềm tập trung vào việc xây dựng một bản mẫu (prototype) nhanh chóng để minh họa các chức năng cơ bản, từ đó thu thập phản hồi của người dùng và cải thiện phần mềm dần dần.



Hình a: Mô hình bản mẫu

* Khi tạo một phần mềm hoặc ứng dụng, điển hình là sử dụng một mô hình nguyên mẫu để cung cấp một phiên bản cũ hơn và đang hoạt động có thể được sử dụng làm bản trình bày hoặc mẫu của dự án.
* Tạo nguyên mẫu là một cách tuyệt vời để nhận đầu vào về các yêu cầu, chức năng và khả năng hoạt động, để quá trình phát triển cuối cùng của sản phẩm có thể diễn ra nhanh chóng và hiệu quả hơn.
* Mô hình nguyên mẫu là một ứng dụng chức năng của sản phẩm đưa ra ý tưởng về tính năng cơ bản của sản phẩm hoặc hệ thống cuối cùng.

**1.2 Phân tích quy trình phát triển phần mềm theo mô hình bản mẫu.**

\_ *Quy trình phát triển phần mềm theo mô hình này thường gồm các bước sau*:

**1. Thu thập yêu cầu ban đầu**

* Nhà phát triển và khách hàng cùng làm việc để xác định các yêu cầu cơ bản của phần mềm.
* Chỉ tập trung vào các yêu cầu chính yếu, chưa cần quá chi tiết hoặc đầy đủ.

**2. Xây dựng bản mẫu ban đầu**

* Dựa trên các yêu cầu ban đầu, nhà phát triển xây dựng một bản mẫu nhanh chóng.
* Bản mẫu này không phải là phiên bản hoàn chỉnh của phần mềm, chỉ nhằm minh họa giao diện, các chức năng cơ bản hoặc quy trình chính.

**3. Đánh giá bản mẫu**

* Người dùng hoặc khách hàng đánh giá bản mẫu.
* Góp ý, phản hồi về các vấn đề như giao diện, chức năng, trải nghiệm sử dụng.

**4. Cải tiến và lặp lại**

* Nhà phát triển cập nhật bản mẫu dựa trên phản hồi của người dùng.
* Quy trình cải tiến lặp lại cho đến khi đạt được sự đồng thuận giữa các bên về tính năng và chất lượng.

**5. Phát triển phần mềm chính thức**

* Sau khi bản mẫu đã hoàn thiện và đáp ứng yêu cầu của khách hàng, nhà phát triển bắt đầu xây dựng sản phẩm phần mềm chính thức dựa trên bản mẫu đã được phê duyệt.
* Kết hợp các chức năng chi tiết hơn và tối ưu hóa hệ thống.

**6. Kiểm thử và triển khai**

* Phần mềm chính thức được kiểm thử để đảm bảo không có lỗi và đáp ứng các yêu cầu kỹ thuật.
* Sau đó, phần mềm được triển khai đến người dùng cuối.
* Sau khi triển khai, phần mềm tiếp tục được duy trì và cập nhật để sửa lỗi hoặc thêm các tính năng mới.

|  |  |
| --- | --- |
| Ưu điểm | Nhược điểm |
| Chúng tôi có thể dễ dàng phát hiện các chức năng bị thiếu. | Đó là một quá trình tốn thời gian vì nếu khách hàng thay đổi trong nguyên mẫu.Và nó cũng sẽ lãng phí thời gian của chúng ta bằng cách thay đổi lặp đi lặp lại trong giả (nguyên mẫu), điều này sẽ làm trì hoãn hoạt động của dự án thực. |
| Trong điều này, nhóm phát triển và khách hàng có sự trao đổi rõ ràng về các yêu cầu và kết quả của sản phẩm. | Không có đánh giá yêu cầu, nhưng đánh giá nguyên mẫu ở đó. |
| Trong điều này, sự hài lòng của khách hàng tồn tại. | Không có giao hàng song song, có nghĩa là hai đội không thể làm việc cùng nhau. |
| Chúng tôi có thể sử dụng lại nguyên mẫu trong giai đoạn thiết kế và cho các ứng dụng tương tự. | Đôi khi, ứng dụng một phần có thể khiến phần mềm không được sử dụng như hệ thống hoàn chỉnh đã được thiết kế. |
| Trong mô hình này, sự từ chối của khách hàng ít hơn so với các mô hình khác. | Phân tích vấn đề không đầy đủ hoặc một phần. |
| Các vấn đề có thể được xác định trong giai đoạn đầu. | Chúng tôi cũng có thể mất sự chú ý của khách hàng nếu họ không hài lòng với sản phẩm cuối cùng hoặc nguyên mẫu ban đầu. |

