

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG.....**



## **Luận văn**

**Ứng dụng phương pháp phân tích thiết kế  
hướng cấu trúc vào bài toán xây dựng Website  
đăng ký và theo dõi sinh viên thực tập tốt  
nghệ tại trường đại học Dân Lập Hải Phòng**



## LỜI CẢM ƠN

Trước hết em xin bày tỏ tình cảm và lòng biết ơn đối với Th.S Nguyễn Thị Thanh Thoan – Bộ môn Công nghệ thông tin – Trường Đại học Dân Lập Hải Phòng, người đã dành cho em rất nhiều thời gian quý báu, trực tiếp hướng dẫn tận tình giúp đỡ, chỉ bảo em trong suốt quá trình làm đồ án tốt nghiệp.

Em xin chân thành cảm ơn tất cả các thầy cô giáo trong bộ môn công nghệ thông tin – Trường Đại học Dân lập Hải Phòng, chân thành cảm ơn các thầy giáo, cô giáo tham gia giảng dạy và truyền đạt những kiến thức quý báu trong suốt thời gian em học tập tại trường, đã đọc và phản biện đồ án của em giúp em hiểu rõ hơn vấn đề mình nghiên cứu, để em có thể hoàn thành tốt đồ án này.

Em xin chân thành cảm ơn GS.TS. NGUYỄN Trần Hữu Nghị hiệu trưởng nhà trường Đại học Dân lập Hải Phòng, ban giám hiệu nhà trường, Bộ môn tin học, các phòng ban nhà trường đã giúp đỡ tạo điều kiện tốt nhất cho em trong suốt thời gian học tập cũng như quá trình làm tốt nghiệp.

Tuy có nhiều cố gắng trong quá trình học tập, trong quá trình thực tập và trong quá trình làm đồ án tốt nghiệp nhưng không thể tránh khỏi những thiếu sót, em rất mong được sự góp ý quý báu của tất cả các thầy cô giáo cũng như tất cả các bạn để kết quả của em được hoàn thiện hơn.

***Em xin chân thành cảm ơn!***

Hải phòng, Ngày 22 tháng 10 năm 2010  
Sinh viên

**Đào Bá Bản**

# MỤC LỤC

<b>CHƯƠNG I: CƠ SỞ LÝ THUYẾT</b> .....	3
1.1 - Tổng quát về HTTT.....	3
1.2 - Phương pháp phân tích thiết kế hướng cấu trúc.....	4
<b>CHƯƠNG II: KHẢO SÁT HIỆN TRẠNG</b> .....	7
2.1 – Khảo sát hiện trạng .....	7
2.2 – Đề xuất.....	10
2.3 – Phát biểu bài toán.....	10
2.4 - Sơ đồ hoạt động.....	12
2.5 - Biểu đồ ngữ cảnh của hệ thống .....	14
2.6 - Biểu đồ phân rã chức năng nghiệp vụ hệ thống .....	16
2.7 - Danh sách hồ sơ dữ liệu sử dụng.....	17
2.8 - Ma trận thực thể chức năng .....	18
<b>CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG</b> .....	19
3.1 Sơ đồ luồng dữ liệu mức 0.....	19
3.2 - Sơ đồ luồng dữ liệu mức 1 .....	20
3.3 – Sơ đồ luồng dữ liệu mức 2.....	23
3.4 - Phân tích dữ liệu hệ thống Website đăng ký và theo dõi sinh viên TT.....	24
3.4.1 - Các khái niệm cơ bản.....	24
3.4.2 - Các thực thể .....	24
3.4.3 – Mô hình ER.....	25
3.5 – Thiết kế cơ sở dữ liệu: .....	26
3.5.1 - Các bảng trong Cơ sở dữ liệu: .....	26
3.5.2 – Mô hình quan hệ: .....	27
3.5.3 - Cấu trúc vật lý của các bảng cơ sở dữ liệu: .....	28
3.6 - Thiết kế giao diện.....	31
<b>CHƯƠNG IV: CÀI ĐẶT CHƯƠNG TRÌNH</b> .....	38
4.1 – Tổng quan về ASP .....	38
4.2. VISUALBASIC SCRIPT LANGUAGE (VBSCRIPT) .....	50
4.3: QUẢN LÝ CƠ SỞ DỮ LIỆU VỚI ASP .....	57
4.4 SQL SERVER.....	62
4.5 - XÂY DỰNG CHƯƠNG TRÌNH.....	65
<b>KẾT LUẬN</b> .....	71
<b>Tài liệu tham khảo</b> .....	72

# CHƯƠNG I

## CƠ SỞ LÝ THUYẾT

### 1.1 - Tổng quát về HTTT

Phát triển một hệ thống thông tin (HTTT) là quá trình tạo ra một HTTT cho một tổ chức. Quá trình đó bắt đầu từ khi nêu vấn đề cho đến khi đưa hệ thống vào vận hành trong tổ chức. Với thực tiễn của nhiều năm xây dựng các HTTT, những cơ sở phương pháp luận phát triển HTTT đã không ngừng được hoàn thiện và bổ sung cho phù hợp với sự phát triển của công nghệ và các điều kiện môi trường đã biến đổi. Cho đến nay, phương pháp phát triển HTTT hướng cấu trúc đã đạt đến mức hoàn hảo.

\* Khái niệm và định nghĩa.

Có nhiều định nghĩa về hệ thống thông tin khác nhau và các khái niệm liên quan. Về mặt kỹ thuật, HTTT được xác định như một tập hợp các thành phần được tổ chức để thu thập, xử lý, lưu trữ, phân phối và biểu diễn thông tin, trợ giúp việc ra quyết định và kiểm soát hoạt động trong một tổ chức. Ngoài các chức năng kể trên, nó còn có thể giúp người quản lý phân tích các vấn đề, cho phép nhìn thấy được một cách trực quan những đối tượng phức tạp, tạo ra các sản phẩm mới. Liên quan đến HTTT là các khái niệm sẽ đề cập đến như dữ liệu, thông tin, hoạt động thông tin, xử lý dữ liệu, giao diện,...

Dữ liệu (Data) là những mô tả về sự vật, con người và sự kiện trong thế giới mà chúng ta gặp bằng nhiều cách thể hiện khác nhau, như bằng ký tự, chữ viết, biểu tượng, hình ảnh, âm thanh, tiếng nói,...

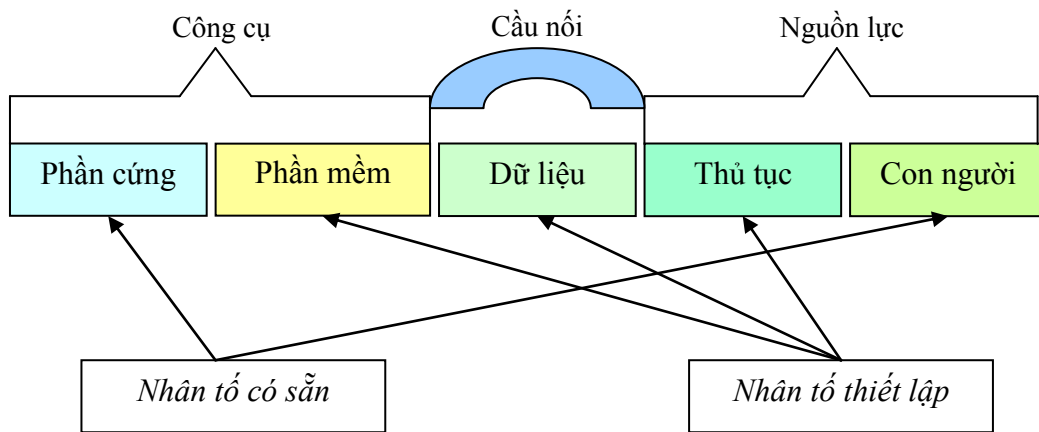
Thông tin (Information) cũng như dữ liệu, đã có nhiều cách định nghĩa khác nhau. Một định nghĩa bao trùm hơn cả, xem thông tin là dữ liệu được đặt vào một ngữ cảnh với một hình thức thích hợp và có lợi cho người sử dụng cuối cùng.

Các hoạt động thông tin (Information Activities) là các hoạt động xảy ra trong một HTTT, bao gồm việc nắm bắt, xử lý, phân phối, lưu trữ, trình diễn dữ liệu và kiểm tra các hoạt động trong HTTT.

Xử lý (processing) dữ liệu được hiểu là các hoạt động tác động lên dữ liệu như tính toán, so sánh, sắp xếp, phân loại, tổng hợp... làm cho nó thay đổi về nội dung, vị trí hay cách thể hiện.

Giao diện (interface) là chỗ mà tại đó hệ thống trao đổi dữ liệu với hệ thống khác hay môi trường. Ví dụ: giao diện của một HTTT thường là màn hình, bàn phím, chuột, micro, loa, hay card mạng...

Môi trường (environment) là phần của thế giới không thuộc hệ thống mà có tương tác với hệ thống thông qua các giao diện của nó.



Các yếu tố cấu thành của HTTT

Năm yếu tố cấu thành của HTTT quản lý xét ở trạng thái tĩnh là: Thiết bị tin học (máy tính, các thiết bị, các đường truyền, ... phần cứng), các chương trình (phần mềm), dữ liệu, thủ tục, quy trình và con người. Các định nghĩa về HTTT trên đây giúp cho việc định hướng quá trình phân tích, thiết kế hệ thống. Tuy vậy, sự mô tả này là chưa đủ, cần đi sâu phân tích hệ thống cụ thể mới có được sự hiểu biết đầy đủ về một hệ thống thực và cho phép ta xây dựng cơ sở dữ liệu, các chương trình và việc bố trí các thành phần bên trong nó.

\* Hệ thống quản lý.

Trong các HTTT, HTTT quản lý (management information system - MIS) được biết đến sớm và phổ biến nhất. Đối tượng phục vụ của HTTT quản lý thực sự rộng hơn rất nhiều so với ý nghĩa bản thân tên gọi của nó. HTTT quản lý là sự phát triển và sử dụng HTTT có hiệu quả trong một tổ chức.

## 1.2 - Phương pháp phân tích thiết kế hướng cấu trúc.

### 1.2.1- Khái niệm.

Tiếp cận hướng cấu trúc hướng vào việc cải tiến cấu trúc các chương trình dựa trên cơ sở module hoá để dễ theo dõi, quản lý và bảo trì. Phát triển hướng cấu trúc đề

cập đến quá trình sử dụng một cách có hệ thống và tích hợp các công cụ và kỹ thuật để trợ giúp thiết kế và phân tích HTTT theo hướng module hoá.

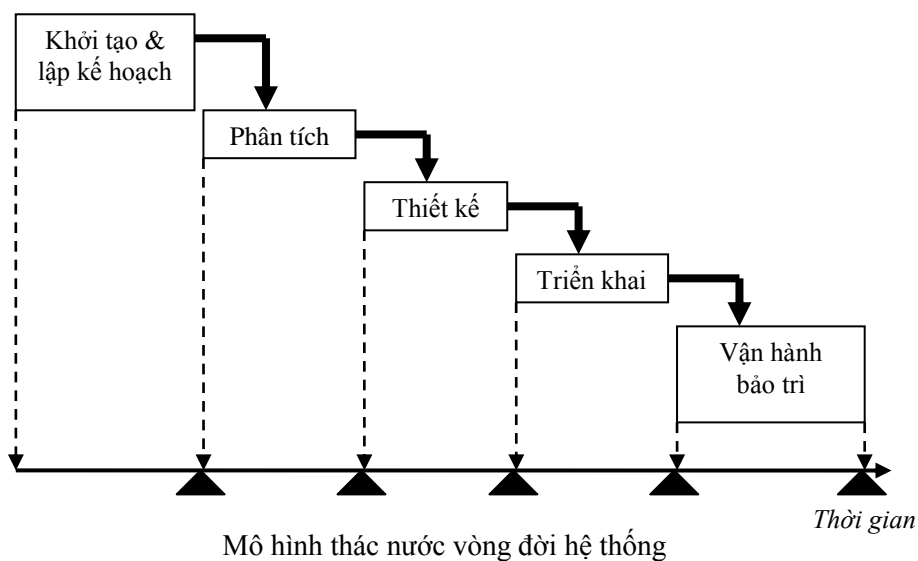
### 1.2.2 - Vòng đời phát triển một HTTT

HTTT được xây dựng là sản phẩm của một loạt các hoạt động được gọi là phát triển hệ thống. Quá trình phát triển HTTT kể từ lúc nó sinh ra đến khi nó tàn lụi được gọi là vòng đời phát triển hệ thống. Vòng đời phát triển hệ thống là một phương pháp luận cho việc phát triển HTTT. Nó được đặc trưng bằng một số pha chủ yếu phân biệt nhau của quá trình đó: xác định yêu cầu, phát triển và tiến hoá HTTT.

Nhiều mô hình vòng đời được sắp xếp các bước phát triển hệ thống theo mô hình bậc thang, cách biểu diễn này giống với mô hình thác nước. Mô hình này sẽ thể hiện với phương pháp luận chung, và bao gồm các pha: khởi tạo và lập kế hoạch, phân tích, thiết kế, triển khai, vận hành và bảo trì HTTT: ở mỗi pha đều có cái vào và cái ra. Chúng thể hiện mối quan hệ quan trọng giữa các pha với nhau. Đây chính là đặc trưng của quá trình quản lý sự phát triển HTTT.

Việc hình thành dự án như một yêu cầu bắt buộc để có thể tiến hành những bước tiếp theo của quá trình phát triển.

Khởi tạo và lập kế hoạch dự án công việc ban đầu và chính thức về những vấn đề của hệ thống. Một kế hoạch dự án phát triển HTTT được mô tả theo vòng đời phát triển hệ thống, đồng thời cũng đưa ra các ước lượng thời gian và các nguồn lực cần thiết.



- Khả thi kỹ thuật: xem xét khả năng kỹ thuật hiện có, đủ đảm bảo các giải pháp công nghệ thông tin được áp dụng để phát triển hệ thống.

- Khả thi kinh tế: khả năng tài chính, lợi ích mang lại, chi phí thường xuyên cho hệ thống hoạt động.

- Khả thi về thời gian: dự án được phát triển trong thời gian cho phép, và lịch trình thực hiện trong giới hạn đã cho.

- Khả thi pháp lý và hoạt động: hệ thống vận hành tốt trong khuôn khổ tổ chức và điều kiện quản lý mà tổ chức có và khuôn khổ pháp lý hiện hành.

### **1.2.3 - Ưu nhược điểm của phương pháp phân tích hướng cấu trúc:**

Ưu điểm:

- Phát triển hoàn thiện từ rất lâu rồi.
- Các hệ quản trị cơ sở dữ liệu phát triển phục vụ cho lập trình hướng cấu trúc.
- Làm giảm sự phức tạp (nhờ chia nhỏ, môđun hoá).
- Tập trung vào ý tưởng (vào logic, kiến trúc trước khi thiết kế).
- Chuẩn mực hoá (theo các phương pháp, công cụ đã cho).
- Hướng về tương lai (kiến trúc tốt, môđun hoá dễ bảo trì).
- Giảm bớt tính nghệ thuật trong thiết kế (phát triển hệ thống phải tuân thủ một tiến trình xác định với các quy tắc và phương pháp đã cho).

Nhược điểm:

- Chỉ áp dụng cho những bài toán nhỏ. Đối với những bài toán lớn phải dùng phương pháp phân tích hướng đối tượng.

## **CHƯƠNG II**

### **KHẢO SÁT HIỆN TRẠNG**

#### **2.1 – Khảo sát hiện trạng**

Ngày nay, Internet và website đã không còn là những thuật ngữ tin học quá xa lạ với chúng ta. Với tốc độ phát triển mạnh mẽ ngày nay, Internet đã chứng tỏ nó đang trở thành một yếu tố không thể thiếu được trong cuộc sống hiện đại. Với Internet, con người ngày càng có cơ hội để phát triển về mọi mặt. Nhiều hoạt động sống của con người đã, đang và sẽ ngày càng gắn bó với Internet. Internet cung cấp cho bạn những cơ hội giao thương, gặp gỡ đối tác nước ngoài một cách thuận lợi. Internet mang những thông tin quảng bá khuyến mãi đến người sử dụng một cách nhanh chóng và hiệu quả. Internet giúp bạn điều hành công việc và giữ liên lạc mật thiết với những người bạn quan tâm khi khoảng cách là một trở ngại lớn...Hàng ngày, Internet không chỉ mang đến cho chúng ta những thông tin nóng bỏng nhất, những lợi ích kinh doanh tốt nhất, những hình thức giải trí mới lạ, đa dạng, phong phú mà nó còn cung cấp cho chúng ta một cách học tập nâng cao trình độ với những Website sinh động, đặc sắc với các biểu mẫu, đồ họa, âm thanh, ảnh động...

Nhận thức tầm quan trọng của Internet đối với ứng dụng trong các trường đại học và cao đẳng

Qua mỗi khoá học sinh viên phải trải qua quá trình thực tập. Sau khi hoàn thành các trương trình học của nhà trường, những sinh viên có đủ điều kiện sẽ được đi thực tập tốt nghiệp.

+Sinh viên đăng ký thực tập được viết vào phiếu đăng ký thực tập rồi gửi lại cho bộ môn của từng ngành học.


+Trong quá trình thực tập sinh viên phải viết các bài báo cáo cuối tuần để giáo viên theo dõi nhận xét và đánh giá. Đối với những sinh viên có cơ sở thực tập ở xa sinh viên sẽ gặp nhiều khó khăn trong việc báo cáo kết quả thực tập của từng tuần và nhiệm vụ cho tuần tiếp theo.

+Trong quá trình viết nhật ký của sinh viên: những thông tin, bài viết của sinh viên mọi người không theo dõi được.



+Đăng ký và quản lý các danh sách các sinh viên thực tập bằng tay các văn bản giấy tờ thông thường sẽ tốn nhiều công sức mà hiệu quả trong việc quản lý không được cao.

Công tác chuẩn bị cho kỳ đi thực tế tại các cơ sở thực tập một cách tốt nhất để cho sinh viên có thể sau khi ra trường đáp ứng được các công việc được giao một cách hiệu quả nhất.

 <p>ISO 9001:2008</p>	<p>BỘ GIÁO DỤC VÀ ĐÀO TẠO <b>TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG</b></p> <p>Số: ...../...../QĐ-HT</p>	<p>CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM <u>Độc lập – Tự do – Hạnh phúc</u></p> <p>Hải Phòng, ngày ... tháng ... năm .....</p>
--	--	---

## QUYẾT ĐỊNH

**V/v: Cho phép sinh viên khoá .....ngành..... đi thực tập tốt nghiệp**

### HIỆU TRƯỞNG TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

**Căn cứ :**

- Quyết định số 792/QĐ-TTg ngày 24/9/1997 của Thủ tướng Chính phủ về việc thành lập trường Đại học Dân lập Hải Phòng;
- Quyết định số 3027/QĐ-GDĐT ngày 29/9/1997 của Bộ trưởng Bộ Giáo dục và Đào tạo về việc công nhận Hiệu trưởng trường;
- Quyết định số 3028/QĐ-GDĐT ngày 29/9/1997 của Bộ trưởng Bộ Giáo dục và Đào tạo về việc cho phép Trường Đại học Dân lập Hải Phòng chính thức hoạt động;
- Quyết định số 86/2000/QĐ-TTg ngày 18/7/2000 của Thủ tướng Chính phủ về việc ban hành Quy chế Đại học dân lập;
- Quyết định số 25/2006/QĐ-BGDĐT ngày 26 tháng 6 năm 2006 của Bộ trưởng Bộ giáo dục và Đào tạo ban hành quy chế đào tạo Đại học & Cao đẳng hệ chính quy;
- Quyết định 06/2008/QĐ BGDĐT ngày 13/02/2008 của Bộ trưởng Bộ GD&ĐT về việc ban hành quy định đào tạo liên thông trình độ Cao đẳng, Đại học (*Lưu ý: sử dụng đối với quyết định cho sinh viên liên thông*).
- Kế hoạch đào tạo của ngành.....;

## QUYẾT ĐỊNH:

**Điều 1.** Cho phép ..... sinh viên ngành ..... khoá ..... được đi thực tập tại các cơ sở kể từ ngày .../.../..... đến ngày .../.../..... (có danh sách kèm theo)

**Điều 2.** Các sinh viên có nhiệm vụ thực hiện đầy đủ, nghiêm túc các quy định của Bộ Giáo dục và Đào tạo, của Trường Đại học Dân lập Hải Phòng về thực tập tốt nghiệp; chấp hành nghiêm chỉnh Nội quy, Quy định của cơ sở thực tập; giữ gìn uy tín của nhà trường.


