

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHỆ TP.HCM**

-----oOo-----



# **ĐỒ ÁN TỐT NGHIỆP**

*Đề tài:*

## **TÌM HIỂU VÀ ỨNG DỤNG DRUPAL**

Sinh viên thực hiện:

ĐOÀN HỮU THIẾT – MSSV:103106037

NGUYỄN THỊ THUYẾT LINH – MSSV:104102072

HÀ MẠNH LONG – MSSV:104102077

GVHD:

VĂN THIÊN HOÀNG

**THÀNH PHỐ HỒ CHÍ MINH**

**NĂM 2009**

**BỘ GIÁO DỤC & ĐÀO TẠO  
ĐẠI HỌC KỸ THUẬT CÔNG  
NGHỆ TP.HCM**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  
ĐỘC LẬP – TỰ DO – HẠNH PHÚC**

**\_\_\_\_\_oOo\_\_\_\_\_**

Khoa: Công nghệ thông tin

**NHIỆM VỤ THỰC HIỆN LUẬN VĂN**

Bộ môn: Công nghệ phần mềm

**TỐT NGHIỆP CUỐI KHOA**

Họ và tên: Đoàn Hữu Thiết

MSSV: 103106037

Họ và tên: Nguyễn Thị Thùy Linh

MSSV: 104102072

Họ và tên: Hà Mạnh Long

MSSV: 104102077

Ngành: Công nghệ phần mềm

Lớp: 04CNPM

### **1. Đầu đề đồ án tốt nghiệp :**

Tìm hiểu và ứng dụng Drupal.

### **2. Nhiệm vụ :**

#### **a. Dữ liệu ban đầu :**

- Dựa vào hệ thống website khoa có sẵn.
- Xây dựng cho Khoa Công Nghệ Thông Tin một hệ thống website vừa đáp ứng các yêu cầu của sinh viên, giảng viên và cán bộ công nhân viên của khoa. Ngoài ra còn dễ dàng tích hợp, mở rộng, nâng cấp, đáp ứng các yêu cầu về sau của khoa. Để đáp ứng các yêu cầu trên, nhóm chúng tôi đã sử dụng một sản phẩm mã nguồn mở Drupal để nghiên cứu và ứng dụng nó trong việc phát triển các chức năng mà yêu cầu đề ra.

#### **b. Nội dung**

##### **Phần tìm hiểu công nghệ :**

- Tìm hiểu về phần mềm nguồn mở.

- **Tìm hiểu về Drupal.**
- **Cài đặt và sử dụng Drupal.**
- **Phát triển Drupal.**
  
- **Phần phân tích, thiết kế & xây dựng ứng dụng minh họa :**
  - **Phân tích và thiết kế hệ thống website khoa.**
  - **Phân tích hệ thống Drupal.**
  - **Ứng dụng Drupal để xây dựng trang web khoa.**

**3. Ngày giao nhiệm vụ đồ án: Ngày 06/10/2008.**

**4. Ngày hoàn thành nhiệm vụ: Ngày 04/01/2009.**

**5. Họ tên giáo viên hướng dẫn:**

**Văn Thiên Hoàng.**

Nội dung và yêu cầu đồ án tốt nghiệp đã  
thông qua.

*TP.HCM, ngày ..... tháng ..... năm 2007*

**CHỦ NHIỆM KHOA**

*(Ký và ghi rõ họ tên)*

**GIÁO VIÊN HƯỚNG DẪN CHÍNH**

*(Ký và ghi rõ họ tên)*

**PHẦN DÀNH CHO BỘ MÔN**

**Người duyệt (chấm sơ bộ) :** .....

**Đơn vị :** .....

**Ngày bảo vệ :** .....

**Điểm tổng quát :** .....

# NHẬN XÉT ĐÁNH GIÁ CỦA GIẢNG VIÊN HƯỚNG DẪN

ĐỀ TÀI: TÌM HIỂU VÀ ỨNG DỤNG CỦA DRUPAL.

Sinh Viên:

Đoàn Hữu Thiết                      MSSV: 103106037

Nguyễn Thị Thùy Linh              MSSV: 104102072

Hà Mạnh Long.                      MSSV: 104102077

Chuyên ngành: Công Nghệ Phần Mềm.

.....  
.....  
.....  
.....  
.....  
.....  
.....

Điểm bằng số: .....

Điểm bằng chữ: .....

*Tp. Hồ Chí Minh, ngày tháng năm*

**NGƯỜI NHẬN XÉT**

(Ký tên và ghi rõ họ tên)

# NHẬN XÉT ĐÁNH GIÁ CỦA GIẢNG VIÊN PHẢN BIỆN

ĐỀ TÀI: TÌM HIỂU VÀ ỨNG DỤNG DRUPAL.

Sinh Viên:

Đoàn Hữu Thiết                      MSSV: 103106037

Nguyễn Thị Thùy Linh              MSSV: 104102072

Hà Mạnh Long.                      MSSV: 104102077

Chuyên ngành: Công Nghệ Phần Mềm.

.....  
.....  
.....  
.....  
.....  
.....

Các câu hỏi :

.....  
.....  
.....  
.....  
.....  
.....

Điểm bằng số: .....

Điểm bằng chữ: .....

*Tp. Hồ Chí Minh, ngày tháng năm*

**NGƯỜI NHẬN XÉT**

(Ký tên và ghi rõ họ tên)

# Lời cảm ơn



*Đầu tiên, nhóm chúng em xin chân thành cảm ơn các thầy cô khoa Công nghệ Thông tin, trường Đại học Kỹ Thuật Công Nghệ đã hết lòng trong công tác giảng dạy, đã tận tình truyền đạt kho tàng kiến thức cho em trong những năm ngồi trên ghế giảng đường để nhóm chúng em vững bước trong tương lai.*

*Đặc biệt, nhóm chúng em chân thành cảm ơn thầy Văn Thiên Hoàng đã đưa chúng em đến với đề tài thú vị này. Trong quá trình làm luận văn, thầy đã cho nhóm chúng em những lời khuyên bổ ích, cung cấp những thông tin có giá trị, hỗ trợ cho việc lập kế hoạch, phân tích thiết kế luận văn. Thầy đã giúp nhóm chúng em làm việc một cách khoa học để hoàn thành luận văn đúng tiến độ, đáp ứng được các yêu cầu đặt ra.*

*Cuối cùng, chúng em cũng không quên gửi lòng biết ơn sâu sắc nhất đến cha mẹ, gia đình, là những bậc sinh thành nuôi dưỡng con, là hậu phương vững chắc để con yên tâm hoàn thành luận văn này.*

# MỤC LỤC

Trang

CHƯƠNG I: GIỚI THIỆU PHẦN MỀM NGUỒN MỞ VÀ CMS .....	1
<b>I.</b> Khái niệm phần mềm nguồn mở .....	1
I.1.    Định nghĩa ( <i>David Wheeler</i> ) .....	1
I.2.    Các học thuyết về PMNM .....	1
I.2.1.    Học thuyết FSF .....	1
I.2.2.    Học thuyết OSI .....	1
I.3.    Ưu điểm của phương pháp xây dựng PMNM .....	2
I.4.    Lịch sử của PMNM .....	2
I.5.    Lợi ích của PMNM .....	2
I.6.    Hạn chế của PMNM .....	2
I.7.    Những dự án PMNM thành công .....	3
I.8.    Các loại giấy phép PMNM .....	3
I.8.1.    Giấy phép đại chúng GNU General Public License .....	3
I.8.2.    Giấy phép BSD (Berkeley System Distribution) .....	3
<b>II.</b> Giới thiệu về CMS .....	3
II.1.    Khái niệm .....	3
II.2.    Các yêu cầu .....	4
II.3.    Danh sách các CMS đứng đầu năm 2008 .....	4
CHƯƠNG II: GIỚI THIỆU DRUPAL .....	6
<b>I</b> Tổng quan về Drupal .....	6
I.1    Khái niệm .....	6
I.2    Lịch sử .....	6
I.3    Bản quyền .....	6
I.4    Ưu điểm và nhược điểm .....	6
I.4.1    Ưu điểm .....	6
I.4.2    Nhược điểm .....	8
I.5    Các tính năng chung .....	8
❖ Collaborative Book: Cho phép thiết lập một "sách" và sau đó cho phép các cá nhân đóng góp nội dung .....	8
<b>II</b> Cách làm việc của Drupal .....	10
II.1    Technology Stack ( <i>Chồng công nghệ</i> ) .....	10
II.2    Core ( <i>Nhân của Drupal</i> ) .....	10
II.3    Modules .....	11
II.4    Hooks .....	12
II.5    Themes ( <i>Giao diện</i> ) .....	12
II.6    Nodes .....	12
II.7    Blocks .....	13
<b>III</b> Tổ chức lưu trữ của Drupal .....	13
<b>IV</b> Tính an toàn của Drupal .....	16