\_*Ưu và nhược điểm mô hình bản mẫu*

**Khi nào thì sử dụng mô hình bản mẫu?**

* Khi yêu cầu của hệ thống mong muốn là rõ ràng.
* Khi các chức năng cơ bản của hệ thống mong muốn vẫn chưa được đánh giá.
* Nếu các yêu cầu của hệ thống kết quả cần phải được thay đổi.
* Để hiển thị các chức năng kỹ thuật của sản phẩm mong muốn bằng cách tạo bản mẫu.
* Thời gian phát triển bị giảm bớt
* Các thành phần có thể tái sử dụng
* Đánh giá ban đầu được đưa ra nhanh chóng
* Khách hàng có thể đưa ra phản hồi theo từng bản mẫu

## PHẦN 2: Mô hình luồng dữ liệu và mô hình thực thể liên kết, cho ví dụ

**2.1: Trình bày mô hình luồng dữ liệu. Cho ví dụ**

**1. Định nghĩa Mô hình luồng dữ liệu (DFD)**

Mô hình luồng dữ liệu (DFD – Data Flow Diagram) là một công cụ mô tả mối quan hệ thông tin giữa các công việc .

Ví dụ: Mô hình luồng dữ liệu của hoạt động bán hàng

A diagram of a diagram

Description automatically generated

Hình b: Sơ đồ dòng dữ liệu hoạt động bán hàng

**2. Các thành phần của mô hình luồng dữ liệu**

**(\*). Chức năng (còn gọi là Tiến trình)**

– Định nghĩa: Là một hoạt động có liên quan đến sự biến đổi hoặc tác động lên thông tin như tổ chức lại thông tin, bổ sung thông tin hoặc tạo ra thông tin mới. Nếu trong một chức năng không có thông tin mới được sinh ra thì đó chưa phải là chức năng trong mô hình luồng dữ liệu.

– Cách đặt tên: Động từ + bổ ngữ.

Ví dụ: Chấp nhận nguồn hàng, ghi kho vật liệu…

– Biểu diễn: hình chữ nhật góc tròn hoặc hình tròn

A black and white image of a rectangular object

Description automatically generated

Chú ý : Trong thực tế tên chức năng phải trùng với tên chức năng trong mô hình phân rã chức năng.

**(\*). Luồng dữ liệu**

**–** Định nghĩa: Là luồng thông tin vào hoặc ra khỏi chức năng

– Cách đặt tên : Danh từ + tính từ

– Biểu diễn : là mũi tên trên đó ghi thông tin di chuyển

A white oval with black text

Description automatically generated

Chú ý: Các luồng dữ liệu phải chỉ ra được thông tin logic chứ không phải tài liệu vật lý. Các luồng thông tin khác nhau phải có tên gọi khác nhau.

Ví dụ: Luồng dữ liệu biểu hiện việc trả tiền mang tên là “thanh toán” chứ không mang tên là “tiền” hay “sec”.

**(\*). Kho dữ liệu**

Kho dữ liệu là nơi biểu diễn thông tin cần lưu giữ, để một hoặc nhiều chức năng sử dụng chúng.

Cách đặt tên kho dữ liệu như sau : danh từ + tính từ. Tên kho phải chỉ rõ nội dung dữ liệu trong kho.

Kho dữ liệu được biểu diễn bằng cặp đường thẳng song song chứa tên kho cần cất giữ.