**Điều 3.** Các ông, bà Trưởng phòng Đào tạo, Ban Khảo thí và đảm bảo chất lượng, Chủ nhiệm Bộ môn ....., Trưởng phòng Hành chính Tổng hợp, phòng Kế hoạch tài chính, Ban Công tác sinh viên và các sinh viên có tên trong danh sách chịu trách nhiệm thi hành quyết định này./.

**HIỆU TRƯỞNG**

*Nơi gửi:*

- Như điều 3 (để thực hiện);
- Các cơ sở thực tập (để phối hợp);
- Lưu : VT, ĐT.

**GS.TS.NGŨT Trần Hữu Nghị**

 Số 9001/2008	BỘ GIÁO DỤC VÀ ĐÀO TẠO <b>TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG</b>	CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM <i>Độc lập - Tự do - Hạnh phúc</i>
	Số: .....2009/TT-HT	

Hải Phòng, ngày ..... tháng ..... năm .....

## GIẤY GIỚI THIỆU

**Kính gửi:** .....

Trân trọng giới thiệu sinh viên: ..... Lớp: .....

Ngành Công nghệ thông tin, được cử đến Quý cơ quan thực tập Tốt nghiệp.

Thời gian từ : ..... đến .....

Đề nghị Quý cơ quan tạo điều kiện giúp đỡ, tiếp nhận sinh viên:.....

được thực tập, góp phần giúp Nhà trường hoàn thành nhiệm vụ đào tạo nguồn nhân lực cho đất nước

**HIỆU TRƯỞNG**

**GS.TS.NGŨT. Trần Hữu Nghị**

Giấy này có giá trị đến ngày ...../...../.....

## **2.2 – Đề xuất**

Công tác chuẩn bị cho một kỳ thực tập là rất quan trọng. Để giải quyết được các vấn đề trên website thực tập đã đáp ứng được các yêu cầu :

- + Sinh viên đăng ký nhanh, hiệu quả.
- + Các cơ sở đăng ký nhận sinh viên thực tập.
- + Quản lý danh sách các sinh viên thực tập.
- + Sinh viên viết nhật ký và nộp báo cáo thông qua website .
- + Theo dõi quá trình sinh viên thực tập.

## **2.3 – Phát biểu bài toán**

Sau khi sinh viên hoàn thành khoá học sinh viên bắt đầu bước vào kỳ thực tập tốt nghiệp để cho sinh viên phát huy được tính chủ động, sáng tạo, học hỏi sau khi ra trường.

+ Đối với sinh viên được đào tạo theo liên chế: Sau khi hoàn thành hết chương trình học, sinh viên đủ điều kiện được xét đi thực tập tốt nghiệp.

+ Đối với sinh viên đào tạo theo tiến chỉ: Căn cứ vào Quy chế 43 của bộ giáo dục và đào tạo sinh viên đã hoàn thành xong tất cả các tiến chỉ sinh viên đủ điều kiện được xét đi thực tập tốt nghiệp.

Khi đã đủ điều kiện để đi thực tập tốt nghiệp:

+ Sinh viên đăng ký thực tập bao gồm thông tin sau: Mã sinh viên, tên sinh viên, lớp, ngày sinh, Email, ngành, ghi chú.

+ Sinh viên có thể tự liên hệ cơ sở thực tập. Đối với sinh viên chưa liên hệ được với cơ sở thực tập thì nhà trường sẽ liên hệ với các cơ sở thực tập cho sinh viên đăng ký hoặc sinh viên có thể chọn cơ sở thực tập trên website do các cơ sở đăng ký thông tin nhận các sinh viên thực tập ở cơ sở đó.

+ Thông tin của các cơ sở đăng ký bao gồm: Mã công ty, tên công ty, thời gian, vị trí tuyển dụng, địa chỉ, số điện thoại, liên hệ, số lượng, lĩnh vực, mô tả.

Các sinh viên đăng ký thực tập tại cơ sở phải phù hợp với từng ngành học của sinh viên (Phải có ứng dụng của ngành học vào cơ sở đó). Tại các cơ sở thực tập sẽ giới hạn số lượng sinh viên thực tập tại cơ sở đó.

Khi sinh viên hoàn thành thủ tục đăng ký thực tập bộ môn của từng ngành sẽ lập quyết định thực tập cho sinh viên của ngành đó. Thông tin của quyết định thực tập bao gồm: Mã số quyết định, quyết định, ngành, thời gian. Quyết định thực tập chuyển

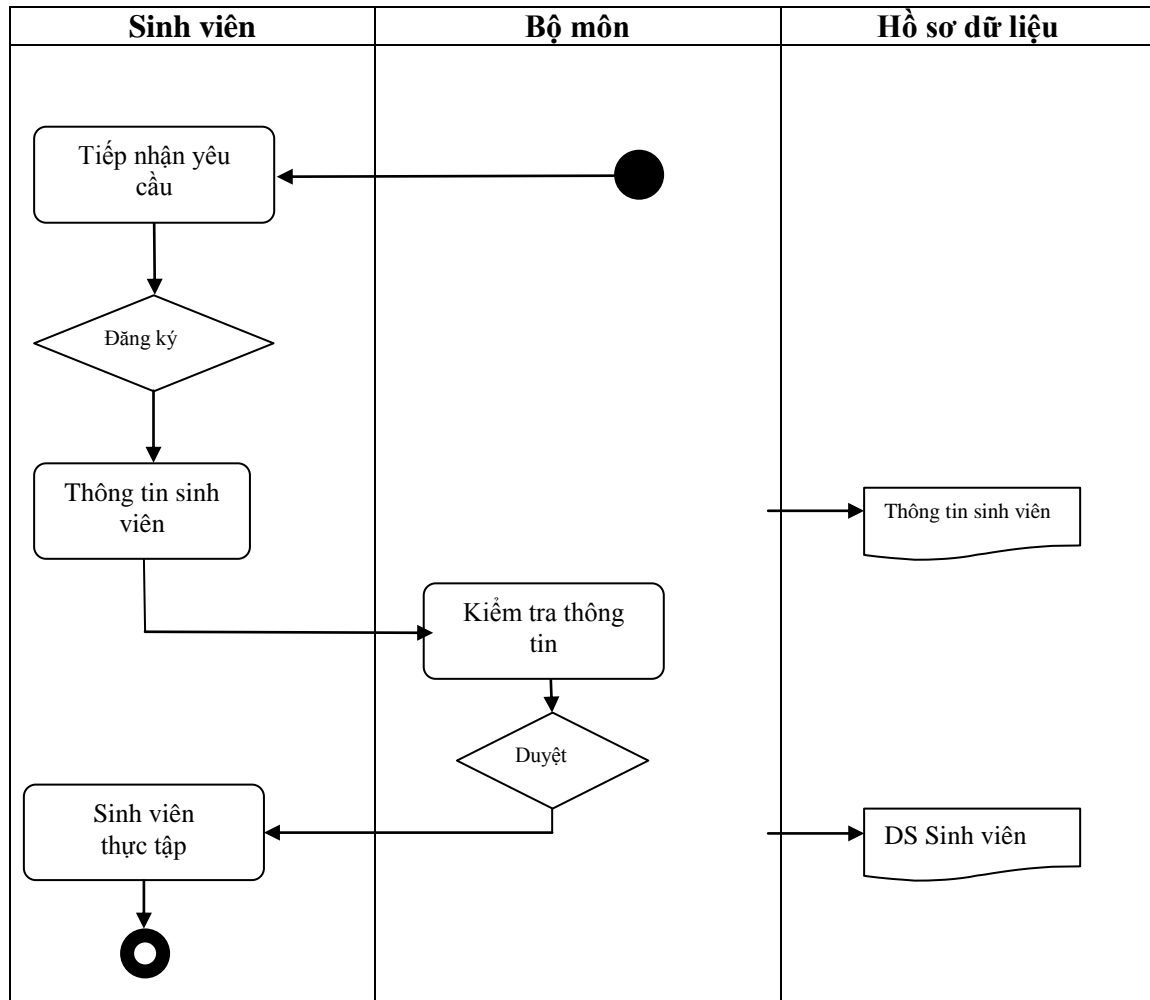
về Ban Giám Hiệu nhà trường ký quyết định thực tập, Ban Giám Hiệu nhà trường gửi trở lại quyết định thực tập về bộ môn. Thông tin của người ký quyết định thực tập bao gồm: họ tên, chức vụ, ghi chú. Danh sách các sinh viên thực tập sẽ được gửi cho giáo viên theo dõi và cơ sở thực tập tiếp nhận sinh viên. Thông tin giáo viên theo dõi bao gồm: Mã giáo viên, họ tên giáo viên, ngành, số lượng sinh viên, ghi chú. Các sinh viên liên hệ với cơ sở thực tập, tiếp nhận yêu cầu thực tập và đang tiến hành thực tập theo đúng tiến độ đã đề ra để đảm bảo chất lượng và theo đúng tiến độ.

Trong quá trình thực tập sinh viên hàng tuần phải viết nhật ký thực tập bằng văn bản về cho giáo viên theo dõi tình hình thực tập tại cơ sở và các vấn đề phát sinh (nếu có) để giáo viên theo dõi và bộ môn của ngành đó kịp thời giải quyết.

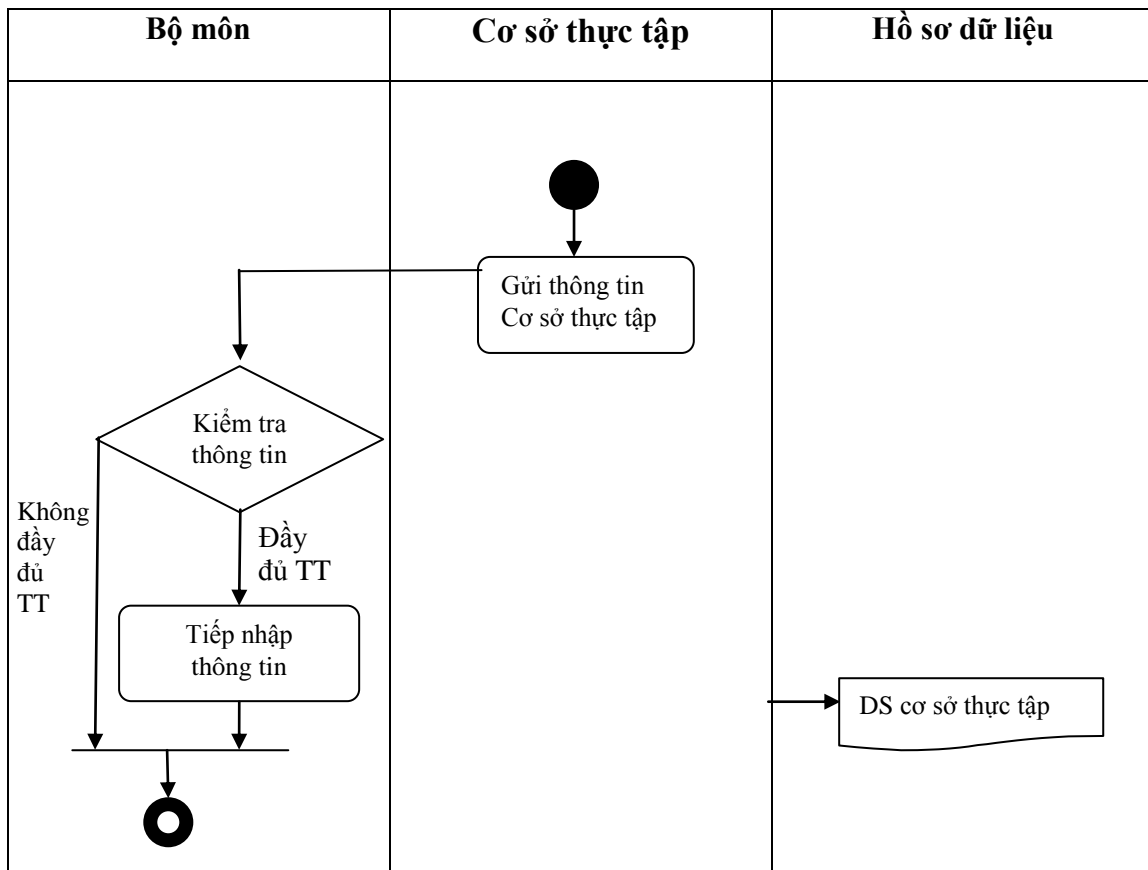
Sau khi hoàn thành viết nhật ký do giáo viên hướng dẫn đề ra, sinh viên báo cáo tình hình thực tập của sinh viên đó tại nơi sinh viên thực tập. Thông tin nội dung báo cáo bao gồm: Mã đề tài báo cáo, tên đề tài báo cáo, nội dung báo cáo, thời gian báo cáo, nhận xét báo cáo.

## 2.4 - Sơ đồ hoạt động

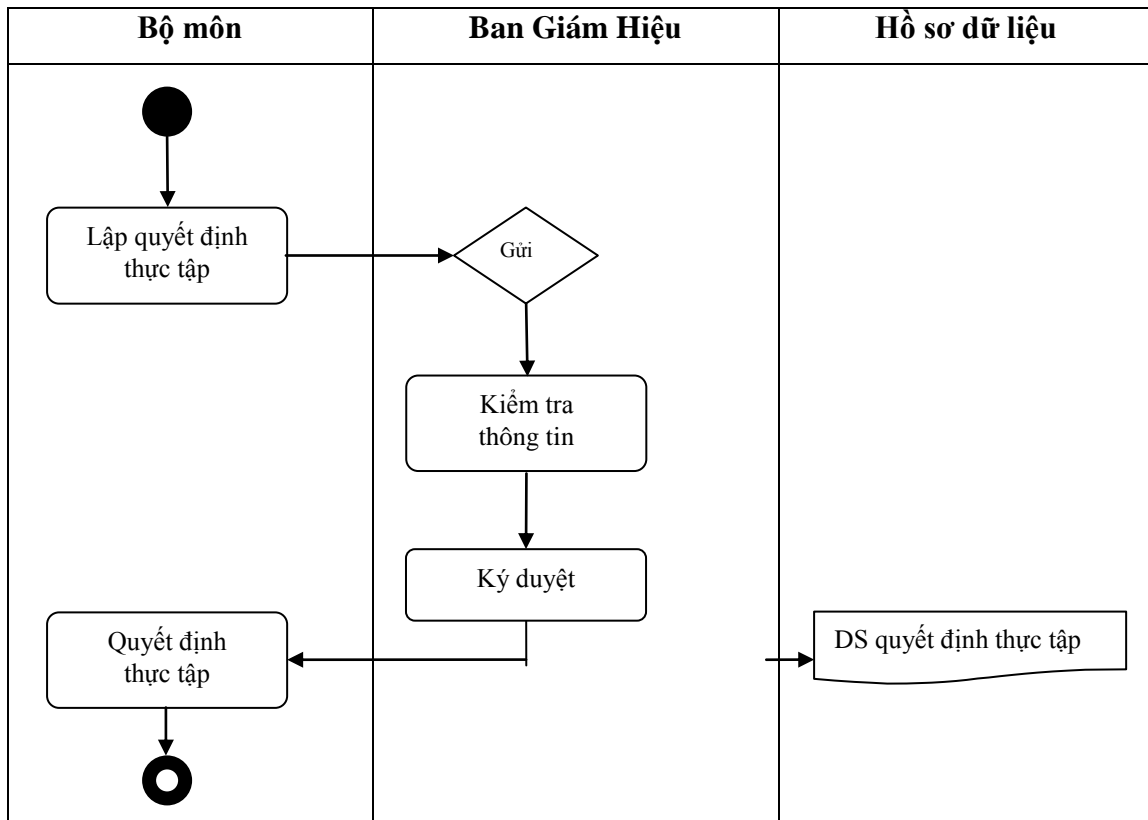
### 2.4.1 - Sơ đồ hoạt động đăng ký của sinh viên thực tập



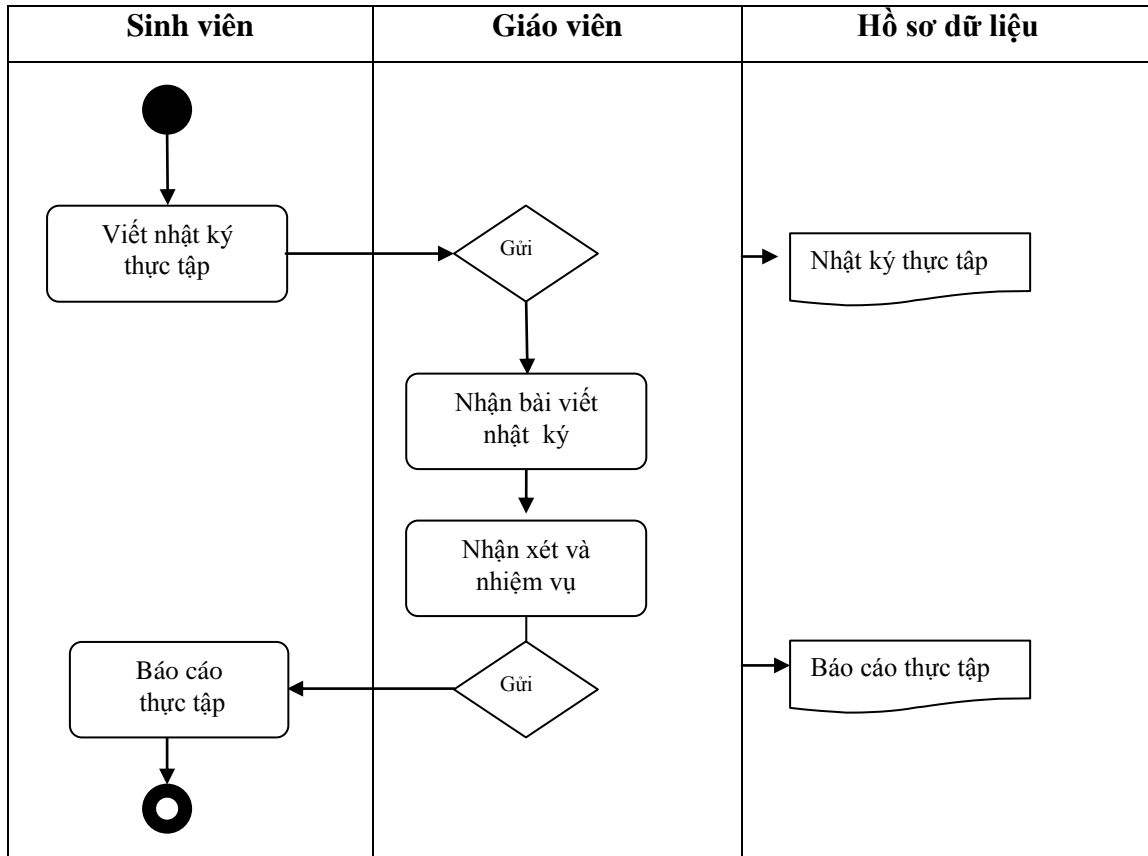
### 2.4.2 - Sơ đồ hoạt động đăng ký của cơ sở thực tập