IV.1	An toàn đối với người dùng .....	16
IV.2	An toàn đối với mã nguồn.....	16
IV.3	Tính an toàn đối với dữ liệu.....	17
<b>V</b>	<b>Drupal so với các hệ CMS khác.....</b>	<b>18</b>
V.1	Sự khác biệt giữa Drupal với Joomla .....	18
V.1.1	Giống.....	18
V.1.2	Khác.....	18
<b>VI</b>	<b>Phát triển Drupal .....</b>	<b>20</b>
VI.1	Làm việc với cơ sở dữ liệu .....	20
VI.1.1	Lớp Database Abstraction Layer.....	20
VI.1.2	Kết nối với cơ sở dữ liệu .....	21
VI.1.3	Thực thi các truy vấn đơn giản.....	21
VI.1.4	Viết mới lớp cơ sở dữ liệu trừu tượng (Database Abstract Layer) .....	24
VI.2	Xây dựng giao diện (Theme) .....	24
VI.3	Xây dựng Module .....	28
VI.4	Các làm việc của Taxonomy .....	34
<b>CHƯƠNG III: MÔ TẢ HỆ THỐNG WEBSITE .....</b>		<b>40</b>
<b>I.</b>	<b>Phân tích.....</b>	<b>40</b>
I.1.	Mục đích.....	40
I.2.	Một số yêu cầu từ người dùng.....	40
I.3.	Mô tả bài toán.....	40
I.4.	Mô tả usecase.....	41
I.4.1.	User .....	41
I.4.2.	Giảng viên .....	42
I.4.3.	Thư ký .....	43
I.4.4.	Người quản trị.....	45
I.4.5.	Câu lạc bộ.....	46
I.4.6.	Diễn đàn.....	47
<b>II.</b>	<b>Đặc tả Usecase .....</b>	<b>48</b>
II.1.	Đăng nhập.....	48
II.2.	Quản lý câu lạc bộ .....	48
II.3.	Quản lý diễn đàn .....	49
II.4.	Đổi mật khẩu.....	50
II.5.	Upload tài liệu .....	50
II.6.	Quản lý thông tin.....	51
II.7.	Tham gia diễn đàn.....	52
II.8.	Gửi liên hệ.....	53
II.9.	Xem thông tin .....	53
II.10.	Download tài liệu .....	54
II.11.	Quản lý tài khoản .....	54
II.12.	Quản lý quyền.....	55
II.13.	Quản lý giao diện .....	56
<b>CHƯƠNG IV: ỨNG DỤNG DRUPAL – XÂY DỰNG WEBSITE KHOA CÔNG</b>		
<b>NGHỆ THÔNG TIN .....</b>		<b>67</b>
I.	Cài đặt.....	67

<i>I.1</i>	<i>Yêu cầu phần mềm</i>	67
<i>I.2</i>	<i>Tiến hành cài đặt</i>	67
II.	<i>Các sơ đồ</i>	68
<i>II.1</i>	<i>Sơ đồ Website</i>	68
<i>II.2</i>	<i>Sơ đồ chức năng tạo nội dung trang</i>	72
III.	<i>Xây dựng</i>	72
<i>III.1</i>	<i>Tạo giao diện theo phong cách riêng</i>	72
III.1.1	<i>Trang chủ</i>	73
III.1.2	<i>Sinh viên tương lai</i>	74
III.1.3	<i>Sinh viên hiện tại</i>	75
III.1.4	<i>Giảng viên</i>	76
III.1.5	<i>Cựu sinh viên</i>	76
<i>III.2</i>	<i>Module thêm vào</i>	77
<i>III.3</i>	<i>Danh sách người dùng</i>	78
IV.	<i>Tổng kết</i>	78

## Danh sách hình

	<b>Trang</b>
Hình 2.1. Chồng công nghệ của Drupal. ....	10
Hình 2.2. Tổng quan về nhân Drupal (Không bao gồm tất cả các chức năng). ....	11
Hình 2.3. Module có thể được thêm vào để tăng cường các chức năng ....	12
Hình 2.4. Kiểu node được bắt nguồn từ một node cơ bản và có thể thêm các trường. ....	14
Hình 2.5. Mọi quan hệ giữa các bảng trong cơ sở dữ liệu của Drupal. ....	14
Hình 2.6. Nội dung mã nguồn của Drupal. Nguồn ohloh ....	20
Hình 2.7. Nội dung mã nguồn của Joomla. Nguồn ohloh ....	20
Hình 2.8. Drupal xác định tập tin cơ sở dữ liệu. ....	21
Hình 2.9. Kết quả mà drupal hiển thị. ....	28
Hình 2.10. Kết quả cấu hình kiểu node. ....	31
Hình 2.11. Kết quả tạo form cho node. ....	32
Hình 2.13. Một từ vựng phân cấp có mối quan hệ cha con giữa các term. ....	36
Hình 2.15. Trong một từ vựng đa phân cấp thì các term có thể có nhiều hơn một cha. .....	36
Hình 2.16. Các bảng của taxonomy của Drupal. Các khóa chính) được in đậm. ....	38
Hình 3.1. Usecase Diagram của User. ....	41
Hình 3.2. Usecase Diagram của Giảng viên. ....	43
Hình 3.3. Usecase Diagram của Thư ký. ....	44
Hình 3.4. Usecase Diagram của Người quản trị. ....	46
Hình 3.5. Usecase Diagram của Người quản lý câu lạc bộ. ....	47
Hình 3.6. Usecase Diagram của Người quản lý diễn đàn. ....	47
Hình 3.7. Mô hình ERD của khoa. ....	58
Hình 3.8. Các bảng cơ sở dữ liệu cơ bản của Drupal. ....	64
Hình 3.9. Bảng node và node_revisions trong cơ sở dữ liệu của Drupal. ....	65
Hình 3.10. Các bảng trong cơ sở dữ liệu Drupal. ....	66
Hình 4.1. Sơ đồ mô tả chức năng tổng quát ....	68
Hình 4.2. Sơ đồ của mô tả chức năng của Sinh Viên Tương Lai. ....	69
Hình 4.4. Sơ đồ mô tả chức năng của Giảng Viên. ....	69
Hình 4.5. Sơ đồ mô tả chức năng của Cựu Sinh Viên. ....	69
Hình 4.3. Sơ đồ mô tả chức năng của Sinh Viên Hiện Tại. ....	70
Hình 4.6. Sơ đồ mô tả chức năng Menu Chính. ....	71
Hình 4.7. Sơ đồ mô tả chức năng Menu Phụ. ....	71
Hình 4.8. Sơ đồ mô tả chức năng Liên Kết Ngoài. ....	71
Hình 4.9. Sơ đồ chức năng tạo chức năng trang. ....	72
Hình 4.10. Sơ đồ của người quản trị. ....	72
Hình 4.11. Giao diện trang chủ. ....	73
Hình 4.12. Giao diện trang Sinh Viên Tương Lai. ....	74
Hình 4.13. Giao diện trang Sinh Viên Hiện Tại. ....	75
Hình 4.14. Giao diện trang Giảng Viên. ....	76
Hình 4.15. Giao diện trang Cựu Sinh Viên. ....	77

## Danh sách bảng

### Trang

Bảng 2.1. So sánh Drupal và Joomla.....	19
Bảng 2.2. Các placeholder và ý nghĩa.....	22
Bảng 2.3. Các term đơn giản trong một từ vựng.....	35
Bảng 2.4. Các term phân cấp trong một từ vựng.....	35
<b>Bảng 2.5. Một từ vựng phân cấp.....</b>	<b>37</b>
Bảng 3.1 Mô tả Usecase của User.....	42
Bảng 3.2 Mô tả Usecase của Giảng viên.....	43
Bảng 3.3. Mô tả Usecase của Thư ký.....	45
Bảng 3.4. Mô tả Usecase của Người quản trị.....	46
Bảng 3.5. Mô tả Usecase của Người quản lý câu lạc bộ.....	47
Bảng 3.6. Mô tả Usecase của Người quản lý diễn đàn.....	48
Bảng 3.7. Đặc tả Usecase Đăng nhập.....	48
Bảng 3.7. Đặc tả Usecase Quản lý câu lạc bộ.....	49
Bảng 3.8. Đặc tả Usecase Quản lý diễn đàn.....	50
Bảng 3.9. Đặc tả Usecase Đổi mật khẩu.....	50
Bảng 3.10. Đặc tả Usecase Upload tài liệu.....	51
Bảng 3.11. Đặc tả Usecase Quản lý thông tin.....	52
Bảng 3.12. Đặc tả Usecase Tham gia diễn đàn.....	53
Bảng 3.13 Đặc tả Usecase Gửi liên hệ.....	53
Bảng 3.14. Đặc tả Usecase Xem thông tin.....	53
Bảng 3.15 Đặc tả Usecase Download tài liệu.....	54
Bảng 3.16. Đặc tả Usecase Quản lý tài khoản.....	55
Bảng 3.17. Đặc tả Usecase Quản lý quyền.....	56
Bảng 3.18. Đặc tả Usecase Quản lý giao diện.....	57
Bảng 3.19. Bảng de_tai.....	58
Bảng 3.20. Bảng giang_vien.....	59
Bảng 3.21. Bảng anh_hoat_dong.....	59
Bảng 3.22. Bảng lien_he.....	59
Bảng 3.23. Bảng bieu_mau.....	59
Bảng 3.24. Bảng thong_bao.....	60
Bảng 3.25. Bảng thoi_khoa_bieu.....	60
Bảng 3.26. Bảng he.....	60
Bảng 3.27. Bảng bang_diem.....	61
Bảng 3.28. Bảng khoa_hoc.....	61
Bảng 3.29. Bảng cuu_sinh_vien.....	61
Bảng 3.30. Bảng bang_vang.....	61
Bảng 3.31. Bảng nguoi_dung.....	62
Bảng 3.32 Bảng cuoc_thi.....	62
Bảng 3.33 Bảng file_dinh_kem.....	62
Bảng 3.34. Bảng doi_tac.....	63
Bảng 3.35. Bảng loai_de_tai.....	63

Bảng 4.1 Danh sách địa chỉ người dùng.....	78
--	----

# Mở đầu

Sự phát triển mạnh mẽ của Internet ngày một kéo theo sự phát triển của các ngành, các lĩnh vực, các thành phần kinh tế, xã hội. Với sự ra đời của các trang web thương mại điện tử đã làm thay đổi diện mạo của các tập đoàn kinh tế lớn, nhỏ.