Ví dụ 1: biểu diễn kho hóa đơn như sau:

A black text on a white background

Description automatically generated

Quan hệ giữa kho dữ liệu, chức năng và luồng dữ liệu được biểu diễn như sau:

A diagram of a circle with a line and a arrow

Description automatically generated with medium confidence

Ví dụ 2: Một người muốn tra cứu một danh sách giá cả: phải lấy thông tin từ kho dữ liệu danh sách giá cả (hình a), còn muốn sửa đổi giá cả thì luồng dữ liệu đi từ tiến trình tới kho dữ liệu (hình b). Để biểu thị việc kiểm tra giá cả mặt hàng và sửa đổi các giá không phù hợp thì dùng mũi tên hai chiều để biểu thị luồng dữ liệu từ tiến trình tới kho

A black and white text

Description automatically generated

**(\*)**. **Tác nhân bên ngoài**

Định nghĩa: Là một người hoặc một nhóm người nằm ngoài hệ thống nhưng có trao đổi trực tiếp với hệ thống. Sự có mặt của các nhân tố này trên sơ đồ chỉ ra giới hạn của hệ thống, định rõ mối quan hệ của hệ thống với thế giới bên ngoài

– Tên : Danh từ

– Biểu diễn : hình chữ nhật

A black and white rectangle with a black line

Description automatically generated

**(\*). Tác nhân bên trong**

Là một chức năng hoặc một hệ thống con của hệ thống đang xét nhưng được trình bày ở một trang khác của mô hình.

Mọi sơ đồ luồng dữ liệu đều có thể bao gồm một số trang, thông tin truyền giữa các quá trình trên các trang khác nhau được chỉ ra nhờ kí hiệu này.

– Tên: động từ + bổ ngữ

– Biểu diễn:

**A black text on a white background

Description automatically generated**

3. Một số quy tắc

Khi vẽ biểu đồ luồng dữ liệu ta phải thực hiện theo các quy tắc sau:

– Các luồng dữ liệu vào của một tiến trình cần khác với các luồng dữ liệu ra của nó. Tức là các dữ liệu qua một tiến trình phải có thay đổi. Ngược lại, tiến trình là không cần thiết vì không tác động gì đến các luồng thông tin đi qua nó

– Các đối tượng trong một mô hình luồng dữ liệu phải có tên duy nhất: mỗi tiến trình phải có tên duy nhất. Tuy nhiên, vì lí do trình bày cùng một tác nhân trong, tác nhân ngoài và kho dữ liệu có thể được vẽ lặp lại.

– Các luồng dữ liệu đi vào một tiến trình phải đủ để tạo thành các luồng dữ liệu đi ra.

– Nói chung tên luồng thông tin vào hoặc ra kho trùng với tên kho vì vậy không cần viết tên luồng. Nhưng khi ghi hoặc lấy tin chỉ tiến hành một phần kho thì lúc đó phải đặt tên cho luồng

– Không có một tiến trình nào chỉ có cái ra mà không có cái vào. Đối tượng chỉ có cái ra thì có thể là tác nhân ngoài (nguồn)

– Không một tiến trình nào mà chỉ có cái vào mà không có cái ra. Một đối tượng chỉ có cái vào thì chỉ có thể là tác nhân ngoài (đích)

– Không thể xảy ra các trường hợp biểu diễn sau:

A black and white image of a couple of rectangular objects

Description automatically generated

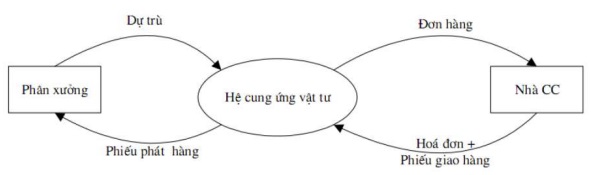
**4. Trình tự xây dựng sơ đồ dòng dữ liệu**

* **Bước 1**: Xây dựng mô hình luồng dữ liệu mức khung cảnh (mức 0)

– Mô hình luồng dữ liệu mức khung cảnh gồm một chức năng duy nhất biểu thị toàn bộ hệ thống đang nghiên cứu, chức năng này được nối với mọi tác nhân ngoài của hệ thống.