### 2.4.3 - Sơ đồ hoạt động lập quyết định thực tập



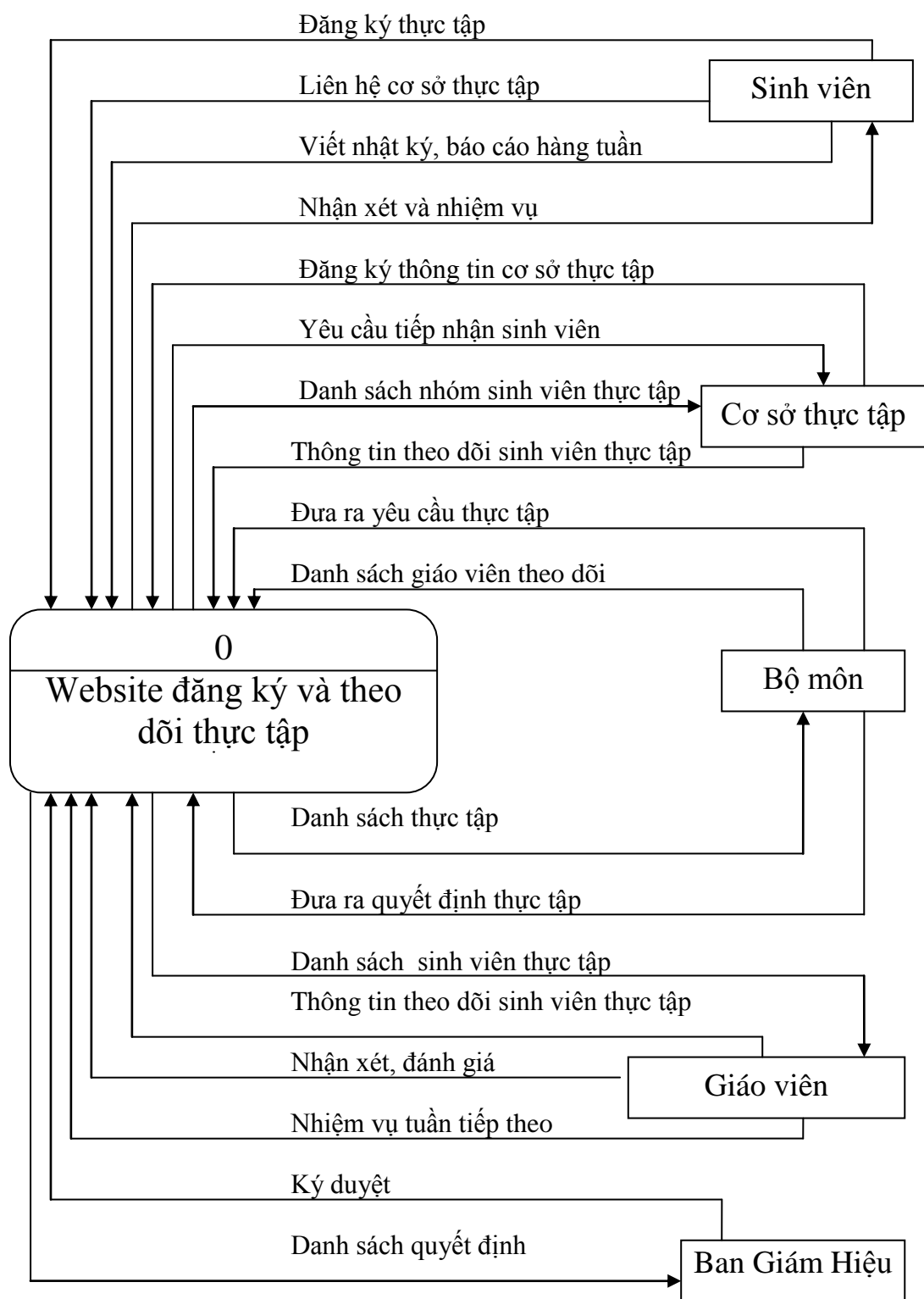
#### 2.4.4 - Sơ đồ hoạt động nhật ký báo cáo



#### 2.5 - Biểu đồ ngữ cảnh của hệ thống

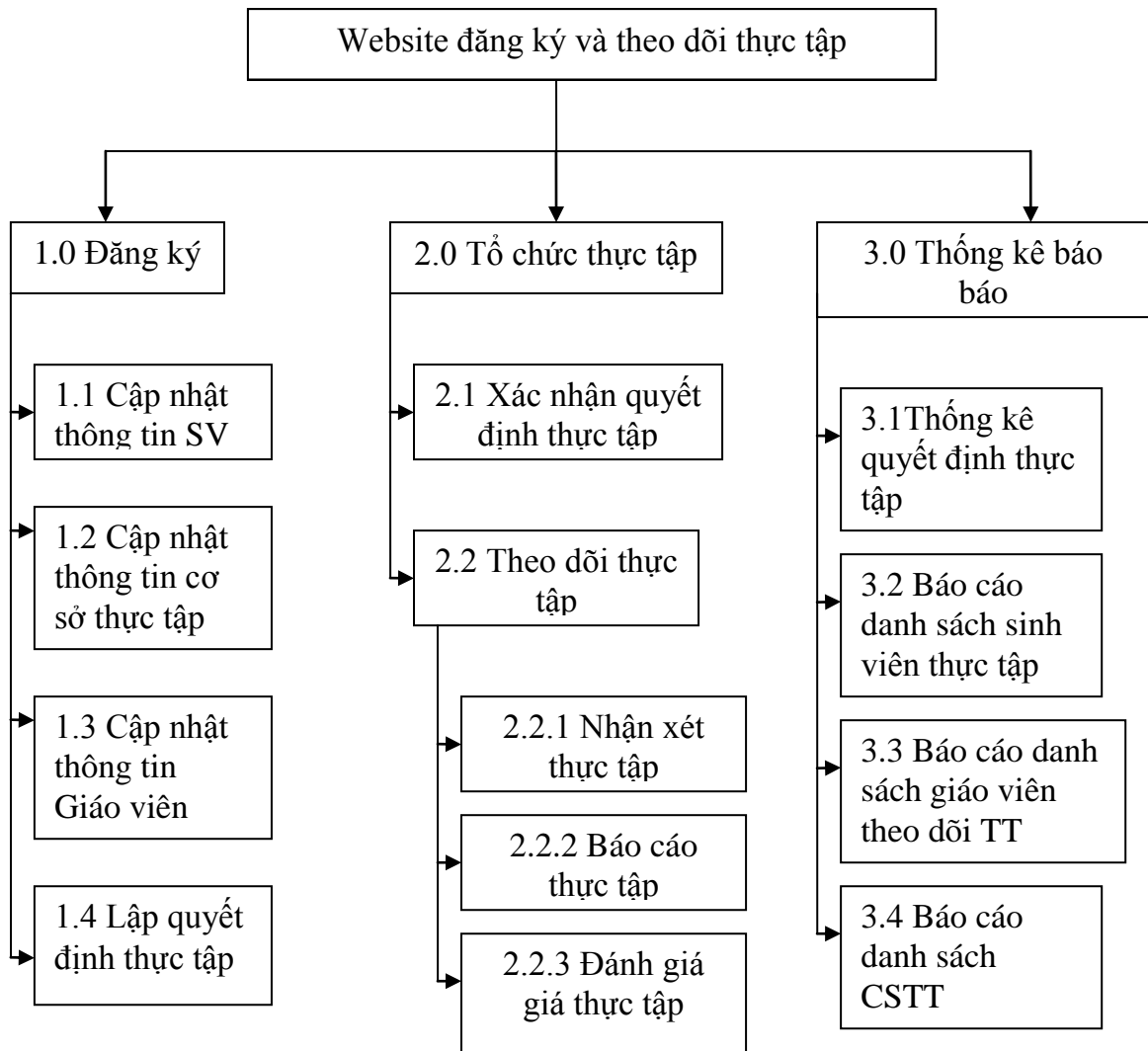
Biểu đồ ngữ cảnh hệ thống cho ta cái nhìn tổng quan về hệ thống. Bao gồm ba thành phần:

- Một tiến trình duy nhất: Website đăng ký và theo dõi thực tập.
- Có bốn tác nhân: Sinh viên, Giáo viên, Cơ sở thực tập và Bộ môn của hệ thống
- Các luồng dữ liệu đi từ các tác nhân vào hệ thống và từ hệ thống ra các tác nhân.





## 2.6 - Biểu đồ phân rã chức năng nghiệp vụ hệ thống



## 2.7 - Danh sách hồ sơ dữ liệu sử dụng

<b>Ký hiệu</b>	<b>Tên hồ sơ dữ liệu</b>
D1	Danh sách sinh viên thực tập
D2	Danh sách giáo viên
D3	Danh sách cơ sở thực tập
D4	Quyết định thực tập
D5	Nhật ký thực tập
D6	Báo cáo thực tập
D7	Bản nhận xét thực tập
D8	Bản đánh giá thực tập
D9	Báo cáo quyết định thực tập
D10	Báo cáo sinh viên thực tập
D11	Báo cáo giáo viên theo dõi thực tập
D12	Báo cáo cơ sở thực tập

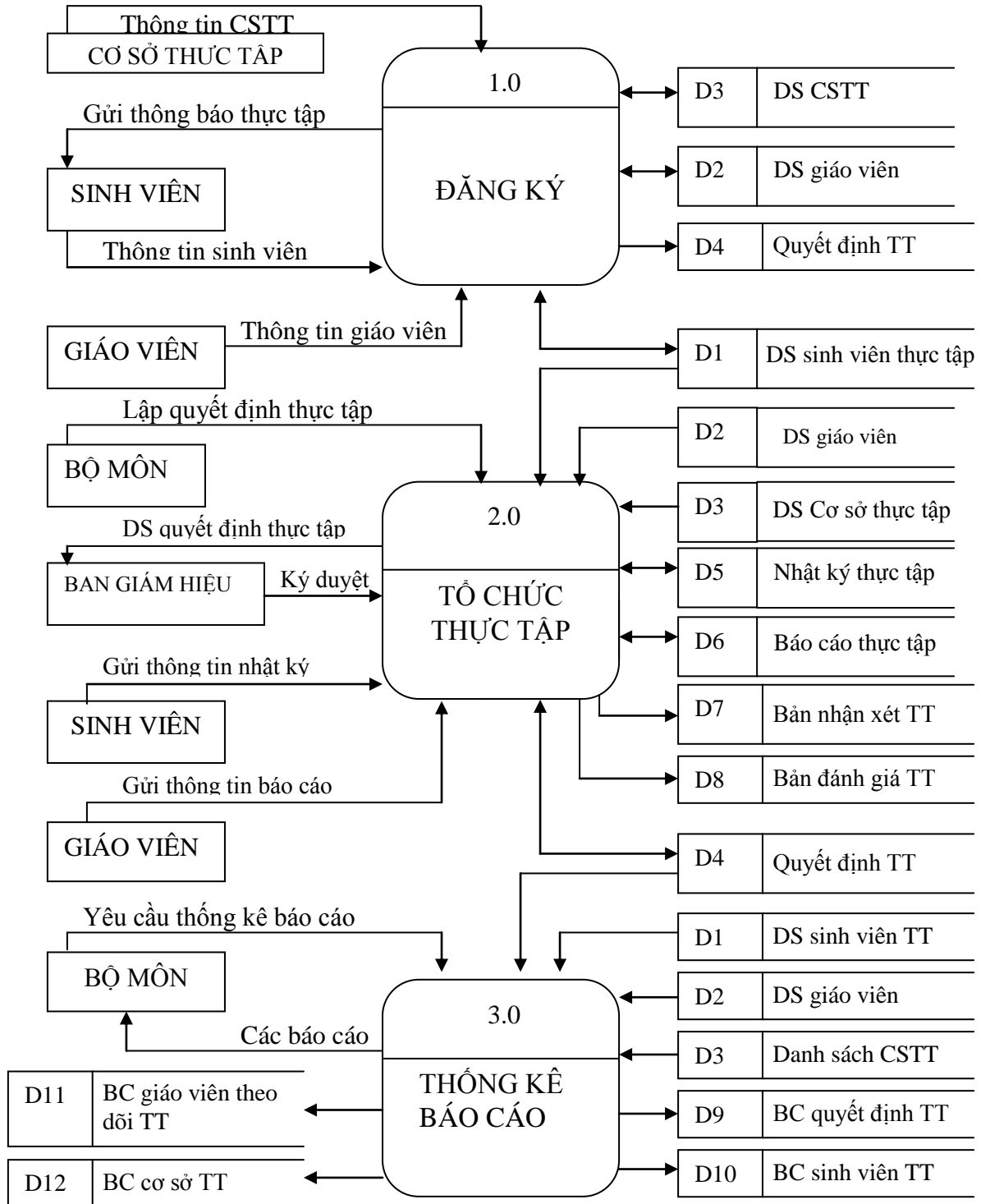
## 2.8 - Ma trận thực thể chức năng

Tên hồ sơ dữ liệu												
D1: Danh sách sinh viên thực tập												
D2: Danh sách giáo viên												
D3: Danh sách cơ sở thực tập												
D4: Quyết định thực tập												
D5: Nhật ký thực tập												
D6: Báo cáo thực tập												
D7: Bản nhận xét thực tập												
D8: Bản đánh giá thực tập												
D9: Báo cáo quyết định thực tập												
D10: Báo cáo sinh viên thực tập												
D11: Báo cáo giáo viên theo dõi thực tập												
D12: Báo cáo cơ sở thực tập												
Các chức năng nghiệp vụ	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12
1. Đăng ký	U	U	U	C								
2. Tổ chức thực tập	R	R	R	U	U	U	C	C				
3. Thống kê báo cáo	R	R	R	R					C	C	C	C

# CHƯƠNG III

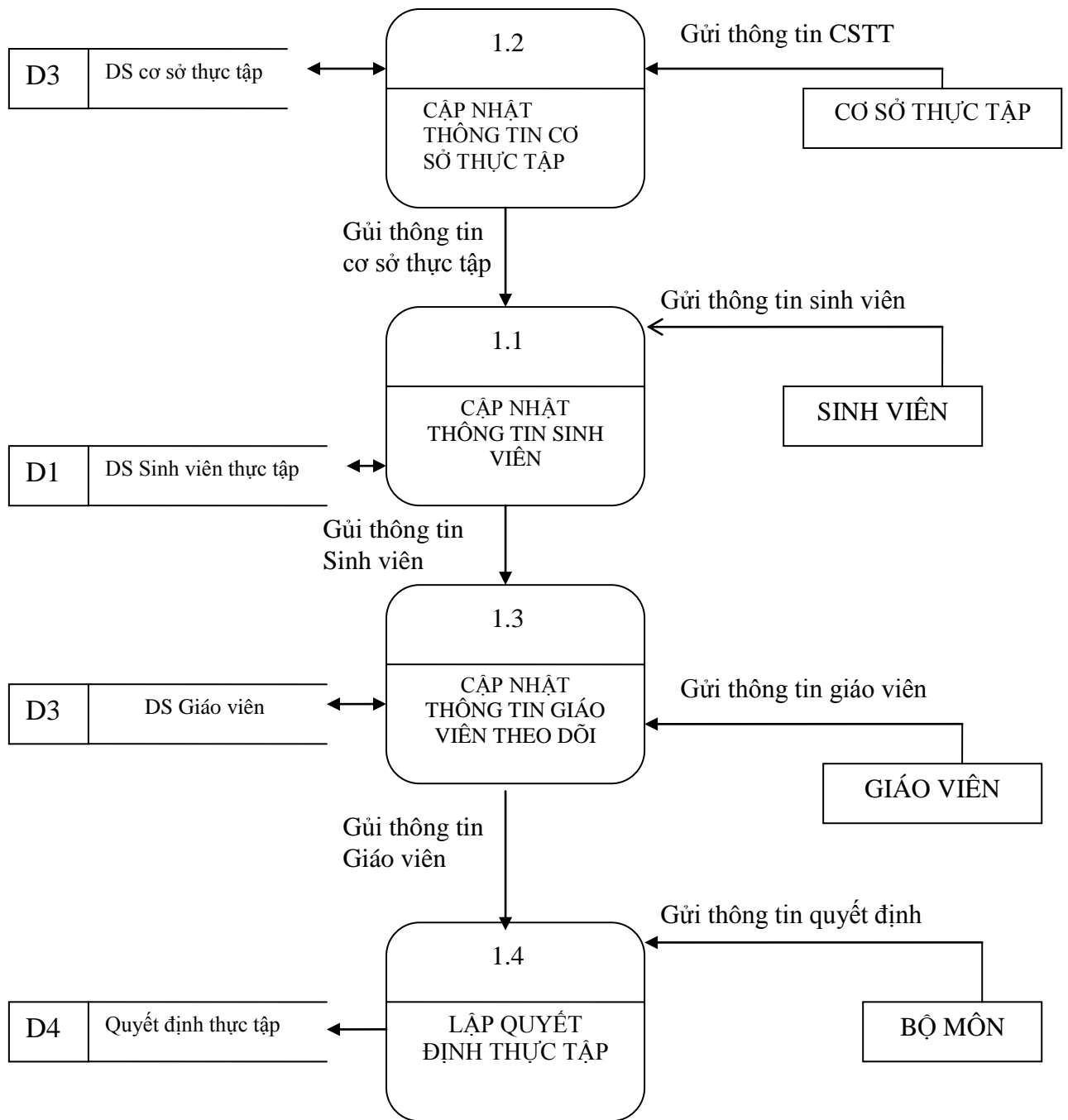
## PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 3.1 Sơ đồ luồng dữ liệu mức 0