Cùng với việc phát triển đó là hàng triệu trang web của các tổ chức, cá nhân ra đời, thông tin ngày càng phong phú và đa dạng. Nhưng để xây dựng và quản lý những trang web thì tốn rất nhiều thời gian, kinh tế.

Phần mềm nguồn mở và các hệ quản trị nội dung ra đời giúp người dùng có thể xây dựng những trang web một cách nhanh chóng mà không cần phải biết các ngôn ngữ lập trình. Đồng thời về mặt kinh phí hoàn toàn là miễn phí.

Ngày nay có rất nhiều hệ quản trị nội dung như là: DotNetNuke (ASP.Net+VB/C#), Drupal (PHP), Joomla/Mambo (PHP), Kentico CMS (ASP.Net + VB/C#), PHP-Nuke (PHP), Rainbow (ASP.NET +C#), Typo3 (PHP), Xoops (PHP). Trong đó Drupal được xếp hạng đầu danh sách (NXB Packt bầu chọn năm 2008).

Vì vậy, nhóm chúng em đã quyết định chọn và thực hiện đề tài “Tìm hiểu và ứng dụng Drupal

Qua khoá luận này, nhóm chúng em hy vọng sẽ tìm hiểu được các phần sau :

- Khái niệm phần mềm nguồn mở.
- Khái niệm hệ quản trị nội dung.
- Tìm hiểu Drupal.
- Phân tích hệ thống Website khoa công nghệ thông tin của trường đại học Kỹ Thuật Công Nghệ.
- Dùng Drupal để xây dựng Website khoa công nghệ thông tin.

Mặc dù đã cố gắng hoàn thành khoá luận trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không khỏi những thiếu sót nên nhóm chúng em rất mong nhận được sự cảm thông góp ý tận tình từ quý thầy cô và các bạn.

# CHƯƠNG I: GIỚI THIỆU PHẦN MỀM NGUỒN MỞ VÀ CMS

## I. Khái niệm phần mềm nguồn mở

### I.1. Định nghĩa (David Wheeler)

Chương trình phần mềm nguồn mở (PMNM) là những chương trình mà quy trình cấp phép sẽ cho người dùng quyền tự do chạy chương trình theo bất kỳ mục đích nào, quyền nghiên cứu và sửa đổi chương trình, quyền sao chép và tái phát hành phần mềm gốc hoặc phần mềm đã sửa đổi (mà không phải trả tiền bản quyền cho những người lập trình trước).

### I.2. Các học thuyết về PMNM

Hai học thuyết PMNM chủ đạo:

- ❖ Tổ chức phần mềm tự do FSF (Free Software Foundation).
- ❖ Chương trình sáng kiến nguồn mở OSI (Open Source Initiative).

#### I.2.1. Học thuyết FSF

Phần mềm miễn phí nhằm mục đích bảo vệ 4 quyền tự do của người dùng:

- Quyền tự do chạy một chương trình với bất kỳ mục đích nào.
- Quyền tự do nghiên cứu cách thức vận hành của một chương trình và thích ứng nó cho phù hợp với nhu cầu của mình.
- Quyền tự do phân phối các phiên bản của phần mềm để giúp đỡ những người xung quanh.
- Quyền tự do thêm mới các chức năng cho một chương trình và công bố những tính năng mới đó đến với công chúng để toàn cộng đồng được hưởng lợi.

#### I.2.2. Học thuyết OSI

Chú trọng giá trị kỹ thuật của việc tạo ra những phần mềm mạnh, có độ tin cậy cao và phù hợp với giới kinh doanh, đặc biệt là lợi ích thực tiễn của phương pháp xây dựng và quảng bá PMNM.



### *1.3. Ưu điểm của phương pháp xây dựng PMNM*

- Giảm sự trùng lặp nguồn lực.
- Tiếp thu kế thừa.
- Quản lý chất lượng tốt hơn.
- Giảm chi phí duy trì.

### *1.4. Lịch sử của PMNM*

#### ❖ Các cột mốc đáng nhớ:

- 1984: Richard Stallman sáng lập dự án GNU (GNU Not Unix).
- 1991: Linus Torvalds viết thành công lõi Linux.
- 1997: GNU/Linux chiếm 25% thị trường máy chủ.
- 1998: Netscape công bố mã nguồn Navigator.

Thuật ngữ “Nguồn mở” ra đời.

Thành lập sáng kiến nguồn mở OSI.

### *1.5. Lợi ích của PMNM*

- Tính kinh tế.
- Tính an toàn.
- Tính ổn định.
- Sử dụng chuẩn mở.
- Giảm phụ thuộc vào nhập khẩu.
- Phát triển năng lực ngành CNPM địa phương.
- Giảm tình trạng vi phạm bản quyền.

### *1.6. Hạn chế của PMNM*

- Thiếu các ứng dụng kinh doanh đặc thù.
- Tính tương hỗ với các phần mềm đóng kém.

- Giao diện người dùng chưa tốt.

### *I.7. Những dự án PMNM thành công*

- BIND (máy chủ tên miền DNS).
- Apache (máy chủ Web).
- Sendmail (máy chủ thư điện tử).
- Open Office (bộ ứng dụng văn phòng).

### *I.8. Các loại giấy phép PMNM*

Các loại giấy phép thông dụng:

#### I.8.1. Giấy phép đại chúng GNU General Public License

- Người phổ biến một chương trình đã được cấp phép đại chúng phải đồng thời phổ biến luôn cả mã nguồn cho người nhận.
- Nếu người phổ biến chương trình đã thực hiện một sửa đổi gì đó cho phần mềm thì những sửa đổi đó cũng phải được cấp phép theo chế độ giấy phép đại chúng.
- Người phổ biến chương trình không áp dụng với người nhận bất cứ hạn chế nào không thuộc phạm vi giấy phép đại chúng.
- Người nhận phần mềm đã cấp phép đại chúng sẽ được trao y nguyên mọi quyền như người phổ biến gốc, tức là quyền sao chép, chỉnh sửa và phổ biến phần mềm.

#### I.8.2. Giấy phép BSD (Berkeley System Distribution)

- Ghi nhận công lao của tác giả đầu tiên làm ra phần mềm bằng cách đưa vào tập tin mã nguồn các thông tin bản quyền gốc.
- Người phát hành ban đầu sẽ không chịu trách nhiệm trước pháp luật về bất cứ thiệt hại nào phát sinh do sử dụng những phần mềm nguồn mở đã được chỉnh sửa.

## **II. Giới thiệu về CMS**

### *II.1. Khái niệm*

CMS viết tắt của *Content Management System* nghĩa là hệ thống quản trị nội dung. Là phần mềm để tổ chức và tạo môi trường cộng tác thuận lợi nhằm mục đích xây dựng một hệ thống tài liệu và các nội dung khác nhau một cách thống nhất. CMS đã được áp dụng vào các chương trình quản lý nội dung website.

## II.2. Các yêu cầu

Phải đảm bảo hệ thống module chuyên biệt và cho phép nhà phát triển có thể viết thêm các module mà họ muốn và tích hợp vào hệ thống CMS đó theo chuẩn quy định bởi những người tạo ra CMS đó. Ví dụ, cụ thể như Rainbow CMS cho phép người dùng tạo module tích hợp vào hệ thống của họ bằng các custom control (điều khiển chỉnh sửa) dạng .ascx, hay như Joomla thì lại hỗ trợ PHP và MySQL.

Phải có kênh quản trị trực quan và dễ dùng, đảm bảo những người có thẩm quyền (admin) có thể thêm, bớt, chỉnh sửa và tùy biến các module theo ý của họ. Và không thể thiếu chế độ soạn thảo trực quan (*WYSIWYG*).

Giao diện có thể chỉnh sửa tùy ý, điều này có thể xem như một giới hạn của CMS. Bởi vì bạn sẽ cảm thấy thật khó chịu nếu như không thể "tùy chỉnh" sáng tạo giao diện trang web của mình khi bị bó buộc vào cấu trúc cứng nhắc đã định sẵn của CMS. Điều này dẫn đến lý do tại sao có những trang có cấu trúc khá giống nhau vì xây dựng trên cùng một CMS, khác chẳng chỉ là màu sắc. Và cũng từ đó người ta mới hiểu ra rằng chỉ áp dụng CMS vào những trang có nội dung chuyên biệt và không quan trọng giao diện như những trang khác. Tuy nhiên do sự phát triển của CMS điều này đã được khắc phục.

## II.3. Danh sách các CMS đứng đầu năm 2008

Cuộc bầu chọn CMS năm 2008 của NXB Packt như sau:

- ❖ CMS nguồn mở tốt nhất:
  - Drupal.
  - Joomla!
  - DotNetNuke.

- ❖ CMS nguồn mở khác (không dùng PHP/MySQL) tốt nhất:
  - Plone.
  - dotCMS.
  - DotNetNuke.
- ❖ CMS nguồn mở PHP tốt nhất:
  - Drupal.
  - Joomla! và CMS Made Simple đồng hạng.
- ❖ CMS nguồn mở hứa hẹn nhất:
  - SilverStripe.
  - CMS Made Simple.
  - ImpressCMS và MiaCMS đồng hạng.

## CHƯƠNG II: GIỚI THIỆU DRUPAL

### I Tổng quan về Drupal

#### *I.1 Khái niệm*

Drupal là một hệ thống quản trị nội dung miễn phí và mã nguồn mở, một khung sườn phát triển phần mềm hướng module. Cũng giống như các hệ thống quản trị nội dung hiện đại khác, Drupal cho phép người quản trị hệ thống tạo và tổ chức dữ liệu, tùy chỉnh cách trình bày, tự động hóa các tác vụ điều hành và quản lý hệ thống. Drupal có cấu trúc lập trình rất tinh vi. Dựa trên đó, hầu hết các tác vụ phức tạp có thể được giải quyết với rất ít đoạn mã được viết, thậm chí không cần. Đôi khi, Drupal cũng được gọi là “khung sườn phát triển ứng dụng web” vì kiến trúc thông minh và uyển chuyển của nó.