– Các luồng dữ liệu giữa chức năng và tác nhân ngoài chỉ thông tin vào và ra của hệ thống

Ví dụ: Mô hình dữ liệu mức khung cảnh của hệ cung ứng vật tư



Hình c:Mô hình dữ liệu mức khung cảnh của hệ cung ứng vật tư

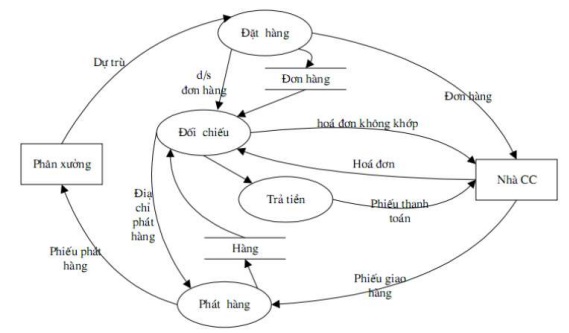
* **Bước 2**: Xây dựng mô hình luồng dữ liệu mức đỉnh (mức 1)

– Với mức đỉnh các tác nhân ngoài của hệ thống ở mức khung cảnh được giữ `nguyên với các luồng thông tin vào ra.

– Hệ thống được phân rã thành các chức năng mức đỉnh là các tiến trình chính bên trong hệ thống theo mô hình phân rã chức năng mức 1.

– Xuất hiện thêm các kho dữ liệu và luồng thông tin trao đổi giữa các chức năng mức đỉnh.

Ví dụ: Mô hình luồng dữ liệu mức đỉnh của hệ cung ứng vật tư



Hình d:Sơ đồ luồng dữ liệu mức đỉnh của hệ cung ứng vật tư

* **Bước 3**: Xây dựng mô hình luồng dữ liệu mức dưới đỉnh (mức 2 và dưới 2)

– Ở mức này thực hiện phân rã đối với mỗi chức năng của mức đỉnh.

– Khi thực hiện mức phân rã này vẫn phải căn cứ vào mô hình phân rã chức năng để xác định các chức năng con sẽ xuất hiện trong mô hình luồng dữ liệu.

– Việc phân rã có thể tiếp tục cho đến khi đủ số mức cần thiết

– Khi phân rã các chức năng phải đảm bảo tất cả các luồng thông tin vào ra ở chức năng mức cao phải có mặt trong các chức năng mức thấp hơn và ngược lại.

Chú ý:

+ Các kho dữ liệu không xuất hiện ở DFD mức khung cảnh.

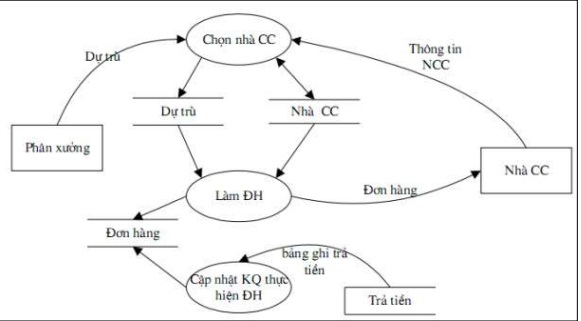
+ Nên đánh số các chức năng theo sự phân cấp.

+ Các kho dữ liệu, các tác nhân ngoài có thể xuất hiện nhiều lần.

+ Số mức phụ thuộc vào độ phức tạp của hệ thống.

Ví dụ 1: Mô hình luồng dữ liệu của hệ thống cung ứng vật tư mức dưới đỉnh của

Chức năng đặt hàng



Hình e:Mô hình luồng dữ liệu mức dưới đỉnh của chức năng đặthàng

**Ví dụ 2: vẽ biểu đồ phân cấp chức năng và biểu đồ luồng dữ liệu phân mức của một cơ sở tín dụng:**

**– Biểu đồ phân cấp chức năng:**

**A diagram of a company

Description automatically generated**

**– Biểu đồ luồng dữ liệu:**

**+ Mức khung cảnh:**

**A diagram of a person's face

Description automatically generated**

**+ Mức đỉnh:**

**A diagram of a diagram

Description automatically generated with medium confidence**

**+ Mức dưới đỉnh**

**A diagram of a flowchart

Description automatically generated**

**5. Chuyển từ mô hình luồng dữ liệu vật lý sang mô hình luồng dữ liệu logic**

Việc chuyển từ mô hình luồng dữ liệu vật lý sang mô hình luồng dữ liệu logic có tác dụng sau:

– Xác định nhu cầu thông tin ở mỗi chức năng

– Cho một thiết kế sơ bộ về thực hiện chức năng

– Là phương tiện giao tiếp giữa người phân tích thiết kế và người sử dụng

– Luôn có hai mức diễn tả vật lý và logic. Mức vật lý trả lời câu hỏi như thế nào, mức lôgíc trả lời câu hỏi làm gì.