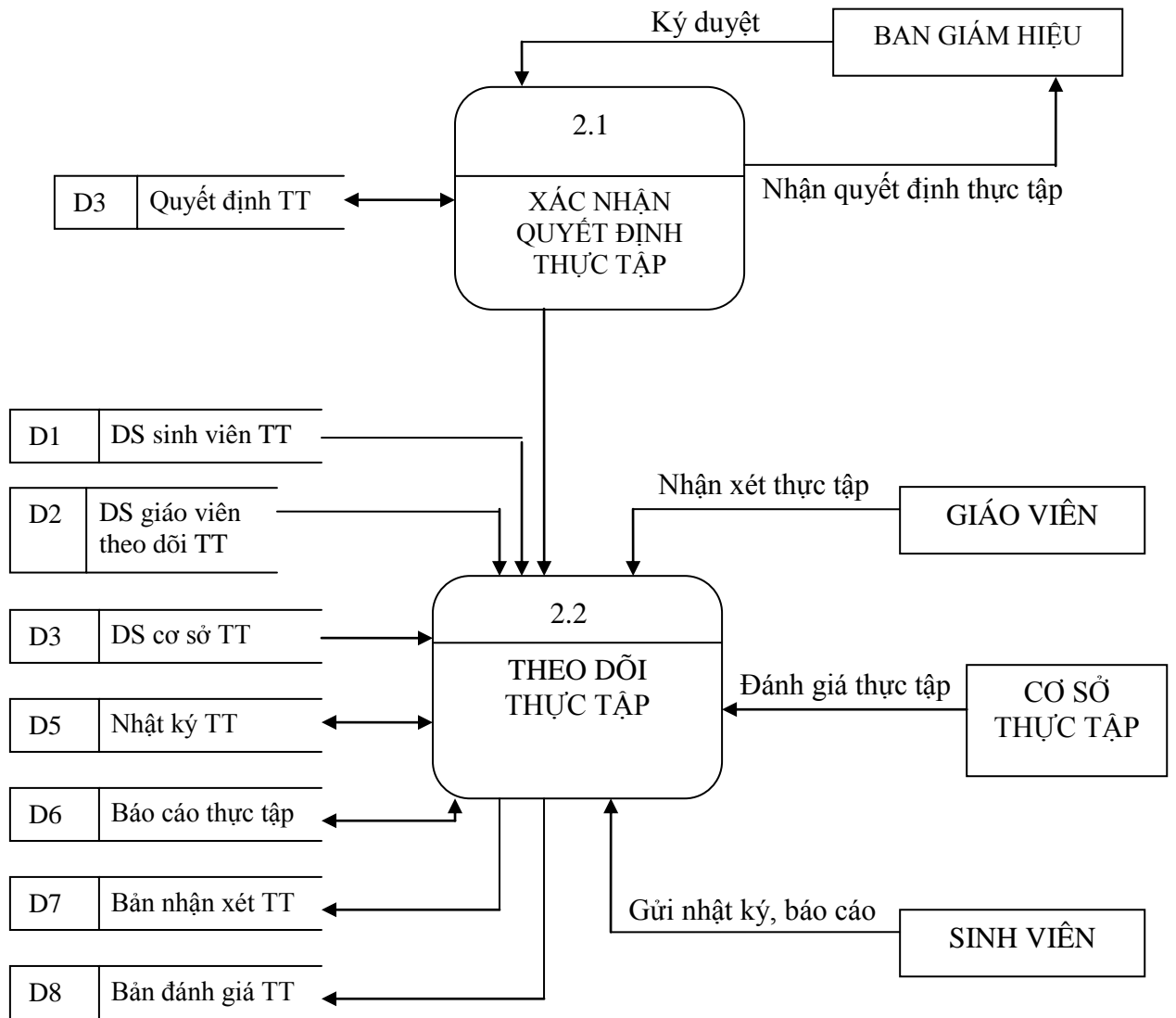


### 3.2 - Sơ đồ luồng dữ liệu mức 1

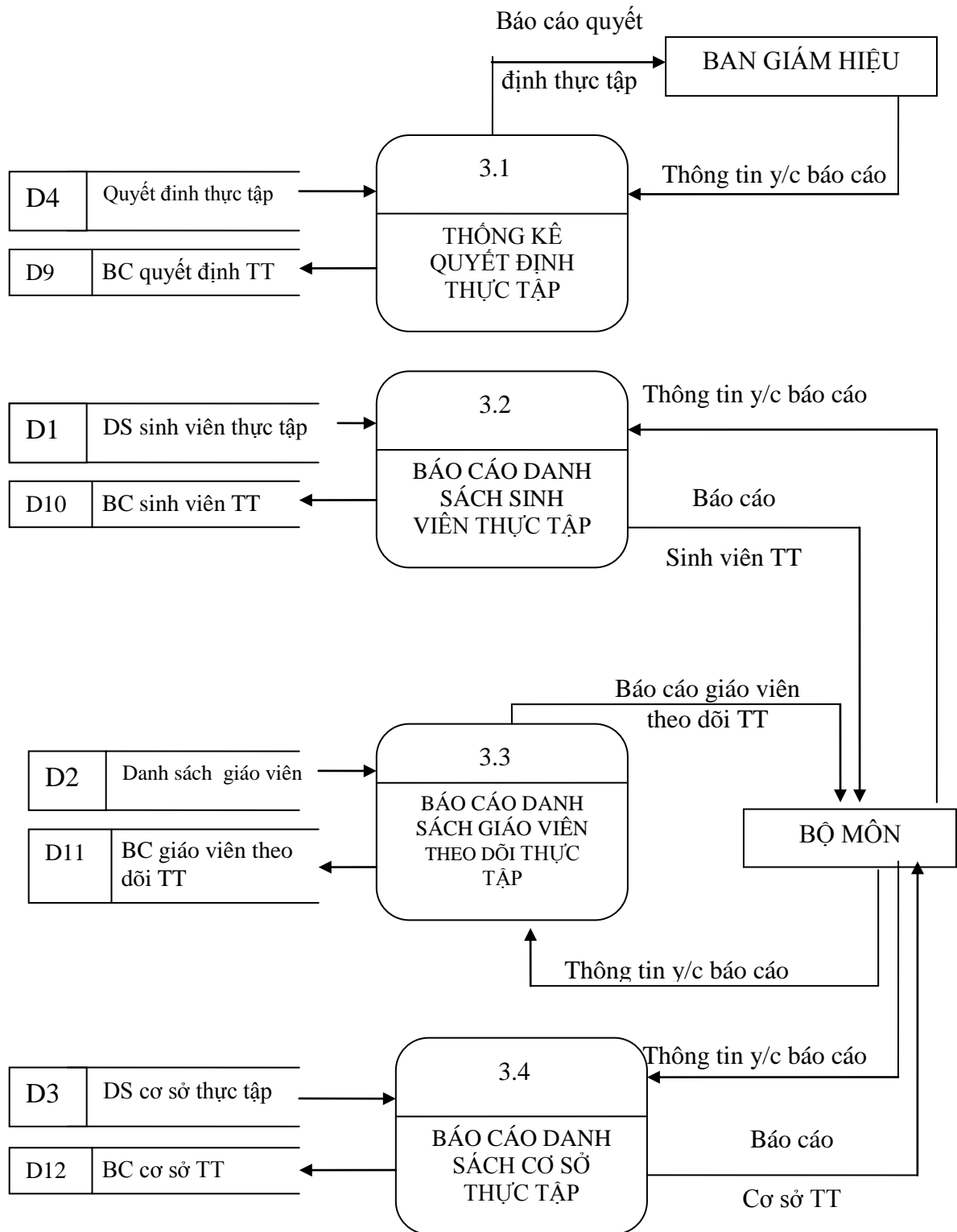
#### 3.2.1 - Phân rã tiến trình Đăng ký



### 3.2.2 - Phân rã tiến trình Tổ chức thực tập

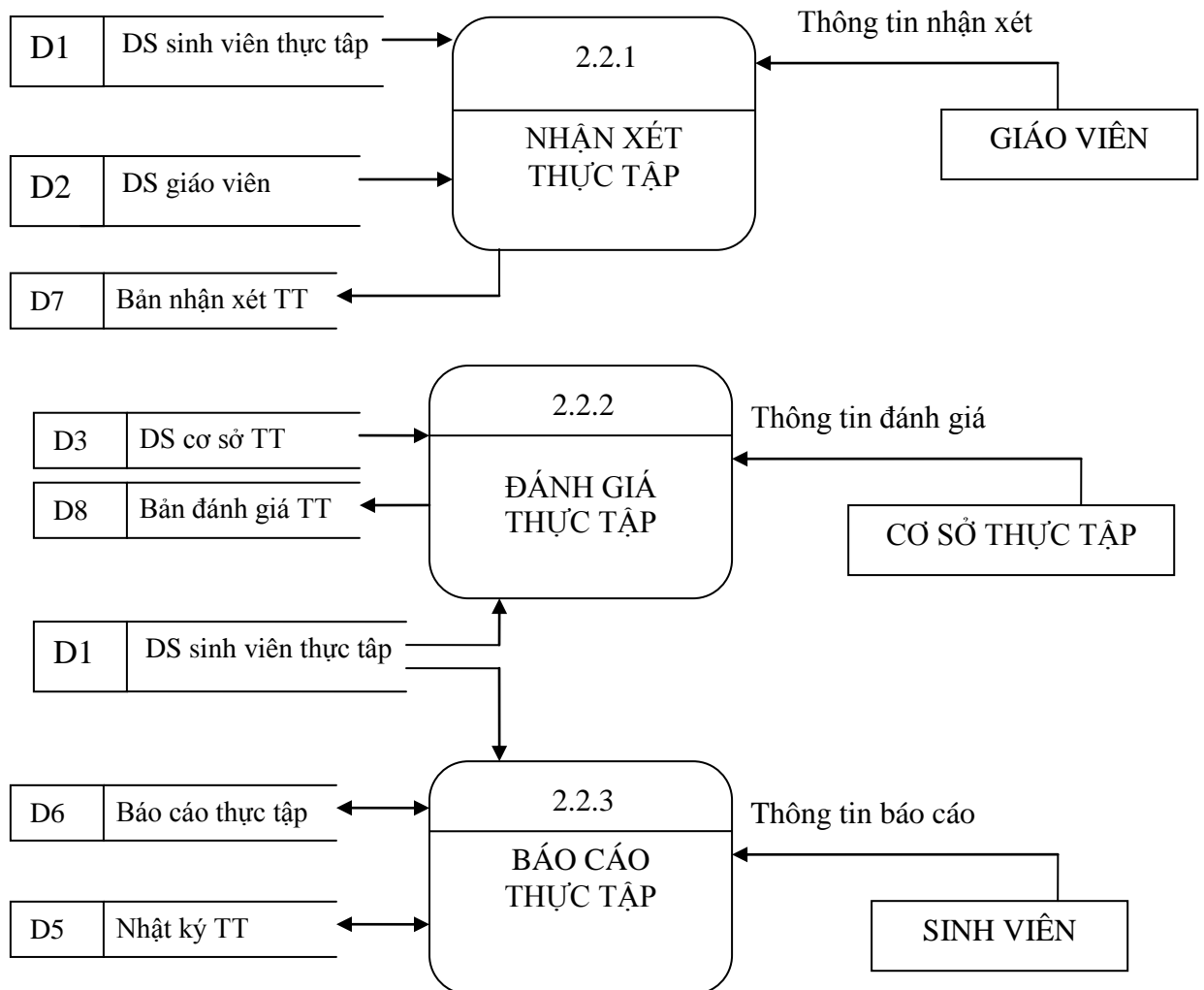


### 3.2.3 - Phân rã tiến trình thông kê báo cáo



### 3.3 – Sơ đồ luồng dữ liệu mức 2

#### 3.3.1 – Phân rã tiến trình Theo dõi thực tập





### 3.4 - Phân tích dữ liệu hệ thống Website đăng ký và theo dõi sinh viên thực tập

#### 3.4.1 - Các khái niệm cơ bản

Thực thể (Entity): Là khái niệm chỉ một tầng các đối tượng cụ thể hay các khái niệm có cùng những đặc trưng mà ta quan tâm. Ký hiệu:

Tên thực thể

Thuộc tính: Là các đặc trưng của các thực thể. Thuộc tính khoá là thuộc tính xác định tính duy nhất của một thực thể

Các liên kết : Là các mối quan hệ gắn kết các thực thể trong mô hình thực thể liên kết E-R. Các thực thể có thể tham gia với số lượng khác nhau trong mỗi liên kết với thực thể khác.

#### 3.4.2 - Các thực thể

**SINH VIÊN:** Mã SV, Họ tên SV, Ngày sinh, Lớp, Ngành, Địa chỉ, Email, Ghi chú.

**GIÁO VIÊN:** Mã GV, Họ tên GV, Ngành, Email, Ghi chú.

**CƠ SỞ THỰC TẬP:** Mã CSTT, Tên CSTT, Địa chỉ, Số ĐT, Lĩnh vực, Mô tả.

**BỘ MÔN:** Mã BM, Tên BM.

**QUYẾT ĐỊNH THỰC TẬP:** Mã QD, Tên QD, Nội dung Thời gian.

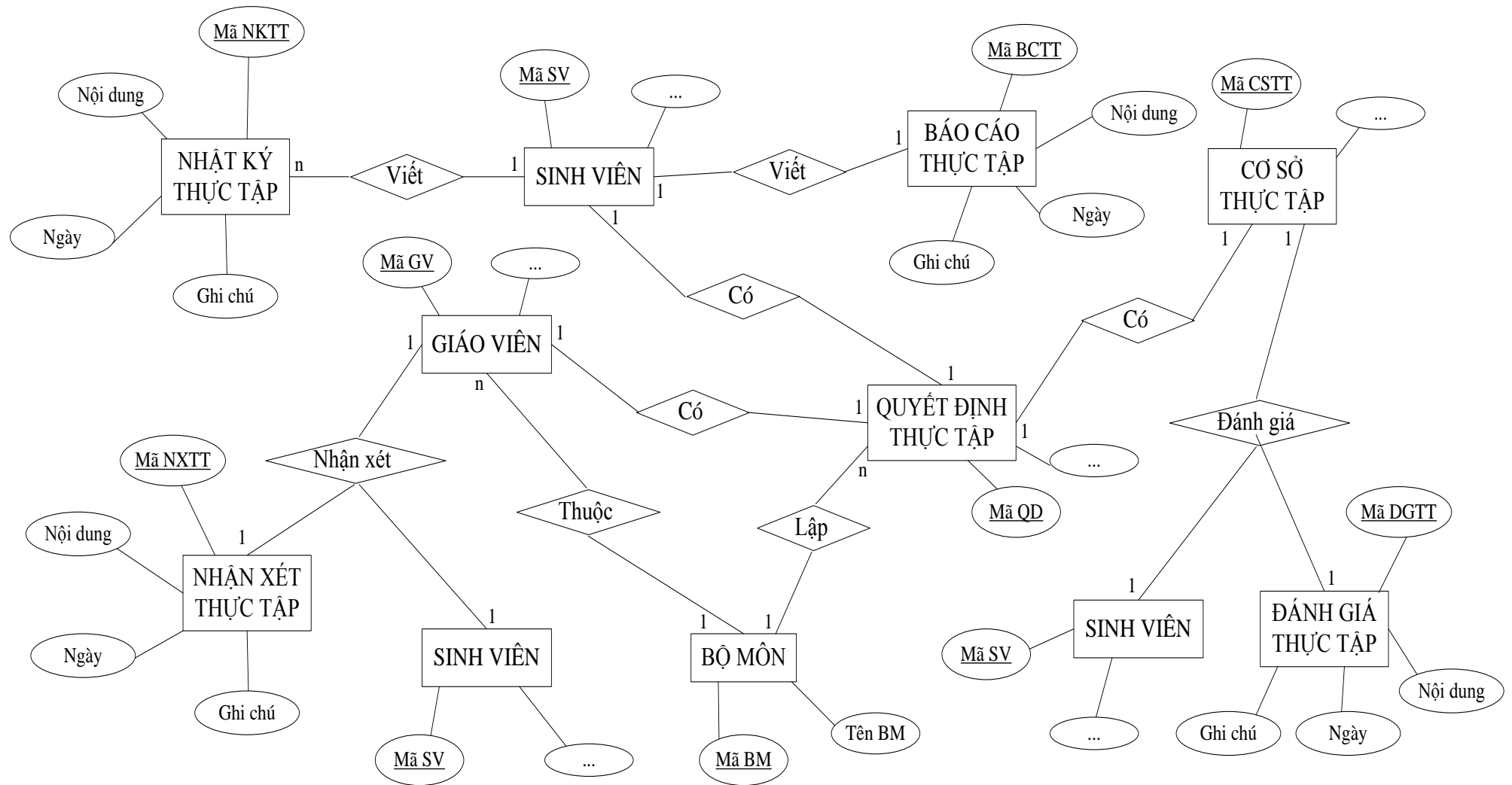
**NHẬT KÝ THỰC TẬP:** Mã NKTT, Nội dung, Ngày, Ghi chú.

**BÁO CÁO THỰC TẬP:** Mã BCTT, Nội dung, Ngày, Ghi chú.

**NHẬN XÉT THỰC TẬP:** Mã NXTT, Nội dung, Ngày, Ghi chú.

**ĐÁNH GIÁ THỰC TẬP:** Mã DGTT, Nội dung, Người đánh giá, Ngày, Ghi chú

### 3.4.3 – Mô hình ER

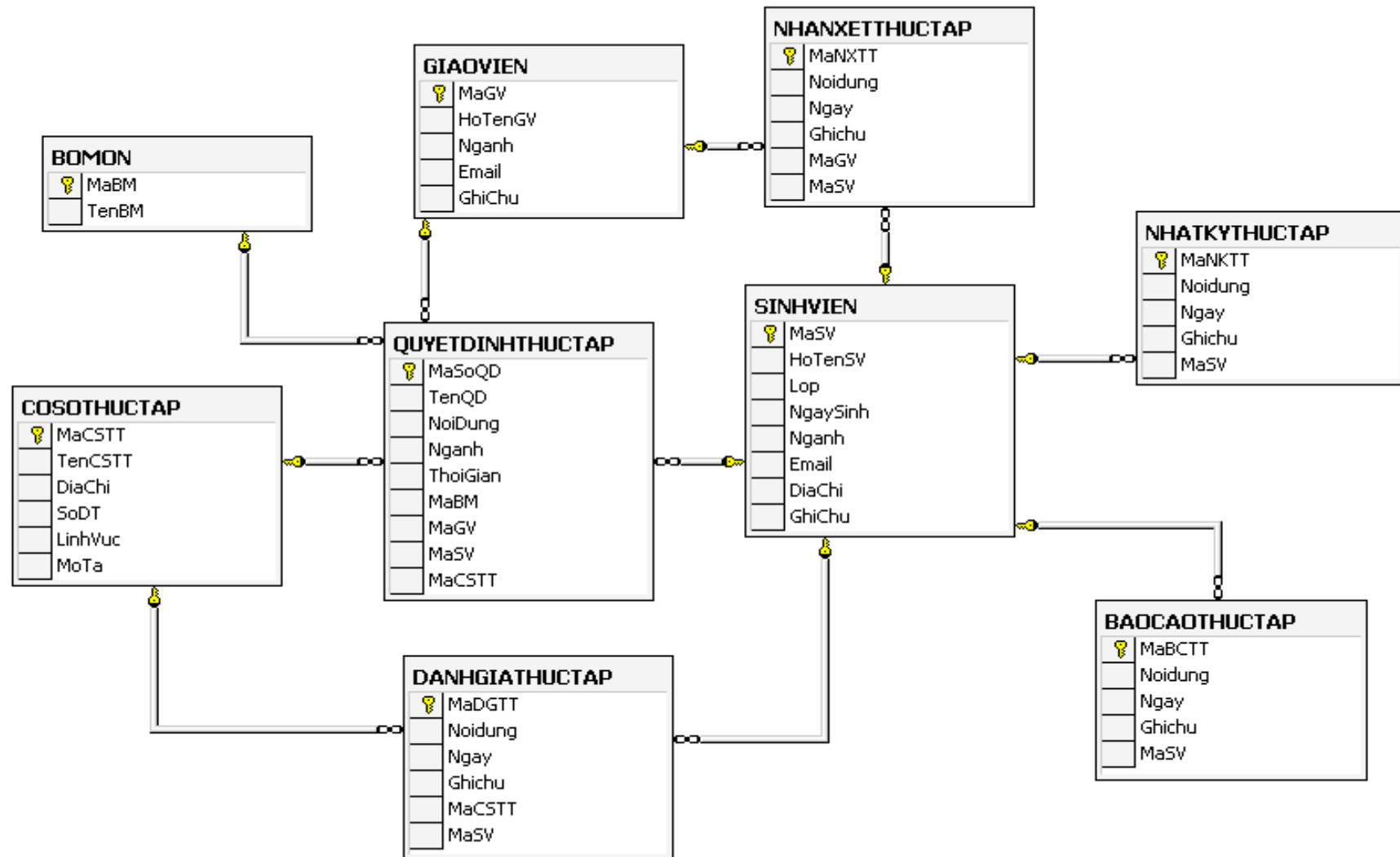


### **3.5 – Thiết kế cơ sở dữ liệu:**

#### **3.5.1 - Các bảng trong Cơ sở dữ liệu:**

1. SINHVIEN (MaSV, HotenSV, Ngaysinh, Lop, Ngay, Diachi, Email, Ghichu)
2. GIAOVIEN (MaGV, HotenGV, Nghanh, Email, Ghichu, MaBM)
3. COSOTHUCTAP (MaCSTT, TenCSTT, Diachi, SoDT, Linhvuc, Mota)
4. BOMON (MaBM, TenBM)
5. QUYETDINHTHUCTAP (MaQD, TenQD, Noidung, Thoigian, MaBM, MaGV, MaSV, MaSCTT)
6. NHATKYTHUCTAP (MaNKTT, Noidung, Ngay, Ghichu, MaSV)
7. BAOCAOTHUCTAP (MaBCTT, Noidung, Ngay, Ghichu, MaSV)
8. NHANXETTHUCTAP (MaNXTT, Noidung, Ngay, Ghichu, MaGV, MaSV)
9. DANHGIATHUCTAP (MaDGTT, Noidung, Ngay, Ghichu, MaCSTT, MaSV)

### 3.5.2 – Mô hình quan hệ:



### 3.5.3 - Cấu trúc vật lý của các bảng cơ sở dữ liệu:

#### 3.5.3.1 - Bảng 1: SINHVIEN

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaSV	char	15	no	PK
HotenSV	nvarchar	30	no	
Lop	nvarchar	20	no	
Ngaysinh	datetime	8	no	
Nganh	nvarchar	30	no	
Email	nvarchar	30	yes	
Diachi	nvarchar	100	no	
Ghichu	nvarchar	200	yes	

#### 3.5.3.2 - Bảng 2: GIAOVIEN

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaGV	char	15	no	PK
HoTenGV	nvarchar	30	no	
Nganh	nvarchar	30	no	
Email	nvarchar	30	yes	
Ghichu	nvarchar	200	yes	
MaBM	char	15	no	FK

#### 3.5.2.3 - Bảng 3: COSOTHUCTAP

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaCSTT	char	15	no	PK
TenCSTT	nvarchar	100	no	
Diachi	nvarchar	100	no	
SoDT	char	20	yes	
Linhvuc	nvarchar	50	yes	
Mota	nvarchar	200	yes	

#### 3.5.3.4 - Bảng 4: BOMON

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaBM	char	15	no	PK
TenBM	nvarchar	100	no	

**3.5.3.5 - Bảng 1: QUYETDINHTHUCTAP**

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaQD	char	15	no	PK
TenQD	nvarchar	50	no	
Noidung	nvarchar	200	yes	
Thoigian	datetime	8	no	
MaBM	char	15	no	FK
MaGV	char	15	no	FK
MaSV	char	15	no	FK
MaCSTT	char	15	no	FK

**3.5.3.6 - Bảng 6: NHATKYTHUCTAP**

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaNKTT	char	15	no	PK
Noidung	nvarchar	1000	no	
Ngay	datetime	8	no	
Ghichu	nvarchar	100	yes	
MaSV	char	15	no	FK

**3.5.3.7 - Bảng 7: BAOCAOTHUCTAP**

Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaBCTT	char	15	no	PK
Noidung	nvarchar	1000	no	
Ngay	datetime	8	no	
Ghichu	nvarchar	100	yes	
MaSV	char	15	no	FK