#### *I.2 Lịch sử*

Drupal được lập trình bởi Dries Buytaert một lập trình viên nguồn mở tạo ra theo dạng một hệ thống bảng tin vào năm 2000. Đến năm 2001, Drupal trở thành một công trình nguồn mở, và hiện nay được một cộng đồng rộng lớn sử dụng và phát triển.

Drupal trong tiếng Anh được chuyển ngữ từ từ liệu "druppel" của tiếng Hà Lan, nghĩa là "giọt nước".

#### *I.3 Bản quyền*

Được cấp giấy phép GNU General Public Licence (GNU GPL hay GPL).

#### *I.4 Ưu điểm và nhược điểm*

##### I.4.1 Ưu điểm

- ❖ Miễn phí theo quy định bản quyền GPL.
- ❖ Drupal có dung lượng nhẹ ( một 1.5MB cài đặt ) và dễ cài đặt.
- ❖ Drupal là một CMS mạnh.

Quản lý nội dung:

Ngoài các kiểu nội dung cơ bản như: Page, story, book, forum, blog, ta mở rộng thêm nhờ hệ thống taxonomy hay bằng cách là thêm vào các module.

Quản lý người dùng dựa trên vai trò:

Hệ thống phân quyền, cơ bản dựa trên bộ ba: module + thao tác + vai trò. Do đó, nó dễ dàng cho việc mở rộng thêm.

Quản lý cấu hình hệ thống:

Cho phép điều chỉnh các tùy chọn cấu hình rất uyển chuyển.

❖ Drupal có Framework tốt

- Nhờ Database Abstraction Layer: Thao tác với cơ sở dữ liệu dễ dàng và an toàn, có thể chạy trên cùng các hệ quản trị cơ sở dữ liệu MySQL và PostgreSQL.
- Hỗ trợ trên nhiều Web Server như: Apache, IIS, lighttpd,...
- Hỗ trợ đa hệ điều hành như: Linux, BSD, Solaris, Windows, Mac OS X.

❖ Module cộng thêm

- Giúp nhanh chóng phát triển các tính năng nhờ vào kiến trúc hook mạnh mẽ và dễ dàng bảo trì.
- Tích hợp Form API: Hỗ trợ tạo và xử lý giao diện nhanh chóng.
- Hỗ trợ PHPTemplate cho phép tạo ra giao diện phong phú, linh động.

❖ Đa ứng dụng

Với Drupal và các module cộng thêm của nó, chúng ta có thể xây dựng rất nhiều ứng dụng mà không cần phải bỏ ra nhiều thời gian và công sức để phát triển.

Các ứng dụng có thể xây dựng:

Hướng cá nhân, tổ chức: Blog, Profolio sites, Calendar,...

Hướng cộng đồng: Multi blog site, Organic Groups, Wiki sites,...

Hướng doanh nghiệp: Webservice, Brochure, Project manament,...

## I.4.2 Nhược điểm

Triết lý “không tương thích ngược” của Drupal làm cho module phải được viết riêng cho từng phiên bản 4.x, 5.x và 6.x tại cùng một thời điểm, Drupal chỉ hỗ trợ 2 phiên bản mới nhất. Điều này làm cho người sử dụng e ngại phải nâng cấp liên tục khi dùng Drupal.

### *1.5 Các tính năng chung*

- ❖ Collaborative Book: Cho phép thiết lập một "sách" và sau đó cho phép các cá nhân đóng góp nội dung.
- ❖ Friendly URLs: Drupal sử dụng mod\_rewrite của Apache để kích hoạt các URL tùy chỉnh được cả người sử dụng và công cụ tìm kiếm một cách thân thiện.
- ❖ Modules: Cho phép mở rộng các tính năng vào nhân Drupal.
- ❖ Online help (Trợ giúp trực tuyến): Trợ giúp này được Drupal phát triển khá tốt.
- ❖ Role based permission system (Hệ thống phân quyền): Phân quyền cho vai trò và sau đó nhóm những người sử dụng tương đồng vào một nhóm vai trò.
- ❖ Searching (Tìm kiếm): Tất cả các nội dung trong Drupal được lập chỉ mục đầy đủ và có thể tìm kiếm ở tất cả các lần nếu ta biết tận dụng lợi thế của việc xây dựng module tìm kiếm.
- ❖ User management (Quản lý người dùng)
  - User authentication (Xác thực người dùng): Người dùng có thể đăng ký tài khoản, xác nhận cục bộ hoặc sử dụng một nguồn xác nhận bên ngoài như Jabber, Blogger, LiveJournal.
- ❖ Content management (Quản lý nội dung)
  - Templating: Hệ thống giao diện của Dupal phân nội dung từ sự trình bày, cho phép bạn kiểm soát xem và cảm nhận trang web. Mẫu được tạo ra từ chuẩn HTML và mã nguồn PHP.

- Version control (Kiểm soát phiên bản): Theo dõi các chi tiết của nội dung cập nhật bao gồm những người thay đổi nó, những gì đã được thay đổi, thời gian các thay đổi nội dung và nhiều hơn nữa.

#### ❖ Blogging (Tạo Blog)

- Blogger API support (Hỗ trợ Blogger API): Blogger API cho phép các trang web Drupal có thể cập nhật bởi nhiều công cụ khác nhau.
- Content syndication (Cung cấp nội dung): Drupal đưa ra và lưu trữ nội dung của trang web trong định dạng RDF (Resource Description Framework – Tiêu chuẩn định dạng dữ liệu) hoặc RSS (Really Simple Syndication – Tiêu chuẩn định dạng tập tin).
- News aggregator (Tập hợp tin tức): Drupal có một News Aggregator xây dựng sẵn mạnh mẽ cho đọc và viết blog tin tức từ các trang web khác. Tập hợp tin tức tạo đệm (caches) cho các bài báo tới cơ sở dữ liệu MySQL và thời gian đệm bộ nhớ được cấu hình thông qua người dùng.

#### ❖ Platform (Nền tảng)

- Hỗ trợ Apache hoặc IIS, Unix / Linux / BSD / Solaris / Windows / Mac OS X.
- Database independence (Cơ sở dữ liệu độc lập): Drupal được xây dựng trên cơ sở dữ liệu của một lớp trừu tượng cho phép bạn sử dụng Drupal với MySQL và PostgreSQL.
- Multi-language (Đa ngôn ngữ): Drupal được thiết kế để đáp ứng các yêu cầu cho các quốc gia khác nhau và cung cấp một Framework (khung sườn) đầy đủ để tạo các ứng dụng đa ngôn ngữ như blog, hệ thống quản lý nội dung hoặc ứng dụng cộng đồng.

#### ❖ Administration and analysis (Quản lý và phân tích)

- Logging and Reporting (Đăng nhập và bản ghi): Kiểm soát đăng nhập và bản ghi giám sát, bắt giữ những sự kiện trong hệ thống và ghi lại trong bản ghi.



- Web based administration (Quản trị trang web): Drupal có thể được quản lý bằng cách sử dụng trình duyệt web, để có thể truy cập nó từ khắp nơi trên thế giới và không yêu cầu phải cài đặt thêm bất cứ phần mềm nào.

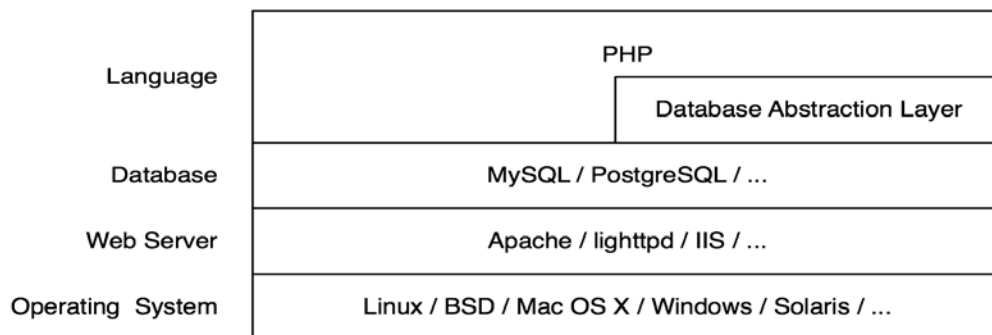
❖ Community features (Các tính năng cộng đồng)

- Discussion forums (Diễn đàn thảo luận): Cung cấp đầy đủ các tính năng để có thể tham gia trao đổi, thảo luận, gửi bài, và đăng ký thành viên.

## II Cách làm việc của Drupal

### II.1 Technology Stack (Chồng công nghệ)

Mục đích thiết kế của Drupal có thể chạy tốt trên các máy chủ web và có thể đảm bảo về hiệu năng và độ an toàn bằng việc sử dụng các công nghệ phổ biến:

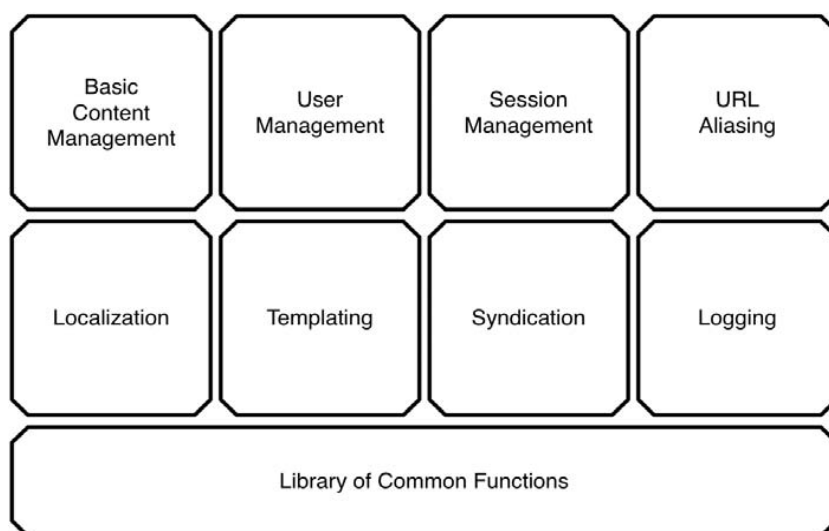


Hình 2.1. Chồng công nghệ của Drupal.