Trong thực tế người ta thấy rằng việc tạo ra một mô hình luồng dữ liệu cho hệ thống thực dưới dạng vật lý không có lợi vì những lý do sau:

+ Tốn nhiều thời gian và tiêu tốn nguồn tài nguyên phát triển dự án một cách không cần thiết. Có thể xem quá trình này là việc sao chép công việc của kỹ thuật viên điều tra, sao chép tất cả những gì đang thực hiện hiện tại.

+ Khi tạo ra mô hình thì phải tạo ra những điều chỉnh tượng trưng cho nó, xử lý nó như mô hình logic, kết quả là hệ thống mới chỉ đơn thuần là tin học hoá hệ thống cũ với rất nhiều lỗi mà cái ta cần cuối cùng là mô hình DFD logic.

Mô hình luồng dữ liệu logic loại bỏ những ràng buộc và các yếu tố vật lý, mô hình này chỉ quan tâm chức năng nào là cần cho hệ thống và thông tin nào là cần để thực hiện cho chức năng đó.

Các yếu tố vật lý cần loại bỏ là:

+ Các phương tiện, phương thức: tự động, thủ công, bàn phím, màn hình,..

+ Các giá mang thông tin: các tệp, chứng từ

+ Các chức năng xử lý gắn với các công cụ hay cách thức cài đặt cụ thể

+ Tiến hành các loại bỏ và chỉnh đốn lại cấu trúc. Loại bỏ: loại bỏ các ngôn từ, hình vẽ biểu diễn các phương tiện, giá mang tin,.. giữ lại các chức năng và nội dung thông tin

– Chú ý:

+ Nên xây dựng mô hình logic cần có bằng cách điều chỉnh mô hình logic thực tại.

+ Không có sự phân chia rõ rệt giữa logic và vật lý. Mô hình càng phân rã ở mức thấp thì càng thêm nhiều yếu tố vật lý.

+ Càng giữ cho mô hình của mình được logic nhiều nhất khi đi sâu vào chi tiết càng tốt.

**6. Chuyển từ mô hình luồng dữ liệu của hệ thống cũ sang mô hình luồng dữ liệu của hệ thống mới**

– Giai đoạn này có ý nghĩa vô cùng quan trọng ảnh hưởng to lớn đến sự thành công của hệ thống mới.

– Trong giai đoạn này nhà quản lý và nhà phân tích phải hợp tác chặt chẽ để tìm cách hoà hợp cơ cấu tổ chức, nhận thức được vai trò của máy tính để thay đổi hệ thống cũ.

– Để chuyển từ mô hình luồng dữ liệu của hệ thống cũ sang mô hình luồng dữ liệu của hệ thống mới trước tiên phải xác định các mặt yếu kém cần cải tiến, thay đổi trong hệ thống cũ.

– Các yếu kém chủ yếu do sự thiếu vắng gây ra : thiếu vắng về cơ cấu tổ chức hợp lý, thiếu vắng các phương tiện hoạt động từ đó dẫn đến hiệu quả hoạt động thấp, chi phí hoạt động cao.

– Xem lại mô hình luồng dữ liệu

+ Nếu thiếu vắng thì bổ sung

+ Nếu thừa thì loại bỏ

+ Nếu thay đổi bắt đầu từ mức đỉnh

* Khoanh vùng vùng sẽ được thay đổi
* Giữ nguyên các luồng vào và luồng ra của vùng
* Xác định chức năng tổng quát của vùng
* Xoá bỏ mô hình luồng dữ liệu bên trong vùng được khoanh, lập lại các chức năng từ mức thấp nhất.
* Thành lập kho dữ liệu và luồng dữ liệu cần thiết.

– Sửa lại mô hình phân rã chức năng theo mô hình luồng dữ liệu.

– Kiểm tra lại các mô hình dữ liệu điều chỉnh lại cho hợp lý.