**3.5.3.8 - Bảng 8: NHANXETTHUCTAP**

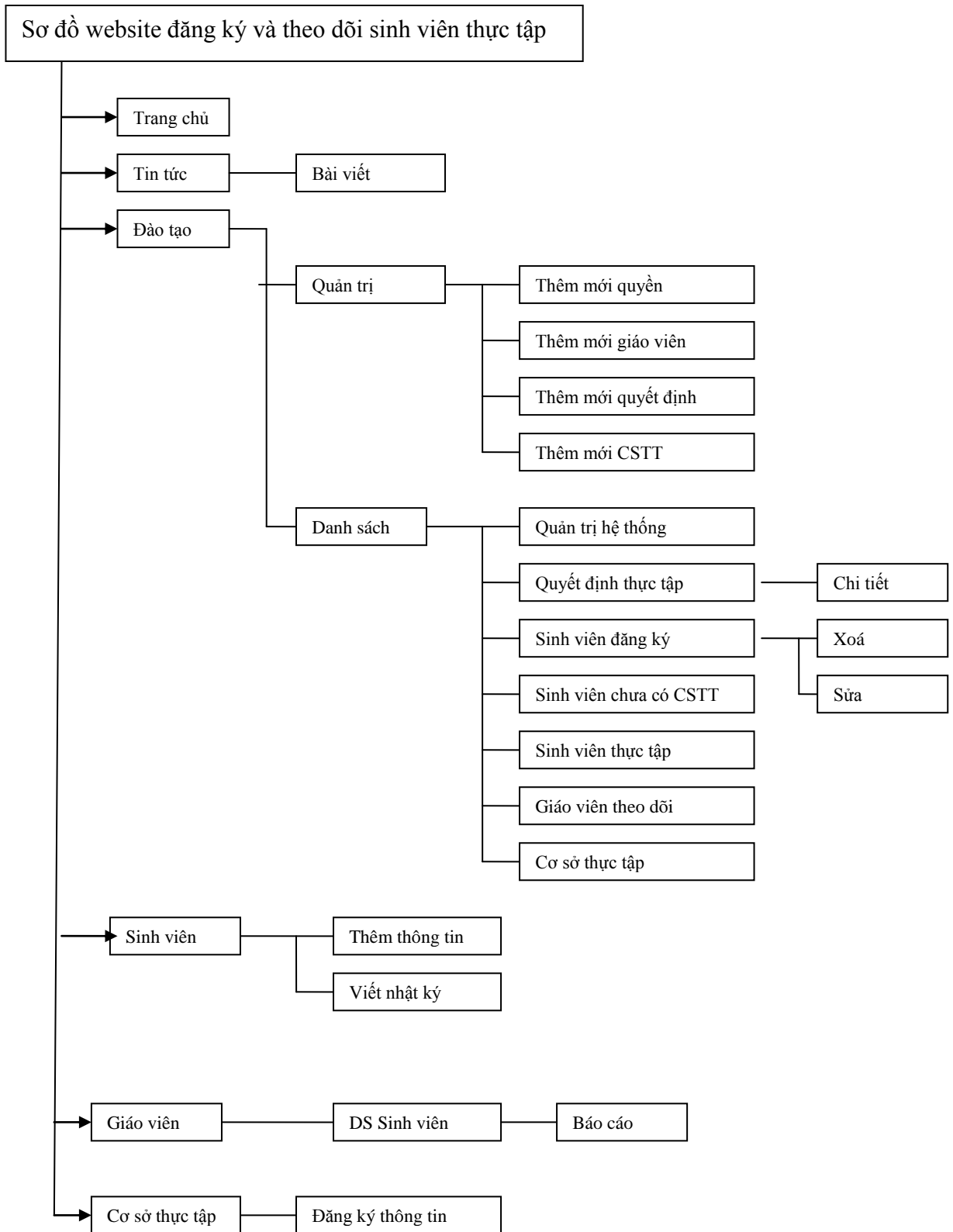
Tên cột	Kiểu dữ liệu	Số byte	NULL	Ràng buộc
MaNXTT	char	15	no	PK
Noidung	nvarchar	1000	no	
Ngay	datetime	8	no	
Ghichu	nvarchar	100	yes	
MaGV	char	15	no	FK
MaSV	char	15	no	FK

**3.5.3.9 - Bảng 9: DANHGIATHUCTAP**

<b>Tên cột</b>	<b>Kiểu dữ liệu</b>	<b>Số byte</b>	<b>NULL</b>	<b>Ràng buộc</b>
MaDGTT	char	15	no	PK
Noidung	nvarchar	1000	no	
Ngay	datetime	8	no	
Ghichu	nvarchar	100	yes	
MaCSTT	char	15	no	FK
MaSV	char	15	no	FK

### 3.6 - Thiết kế giao diện

#### 3.6.1 - Giao diện chính của Website





### 3.6.2 - Trang quản lý hệ thống

<a href="#">Trang chủ</a>	<a href="#">Tin Tức</a>	<a href="#">Cơ quan thực tập</a>
Đăng nhập hệ thống		
NỘI DUNG CHÍNH CỦA WEBSITE		

#### 3.6.2.1- Trang thêm mới Accounts đăng nhập hệ thống .

THÊM MỚI QUYỀN QUẢN TRỊ			
Mã QT :	<input type="text"/>	Tên DN :	<input type="text"/>
Tên QT :	<input type="text"/>	Mật khẩu :	<input type="text"/>
Email :	<input type="text"/>	Ngành :	<input type="text"/> M
		Quyền :	<input type="text"/> M
Thông tin :	<input type="text"/>		

### 3.6.2.2- Trang cập nhật quyết định thực tập

THÊM MỚI QUYẾT ĐỊNH THỰC TẬP			
Mã số quyết định :	<input type="text"/>	Ngày ra QD :	<input type="text"/>
Tên quyết định :	<input type="text"/>	Ngành :	<input type="text" value="CNTT"/>
Thông tin :	<input type="text"/>		

### 3.6.2.3- Trang cập nhật thông tin giáo viên theo dõi.

THÊM MỚI GIÁO VIÊN THEO DÕI			
Mã Giáo viên :	<input type="text"/>	Email :	<input type="text"/>
Họ Tên Giáo viên :	<input type="text"/>	Ngành :	<input type="text" value="CNTT"/>
Thông tin :	<input type="text"/>		

### 3.6.2.4-Trang thống kê danh sách Accountst.

THỐNG KÊ DANH SÁCH QUYỀN									
Chọn ngành :			<input type="text" value="CNTT"/>	Chọn quyền :			<input type="text" value="Admin"/>		
<input type="button" value="Thực hiện thống kê"/>									
DANH SÁCH QUẢN TRỊ WEBSITE									
MaQT	Họ Tên	Ngành	Tên DN	Mật Khẩu	Quyền	Email	Mô Tả	Lựa Chọn	

### 3.6.2.5- Trang thống kê danh sách giáo viên theo dõi.

THỐNG KÊ DANH GIÁO VIÊN THEO DÕI THỰC TẬP				
Chọn ngành :		<input type="text" value="CNTT"/>	<input type="button" value="Thực hiện thống kê"/>	
DANH SÁCH GIÁO VIÊN THEO DÕI				
Mã Giáo viên	Họ Tên Giáo viên	Ngành	Email	Liên hệ

### 3.6.2.6-Trang thống kê danh sách sinh viên đăng ký

<b>THỐNG KÊ DANH SINH VIÊN ĐĂNG KÝ</b>						
Chọn ngành : <input type="text" value="CNTT"/> <input checked="" type="checkbox"/> <input type="checkbox"/>						
<input type="button" value="Thực hiện thống kê"/>						
<b>DANH SÁCH SINH VIÊN ĐĂNG KÝ THỰC TẬP</b>						
Ma SV	Họ Tên sinh viên	Ngày sinh	Ngành	Địa chỉ	Email	Liên hệ

### 3.6.2.7-Trang thống kê danh sách sinh viên thực tập

<b>THỐNG KÊ DANH SINH VIÊN THỰC TẬP TẠI CƠ SỞ</b>						
Chọn ngành : <input type="text" value="CNTT"/> <input checked="" type="checkbox"/> <input type="checkbox"/>						
<input type="button" value="Thực hiện thống kê"/>						
<b>DANH SÁCH SINH VIÊN THỰC TẬP TẠI CƠ SỞ</b>						
Ma SV	Họ Tên sinh viên	Ngày sinh	Ngành	Địa chỉ	Họ tên GV	Tên SCTT

### 3.6.3-Trang quản lý sinh viên.

#### 3.6.3.1-Trang đăng ký thông tin sinh viên

CẬP NHẬT THÔNG TIN SINH VIÊN			
Mã sinh viên :	<input type="text"/>	Email :	<input type="text"/>
Ngày sinh :	<input type="text"/>	Ngành :	<input type="text" value="CNTT"/> <input checked="" type="checkbox"/>
Địa chỉ :	<input type="text"/>	Lớp học :	<input type="text"/>
Liên hệ :	<input type="text"/>		

#### 3.6.3.2-Trang sinh viên viết nhật ký.

VIẾT NHẬT KÝ THỰC TẬP			
Mã bài viết :	<input type="text"/>	Mã bài viết :	<input type="text"/>
Tên viết :	<input type="text"/>		
Nội dung :	<input type="text"/>		
<input type="button" value="Gửi bài"/>		<input type="button" value="Làm lại"/>	

### 3.6.4-Trang quản lý Giáo viên.

VIẾT NHẬT KÝ THỰC TẬP	
Tên bài viết :	<input type="text" value="Tiêu đề bài báo cáo"/>
Nội dung :	<input type="text" value="Nội dung bài báo cáo"/>
Nội dung :	<input type="text"/>
<input type="button" value="Gửi nhận xét"/> <input type="button" value="Làm lại"/>	

# CHƯƠNG IV

## CÀI ĐẶT CHƯƠNG TRÌNH

### 4.1 – Tổng quan về ASP

#### 4.1.1 - Tìm hiểu công nghệ ASP:

##### Giới thiệu:

- ASP (Microsoft Active Server Pages) là một môi trường kịch bản trên máy chủ dùng để tạo ra và chạy các ứng dụng Web động, tương tác và có hiệu quả cao. ASP hoạt động dựa vào các script do người lập trình tạo ra.

- ASP xây dựng sẵn một thư viện cho phép giao tiếp với cơ sở dữ liệu thông qua ODBC (Open Database Connectivity) gọi là ADO (ActiveX Data Object). Hiện nay thư viện ADO là một trong những công cụ mạnh nhất trong việc phát triển các ứng dụng Web.

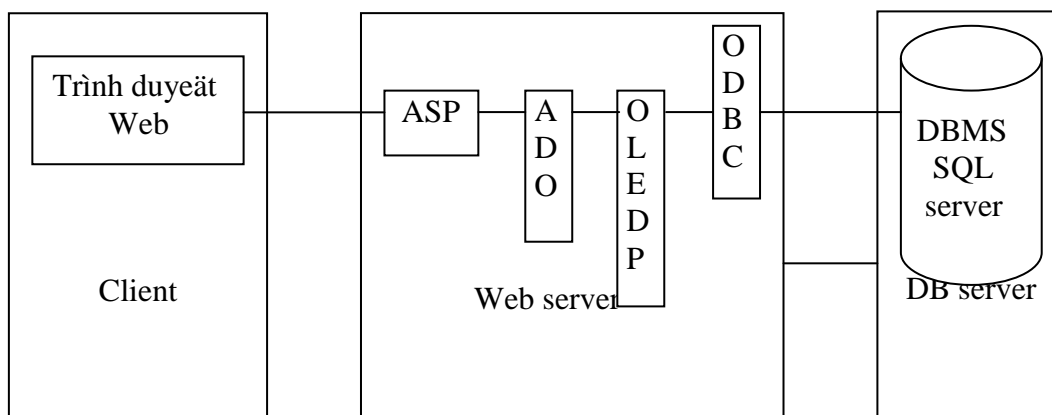
##### Hoạt động của trang ASP:

- Trang ASP là trang Web trong đó có kết hợp các thành phần HTML, ActiveX Component và Script ASP. Có thể xem trang ASP như một trang HTML có bổ sung các lệnh kịch bản ASP.

- Khi một trang ASP được trình duyệt Web yêu cầu, đầu tiên Web server sẽ duyệt tuần tự trang ASP này và chỉ thực hiện dịch những câu lệnh kịch bản ASP, kết quả trả về cho trình duyệt dưới dạng một trang HTML.

#### 4.1.2 - Mô hình ứng dụng cơ sở dữ liệu trên Web qua công nghệ ASP

- Thao tác giữa client và server trong một ứng dụng Web có thể được thể hiện khái quát như sau:



- Web server:

Là nơi tiếp nhận và trả lời các yêu cầu của người dùng Web (client), đồng thời cũng thực hiện việc kết nối đến hệ quản trị cơ sở dữ liệu (DBMS) trên Database server theo yêu cầu truy cập dữ liệu của trang ASP. ADO cung cấp giao diện lập trình cho người phát triển xây dựng các lệnh truy cập cơ sở dữ liệu. Các lệnh này được chuyển đến cho hệ DBMS để thi hành thông qua các thành phần OLEDB (và ODBC). Kết quả truy vấn dữ liệu sẽ được Web server đưa ra hiển thị trên trình duyệt.

- Database server:

Là nơi diễn ra việc thực thi các thao tác trên cơ sở dữ liệu như truy vấn, cập nhật, cũng như đảm bảo tính toàn vẹn dữ liệu của hệ DBMS.

- Browser:

Giao diện với người dùng là nơi tiếp nhận yêu cầu của người dùng, cũng như hiển thị kết quả yêu cầu. Ngoài, ra trình duyệt còn là nơi kiểm tra sơ bộ tính hợp lệ của dữ liệu trước khi chuyển đến cho Web server.

#### **4.1.3 - Cấu trúc và cách tạo trang ASP**

- Trang ASP đơn giản là trang văn bản với phần mở rộng .asp, gồm có ba phần:

+ Văn bản (Text)

+ Thẻ HTML

+ Các đoạn kịch bản ASP

- Sử dụng trang HTML có sẵn để tạo trang ASP bằng cách thêm vào các lệnh kịch bản ASP cần thiết và đổi phần mở rộng của tên tập tin thành .asp.

- Khi thêm một đoạn kịch bản vào trang HTML để đánh dấu nơi kịch bản ASP bắt đầu và kết thúc, dùng cặp lệnh `<% ... %>`.

- Những lệnh nằm giữa `<% ... %>` phải sử dụng ngôn ngữ viết kịch bản chính thức quy định cho trang đó. VBScript là ngôn ngữ mặc định của ASP. Nếu muốn sử dụng một ngôn ngữ khác (như JavaScript chẳng hạn) thì cần phải định nghĩa ngôn ngữ tại đầu mỗi trang ASP như sau:

`%@LANGUAGE = tên ngôn ngữ %`.

- Các biến không cần khai báo mà tự xác định khi gán giá trị cho nó.

- Muốn in ra các biến trong Web, sử dụng mã sau: `<%=s %>`

- Để làm việc với hệ thống tập tin, thao tác với cơ sở dữ liệu, cần phải sử dụng các đối tượng ActiveX của ASP.



## 4.1.4 - Các đối tượng và thành phần xây dựng sẵn của ASP

### 4.1.4.1 - Các đối tượng (object) của ASP

Đối tượng là những đoạn chương trình có khả năng thực hiện một số công việc cơ bản nào đó. Mỗi đối tượng là một kết hợp giữa lập trình và dữ liệu. Các đối tượng ASP cho phép giao tiếp, tương tác với cả máy chủ (Web server) lẫn trình duyệt (Browser).

Có 6 đối tượng cơ bản trong ASP:

- Request: cho phép lấy thông tin thông qua một yêu cầu HTTP. Những thông tin này gồm có các tham số của Form khi được Submit bằng phương thức POST hay GET, hoặc các tham số được chia cùng với trang ASP trong lời gọi đến trang đó. Dùng đối tượng Request để chia sẻ thông tin qua lại giữa các trang ASP trong một ứng dụng với nhau.

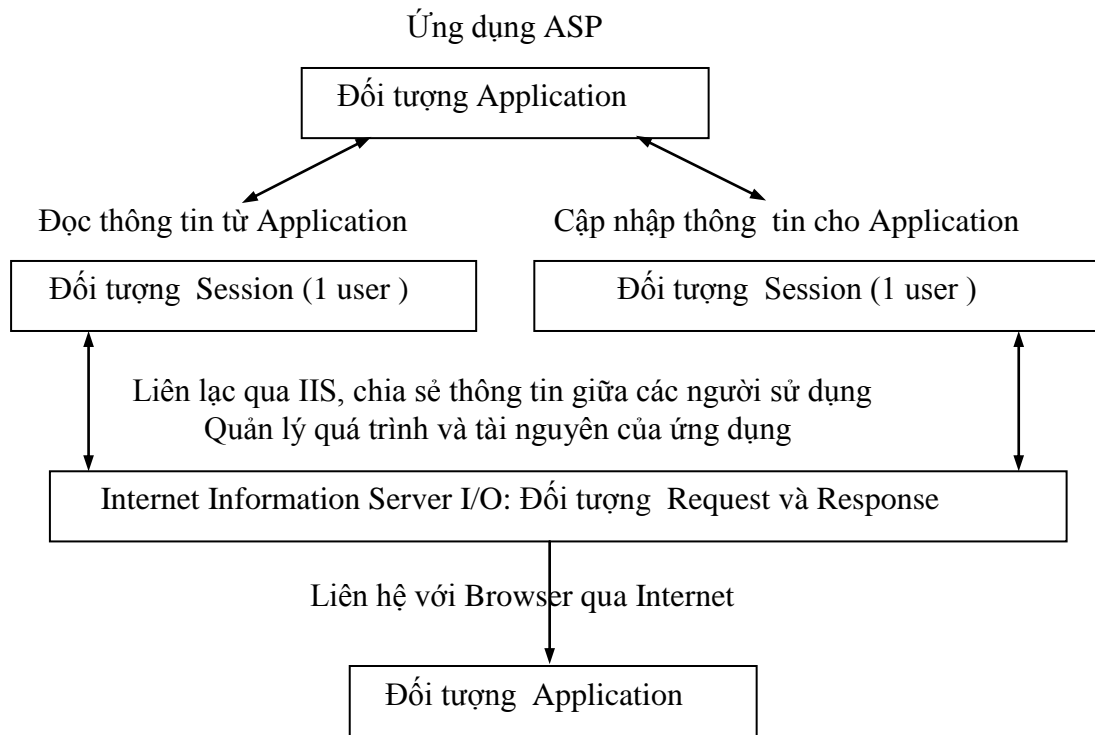
- Response: gửi thông tin tới người dùng, là đại diện cho phần thông tin do server trả về cho trình duyệt Web.

- Application: giúp điều khiển các đặc tính liên quan đến việc khởi động và khai thác ứng dụng cũng như việc lưu trữ thông tin có thể được truy nhập bởi các ứng dụng nói chung. Nói cách khác đối tượng Application dùng để chia sẻ thông tin giữa các người dùng trong cùng một ứng dụng, chẳng hạn đếm số lần truy cập đến ứng dụng của các người dùng.

- Server: cung cấp phương tiện truy cập đến những phương thức và thuộc tính trên server. Thường sử dụng phương thức Server.CreateObject để khởi tạo phiên bản của một đối tượng ActiveX trên trang ASP.

- Session: dùng lưu trữ thông tin cần thiết trong phiên làm việc của người dùng đã truy nhập vào trang Web. Những thông tin lưu trữ trong phiên làm việc không bị mất đi khi người dùng di chuyển qua các trang của ứng dụng.

- ObjectContext: dùng để chấp thuận hay hủy bỏ một giao dịch do một kịch bản ASP khởi tạo.



Sơ đồ môi quan hệ giữa các đối tượng ASP trong ứng dụng ASP

Cú pháp của các đối tượng:

Cú pháp của các đối tượng không phụ thuộc vào ngôn ngữ kịch bản mà người lập trình sử dụng.

Để truy xuất thông tin của một đối tượng ta dùng các phương thức (method) và các thuộc tính (property).

Sử dụng các phương thức:

Một phương thức (method) là một procedure hoạt động trên một đối tượng nào đó. Cú pháp tổng quát của method là:

Object.Method [parameters]

parameter : là các thông số của method.

Sử dụng các thuộc tính:

Một thuộc tính là một tính chất được đặt tên của một đối tượng. Chúng định nghĩa các tính chất của đối tượng như kích thước, màu, vị trí trên màn hình hay thể hiện các trạng thái của đối tượng như được phép (enable) hay không được phép (disable). Cú pháp tổng quát của property là:

Object.Property [parameters]

Parameter: là các thông số của Property.

Parameter thường là các biến, dữ liệu, chuỗi ký tự, hoặc là một URL.

- Đối tượng Request:

- Các tập hợp (Collection) của đối tượng Request: Đối tượng Request cung cấp 5 collection cho phép truy xuất tất cả các loại thông tin về yêu cầu của Browser đối với Server.

•QueryString: Tập hợp này nhận tất cả giá trị trong chuỗi truy vấn, đó là giá trị được gửi lên theo sau dấu chấm hỏi (?) trong câu lệnh request.

Cú pháp: Request.QueryString (Tên biến)[(chỉ mục)].Count

•Form: Nhận các giá trị của các thành phần trong Form được gửi bằng phương thức POST thông qua một HTTP request.

Cú pháp: Request.Form (element)[(index) | .Count]

Trong đó:

Element: Tên của một thành phần trong Form.

Index: Tham số tùy chọn cho phép chúng ta truy cập một trong những giá trị của một tham số (parameter).

•Cookie: Là một tập tin lưu trên máy của client dưới dạng một tập tin nhỏ. Cookie được trình duyệt của client gửi kèm trong HTTP request.

Cú pháp: Request.Cookies (cookie)[(key)|.attribute]

Trong đó:

Cookie: Chỉ định cookie để nhận giá trị.