Theo Hình 2.1, lớp thấp nhất là hệ điều hành: Drupal có thể chạy trên đa nền như Linux, BSD, Mac OS X, Windows, Solaris. Kế đến là lớp Web Server: Drupal có thể chạy trên nhiều máy chủ web khác nhau được sử dụng phổ biến là Apache và còn hỗ trợ cả IIS. Lớp Database (cơ sở dữ liệu): Drupal hỗ trợ MySQL và PostgreSQL (MySQL, PostgreSQL là hệ quản trị cơ sở dữ liệu). Lớp trên cùng là Language (lớp ngôn ngữ): Drupal sử dụng ngôn ngữ PHP có hỗ trợ lớp cơ sở dữ liệu trừu tượng (Database Abstraction Layer). Nhờ lớp này, làm cho PHP có thể kết nối và truy vấn đến cơ sở dữ liệu dễ dàng và an toàn hơn.

### II.2 Core (Nhân của Drupal)

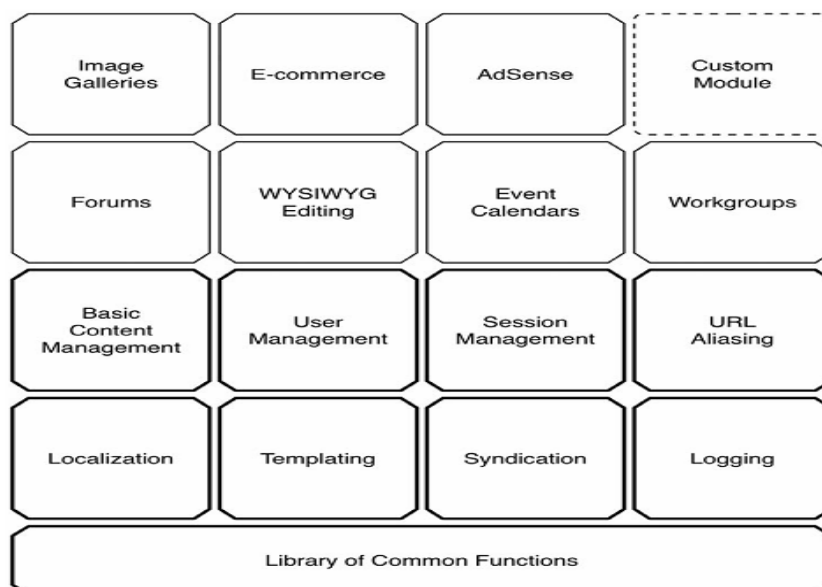
Một khung sườn nhẹ được tích hợp thành nhân của Drupal: Cung cấp các chức năng cơ bản như quản lý người dùng, quản lý nội dung có hỗ trợ phân loại kiểu nội dung (taxonomy) và các template (mẫu giao diện). Hệ thống sẽ thực hiện các yêu cầu của người dùng và một thư viện dùng chung sẽ được sử dụng cùng với các thành phần Drupal, như hình bên dưới:



*Hình 2.2. Tổng quan về nhân Drupal (Không bao gồm tất cả các chức năng).*

### *II.3 Modules*

Module là phần mở rộng có thể cài đặt vào Drupal để thêm các ứng dụng hay các chức năng cho Drupal, có thể bật hoặc tắt các module có sẵn trong Drupal. Ngoài ra, ta có thể tìm các module mở rộng để thích hợp vào Drupal tại <http://api.drupal.org/> hoặc tự viết các module mới cho trang web.



Hình 2.3. Module có thể được thêm vào để tăng cường các chức năng.

#### II.4 Hooks

Hook được xem như là các sự kiện (event) bên trong bộ nhân Drupal hoặc được gọi là callbacks. Hệ thống hook đa dạng giúp người lập trình can thiệp vào mọi hoạt động mà không phải sửa bất kì dòng lệnh nào của nhân hay các phần khác.

#### II.5 Themes (Giao diện)

Khi tạo ra một trang web gửi tới trình duyệt, nó có 2 phần chính là một nhóm các dữ liệu tương ứng (nội dung) và đánh dấu dữ liệu trên web. Trong Drupal, lớp chủ đề chịu trách nhiệm tạo ra HTML để cung cấp cho trình duyệt web.

Đặc biệt Drupal khuyến khích tách biệt giữa nội dung và đánh dấu nội dung.

Drupal cho phép người quản trị giao diện có thể hiệu chỉnh và thay thế giao diện cho trang web, một cách đơn giản nhất là dùng CSS (Cascading Style Sheet) để thay đổi các mẫu giao diện có sẵn.

#### II.6 Nodes

Các kiểu nội dung trong Drupal đều kế thừa từ kiểu cơ bản là node. Cũng có thể nói hạt nhân của Drupal là node với cấu trúc đơn giản. Để phát triển các kiểu nội dung phức tạp, mỗi node sẽ được liên kết với một nội dung có kiểu khác nhau. Node được

quản lý nhờ hệ thống phân loại kiểu nội dung (taxonomy), đây cũng là đặc trưng của Drupal. Node chứa tất cả các mục đề, nội dung, ngày tháng, ...

Cách tương tác với node là mở rộng chúng. Có thể thêm vào các node các chức năng mở rộng như: đánh giá bài viết, bình luận, đính kèm file, ...

Mỗi node có một địa chỉ URL xác định, dựa vào *node\_id*, cấu trúc của nó như sau:

<địa chỉ trang web>?q=node/node\_id

Nếu *node\_id* không được xác định thì hệ thống sẽ chuyển về trang chủ.

## II.7 Blocks

Block là thông tin mà ta có thể kích hoạt hoặc vô hiệu hóa trong một vị trí xác định trên mẫu sử dụng của trang web. Các block thường được đặt ở các vị trí: left-sidebar, right-sidebar, header hoặc footer.

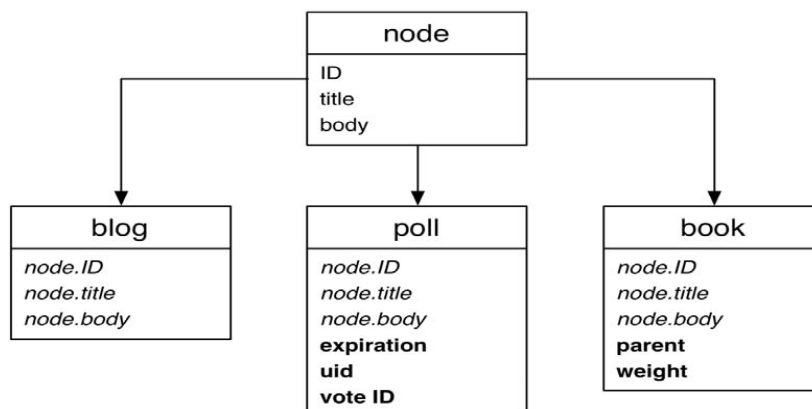
Thông thường, các block được sử dụng để mô tả thông tin được điều chỉnh tới người dùng hiện hành. Ví dụ: Block điều hướng (navigation) chứa các liên kết tới chức năng quản trị mà người dùng có thể truy cập. Sự sắp xếp và vùng hiển thị các block được quản lý thông qua giao diện quản trị trang web.

## III Tổ chức lưu trữ của Drupal

Như Drupal, kiểu node được xem là một đối tượng và một node riêng lẻ thì được coi là một thể hiện của đối tượng. Và node cũng được xem là đơn vị dữ liệu. Do đó, mọi thông tin đều được lưu trữ trong node (ngoại trừ users, blocks, comments).

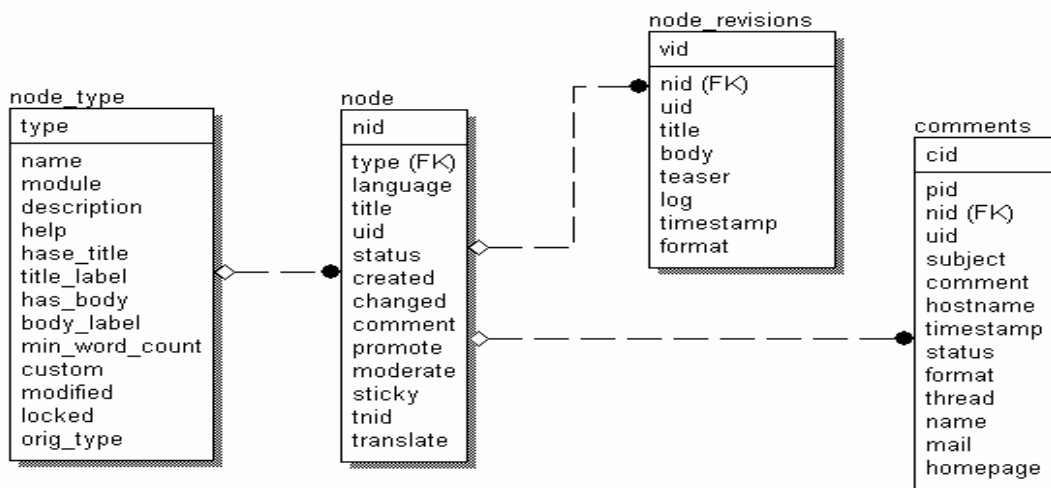
Có nhiều kiểu node khác nhau. Một số kiểu node chung như blog, poll, page, story, book. Và kiểu nội dung được sử dụng như là kiểu node, mặc dù một kiểu node thì thật sự là một kiểu trừu tượng.