Ví dụ: Hệ cung ứng vật tư

– Nhược điểm: thiếu kho hàng thông dụng

+ Tốc độ chậm vì có khâu đối chiếu thủ công

+ Theo dõi thực hiện đơn hàng còn nhiều sai sót

+ Lãng phí do đối chiếu thủ công

– Sửa mô hình luồng dữ liệu

+ Bổ sung: Kho chứa vật tư

+ Sửa lại mô hình luồng dữ liệu của hệ thống

**7. Hoàn chỉnh mô hình luồng dữ liệu**

Khi đã hoàn thành sơ đồ luồng dữ liệu cần kiểm tra về tính đầy đủ và nhất quán của nó. Phải làm cho sơ đồ đơn giản, chính xác và logic nhất có thể được. Nên tránh để xảy ra các tình huống sau:

– Tình huống 1: Hiệu ứng “mặt trời bừng sáng” tức là một chức năng có quá nhiều dòng vào, ra. Cách khắc phục tình huống này như sau: Gom nhóm hoặc phân rã tiếp một số chức năng chưa hợp lý.

– Tình huống 2: Thông tin đi qua một chức năng mà không bị thay đổi

Cách khắc phục tình huống này như sau: xoá bỏ chức năng không biến đổi thông tin.

Nếu xuất hiện một chức năng có các chức năng con không có liên quan về dữ liệu (không có dòng thông tin nội bộ gắn với nhau hoặc không sử dụng kho dữ liệu chung) thì việc phân bố sơ đồ phân rã chức năng là chưa hợp lý, lúc này cần phải xem xét lại. Ví dụ:

A diagram of a company structure

Description automatically generated

Tác dụng của việc hoàn chỉnh mô hình luồng dữ liệu:

– Xác định nhu cầu thông tin ở mỗi chức năng

– Cho một thiết kế sơ bộ về thực hiện chức năng

– Là phương tiện giao tiếp giữa người phân tích thiết kế và người sử dụng

– Luôn có hai mức diễn tả vật lý và lôgíc. Mức vật lý trả lời câu hỏi như thế nào, mức lôgíc trả lời câu hỏi làm gì

**2.2 Trình bày mô hình thực thể liên kết. Cho ví dụ**

**(\*).Một số nét cơ bản của mô hình dữ liệu thực thể - liên kết**

Mô hình dữ liệu thực thể - liên kết (gọi tắt là mô hình thực thể liên kết) sử dụng lại một số khái niệm của mô hình quan hệ của Codd và đưa thêm một số khái niệm vào để mô hình được đơn giản hơn, dễ hiểu hơn, dễ triển khai hơn. Một cách khái quát mô hình thực thể - liên kết được xây dựng như sau:

* Dữ liệu được tổ chức theo các thực thể : CSDL gồm các bảng, mỗi bảng chứa các dữ liệu của một thực thể.
* Mỗi bảng gồm nhiều cột, mỗi cột tương ứng với một thuộc tính của thực thể.
* Mỗi bảng gồm nhiều dòng, mỗi dòng tương ứng với một cá thể.
* Giữa các thực thể có mối liên kết với nhau. Để thực hiện mối liên kết này, trong mỗi thực thể chọn một thuộc tính làm khóa chính. Thuộc tính này sẽ được đưa vào bảng có liên kết làm khóa ngoại (các thuộc tính khóa chính và khóa ngoại sẽ được trình bày chi tiết hơn trong phần sau).
* Mô hình của CSDL được thể hiện trên một hình vẽ.

**(\*).Lược đồ của mô hình thực thể - liên kết**

Một số quy ước

Dưới đây, chúng tôi đề xuất một số quy ước sử dụng trong phần cơ sở dữ liệu của website này. Những quy ước này được lựa chọn dựa theo sự phổ biển trong các tài liệu và phần mềm mà chúng tôi tham khảo được. Các quy ước này được minh họa trong Hình 1.