Key: Tham số tùy chọn dùng để nhận các khoản mục có trong cookie.

Attribute: Chỉ định thông tin về bản thân cookie. Tham số thuộc tính này có thể là Name hay HasKeys.

• ServerVariable: Nhận các giá trị của các biến môi trường.

•ClientCertificate: Nhận certificate fields từ yêu cầu của Trình duyệt Web.

Cú pháp: Request.ClientCertificate (Key [SubField])

- Thuộc tính (property) của đối tượng Request: Đối tượng Request chỉ có 1 thuộc tính cung cấp thông tin về số byte dữ liệu mà người dùng chuyển lên trình chủ.

•TotalByte: Trả lại tổng số byte trong đối tượng Request được gửi lên từ trình khách.

- Phương thức (method) của đối tượng Request: Đối tượng Request chỉ có một phương thức cho phép truy xuất toàn bộ nội dung của yêu cầu của người dùng được POST lên server qua phân đoạn <FORM> của một trang Web.

• BinaryRead (count): nhận count byte dữ liệu từ yêu cầu client khi dữ liệu được gửi đến server là một phần của request POST. Nó trả về một mảng Variant.

Cú pháp tổng quát khi sử dụng đối tượng Request:

Request.CollectionName(variable)

Variable: là tên biến trong Collection muốn truy xuất thông tin. Cũng có thể sử dụng cú pháp:

Request (variable)

Khi đó thứ tự tìm kiếm sẽ tuân theo danh sách các Collection trên và khi gặp biến đầu tiên trùng tên thì trả về giá trị của biến đó. Vì thế nếu có nhiều hơn một biến với cùng một tên trong các Collection khác nhau thì nên sử dụng dạng cú pháp tổng quát.

Sau đây là một số ứng dụng của đối tượng Request:

Lấy thông tin từ FORM

HTML Form là cách thức thông thường để trao đổi thông tin giữa Webserver và user. HTML Form cung cấp nhiều cách nhập thông tin của user như: text box, radio button, check box, submit, reset, password, ... và hai phương thức gửi thông tin là POST hoặc GET.

Sử dụng QueryString

QueryString Collection chứa toàn bộ thông tin được gửi tới từ Form bằng phương thức GET. Các giá trị gửi từ Form chứa trong một chuỗi truy vấn (query string) và được add vào URL (bắt đầu từ dấu chấm hỏi (?) trở về sau là phần query string được thêm vào).

Ví dụ:

```
<FORM NAME="frmDetails" ACTION="getDetails.asp"
METHOD="GET">
Name: <INPUT TYPE=TEXT NAME="txtName">
Age: <INPUT TYPE=TEXT NAME="txtAddress">
<INPUT TYPE=SUBMIT>
</FORM>
```

Khi user nhập giá trị vào 2 hộp Name và Age sau đó nhấn Submit thì hộp Address của trình duyệt sẽ có dạng như sau:

http://company/asptest/GetDetails.asp?txtName=Minh&txTAGe=30

Hạn chế của phương thức GET: Chiều dài tối đa của URL là 1000 ký tự, như vậy nếu gửi một loạt thông tin từ Form, thông tin sẽ bị cắt bớt.

Sử dụng Form collection

Thay vì add thông tin gửi đi từ Form vào URL như một querystring, một phương thức khác là POST, đặt thông tin gửi đi vào bên trong HTTP HEADER.

Ví dụ:

```
<FORM NAME="frmDetails" ACTION="getDetails.asp"
METHOD="POST">
Name: <INPUT TYPE=TEXT NAME="txtName">
Age: <INPUT TYPE=TEXT NAME="txtAddress">
<INPUT TYPE=SUBMIT>
</FORM>
```

Khi user nhập giá trị vào 2 hộp Name và Age sau đó nhấn Submit thì hộp Address của trình duyệt sẽ có dạng như sau:

```
http://company/asptest/GetDetails.asp
```

Tại thời điểm này, các giá trị gửi từ Form không thể truy xuất bằng QueryString Collection. Thay vào đó, Active Server Pages hiện thực một Form Collection chứa toàn bộ thông tin gửi từ Form sử dụng phương thức POST.

- Đối tượng Response:

- Các tập hợp của đối tượng Response: Đối tượng Response cung cấp loại đối tượng tập hợp dùng để xác lập giá trị của bất kỳ cookie nào ta muốn đặt trên hệ thống của trình khách client. Nó tương đương với tập Request.Cookies.

• Cookies: Một tập chứa giá trị của tất cả các cookie sẽ được gửi ngược lại client trong đáp ứng hiện hành. Các thành phần của tập đều là giá trị chỉ ghi.

Cú pháp: Response.Cookies(cookie)[(key)].attribute = value

Cookie: Tên của cookie muốn tạo.

Key: Là tham số tùy chọn. Nếu giá trị này được thiết lập thì cookie này được xem là cookie từ điển.

Attribute: Bao gồm những thông tin liên quan đến cookie.

- Các thuộc tính của đối tượng Response: Đối tượng Response cung cấp một số thuộc tính mà chúng ta có thể đọc và xác lập theo yêu cầu:

- Buffer: Là một biến kiểu boolean. Nếu thuộc tính này là True thì Web server chỉ trả kết quả cho client khi tất cả các kịch bản được xử lý hoặc phương thức Response.Flush hay Response.End được gọi. Giá trị mặc định là False.

Cú pháp: Response.Buffer = [True, False]

- ContentType: Quy định dạng dữ liệu cho Web server trả về cho client. Giá trị mặc định là Text/ HTML.

Cú pháp: Response.ContentType [ = ContentType ]

- ExpiresAbsolute #date[time]# : Chỉ định ngày và giờ tuyệt đối một trang sẽ hết hạn và không còn hợp lệ nữa.

- Expires minutes: Chỉ định chiều dài thời gian tính bằng phút một trang còn hợp lệ.

- IsClientConnected: Trả lại một chỉ định client có còn kết nối và tải trang từ server xuống hay không? Có thể sử dụng để kết thúc quá trình (với phương thức Response.End) nếu client di chuyển đến trang khác trước khi trang hiện hành được xử lý xong.

- Status = “code message”: Chỉ định giá trị trạng thái và thông điệp sẽ được gửi đến client trong các header HTTP của đáp ứng để một lỗi hay xử lý trang thành công.

- Các phương thức của đối tượng Response:

- AddHeader (tên, nội dung): Thêm một HTML header với một giá trị được chỉ định. Phương thức này luôn luôn thêm mới một header vào Response. Nó sẽ không thay thế những header có sẵn cùng tên với header mới.

- AppendToLog (“string”): Thêm một chuỗi vào cuối file Log của Web server cho Request này.

- BinaryWrite (Array): Xuất thông tin ra output HTML dạng binary.

- Clear (): Huỷ bất cứ nội dung trang nào trong bộ đệm khi Response.Buffer bằng True. Không huỷ các header HTTP. Có thể sử dụng để bỏ một trang chưa hoàn thành.

- End (): Dừng xử lý file .asp và trả về kết quả hiện tại.

- Flush (): Gửi thông tin trong bộ đệm IIS đến client nếu Response.Buffer = True. Có thể sử dụng để gửi từng phần riêng của một trang dài đến client.

• Redirect (“URL”): Gởi một thông báo cho browser định hướng lại đến một URL khác.

• WriteString (“string”): Ghi một biến ra HTML output như là một chuỗi.

- Đối tượng Session:

- Các thuộc tính của đối tượng Session:

SessionID: trả về số định danh session cho user. Mỗi session sẽ được server cho một số định danh duy nhất khi nó được tạo ra.

Timeout: chỉ thời gian sống của đối tượng session, giá trị này tính bằng phút. Nếu người dùng không cập nhật hay yêu cầu một trang Web của ứng dụng trong khoảng thời gian lớn hơn thời gian timeout thì phiên làm việc sẽ kết thúc. Giá trị mặc định là 20 phút.

- Các phương thức của đối tượng Session:

Abandon: Dùng để huỷ đối tượng session và giải phóng tài nguyên. Web server sẽ tự thực hiện phương thức này khi phiên làm việc hết hiệu lực (timeout).

Ví dụ: Ta có thể tạo các biến trong đối tượng Session để lưu thông tin cho mỗi kết nối đến Server.

+ Session("Login"): Cho biết người yêu cầu truy xuất đến trang có login chưa.

+ Session("Username"): Tên của Account tạo ra session hiện tại.

+ Session("SelectedTopic"): Tên chủ đề đang được chọn để thực hiện một thao tác nào đó.

Session\_OnStart: Biến cố này xảy ra khi server tạo một phiên làm việc. Server xử lý kịch bản này lúc thực thi yêu cầu một trang. Những đối tượng xây dựng sẵn tồn tại trong biến cố này là: Application, ObjectContext, Request, Response, Server và Session.

Session\_OnEnd: Biến cố này xảy ra khi phiên làm việc bị huỷ bởi lệnh Abandon hoặc timeout. Trong biến cố này chỉ có các đối tượng xây dựng sẵn: Application, Server, Session của ASP là tồn tại.

Các thông tin lưu giữ trong Session được giữ nguyên trong suốt thời gian session tồn tại và có giá trị trong cả tầm vực của session.

d. Đối tượng Application:

- Các phương thức của đối tượng Application:

- Contents: Chứa tất cả các item do chúng ta thiết lập trong đối tượng Application mà không dùng tag <OBJECT>

Cú pháp: Application.Contents (Key)

Key: Tên thuộc tính (property) cần nhận về.

- Lock: Phương thức Lock ngăn chặn các client khác thay đổi giá trị của biến dùng chung trong đối tượng Application, đảm bảo tại một thời điểm chỉ có một client được phép thay đổi và truy xuất các biến của đối tượng này. Nếu phương thức Unlock không được gọi thì server tự động thực hiện lệnh này khi xử lý xong các script hoặc time out.

Cú pháp: Application.Lock

- Unlock: Phương thức Unlock cho phép client thay đổi giá trị của các biến lưu trong đối tượng Application đã bị khóa trước kia.

Cú pháp: Application.Unlock

- Application\_OnStart: Biến cố này xảy ra trước khi phiên làm việc (session) đầu tiên được tạo, nghĩa là nó xảy ra trước biến cố Session\_OnStart. Trong biến cố này chỉ tồn tại đối tượng Application và Session.

- Application\_OnEnd: Biến cố này xuất hiện khi thoát ứng dụng. Nó xảy ra sau biến cố Session\_OnEnd. Trong biến cố này chỉ tồn tại đối tượng Application và Session.

- Đối tượng Server:

- Thuộc tính của đối tượng Server:

- ScriptTimeout: Thuộc tính quy định khoảng thời gian lớn nhất mà các script còn được thực hiện. Giá trị mặc định là 90 giây. Giá trị Timeout sẽ không hiệu lực khi server thực hiện kịch bản.

Cú pháp: Server.ScriptTimeout = NumSeconds

NumSecond: Thời gian tối đa kịch bản hết hiệu lực.

- Các phương thức của đối tượng Server:

- CreateObject: Phương thức này tạo một phiên bản của thành phần ActiveX. Những đối tượng được tạo bởi phương thức này sẽ giải phóng khi server thực hiện xong các kịch bản. Phương thức này không dùng để tạo các đối tượng xây dựng sẵn của ASP.

Cú pháp: Server.CreateObject (ProgID)



ProgID (program identifier): Kiểu đối tượng cần tạo.

- HTMLEncode : Phương thức này dùng để mã hóa một chuỗi.

Cú pháp: Server.HTMLEncode (string)

- MapPath: Phương thức này cho biết thông tin về đường dẫn vật lý của một thư mục ảo trên Web server.

- URLEncode (string): Mã hóa một string thành dạng URL

- Execute (path): Thực thi trang .asp trong đường dẫn path.

- Đối tượngObjectContext:

-Các phương thức:

- SetComplete: Nếu tất cả thành phần có trong giao dịch gọi phương thức này thì giao dịch sẽ kết thúc. Phương thức SetComplete bỏ qua bất kỳ lệnh SetAbort đã được gọi trước đó.

-Cú pháp: ObjectContext.SetComplete

- SetAbort: Phương thức này khai báo tất cả thao tác thực hiện trong giao dịch sẽ không hoàn thành và tài nguyên có thể không được cập nhật.

-Cú pháp: ObjectContext.SetAbort

- OnTransactionCommit: Biến cố này xảy ra sau khi tất cả kịch bản giao tác kết thúc. Web server sẽ xử lý hàm này nếu nó tồn tại.

- OnTransactionAbort: Tương tự như biến cố OnTransactionCommit.

#### **4.1.4.2- Các thành phần của ASP.**

Thành phần ASP là điều khiển ActiveX ghép nối với ASP để đơn giản hóa các thủ tục thông thường.

- Các thành phần ActiveX thông dụng:

- BrowserType (Browser capabilities object): Thành phần này cho phép xác định trình duyệt của người dùng là gì và những tính năng nào được hỗ trợ bởi trình duyệt đó như tên, phiên bản, khả năng hỗ trợ Frame, Table,...

- CDO (Collaboration Data Object): Liên kết chặt chẽ với IIS SMTP server. CDO hỗ trợ chúng ta gửi và nhận mail.

- Database Access: Một trong những khả năng mạnh mà ASP có được là khả năng thâm nhập vào các cơ sở dữ liệu. ASP thường làm việc với Access và hệ quản trị cơ sở dữ liệu SQL. Thành phần này rất hữu ích, giúp chúng ta có thể kết nối vào một

cơ sở dữ liệu bằng cách sử dụng ActiveX Data Object để viết nội dung lên màn hình trình duyệt và tạo lập hoặc cập nhật các tập tin cơ sở dữ liệu.

- File Access: Thành phần này chứa đựng các phương thức và thuộc tính có thể sử dụng để truy cập tập tin trên máy tính. Thông qua thành phần này, chúng ta có thể tạo ra một đối tượng là FileSystemObject, giúp thực hiện một số công việc như tạo, đọc tập tin...

Ngoài ra còn có những thành phần cung cấp miễn phí hoặc mua từ các nhà phân phối khác.

- ADO (ActiveX Data Object): Đối tượng truy cập dữ liệu. ADO cung cấp giao diện lập trình quen thuộc với những thuộc tính và phương thức theo mô hình đối tượng.

- AdRotator (AD Rotator Object): Luân phiên hiển thị dãy các hình ảnh cũng như liên kết từ hình ảnh được hiển thị đến một URL. Thông tin về hình ảnh và liên kết tương ứng được lưu trữ trong tập tin văn bản (text).

- Page Counter: Đếm và hiển thị số lần trang Web được yêu cầu.

- MyInfo, Status, System, Tools Object: các thông tin về Web site.

#### **4.1.5. File cấu hình khởi động global.asa:**

- Ứng dụng Web là một tập các trang Web, asp và html trong một thư mục Web và các thư mục con.

- Tập tin global.asa xác định khởi đầu và kết thúc của một ứng dụng Web cũng như của các phiên làm việc (session) thuộc từng người dùng đối với ứng dụng.

- ASP cung cấp file cấu hình global.asa, nơi có thể đặt các đoạn mã script xử lý sự kiện hay triệu gọi các hàm, thủ tục, biến mang tính toàn cục. File global.asa được đặt trong thư mục gốc của ứng dụng. Mỗi ứng dụng Web chỉ được phép có duy nhất một file global.asa. Khi lần đầu tiên trang ASP của ứng dụng được yêu cầu, trình chủ IIS sẽ đọc và nạp thông tin trong file global.asa, phát sinh và xử lý các sự kiện được cài đặt trong file global.asa sau đó chuyển giao quyền xử lý lại cho trang ASP.

File global.asa cho phép cài đặt và xử lý các sự kiện sau:

- Application\_OnStart: Sự kiện này phát sinh khi người dùng đầu tiên triệu gọi bất kì trang nào trong ứng dụng Web. Khi trình chủ IIS khởi động lại hoặc khi nội dung file global.asa bị hiệu chỉnh thì sự kiện này sẽ được phát sinh trở lại. Sau khi sự kiện này kết thúc quá trình xử lý, trình chủ IIS sẽ bắt đầu thực thi đến sự kiện

Session\_OnStart chuẩn bị cho phiên kết nối. Các biến Application thường được khởi tạo bên trong sự kiện này.

- Session\_OnStart: Sự kiện này phát sinh mỗi khi có một người dùng mới yêu cầu trang ASP của ứng dụng Web lần đầu tiên.

- Session\_OnEnd: Sự kiện này được gọi khi phiên làm việc Session của người dùng chấm dứt. Phiên làm việc được xem là chấm dứt khi nó hết hạn mặc định cho thời gian hết hạn của Session là 20 phút.

- Application\_OnEnd: Sự kiện này phát sinh khi không còn người dùng nào tương tác với ứng dụng Web của chúng ta nữa. Thông thường sự kiện này được gọi khi trình chủ IIS ngừng hoạt động. Sự kiện này giúp ứng dụng Web lưu các thông tin trạng thái cần thiết xuống đĩa cứng phục vụ cho quá trình hoạt động trở lại của trình chủ sau đó.

Thủ tục xử lý các sự kiện này trong file global.asa được cài đặt theo mẫu sau:

```
<script language = "vbscript" runat = "server">  
sub Application_OnStart  
.....Code  
end sub  
sub Application_OnEnd  
.....Code  
end sub  
sub Session_OnEnd  
.....Code  
end sub  
</script>
```

- Khai báo biến trong global.asa: các biến cũng phải ở dạng biến Application hoặc biến Session. Có thể đặt vào một đoạn bất kỳ hoặc trong các thủ tục Application\_OnStart, Session\_OnStart, giải phóng biến ở Application\_OnEnd hoặc Session\_OnEnd.

## **4.2. VISUALBASIC SCRIPT LANGUAGE (VBSCRIPT)**

### **4.2.1. Giới thiệu về VBScript:**

- VBScript là một thành phần mới nhất trong họ ngôn ngữ lập trình Visual Basic, cho phép tạo ra những script sử dụng được trên nhiều môi trường khác nhau

như các script chạy trên Trình duyệt của client (Ms Internet Explorer 3.0) hay trên Web server (Ms Internet Information Server 3.0).

- Cách viết VBScript tương tự như cách viết các ứng dụng trên Visual Basic hay Visual Basic for Application. VBScript giao tiếp với các ứng dụng chủ (host application) bằng cách sử dụng các ActiveX Scripting.

#### 4.2.2. Các kiểu dữ liệu của VBScript:

VBScript chỉ có một loại dữ liệu được gọi là Variant. Variant là một kiểu dữ liệu đặc biệt có thể chứa đựng những loại thông tin khác nhau tùy theo cách sử dụng. Dĩ nhiên nó cũng là kiểu dữ liệu được trả về bởi tất cả các hàm. Ở đây đơn giản nhất một Variant có thể chứa thông tin số hoặc chuỗi tùy theo ngữ cảnh sử dụng. Các loại dữ liệu (subtype) mà Variant có thể biểu diễn được trình bày trong bảng sau:

Kiểu dữ liệu	Diễn giải
Empty	Dữ liệu trống.
Null	Dữ liệu không hợp lệ.
Boolean	Giá trị logic True hoặc False.
Byte	Số nguyên từ 0 đến 255.
Integer	Số nguyên từ -32768 đến 32767.
Currency	Kiểu tiền tệ từ -922,337,203,685,477.5808 đến 922,337,203,685,477.5807
Long	Kiểu số nguyên dài từ -2,147,483,648 đến 2,147,483.647.
Single	Kiểu số chính xác đơn.
Double	Kiểu số chính xác kép.
Date (Time)	Kiểu ngày từ 01/01/100 đến 31/12/9999
String	Kiểu chuỗi, tối đa 2 tỉ ký tự.
Object	Kiểu đối tượng.
Error	Chứa một số lỗi.

#### 4.2.3. Biến trong VBScript:

- Một biến là một tên tham khảo đến một vùng nhớ, là nơi chứa thông tin của chương trình mà thông tin này có thể được thay đổi trong thời gian script chạy.

Ví dụ: có thể đặt một biến tên là ClickCount để đếm số lần người sử dụng click vào một đối tượng trên một trang Web nào đó.

- Vị trí của biến trong bộ nhớ không quan trọng, ta chỉ truy xuất đến nó thông qua tên mà thôi. Trong VBScript biến luôn có kiểu là Variant.

Khai báo biến: dùng phát biểu Dim, Public (cho biến toàn cục) hay Private (cho biến cục bộ).

Ví dụ: Dim ClickCount

Tên Biến phải bắt đầu bằng một kí tự chữ, trong tên biến không chứa dấu chấm, chiều dài tối đa là 255 kí tự và 1 biến là duy nhất trong tầm vực mà nó được định nghĩa.