Tất cả kiểu nội dung đều là node, chúng dựa trên cùng kiến trúc dữ liệu. Do vậy, giúp dễ dàng thao tác như tìm kiếm, tạo mới, điều chỉnh, xóa và quản lý nội dung trên node.



Hình 2.4. Kiểu node được bắt nguồn từ một node cơ bản và có thể thêm các trường.

Các kiểu node được mở rộng trên một node cơ bản, thông thường được thêm vào các thuộc tính dữ liệu riêng của nó. Chẳng hạn, một node kiểu poll chứa các sự lựa chọn (voting) và ngày hết hạn (expiration date) của poll. Một node kiểu book chứa thuộc tính parent, nhờ vào node ID cho mỗi node nó xác định đâu là nội dung của node phù hợp. Một kiểu node khác như blog, thì không thêm thuộc tính nào khác. Thay vào đó, chúng lại tạo ra những khung nhìn (view) khác nhau cho dữ liệu bằng cách tạo ra những blog cho mỗi người dùng và RSS feeds (nguồn cấp dữ liệu RSS) cho mỗi blog.



Hình 2.5 Mối quan hệ giữa các bảng trong cơ sở dữ liệu của Drupal.

Tất cả các node có các thuộc tính bên dưới được chứa trong bảng cơ sở dữ liệu node và node\_revisions:

- nid: Là unique ID (duy nhất) cho một node.

- vid: Là thuộc tính khóa duy nhất cho một node, giúp xác định nội dung của node\_revisions. vid là unique cho cả node và node\_revisions.
- type: Mỗi node có một kiểu node. Ví dụ: blog, story, article, image, ...
- title: Tiêu đề của một node cho phép chuỗi dài 128 ký tự.
- uid: Thuộc tính khóa cho tác giả (author). Mặc định, một tác giả có nhiều node.
- status: Giá trị bằng 0 là chưa được công bố, có nghĩa là nội dung được ẩn đi vì chưa được sự cho phép của người quản trị. Giá trị bằng 1 là node đã được công bố và nội dung được hiển thị cho người dùng với sự cho phép truy cập nội dung.
- created: Cho biết thời gian khi một node được tạo.
- changed: Ghi nhận thời gian khi có sự điều chỉnh cuối cùng.
- comment: Có kiểu integer, mô tả trạng thái các comment của node, có 3 giá trị:
  - 0: Các comment bị vô hiệu hóa cho node hiện hành. Đây là giá trị mặc định khi module comment được kích hoạt.
  - 1: Không cho phép thêm comment vào node hiện hành.
  - 2: Các comment được hiển thị và có thể tạo ra comment mới.
- promote: Một trường kiểu integer, xác định cho phép mô tả node trên trang đầu hay không, với 2 giá trị:
  - 1: Cho phép node hiển thị trên trang đầu của trang web. Trang sẽ chứa tất cả các node mà có trường promote bằng 1, [http:// example.com/?q=node](http://example.com/?q=node) là trang đầu mặc định.
  - 0: Node sẽ không được hiển thị ở trang <http://example.com/?q=node>.
- moderate: Có kiểu integer, nếu giá trị bằng 0 là moderate bị vô hiệu hóa và giá trị bằng 1 là cho phép moderate.
- sticky: Khi Drupal mô tả danh sách các node trên một trang hành vi mặc định được liệt kê đầu tiên với những node được đánh dấu là sticky và danh sách còn lại là các

node “unsticky” sẽ nằm trong danh sách theo ngày tạo ra. Giá trị bằng 1 là sticky và bằng 0 là unsticky. Có thể có nhiều node sticky với một danh sách giống nhau.

## **IV Tính an toàn của Drupal**

### *IV.1 An toàn đối với người dùng*

- Người quản trị có thể quản lý người dùng qua cài đặt vai trò, quản lý tài khoản.
- Vai trò cho phép điều chỉnh, bảo mật và quản trị hệ thống. Một vai trò có thể xác định một nhóm người dùng có các đặc quyền cụ thể được định nghĩa ở trang điều khiển truy cập.
  - o Mặc định hệ thống có hai vai trò người dùng:
    - Người dùng nặc danh: Vai trò này được sử dụng cho những người dùng không có tài khoản người dùng hoặc chưa được xác thực.
    - Người dùng đã xác thực: Vai trò này được tự động gán cho tất cả những người dùng đã đăng nhập.
  - o Với mỗi vai trò có thể điều khiển truy cập tương ứng: Thiết lập quyền hạn cho phép điều khiển những việc mà người dùng có thể thực hiện trong hệ thống. Mỗi vai trò người dùng có một tập hợp các quyền hạn riêng.

### *IV.2 An toàn đối với mã nguồn.*

Mặc dù mã nguồn của Drupal hoàn toàn có thể xem và chỉnh sửa một cách dễ dàng nhưng nó cũng có những cách riêng để ngăn chặn những mối nguy hại chính:

- o Viết đoạn mã tấn công bằng cách kiểm tra dữ liệu xuất hợp lệ.
- o Đưa câu truy vấn SQL để tấn công bằng cách sử dụng lớp trừu tượng (database abstraction layer).
- o Giới hạn truy cập node bị loại bỏ bằng cách sử dụng `db_rewrite_sql`.
- o Cần sử dụng lớp Database Abstraction Layer hợp lý. Không bao giờ, viết dữ liệu người dùng vào trong đoạn mã SQL. Dưới đây là một đoạn SQL không an toàn:

```
db_query('SELECT foo FROM {table} t WHERE t.name = ' . $_GET['user']');
```

thay vào đó ta phải sử dụng:

```
db_query("SELECT foo FROM {table} t WHERE t.name = '%s' ", $_GET['user']);
```

### IV.3 Tính an toàn đối với dữ liệu

Khi xử lý và xuất văn bản trong HTML, cần phải quan tâm đến việc sàn lọc dữ liệu. Mặt khác, có thể gây lỗi khi người dùng sử dụng các dấu đặc biệt như <, &, hoặc tệ hơn người dùng có thể mở và khai thác XSS (Cross-site scripting: Một dạng mà có thể làm ảnh hưởng tới sự an toàn).

Khi xử lý dữ liệu, nguyên tắc quan trọng là chứa chính xác những gì mà người dùng thao tác nội dung. Trong trường hợp người dùng điều chỉnh một nội dung đã được tải lên, thì form được thao tác trước đó nên giống với form cần điều chỉnh lại.

Khi thao tác trên chuỗi như nối chuỗi phải đảm bảo các chuỗi cùng định dạng.

– An toàn đối với module được thêm vào

Các module sẽ được nhóm Drupal Security Team kiểm tra và thông báo về tình trạng an toàn của các module.

Ví dụ:

- Advisory ID: DRUPAL-SA-2008-075
- Project: Views
- Versions: 6.x
- Date: 2008-December-16
- Security risk: Moderately critical
- Exploitable from: Remote
- Vulnerability: SQL injection

Ở module Views phiên bản 6.x, cho ta biết tình trạng an toàn là ở mức độ trung bình, có thể xâm nhập từ xa thông qua truy vấn SQL đưa vào.



Và thông qua các thông tin về sự an toàn của từng module giúp có thể lựa chọn cài đặt và điều chỉnh một cách phù hợp.

– Ngoài ra Drupal còn cung cấp một số cơ chế mã hóa như MD5, SHA1, ENCRYPT.

Kết luận: Với các cơ chế bảo mật được hỗ trợ Drupal được xem là một CMS có tính bảo mật tốt, giúp người quản trị và người dùng có thể tin cậy.

## V Drupal so với các hệ CMS khác

Vì Drupal và Joomla là hai hệ quản trị nội dung web được đánh giá là dẫn đầu so với các hệ CMS khác. Cho nên chúng ta sẽ thấy được điểm mạnh của Drupal thông qua sự so sánh với Joomla.

### V.1 Sự khác biệt giữa Drupal với Joomla

#### V.1.1 Giống

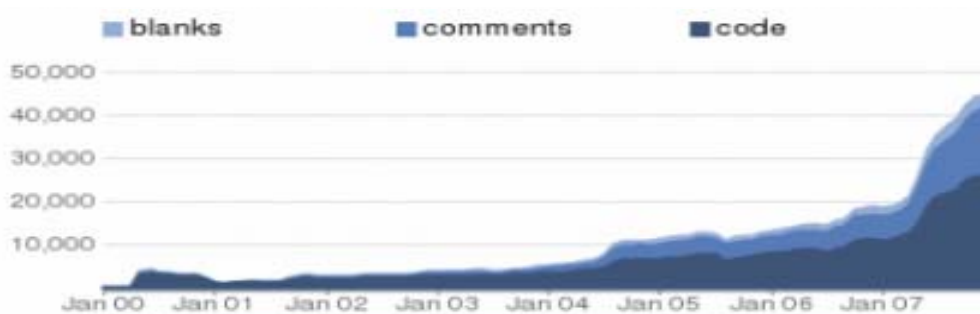
- ❖ Drupal và Joomla là hai hệ quản trị miễn phí và mã nguồn mở.
- ❖ Sử dụng ngôn ngữ PHP có thể tương thích với nhiều hệ điều hành khác nhau.
- ❖ Dễ sử dụng, phát triển và có hiệu năng cao nhờ bộ đệm caching.
- ❖ Có tính bảo mật tốt.