* Thực thể được biểu diễn bằng một hình chữ nhật có ghi tên của thực thể (Hình 1a và 1b). Tùy theo điều kiện không gian, có thể ghi tên các khóa, các thuộc tính hay không.
* Liên kết được biểu diễn bằng một hình thoi nối đến các thực thể tham gia liên kết (Hình 1b).
* Bản số của thực thể tham liên kết được biểu diễn bằng một cặp số như 0:N, trong đó số đầu và số sau lần lượt là số cá thể bé nhất, và lớn nhất tham gia vào liên kết (Hình 1a).
* Đối với những cơ sở dữ liệu phức tạp, có nhiều thực thể, nhiều liên kết, người ta thường sử dụng lược đồ ở dạng rút gọn bằng cách bỏ các hình thoi biểu diễn các liên kết. Liên kết lúc bấy giờ được biểu diễn bằng một hay vài đoạn thẳng nối hai thực thể có liên kết (Hình 1b). Đồng thời, thay vì một cặp số như trên Hình 1a, bản số được trình bày theo ký hiệu "vết chân chim" (crow's foot) như trên Hình 1b. Các quy ước về bản số theo cách này được trình bày trên Hình 1c.

A diagram of a computer

Description automatically generated

Hình 1 Một số quy ước về lược đồ của mô hình dữ liệu thực thể - liên kết

**(\*). thí dụ**

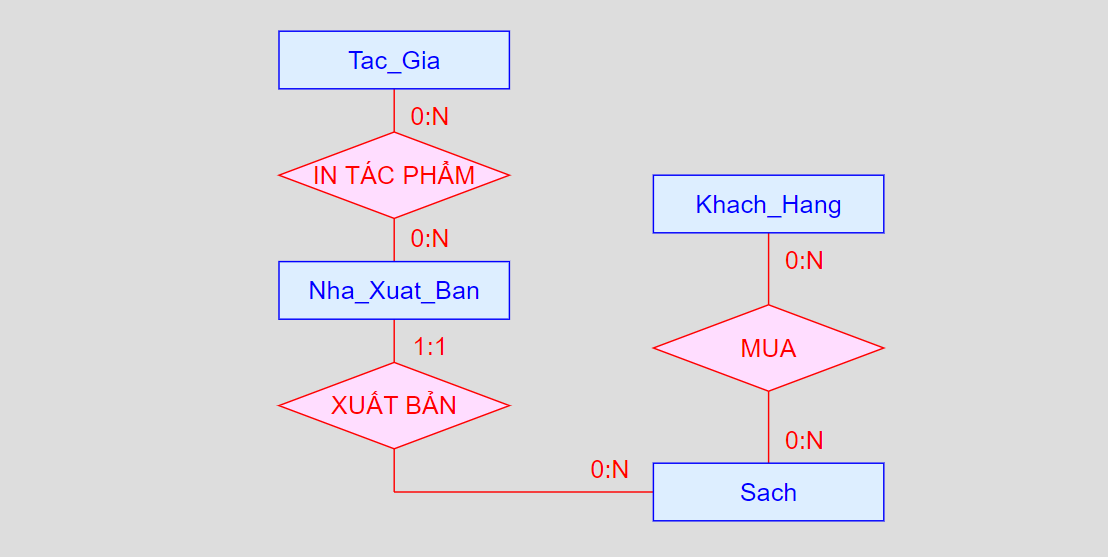
Để minh họa cho những điều vừa trình bày trên, ta xét một nhà sách bán hàng qua mạng với một vài nghiệp vụ đơn giản. Các tác giả sáng tác và gửi cho nhà xuất bản, nhà xuất bản in rồi cung cấp cho nhà sách, và nhà sách bán cho khách hàng.

Dựa vào sự mô tả này, để xây dựng cơ sở dữ liệu, ta có thể chọn ra các thực thể là Sach, Tac\_Gia, Nha\_Xuat\_Ban, Khach\_Hang. Trong bước đầu này, ta chỉ tập trung vào thực thể và liên kết, tạm thời chưa xem xét đến thuộc tính.

Giữa các thực thể này có những mối quan hệ được thể hiện qua các liên kết sau:

* Giữa tác giả và nhà xuất bản có liên kết IN TÁC PHẨM. Vì mỗi tác giả có thể có sách được in bởi một số nhà xuất bản và mỗi nhà xuất bán có thể in tác phẩm từ nhiều tác giả khác nhau nên đây là quan hệ nhiều-nhiều. Tuy vậy có những nhà xuất bản không in được tác phẩm từ bất kỳ tác giả nào nên bản số ở phía Tac\_Gia là 0:N. Và cũng có những tác giả không được bất kỳ nhà xuất bản nào in sách nên bản số ở phía Nha\_Xuat\_Ban cũng là 0:N.
* Giữa nhà xuất bản và sách có quan hệ là XUẤT BẢN. Vì mỗi nhà xuất bản có thể xuất bản nhiều tựa sách nhưng mỗi tựa sách chỉ có thể xuất bản bởi một nhà xuất bản nên quan hệ này là quan hệ một-nhiều. Do mỗi tựa sách phải được xuất bản bởi một nhà xuất bản nên bản số phía Nha\_Xuat\_Ban là 1:1. Nhưng cũng có nhà xuất bản không xuất bản được tựa sách nào nên bản số phía Sach là 0:N.
* Giữa khách hàng và sách có quan hệ là MUA. Vì mỗi khách hàng có thể mua nhiều tựa sách và mỗi tựa sách có thể được bán cho nhiều khách hàng nên liên kết này là liên kết nhiều-nhiều. Do có những tựa sách không được khách hàng nào mua nên bản số phía Khach\_Hang là 0:N. Và cũng có những khách hàng không mua tựa sách nào nên bản số phía Sach cũng là 0:N.

Từ những suy luận trên, bước đầu ta có lược đồ của cơ sở dữ liệu này theo mô hình thực thể - liên kết được thể hiện trên Hình 2. Do chưa xem xét đến các thuộc tính nên trong lược đồ này chỉ trình bày các thực thể và liên kết.



Hình 2: Lược đồ cơ sở dữ liệu theo mô hình thực thể - liên kết

Nếu ta trình bày lại lược đồ trên ở dạng rút gọn, ta có Hình 3.

A diagram of a computer network

Description automatically generated

Hình 3: Lược đồ cơ sở dữ liệu theo mô hình thực thể - liên kết ở dạng rút gọn

**PHẦN 3: Phân tích và thiết kế website bán đồ ăn quê**

### **3.1: Xác định các chức năng của hệ thống website bán đồ ăn quê**

#### Chức năng chính của hệ thống:

* **Quản lý người dùng**:

**Đăng ký người dùng**: Cho phép người dùng tạo tài khoản mới.

**Đăng nhập/Đăng xuất**: Người dùng có thể đăng nhập và thoát khỏi hệ thống.

**Cập nhật thông tin cá nhân**: Người dùng có thể thay đổi thông tin cá nhân, như địa chỉ giao hàng hoặc mật khẩu.

* + **Quản lý sản phẩm**:

**Hiển thị danh mục sản phẩm**: Danh sách các món ăn quê, bao gồm tên món, giá, hình ảnh.

**Lọc sản phẩm**: Cho phép người dùng lọc món ăn theo loại (mặn, ngọt, chay).

**Chi tiết sản phẩm**: Cung cấp thông tin chi tiết như mô tả, giá, thành phần, hình ảnh.

* + **Giỏ hàng**:

**Thêm món vào giỏ**: Người dùng có thể chọn món ăn và thêm vào giỏ hàng.

**Cập nhật số lượng**: Người dùng có thể thay đổi số lượng món ăn trong giỏ hàng.

**Xóa món khỏi giỏ**: Người dùng có thể xóa món ăn khỏi giỏ hàng.

* **1.4 Đặt hàng và thanh toán**:

**Đặt hàng**: Người dùng có thể chọn món ăn, nhập địa chỉ giao hàng và đặt hàng.

**Thanh toán**: Người dùng thanh toán qua các phương thức như chuyển khoản, thanh toán khi nhận hàng.

* **1.5 Quản lý đơn hàng**:

**Theo dõi đơn hàng**: Người dùng có thể xem trạng thái đơn hàng (chờ xử lý, đang giao, đã giao).

**Lịch sử đơn hàng**: Người dùng có thể xem lại các đơn hàng đã đặt.

* **1.6 Quản lý giao hàng và nhân viên**:

**Quản lý đơn hàng**: Nhân viên quản lý đơn hàng, xác nhận đơn hàng và giao hàng.

**Cập nhật trạng thái giao hàng**: Nhân viên giao hàng cập nhật tình trạng (đang giao, đã giao, hủy).

**Mô hình phân rã chức năng**

**3.2.Mô hình dữ liệu mức khung cảnh**

Cơ sở dữ liệu

Lưu trữ thông tin khách

hàng, sản phẩm