Tầm vực và thời gian sống của một biến: có 2 loại biến là procedure-level và script-level tương ứng với 2 cấp tầm vực là local và script-level. Thời gian sống của một biến script-level được tính từ khi nó được khai báo đến khi script kết thúc, đối với biến local là từ khi nó được khai báo đến khi procedure chứa nó kết thúc.

Biến trong VBScript có thể là biến đơn hay là dãy. Khi khai báo Dim A(10) thì VBScript tạo ra một dãy có 11 phần tử (vì phần tử đầu có chỉ số là 0). Một biến dãy có thể mở rộng tối đa đến 60 chiều, nhưng thường dùng từ 2 đến 4 chiều. Có thể thay đổi kích thước một dãy trong thời gian chạy bằng cách dùng phát biểu ReDim.

Ví dụ: Dim MyArray(25)...

ReDim MyArray(30) hay

ReDim Preserve MyArray(30) ‘giữ lại các giá trị trong dãy cũ.

#### **4.2.4. Hằng trong VBScript:**

- Hằng là một tên có nghĩa đại diện cho 1 số hay chuỗi và không thể thay đổi trong quá trình chạy. Tạo một hằng bằng phát biểu Const.

Ví dụ: Const MyString = “This is my string “

#### **4.2.5. Các toán tử trong VBScript:**

- VBScript có các toán tử khác nhau như số học, luận lý, so sánh. Nếu muốn chỉ định thứ tự ưu tiên của toán tử một cách rõ ràng thì dùng dấu ngoặc ( ), còn không thì thứ tự ưu tiên như sau (từ trên xuống dưới, từ trái sang phải):

- So sánh: =, <>, <, >, <=, >=, Is.

- Luận lý: Not, And, Or, Xor, Eqv, Imp.

Toán tử \* và /, + và - có cùng độ ưu tiên và được thực hiện từ trái sang phải.

Toán tử	Diễn giải
+	Cộng hai số
&	Toán tử And. Nối hai biểu thức logic. Trả về kết quả True nếu hai biểu thức đều True, ngược lại là False. Ngoài ra cũng có thể dùng để nối hai chuỗi.
/	Chia hai số.
^	Lũy thừa.
\	Phép chia nguyên.
IS	So sánh hai biến tham chiếu đối tượng.
MOD	Phép chia lấy phần dư.
*	Nhân hai số.
NOT	Phủ định giá trị biểu thức.
OR	Toán tử OR. Nối hai biểu thức logic. Trả về kết quả True nếu có ít nhất một trong hai biểu thức có giá trị True, ngược lại là False.
XOR	Toán tử XOR. Nối hai biểu thức logic. Trả về kết quả True khi cả hai biểu thức đều cùng giá trị True hoặc False. Ngược lại thì trả về giá trị False.

#### 4.2.6. Câu lệnh và cấu trúc điều khiển chương trình

##### - câu lệnh

Tên	Diễn giải
Call	Gọi một hàm hoặc thủ tục con.
Const	Khai báo một giá trị hằng.
Dim	Khai báo biến.
Erase	Giải phóng một mảng động.
Exit	Thoát khỏi một thủ tục hay câu lệnh điều kiện.
Function	Khai báo một hàm.
On Error	Bẫy lỗi.
Option Explicit	Bắt buộc khai báo biến trước khi sử dụng.
Private	Khai báo các biến riêng.

Public	Khai báo các biến dùng chung.
Randomize	Khởi động bộ phát sinh số ngẫu nhiên.
Redim	Định nghĩa lại các biến mảng động.
Rem	Câu lệnh chú thích chương trình.
Set	Gán một đối tượng tham chiếu tới một biến hoặc một thuộc tính.
Sub	Khai báo thủ tục.
'	Câu lệnh chú thích.

- Cấu trúc điều khiển chương trình

<p>Câu lệnh điều khiển IF:</p> <pre>IF &lt;điều kiện&gt; then     &lt;lệnh 1&gt; Else     &lt;lệnh 2&gt; End if</pre>	<p>Câu lệnh IF lồng nhau:</p> <pre>IF &lt;điều kiện 1&gt; then     &lt;lệnh 1&gt; Else if &lt; điều kiện 2&gt; then     &lt;lệnh 2&gt; Else     &lt;Câu lệnh khác&gt; End if</pre>	
<p>Cấu trúc chọn lựa Select Case:</p> <pre>Select Case biểu thức     Case danh sách giá trị         Các câu lệnh     Case Else         Câu lệnh khác End Select</pre>	<p>Cấu trúc lặp For ... Next (lặp với số lần lặp xác định)</p> <pre>For counter = start To end Step stepsize     Các câu lệnh Exit For     Các câu lệnh Next</pre>	
<p>Cấu trúc lặp For Each ... Next (lặp với từng phần tử thuộc nhóm)</p> <pre>For Each phần tử In nhóm     Các câu lệnh Exit For</pre>	<p>Do While điều kiện</p> <pre>Các câu lệnh Exit Do Loop</pre>	<p>Do Until điều kiện</p> <pre>Các câu lệnh Exit Do Loop</pre>

Next	Do Các câu lệnh Exit Do Loop While Điều khiển	Do Các câu lệnh Exit Do Loop Until Điều khiển
------	---	---

#### 4.2.7. Đối tượng VBScript:

VBScript cung cấp bốn đối tượng xây dựng sẵn.

Tên	Diễn giải
Dictionary	Đối tượng lưu khoá và nội dung dữ liệu.
Err	Chứa các thông báo lỗi khi chạy kịch bản.
FileSystemObject	Cung cấp các truy xuất tới một hệ thống tập tin máy tính.
TextStream	Cung cấp truy xuất tập tin tuần tự.

#### 4.2.8. Phương thức và thuộc tính VBScript:

Phương thức	Diễn giải
Add	Thêm cặp khóa và nội dung tới đối tượng Dictionary.
Clear	Xóa tất cả các xác lập của đối tượng Err.
Close	Đóng một tập tin TextStream đang mở.
CreateTextFile	Tạo tên tập tin và trả lại một đối tượng TextStream.
Exists	Trả lại khóa đang tồn tại trong đối tượng Dictionary.
Items	Trả lại mảng chứa nội dung tất cả các mục trong Dictionary.
Keys	Trả lại mảng chứa nội dung tất cả các khóa trong Dictionary.
OpenTextFile	Mở tập tin và trả lại đối tượng TextStream.
Raise	Phát sinh một lỗi khi chạy chương trình.
Read	Đọc một số ký tự từ tập tin TextStream.
ReadAll	Đọc toàn bộ tập tin TextStream.
ReadLine	Đọc một dòng trong tập tin TextStream.



Remove	Gỡ bỏ một cặp khóa và nội dung khỏi đối tượng Dictionary.
RemoveAll	Gỡ bỏ tất cả các khóa và nội dung trong Dictionary.
Skip	Nhảy qua một số ký tự khi đọc dữ liệu trong tập tin TextStream.
SkipLine	Nhảy tới dòng kế tiếp khi đọc dữ liệu trong tập tin TextStream.
Write	Ghi một chuỗi vào tập tin TextStream.
WriteBlankLines	Chèn một số dòng mới xác định tới tập tin TextStream.
WriteLine	Ghi một chuỗi xác định vào một dòng mới trong tập tin TextStream.

Thuộc tính	Diễn giải
AtEndOffline	Con trỏ tập tin có đang ở dòng cuối tập tin TextStream hay không.
AtEndOfStream	Con trỏ tập tin có đang ở cuối tập tin TextStream hay không.
Column	Số cột vị trí của ký tự trong tập tin TextStream.
CompareMode	Chế độ so sánh chuỗi khóa trong đối tượng Dictionary.
Count	Số mục trong đối tượng Dictionary.
Description	Chuỗi thông báo lỗi.
HelpContext	Số ID của một đề mục trong tập tin trợ giúp.
HelpFile	Đường dẫn tới một tập tin trợ giúp.
Item	Nội dung ứng với khóa được xác định trong Dictionary.
Key	Đặt một khóa trong đối tượng Dictionary.
Line	Cho biết dòng hiện hành trong tập tin TextStream.
Number	Giá trị số xác định lỗi.
Source	Tên đối tượng gây ra lỗi.

## 4.3: QUẢN LÝ CƠ SỞ DỮ LIỆU VỚI ASP

### 4.3.1. Khái niệm về ADO:

- ADO (ActiveX Data Object) là một tập hợp các đối tượng cho phép nhà lập trình ASP kết nối với cơ sở dữ liệu.

- ADO là sự kết hợp cơ sở dữ liệu dựa trên DAO (Data Access Object) và RDO (Remote Data Object) được sử dụng trong những công cụ xây dựng ứng dụng như Visual Basic.

- ADO là một phần của OLEDB (Object Linking and Embedding for Databases), là một cách mới để truy xuất và tổ hợp dữ liệu. ADO cho phép tạo nhanh một tập mẫu tin để lấy dữ liệu.

### 4.3.2. Các đối tượng trong ADO

- Đối tượng Connection:

- Cho phép thực hiện việc mở kết nối đến nguồn dữ liệu cần truy xuất. Các thông tin kết nối bổ sung khác như tên đăng nhập cơ sở dữ liệu (username), mật khẩu (password), tên máy chủ (server) ... thường được lưu vào một chuỗi gọi là chuỗi kết nối (connection string).

- Để có thể kết nối và truy xuất vào nguồn dữ liệu, không bắt buộc phải tạo ra đối tượng Connection. Các đối tượng như Recordset, Record... cũng cho phép mở trực tiếp kết nối, tuy nhiên sử dụng đối tượng Connection sẽ cho phép tách biệt thao tác kết nối và thao tác truy cập dữ liệu. Đối tượng này còn cung cấp thêm một số chức năng như cho phép thực thi câu lệnh SQL tác động dữ liệu (INSERT, UPDATE, DELETE...).

- Các phương thức của đối tượng Connection:

• Open: Sử dụng để mở một kết nối với cơ sở dữ liệu. Sau khi tạo một instance cho đối tượng Connection, ta có thể mở một kết nối với nguồn dữ liệu để có thể truy xuất dữ liệu.

Cú pháp: `dataConn.Open ConnectionString, UserID, Password`

Trong đó ConnectionString là chuỗi định nghĩa tên của Data Source (DNS), tên này được khai báo trong ODBC. UserID và Password sẽ thực hiện việc mở Data Source.

• Execute: Phương thức này cho phép thi hành một câu lệnh SQL, tác động lên Data Source.

Theo trình tự sau khi đã có một kết nối tới nguồn dữ liệu, ta có thể sử dụng nó với phương thức Execute với cú pháp trong VBScript như sau:

Set rs = dataConn.Execute (CommandText, RecordAffected, Options)

Thông thường các giá trị trên sẽ được gán cho tên (danh hiệu), hằng và được đặt trong các file include.

- Close: trong các trang ASP, sau khi đã xử lý xong dữ liệu trên Data Source, trước khi kết thúc trang sử dụng phải đóng lại các kết nối đã mở. Việc đóng kết nối thực hiện nhờ phương thức Close.

Cú pháp: dataConn.Close

Nếu chương trình không thực hiện việc này thì ASP sẽ tự động đóng Connection đã mở trong trang, mỗi khi người sử dụng tham khảo qua trang khác (đi khỏi tầm vực của biến instance). Nếu ta tạo instance và mở kết nối trong các thủ tục Application hoặc Session\_OnStart thì việc đóng các Connection này sẽ được thể hiện trong thủ tục OnEnd.

- BeginTrans: Bắt đầu một giao dịch mới.

Cú pháp: intTransactionLevel = dataConn.BeginTrans

- CommitTrans: Lưu các thay đổi trong giao dịch vào cơ sở dữ liệu.

Cú pháp: dataConn.CommitTrans

- RollbackTrans: Bỏ qua các thay đổi trong giao dịch.

Cú pháp: dataConn.RollbackTrans

- Các thuộc tính của đối tượng Connection:

- ConnectionTimeout: Xác định thời gian đợi kết nối trước khi trả lại thông báo lỗi, mặc định là 15 giây.

- Đối tượng Recordset:

- Cung cấp kết quả trả về từ câu lệnh truy vấn một tập các mẫu tin, trang ASP có thể dùng vòng lặp duyệt qua các mẫu tin này và hiển thị dữ liệu kết xuất ra trang Web phía trình duyệt. Ngoài ra Recordset còn cho phép lọc dữ liệu từ tập các mẫu tin, truy xuất đến từng trường cụ thể của mẫu tin thông qua đối tượng Field hoặc danh sách các trường trong mẫu tin thông qua đối tượng Fields.

- Các phương thức của đối tượng Recordset:

- AddNew: Là phương thức dùng để thêm mẫu tin mới với các giá trị xác định.

Cú pháp: rs.AddNew Fields, Values

- Close: Đóng lại đối tượng Recordset và các đối tượng con của nó.

Cú pháp: rs.Close

- Delete: Xóa đi mẫu tin hiện hành trong Recordset.

Cú pháp: `rs.Delete recordAffected`

- Move: Di chuyển con trỏ mẫu tin hiện hành tới số mẫu tin xác định.

Cú pháp: `rs.Move NumRecs, start`

- MoveFirst: Di chuyển con trỏ mẫu tin hiện hành về mẫu tin đầu tiên trong tập mẫu tin.

Cú pháp: `rs.MoveFirst`

- MoveNext: Di chuyển con trỏ mẫu tin hiện hành tới mẫu tin kế tiếp trong tập mẫu tin.

Cú pháp: `rs.MoveNext`

- MoveLast: Di chuyển con trỏ mẫu tin hiện hành về mẫu tin cuối cùng trong tập mẫu tin.

Cú pháp: `rs.MoveLast`

- NextRecordset: Di chuyển tới tập mẫu tin kế tiếp.

Cú pháp: `set newRs = oldrs.NextRecordset (recordsAffected)`

- Open: Mở một Record mới, sau khi mở xong thì con trỏ nằm ở Record đầu tiên trong Recordset.

Cú pháp: `rs.Open (Source, ActiveConnection, CursorType, LockType, Options)`

- Requery: Truy xuất lại cơ sở dữ liệu và cập nhật tập mẫu tin.

Cú pháp: `rs.Requery`

- Update: Cập nhật mẫu tin hiện hành.

Cú pháp: `rs.Update fields, values`

- UpdateBatch: Viết gói mẫu tin được cập nhật lên đĩa.

Cú pháp: `rs.UpdateBatch recordsAffected`

- CancelUpdate: Hủy mẫu tin đang cập nhật.

Cú pháp: `rs.CancelUpdate`

- Supports: Xác định chức năng tập mẫu tin hỗ trợ.

Cú pháp: `set bool = rs.Supports (cursorOption)`

- Các thuộc tính của đối tượng Recordset:

- AbsolutePage: Xác định trang để di chuyển con trỏ mẫu tin hiện hành tới.
- AbsolutePosition: Xác định mẫu tin để di chuyển mẫu tin hiện hành tới.
- ActiveConnection: Xác định đối tượng Connection của tập mẫu tin.

- BOF (Begin Of File): Có giá trị True nếu con trỏ hiện hành nằm ở đầu tập mẫu tin.
- EOF (End Of File): Có giá trị True nếu con trỏ hiện hành nằm ở cuối tập mẫu tin.
- RecordCount: Trả về tổng số mẫu tin trong tập mẫu tin.
- Source: Xác định nguồn tạo mẫu tin (câu lệnh SQL, table hay thủ tục lưu trữ).
- Status: Cho biết trạng thái tập mẫu tin.
- PageSize: Số mẫu tin trong một trang.
- LockType: Xác định loại khóa áp dụng cho tập mẫu tin khi nó được mở.
  - + adLockReadOnly: Tập mẫu tin chỉ đọc.
  - + adLockPessimistic: Khóa mẫu tin khi nó trở thành mẫu tin hiện hành.
  - + adLockOptimistic: Khóa mẫu tin khi gọi phương thức Update.
  - + adLockBatchOptimistic: Khóa gói các mẫu tin khi gọi phương thức BatchUpdate.

- Đối tượng Command:

- Cho phép chuyên tham số vào các lệnh thực thi SQL. Các tham số có thể nhận giá trị trả về sau khi thực thi. Command có thể dùng cho cả hai mục đích: thực thi các câu lệnh SQL không cần nhận kết quả trả về (như INSERT, DELETE, UPDATE, các Store Procedure) hoặc thực thi các lệnh trả về tập mẫu tin như lệnh SELECT.

- Các thuộc tính của đối tượng Command:

- ActiveConnection: Xác định đối tượng Connection dùng cho đối tượng Command.
- CommandText: Tên bảng, tên thủ tục lưu trữ hay câu lệnh SQL cần mở, thi hành.
- CommandTimeout: Số giây đợi lệnh thi hành trước khi dừng và đưa ra lỗi. Mặc định là 30 giây.

• CommandType: Xác định kiểu lệnh chứa trong CommandText.

- + adCmdText: Câu lệnh SQL.
- + adCmdTable: Tên bảng.
- + adStoreProc: Thủ tục lưu trữ.
- + adCmdUnknown: Kiểu lệnh chưa biết.

- Các phương thức của đối tượng Command:

- CreateParameter: Dùng để tạo một đối tượng tham số mới cho tập hợp Parameter.

Cú pháp: Set paramObj = dataCmd.CreateParameter (Name, [Type, Direction, Size, Value])

Type: Xác định loại dữ liệu của tham số.

Direction: Xác định tham số input, output hoặc cả hai.

Size: Kích thước tối đa của tham số (tính bằng byte).

Value: Giá trị tham số.

- Execute: Thi hành lệnh trong thuộc tính CommandText.

Cú pháp: Set dataRS = dataCmd.Execute ([ RecordsAffected,  
Parameters, Options ])

RecordsAffected: Số mẫu tin chịu tác động bởi lệnh.

Parameters: Mảng các tham số được gửi cùng với lệnh.

Options: Xác định loại lệnh tìm thấy trong CommandText.

- Đối tượng và tập hợp Fields:

- Trình bày một cột dữ liệu với kiểu dữ liệu chung.

- Các thuộc tính của Fields:

- ActualSize: Chiều dài thực sự của giá trị trong Field.
- Attributes: Xác định các thuộc tính của Field.
- DefinedSize: Kích thước đã định nghĩa cho Field.
- Name: Trả lại tên Field.
- OriginalValue: Giá trị của Field trước khi nó được cập nhật.
- Type: Kiểu dữ liệu của Field.
- UnderlyingValue: Trả lại giá trị Field trong cơ sở dữ liệu.
- Value: Xác định giá trị của Field.

- Đối tượng và tập hợp Errors:

- Chứa các chi tiết về lỗi truy xuất dữ liệu liên quan với một toán tử đơn liên quan ADO.

- Các thuộc tính của Errors:

- Description: Mô tả ngắn về lỗi phát sinh bởi trình cung cấp hay ADO.
- Number: Trả lại một số nhận dạng duy nhất về lỗi đó.
- Source: Xác định đối tượng hay ứng dụng đã phát sinh lỗi.
- SQLState: Trả lại mã lỗi ANSI SQL.

- Đối tượng và tập hợp Parameters:

- Trình bày tham số hay đối số được kết hợp với một đối tượng Command dựa trên văn bản hay thủ tục lưu trữ.

- Các thuộc tính của Parameters:

- Attributes: Xác định kiểu dữ liệu tham số chấp nhận được.
  - + adParamSigned: Tham số có nhận giá trị dấu hay không.
  - + adParamNullable: Tham số có nhận giá trị Null hay không.
  - + adParamLong: Tham số có nhận giá trị Long.
- Direction: Xác định hướng truy xuất tham số.
  - + adParamInput: Chỉ nhận vào.
  - + adParamOutput: Chỉ xuất ra.
  - + adParamInputOutput: Cả nhận lẫn xuất.
  - + adParamReturnValue: Trả lại giá trị.
- Name: Xác định tên tham số.
- Size: Số byte tối đa tham số có thể nhận.
- Type: Xác định kiểu dữ liệu của tham số.
- Value: Giá trị tham số.
  - Các phương thức của Parameters:
- Append: Thêm đối tượng Parameter mới vào tập hợp Parameters của đối tượng Command.

Cú pháp: `dataCmd.Append ParamObj`

- Delete: Xóa tham số khỏi tập hợp Parameters.

Cú pháp: `dataCmd.Delete Index`

- Đối tượng Property:

- Giới thiệu đặc tính động của đối tượng ADO do nhà cung cấp định nghĩa

## 4.4 SQL SERVER

### 4.4.1. Giới thiệu

SQL Server sử dụng cơ sở dữ liệu quan hệ có dữ liệu bên trong được tổ chức thành các bảng, các bảng được tổ chức bằng cách nhóm dữ liệu theo cùng chủ đề và có chứa các cột và các hàng thông tin. Sau đó các bảng này lại liên hệ với nhau bởi bộ Database Engine khi có yêu cầu.