#### V.1.2 Khác

Joomla	Drupal
- Theo triết lý của Joomla, đơn vị dữ liệu cơ bản là content item (đơn vị nội dung) (ở bản 1.5 đổi thành article (bài báo)) chứa trong category (kiểu), bản thân category được chứa trong section (đoạn). Do đó, dữ liệu trong Joomla tổ chức thành 3 cấp.	- Hạt nhân của Drupal là node với cấu trúc đơn giản.
- Không có nền tảng để hỗ trợ các kiểu	- Để phát triển các kiểu nội dung phức

dữ liệu phức tạp hơn. Do đó, cách tổ chức rất logic và không gây rắc rối cho người dùng mới.	tạp, mỗi node sẽ được liên kết với một nội dung có kiểu khác nhau. Node được quản lý nhờ hệ thống taxonomy, đây cũng là đặc trưng của Drupal.
- Do Joomla được thiết kế dành cho người dùng cuối, nên không cung cấp nhiều phương tiện can thiệp vào hoạt động bên trong.	- Drupal được xây dựng để giúp đỡ cho nhà phát triển. Hệ thống hook đa dạng giúp người lập trình can thiệp vào mọi hoạt động mà không phải sửa bất kì dòng lệnh nào của nhân hay các phần khác.
- Kiến trúc cồng kềnh. Hình 2.6.	- Kiến trúc nhỏ gọn. Hình 2.7.
- Không phân quyền cho người dùng.	- Hỗ trợ phân quyền cho cả một nhóm người dùng.
Yêu cầu hệ thống	
- Chỉ làm việc với MySQL.	- Có thể làm việc với MySQL và Postgres.
- Chỉ làm việc với Apache.	- Có thể làm việc với Apache và IIS.
- Bản quyền: GNU/GPL v2.	- Bản quyền: GNU GPL.

Bảng 2.1. So sánh Drupal và Joomla.



Hình 2.6 Nội dung mã nguồn của Drupal. Nguồn ohloh



Hình 2.7 Nội dung mã nguồn của Joomla. Nguồn ohloh

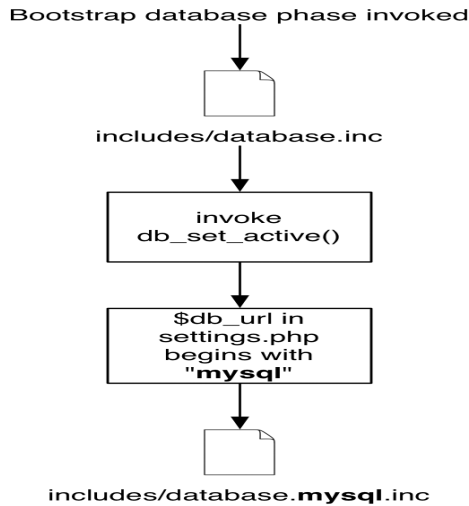
## VI Phát triển Drupal

### VI.1 Làm việc với cơ sở dữ liệu

#### VI.1.1 Lớp Database Abstraction Layer

Lớp này có khối lượng nhẹ và phục vụ hai mục đích chính. Thứ nhất, là giữ liên kết giữa mã nguồn và cơ sở dữ liệu. Thứ hai, cải thiện dữ liệu mà người dùng gửi, để ngăn chặn sự xâm phạm thông qua truy vấn SQL. Lớp này được xây dựng trên nguyên tắc khá đơn giản và tiện dụng là chỉ cần viết các câu lệnh SQL.

Drupal xác định kiểu của cơ sở dữ liệu để kết nối thông qua sự xác định giá trị `$db_url` bên trong file `settings.php`. Ví dụ: Nếu `$db_url` bắt đầu với `mysql` là `includes/database.mysql.inc` và bắt đầu với `pgsql` là `includes/database.pgsql.inc`. Cơ chế này được thể hiện ở hình dưới:



Hình 2.8 Drupal xác định tập tin cơ sở dữ liệu.

Một ví dụ cho thấy sự khác nhau về `db_fetch_object()` giữa lớp trừu tượng MySQL và PostgreSQL :

// Từ `database.mysql.inc`.

```

function db_fetch_object($result) {
    if ($result) {
        return mysql_fetch_object($result);
    } } // Từ database.pgsql.inc.
  
```

```

function db_fetch_object($result) {
    if ($result) {
        return pg_fetch_object($result);
    } }
  
```

### VI.1.2 Kết nối với cơ sở dữ liệu

Drupal thực thi tự động kết nối đến cơ sở dữ liệu được hỗ trợ sẵn thông qua xử lý bootstrap (Drupal hỗ trợ sẵn) bằng việc gọi `include_once('includes/bootstrap.inc')`, rồi gọi `drupal_bootstrap(DRUPAL_BOOTSTRAP_DATABASE)` để phát sinh kết nối. Từ đó, có thể sử dụng các truy vấn SQL.

### VI.1.3 Thực thi các truy vấn đơn giản

Chức năng `db_query()` được sử dụng để thực thi truy vấn đến kết nối cơ sở dữ liệu hỗ trợ các truy vấn cơ bản “select”, “insert”, “update”, “delete”.

Ví dụ:

Lấy tất cả các nội dung của bảng `joke` với giá trị bằng `$node->vid` có kiểu là integer:

```
db_query('SELECT * FROM {joke} WHERE vid = %d', $node->vid);
```

Tương tự, cho phép INSERT, UPDATE, DELETE:

```
db_query("INSERT INTO {joke} (nid, vid, punchline) VALUES (%d, %d, '%s')",  
$node->nid, $node->vid, $node->punchline);
```

Dấu “{}” chỉ tên bảng là duy nhất và tránh được sự trùng lặp trong cơ sở dữ liệu. Như `{joke}`, `joke` là bảng sẽ được thêm vào cơ sở dữ liệu của Drupal khi bảng `joke` chưa có trong Drupal.

Placeholder (nơi giữ giá trị cho biến), ví dụ `%d` sẽ tự động thay thế giá trị của tham số, trong trường hợp này là `$node->vid`. Thêm một placeholder thì thêm một tham số tương ứng.

*Bảng 2.2. Các placeholder và ý nghĩa.*

Placeholder	Ý nghĩa
%s	String
%d	Integer
%f	Float
%b	Binary data
%%	Thêm vào dấu %

Kết quả của câu truy vấn

- Trả về giá trị đơn

```
$sql = "SELECT COUNT(*) FROM {node} WHERE type = 'blog' AND status = 1";
```

```
$total = db_result(db_query($sql));
```

Kết quả trả về số blog được công bố.

- Trả về nhiều giá trị

```
$sql = "SELECT * FROM {node} WHERE type = 'blog' AND status = 1";
```

```
$result = db_query(db_rewrite_sql($sql));
```

```
while ($data = db_fetch_object($result)) {
```

```
    $node = node_load($data->nid);
```

```
    print node_view($node, TRUE);} 
```

Kết quả xuất ra một danh sách các blog được công bố.

- Trả về kết quả có giới hạn

```
$sql = "SELECT * FROM {node} n WHERE type = 'blog' AND status = 1 ORDER BY  
n.created DESC";
```

```
$result = db_query_range(db_rewrite_sql($sql), 0, 10);
```

Kết quả trả về 10 blog mới nhất.

- Trả về kết quả có phân trang

```
$sql = "SELECT * FROM {node} n WHERE type = 'blog' AND status = 1 ORDER BY  
n.created DESC"
```

```
$result = pager_query(db_rewrite_sql($sql), 0, 10);
```

```
while ($data = db_fetch_object($result)) {
```

```
    $node = node_load($data->nid);
```

```
    print node_view($node, TRUE);} 
```

```
print theme('pager', NULL, 10);
```

Kết quả trên mỗi trang sẽ chứa danh sách 10 blog.

- Kết nối với nhiều cơ sở dữ liệu (database)

Trong tập tin settings.php, `$db_url` có thể là một chuỗi hoặc một mảng bao gồm nhiều chuỗi kết nối database. Và đây là cú pháp mặc định, xác định một chuỗi kết nối đơn:

```
$db_url = 'mysql://username:password@localhost/databasename';
```

Ở đây, sử dụng MySQL với databasename sẽ tham chiếu đến tên database sau khi cài đặt Drupal.

Xác định hai chuỗi kết nối: default và legacy.

```
$db_url['default'] = 'mysql://user:password@localhost/drupal5';
```

```
$db_url['legacy'] = 'mysql://user:password@localhost/legacydatabase';
```

Khi kết nối đến một database khác, chỉ cần : *db\_set\_active('legacy');*

```
$result = db_query("SELECT * FROM ldap_user WHERE uid = %d", $user->uid);
```

Sau khi hoàn thành, nên chuyển về mặc định: *db\_set\_active('default');*

#### VI.1.4 Viết mới lớp cơ sở dữ liệu trừu tượng (Database Abstract Layer)

Đầu tiên, sao chép tập tin từ includes/database.mysql.inc và đổi tên theo mong muốn như includes/database.dnabase.inc. Sau đó, thay đổi mã nguồn bên trong mỗi chức năng wapper để tham chiếu đến chức năng của dnabase trong tập tin database.dnabase.inc thay cho chức năng của mysql trong tập tin database.mysql.inc.

#### VI.2 *Xây dựng giao diện (Theme)*

Để theme có thể tương tác được với Drupal thì cần thiết phải có theme engine (máy theme). Thông qua theme engine có thể giao tiếp với ngôn ngữ template (mẫu) và được Drupal hỗ trợ sẵn. Cho nên chỉ cần tạo ra theme mới.