### 4.4.2. Các kiểu dữ liệu

Loại	Kiểu dữ liệu	Chú giải
String	char(n), varchar	Lưu trữ các chuỗi ký tự
Binary	binary(n), varbinary	Lưu trữ thông tin nhị phân trong cặp 2 bytes

Integer	int, smallint, tinyint	Lưu trữ các giá trị nguyên
Approximate Numeric	float, real	Lưu trữ các số gần đúng
Exact Numeric	decimal, numeric	Lưu trữ các số chính xác
Special	bit, text, image	Lưu trữ bit, văn bản nhiều hơn 8000 bytes, hay dữ liệu ảnh
Date and time	datetime, smalldatetime	Lưu trữ ngày và giờ
Money	money, smallmoney	Lưu trữ các giá trị tiền tệ
Auto-incrementing datatypes	identity, timestamp	Lưu trữ các giá trị cho phép sẽ tăng tự động hay được gán trị bởi SQL server.
Synonyms		Kiểu dữ liệu ANSI thành kiểu dữ liệu SQL server
User-defined		Tạo kiểu dữ liệu riêng để lưu trữ thông tin

#### 4.4.3. Truy xuất dữ liệu bằng truy vấn:

- Câu lệnh SELECT:

- Dùng để truy vấn các hàng và cột của bảng trong cơ sở dữ liệu.
- Có 3 thành phần trong câu lệnh SELECT: SELECT, FROM và WHERE.
- Cú pháp câu lệnh SELECT đơn giản:

```
SELECT <column_list>
      FROM <table_list>
      WHERE <search_criteria>
```

-Phần SELECT xác định cột muốn khôi phục, phần FROM xác định bảng có cột được khôi phục và phần WHERE giới hạn hàng được khôi phục.

- Cú pháp đầy đủ của câu lệnh SELECT là:

```
SELECT { ALL | DISTINCT } { TOP n { PERCENT } { WITH TIES } }
      { INTO new_table }
```



```

{ FROM <table_sources> }
{ WHERE <search_condition>}
[ GROUP BY [ALL] group_by_expression [,...n]
[ WITH {CUBE | ROLLUP}]
      [HAVING <search_condition> ]
      [ORDER BY {column_name [ASC | DESC ]} [...n] ]
      [COMPUTE (expression)]

```

Mệnh đề GROUP BY sẽ nhóm dữ liệu tổng gộp trong mệnh đề WHERE để trả ra những dòng đơn. Mệnh đề GROUP BY ALL trả về tất cả các nhóm, bao gồm cả những nhóm không đáp ứng tiêu chuẩn ở mệnh đề WHERE.

Mệnh đề HAVING đặt tiêu chuẩn xác định những dòng nào được trả ra bởi mệnh đề GROUP BY.

Các mệnh đề GROUP BY và HAVING cần thoả mãn những yêu cầu nhất định để phù hợp với tiêu chuẩn ANSI. Một trong những yêu cầu này là mệnh đề GROUP BY phải chứa tất cả các cột không kết hợp trong SELECT column\_list. Yêu cầu khác là những cột tiêu chuẩn trong mệnh đề HAVING chỉ trả về một giá trị.

Mệnh đề ORDER BY dùng để sắp xếp dữ liệu, mệnh đề này không dùng ở cột có kiểu dữ liệu text (văn bản) hoặc image (hình ảnh).

Mệnh đề COMPUTE trả về những dòng chi tiết và dòng tổng số.

- Câu lệnh INSERT:

- Dùng để thêm một hàng vào một bảng tại một thời điểm.
- Cú pháp:

<pre> INSERT [INTO] {&lt;table_or_view&gt;} {column_list} VALUES      {select_statement execute_statement} - Câu lệnh UPDATE:       - Dùng để thay đổi giá trị của các cột trong hàng       - Cú pháp: UPDATE </pre>	<pre> [, column-name2=...] [WHERE search_conditions] Mệnh đề SET cho chỉ định những cột được cập nhật. - Câu lệnh DELETE:       - Cho phép xóa một hay nhiều hàng dữ liệu từ một bảng.       - Cú pháp: DELETE [ FROM ] { table_name   view_name} </pre>
--	--

<pre>{table_name   view_name} SET Column_name1={expression1   NULL   (select_statement)}</pre>	<pre>[ WHERE clause ].</pre>
--	------------------------------

## 4.5 - XÂY DỰNG CHƯƠNG TRÌNH

### 4.5.1- MỘT SỐ GIAO DIỆN CHƯƠNG TRÌNH

#### 4.5.2- Giao diện chính của Website.



#### 4.5.3- Trang đăng nhập hệ thống Website.



ĐẠI HỌC DÂN LẬP HẢI PHÒNG  
Chất lượng đào tạo là sự sống còn của nhà trường

ISO 9001 : 2008

**ĐĂNG NHẬP HỆ THỐNG**

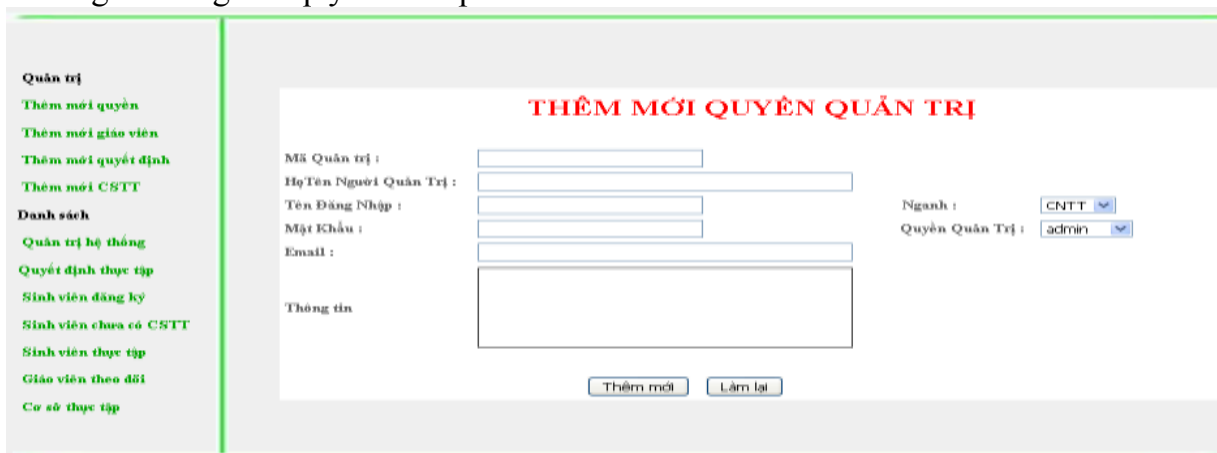
Tên đăng nhập:

Mật khẩu:

- Bộ môn sẽ cấp cho mỗi sinh viên, giáo viên một Accounts để đăng nhập vào hệ thống. Sau khi đăng nhập vào hệ thống Website tự động điều hướng đến trang người dùng.

#### 4.5.4- Trang quản lý của hệ thống Website.

- Trang bổ xung các quyền và cấp Accounts.



**THÊM MỚI QUYỀN QUẢN TRỊ**

Mã Quản trị:

Họ Tên Người Quản Trị:

Tên Đăng Nhập:

Mật khẩu:

Email:

Ngành:

Quyền Quản Trị:

Thông tin:

#### 4.5.5- Trang bổ xung giáo viên theo dõi.



**THÊM MỚI GIÁO VIÊN THEO DÕI**

Mã Giáo Viên:

Họ tên Giáo Viên:

Ngành:

Email:

Thông tin:

#### 4.5.6- Trang bổ xung quyết định thực tập.

**Quản trị**

Thêm mới quyền

Thêm mới giáo viên

Thêm mới quyết định

Thêm mới CSTT

**Danh sách**

Quản trị hệ thống

Quyết định thực tập

Sinh viên đăng ký

Sinh viên chưa có CSTT

Sinh viên thực tập

Giáo viên theo dõi

Cơ sở thực tập

### THÊM MỚI QUYẾT ĐỊNH THỰC TẬP

Mã Số Quyết Định:

Ngày ra OD:  Ngành:

Tên Quyết Định:

Nội Dung:

#### 4.5.7- Thống kê danh sách các quyền(Thống kê theo từng ngành).

### THỐNG KÊ DANH SÁCH QUYỀN

Chọn ngành cần tìm :

Thông kê theo quyền

DANH SÁCH QUẢN TRỊ WEBSITE

Mã QT	Họ Tên Quản trị	Ngành	Tên Đăng Nhập	Mật Khẩu	Quyền	Email	Mô Tả	Chọn
cnm001	Vu Anh Hùng	CNTT	vahung	hung123	admin	vahung@gmail.com		<a href="#">(Xóa)</a>
cnm002	Ngô Trùng Giang	CNTT	ngiang	giang123	admin	ngiang@gmail.com	Phó chủ nhiệm	<a href="#">(Xóa)</a>
cnm008	Nguyễn Thu Xuân Hồng	CNTT	truhong	huong123	admin	truhong@gmail.com	Phó chủ nhiệm	<a href="#">(Xóa)</a>

[Quay về trước](#)

#### 4.5.8- Thống kê danh sách quyết định thực tập.

**Quản trị**

Thêm mới quyền

Thêm mới giáo viên

Thêm mới quyết định

Thêm mới CSTT

**Danh sách**

Quản trị hệ thống

Quyết định thực tập

Sinh viên đăng ký

Sinh viên chưa có CSTT

Sinh viên thực tập

Giáo viên theo dõi

Cơ sở thực tập

### DANH SÁCH QUYẾT ĐỊNH THỰC TẬP

Mã Số OD	Ngành	Tiêu đề	Ngày đăng	Lựa chọn
nh1	DT	Thu	12/8/2010	( Xóa)
s02	CNTT	Hệ	12/2/2010	( Xóa)
s09	QT	Thực tập tốt nghiệp	10/9/2010	( Xóa)
s05	CNTT	Thực tập	12/12/2010	( Xóa)
s11	BA		10/9/2010	( Xóa)

#### 4.5.9-Thống kê danh sách sinh viên đăng ký thực tập.(Thống kê theo từng ngành).

**THỐNG KÊ DANH SÁCH SINH VIÊN ĐĂNG KÝ THỰC TẬP**

Chọn ngành cần tìm :

**DANH SÁCH SINH VIÊN ĐĂNG KÝ THỰC TẬP**

Mã SV	Họ Tên Sinh Viên	Ngày sinh	Ngành	Email	Địa chỉ	Liên hệ	Lựa chọn
bf	h	8/12/1988	CNTT	h@gmail.com	bb	chua co CSTT	<a href="#">(Sửa / Xóa)</a>
cds	sz	12/13/1986	CNTT	g@gmail.com	Chua co CSTT	as	<a href="#">(Sửa / Xóa)</a>
g	54	10/23/2010 9:22:13 PM	CNTT	lopct1001	cd	Chua co CSTT	<a href="#">(Sửa / Xóa)</a>
s04	ni	10/23/2010 8:24:34 AM	CNTT	ni@gmail.com	cds	cong ty phuong a	<a href="#">(Sửa / Xóa)</a>

[Quay về trước](#)

#### 4.5.10-Thống kê danh sách giáo viên theo dõi.(Thống kê theo từng ngành).

**THỐNG KÊ DANH SÁCH GIÁO VIÊN THEO DÕI**

Chọn ngành cần tìm :

**DANH SÁCH GIÁO VIÊN THEO DÕI THỰC TẬP**

Mã GV	Họ Tên Giáo Viên	Ngành	Email	Thông tin
gv001	Nguyễn Trọng Thế	CNTT	trithe@gmail.com	Giáo viên giảng dạy ngoại ngữ.
gv002	Hồ Thị Hương Thơm	CNTT	trithom@gmail.com	Giảng dạy ASP...
gv003	Trần Ngọc Thái	CNTT	trithai@gmail.com	Giảng dạy môn Hệ điều hành,Java..

[Quay về trước](#)

**4.5.11-Thống kê danh sách sinh viên chưa có đi thực tập.(Thống kê theo từng ngành).**

**THỐNG KÊ DANH SÁCH SINH VIÊN ĐĂNG KÝ CHƯA CÓ CSTT**

Chọn ngành cần tìm :

**4.5.12-Thống kê danh sách sinh viên được đi thực tập.(Thống kê theo từng ngành).**

**THỐNG KÊ DANH SÁCH SINH VIÊN THỰC TẬP TẠI CSTT**

Chọn ngành cần tìm :

**DANH SÁCH SINH VIÊN THỰC TẬP TẠI CSTT**

Mã SV	Họ Tên Sinh Viên	Ngày sinh	Ngành	Lớp	Địa chỉ	Họ Tên Giáo Viên	Cơ Sở Thực Tập
tt	hb	3/12/1988	CNTT	tt	hb	tt	hg
oos	bx	12/13/1986	CNTT	s	Chưa có CSTT	hg	tttt

[Quay về trước](#)

#### 4.5.12-Trang thông tin sinh viên.

- Sau khi sinh viên có Accounts đăng nhập vào hệ thống Website. Sinh viên đăng ký thông tin sinh viên trên Website.
- Trang đăng ký thông tin trên Website.

**ĐĂNG KÝ THÔNG TIN SINH VIÊN**

Họ tên Sinh Viên	<input type="text"/>	Ngày Sinh	<input type="text"/>
Email	<input type="text"/>	Ngành	CNTT <input type="button" value="v"/>
Địa chỉ	<input type="text"/>	Lớp Học	<input type="text"/>
Liên Hệ	<input type="text" value="Chưa có CSTT"/>	(Không có CSTT nhập _ko )	

#### 4.5.13- Trang viết nhật ký sinh viên.

**VIẾT NHẬT KÝ THỰC TẬP**

Mã bài viết :	<input type="text"/>	Ngày viết :	<input type="text"/>
Tiêu đề :	<input type="text"/>		
Nội dung	<input type="text"/>		

[Quay lại](#) 4.5.

#### 14-Trang thông tin giáo viên nhận xét.

**NHẬN XÉT GIÁO VIÊN THEO DÕI THỰC TẬP**

Mã đề tài :	<input type="text"/>
Tên đề tài báo cáo :	<input type="text"/>
Nội dung bài báo cáo :	<input type="text"/>
Giáo viên nhận xét :	<input type="text"/>

[Quay lại](#)

## KẾT LUẬN

Qua quá trình thực hiện đồ án tốt nghiệp với đề tài “Ứng dụng phương pháp phân tích thiết kế hướng cấu trúc vào bài toán xây dựng Website đăng ký và theo dõi sinh viên thực tập tốt nghiệp tại trường đại học Dân Lập Hải Phòng bản thân em tự thấy mình đã thu được các kết quả sau:

1. Hiểu biết thêm nhiều về phương pháp phân tích thiết kế hệ thống hướng cấu trúc. Từ đó đã áp dụng để phân tích thiết kế bài toán xây dựng Website đăng ký và theo dõi sinh viên thực tập tốt nghiệp
2. Có được kinh nghiệm thực tế khi được tham gia vào một dự án cụ thể, có thể áp dụng kiến thức đã được học vào thực tiễn, đồng thời thu thập được rất nhiều những kiến thức khác từ quá trình làm đồ án .
3. Tiến hành phân tích thiết kế hoàn thiện hệ thống bằng phương pháp hướng cấu trúc một cách đầy đủ.

Đồ án chỉ thực hiện ở mức độ đơn giản để mô tả các phân tích của bài toán. Để đưa hệ thống Website vào thực hiện, cần phải có các nghiên cứu công nghệ sử dụng và các tích hợp về cơ sở dữ liệu vào trong hệ thống..

Do thời gian thực hiện đề tài có hạn nên không thể tránh được sai sót, em rất mong nhận được các ý kiến đóng góp của thầy cô và các bạn để chương trình hoàn thiện hơn.

*Một lần nữa em xin chân thành cảm ơn!*

*Hải Phòng, ngày...tháng...năm 2010*

Sinh viên thực hiện

Đào Bá Bản



## Tài liệu tham khảo

[1] Nguyễn Văn Vy (2002), *Phân tích thiết kế các hệ thống thông tin hiện đại, hướng cấu trúc và hướng đối tượng*, NXB Thống kê, Hà Nội.

[2] Đoàn Văn Ban (2003), *Phân tích thiết kế hướng đối tượng bằng UML*, NXB Thống kê.

[3] <http://www.ebook.edu.vn>

DANH SÁCH ĐỊA ĐIỂM THỰC TẬP KHÓA 10 SINH VIÊN TỰ LIÊN HỆ						
STT	HỌ	TÊN	MSV	LỚP	ĐỊA ĐIỂM THỰC TẬP	ĐỊA CHỈ LIÊN HỆ
29	Đinh Quang	Khánh	100089	CT1002	Công ty cổ phần nhựa Phú Lâm	
59	Cao Mai	Liên	LT20052	CTL201	Chi cục thuế huyện Tiên Lãng	
64	Nguyễn Thành	Công	100438	CT1002	Công ty AROMA BAY CANDLE	
66	Đỗ Thanh	Tùng	090128	CT1002	Công ty AROMA BAY CANDLE	
66	Nguyễn Văn	Thành	101549	CT1002	Công ty AROMA BAY CANDLE	
30	Nguyễn Đức	Quỳnh	100021	CT1001	Công ty cổ phần vận tải và thương mại CPN Việt Nam	
31	Phạm Đình	Hậu	100018	CT1002	Công ty cổ phần vận tải và thương mại CPN Việt Nam	
7	Trần Hải	Hà	100262	CT1001	Công ty CP ALO	183C- Văn Cao-Đông Giang-Ngô Quyền-HP
8	Nguyễn Quang	Trung	100860	CT1001	Công ty CP ALO	183C- Văn Cao-Đông Giang-Ngô Quyền-HP
43	Trần Thu	Hương	100313	CT1001	Công ty CP công nghiệp Biển Đông	Km6, quốc lộ 5 Hùng Vương, Hồng Bàng, HP
46	Nguyễn Thị Mai	Hương	100376	CT1001	Công ty CP dịch vụ bảo vệ Đại Bình	
45	Nguyễn Văn	Dũng	100330	CT1002	Công ty CP máy tính Đông Nam	
44	Đặng Bá	Hậu	100439	CT1002	Công ty CP Mycam	402 Thanh Sơn, Kiến Thụy, HP
3	Bùi Thị	Bình	101477	CT1001	Công ty CP Sao Mai	01679006662
4	Nguyễn Nam	Khánh	100001	CT1001	Công ty CP Sao Mai	
5	Đặng Thanh	Tâm	101400	CT1001	Công ty CP Sao Mai	0974816634
14	Nguyễn Thị	Thúy	101386	CT1001	Công ty CP thép Cửu Long VINASHIN	Km8-Quán Toan-Hồng Bàng-HP
71	Trịnh Thanh	Hoa	100254	CT1002	Công ty CP thương mại và xuất nhập khẩu Trường Hải	
57	Nguyễn Mạnh	Cường	100462	CT1001	Công ty CP vận tải biển Ngọc Long	378C đường Lê Thánh Tông
62	Bùi Cao	Phát	LT20055	CTL201	Công ty CP vận tải biển Ngọc Long	276C-Lê Thánh Tông-HP
36	Trần Thị	Thúy	101276	CT1001	Công ty CP vận tải Đông Phong	số 630 Lê Thánh Tông, Đông Hải I, Hải An, HP
26	Vũ Minh	Đông	100514	CT1002	Công ty CP vận tải và thương mại CPN Việt Nam	0313540420
27	Nguyễn Thị Hồng	Ngân	100275	CT1002	Công ty giải pháp phần mềm Hữu Phong	
28	Nguyễn Thị Thúy	Ngọc	100259	CT1002	Công ty giải pháp phần mềm Hữu Phong	
37	Nguyễn Ngọc	Duy	400150	CTC401	Công ty Hoàng Minh	Số 77 đường Quang Trung
55	Nguyễn Quốc	Thư	100277	CT1002	Công ty máy tính Sơn Đạt	278A Lạch Tray-Ngô Quyền-HP
56	Vũ Văn	Hưng	101509	CT1002	Công ty máy tính Sơn Đạt	278A Lạch Tray-Ngô Quyền-HP
49	Nguyễn Thị Ngọc	Mai	100348	CT1001	Công ty thương mại đầu tư và phát triển đô thị	
58	Nguyễn Thị Thanh	Nhân	LT20054	CTL201	Công ty TNHH 2D	114-Nguyễn Đình Khiêm-Ngô Quyền-HP
47	Trương Đức	Phúc	100479	CT1001	Công ty TNHH dịch vụ và thương mại Hoàng Gia	
48	Diệp Hữu	Dũng	101416	CT1001	Công ty TNHH dịch vụ và thương mại Hoàng Gia	