Đầu tiên, tạo ra một thư mục theme được đặt tên custom\_theme, từ thư mục này tạo ra hai tập tin:

- page.tpl.php: Tạo template cho website kết hợp với giá trị được Drupal hỗ trợ.
- page.css: Định dạng cho trang page.tpl.php về phong chữ, màu sắc, kích thước,...

Ngoài ra, có thể tạo ra nhiều tập tin khác để hỗ trợ cho theme như: node.tpl.php, block.tpl.php, comment.tpl.php.

Tạo page.tpl.php

```
<html>

<head>

<title><?php print $head_title ?></title>

<link rel="stylesheet" href="page.css" type="text/css" />

</head>

<body>

    <div id="container">

        <div id="header">

            <h1><?php print $site_name ?></h1>

        </div>

        <?php if ($sidebar_left): ?>

            <div id="sidebar-left">

                <?php print $sidebar_left ?>

            </div>

        <?php endif; ?>

        <div id="main">

            <h2><?php print $title ?></h2>

            <?php print $content ?>

        </div>

        <div id="footer">

            <?php print $footer_message ?>

        </div>

    </div>

</body>
```



</html>

Bên dưới là những giá trị của Drupal, cho phép sử dụng một cách linh động:

- \$base\_path: Đường dẫn cài đặt Drupal. Mặc định, là "/" nếu Drupal được cài đặt trong thư mục gốc.
- \$breadcrumb: Trả về HTML để hiển thị breadcrumbs điều hướng trên trang.
- \$closure: Trả về kết quả của hook\_footer() được hiển thị ở cuối trang.
- \$css: Trả về cấu trúc mảng của tất cả CSS được thêm vào trang.
- \$content: Trả về nội dung HTML được hiển thị.
- \$directory: Đường dẫn tới thư mục theme được đặt.

<?php print \$base\_path . \$directory ?>

- \$feed\_icons: Trả về liên kết nguồn dữ liệu RSS.
- \$footer\_message: Trả về thông điệp ở footer.
- \$head: Trả về HTML được thay thế với <head></head>.
- \$head\_title: Trả về tiêu đề trang giữa thẻ <title></title>.
- \$help: Trợ giúp, chủ yếu là xuất hiện ở trang quản trị.
- \$is\_front: TRUE nếu trang trước hiện hành được hiển thị.
- \$language: Ngôn ngữ mà trang mô tả.
- \$layout: Cho phép tạo các kiểu khác nhau của layout (bố trí).
- \$logo: Đường dẫn hình ảnh logo. 
- \$messages: Thông báo lỗi và thành công.
- \$mission: Trả về tác vụ của trang khi \$is\_front là TRUE.
- \$node: Nội dung node.
- \$primary\_links: Một mảng chứa liên kết chính.

<?php print theme('links', \$primary\_links) ?>

- `$scripts`: Trả về HTML bằng việc thêm thẻ `<script>`.
- `$search_box`: Trả về HTML cho form tìm kiếm, được hiển thị khi cấu hình kích hoạt.
- `$secondary_links`: Một mảng chứa các liên kết phụ tương tự như `$primary_links`.

```
<?php print theme('links', $secondary_links) ?>
```

- `$sidebar_left`: Trả về HTML cho sidebar left (khu vực chứa nội dung bên trái).
- `$sidebar_right`: Trả về HTML cho sidebar right (khu vực chứa nội dung bên phải).
- `$site_name`: Tên trang, được hiển thị khi cấu hình kích hoạt.
- `$site_slogan`: Khẩu hiệu của trang, khi được kích hoạt.
- `$styles`: Trả về HTML khi liên kết với CSS cần thiết.
- `$tabs`: Trả về HTML để hiển thị các thẻ như xem hoặc sửa cho các node.
- `$title`: Tiêu đề của node.

Tạo `page.css`

```
#container{
```

```
    width: 90%;
```

```
    margin: 10px auto;
```

```
    background-color: #fff; }
```

```
#container #header {
```

```
    padding: .5em;
```

```
    background-color: #ddd;
```

```
    border-bottom: 1px solid gray; }
```

```
#container #sidebar-left {
```

```
    float: left;
```

```
    width: 160px;
```

```
    margin: 0;
```

```
padding: 1em; }
#container #footer {
margin: 0;
padding: .5em; }
```

### VI.3 Xây dựng Module

Để viết một module ta cần phải tạo ra tối thiểu hai tập tin với định dạng .info và .module. Trong đó, tập tin .info sẽ cung cấp thông tin cho hệ thống và tập tin .module sẽ thực hiện chức năng mà cần xây dựng thông qua các hook được Drupal hỗ trợ sẵn.

Đầu tiên, tạo một thư mục đặt tên là annotate (tên phù hợp với ý nghĩa của module). Tiếp theo, trong thư mục annotate tạo ra 2 tập tin được đặt tên là annotate.info và annotate.module.

Viết tập tin annotate.info:

```
; $Id$
```

name = Annotate → Cho biết tên của module là Annotate.

description = Allows users to annotate nodes. → Mô tả chức năng của module.

package = Example → Tên package là Example.

version = “5.11” → Cho biết phiên bản của module là 5.11.

▼ Example				
Bật	Throttle	Tên	Phiên bản	Mô tả
<input type="checkbox"/>	<input type="checkbox"/>	Annotate	5.11	Allows users to annotate nodes.

Hình 2.9. Kết quả mà drupal hiển thị.

Viết tập tin annotate.module:

Tập tin này được viết bằng ngôn ngữ PHP, điều đặc biệt là viết module trong Drupal chỉ cần thẻ mở (không cần thẻ đóng “?php”):

```
<?php
```

```
// $Id$

/**

* @file

* Lets users add private annotations to nodes.

*

* Adds a text field when a node is displayed

* so that authenticated users may make notes.

*/ → Đây là các dòng chú thích giúp cho api.module phát hiện được tập tin này có chức năng gì.
```

❖ *Thực thi hook*

Mã nguồn sẽ được viết bên trong thẻ mở, bằng cách thực thi các hook mà Drupal hỗ trợ sẵn theo quy tắc kết hợp: tên module và tên hook. Ở Drupal, các phương thức hook không cần phải đăng ký sự kiện, để thực thi thì chỉ cần gọi callback.

```
/* Thực thi hook_menu().*/

function annotate_menu($may_cache) {

  $items = array();

  if ($may_cache) {

    $items[] = array(

      'path' => 'admin/settings/annotate',

      'title' => t('Annotation settings'),

      'description' => t('Change how annotations behave.'),

      'callback' => 'drupal_get_form',

      'callback arguments' => array('annotate_admin_settings'),

      'access' => user_access('administer site configuration')

    );
  }
}
```

```
return $items; }
```

Hook này cho phép module đăng ký đường dẫn (path) để xác định yêu cầu được xử lý. Dựa vào kiểu đăng ký đó, một liên kết được đặt trong khối điều hướng (navigation block) và xuất hiện trong trang quản lý menu (q=admin/menu).

Drupal sẽ gọi hook hai lần: một lần với thiết lập `$may_cache` là TRUE, và một lần được thiết lập là FALSE. Vì vậy, mỗi thành phần menu nên được đăng ký khi `$may_cache` hoặc là TRUE hoặc là FALSE.

#### ❖ Cấu hình cho module

Drupal có nhiều kiểu node như là story, page. Và cần tạo ra kiểu node cho phù hợp với module annotate:

```
/**
```

```
* Định nghĩa form cấu hình.
```

```
*/
```

```
function annotate_admin_settings() {
```

```
  $form['annotate_nodetypes'] = array(
```

```
    '#type' => 'checkboxes',
```

```
    '#title' => t('Users may annotate these node types'),
```

```
    '#options' => node_get_types('names'),
```

```
    '#default_value' => variable_get('annotate_nodetypes', array('story')),
```

```
    '#description' => t('A text field will be available on these node types to  
make user-specific notes.'), );
```

```
  $form['array_filter'] = array('#type' => 'hidden');
```

```
  return system_settings_form($form); }
```

**Users may annotate these node types:**

page

story

A text field will be available on these node types to make user-specific notes.

Save configuration

Reset to defaults

*Hình 2.10. Kết quả cấu hình kiểu node.*

❖ *Thêm form nhập dữ liệu*

Tạo ra giao diện cho phép người dùng có thể nhập dữ liệu:

```
/* Thực thi của hook_nodeapi().*/  
  
function annotate_nodeapi(&$node, $op, $teaser, $page) {  
  switch ($op) {  
    case 'view':  
      global $user;  
  
      // Cho phép người dùng thao tác sau khi xác nhận quyền hạn.  
      if ($teaser || $user->uid == 0) {  
        break; }  
  
      $types_to_annotate = variable_get('annotate_nodetypes', array('story'));  
      if (!in_array($node->type, $types_to_annotate)) {  
        break; }  
  
      // Thêm thành phần nội dung.  
      $node->content['annotation_form'] = array(  
        '#value' => drupal_get_form('annotate_entry_form', $node),  
        '#weight' => 10  
      );  
    }  
  }  
  /* Tạo form nhập cho module annotation.*/
```

```

function annotate_entry_form($node) {
  $form['annotate'] = array(
    '#type' => 'fieldset',
    '#title' => t('Annotations') );
  $form['annotate']['nid'] = array(
    '#type' => 'value',
    '#value' => $node->nid );
  $form['annotate']['note'] = array(
    '#type' => 'textarea',
    '#title' => t('Notes'),
    '#default_value' => $node->annotation,
    '#description' => t('Make your personal annotations about this content here. Only you
    (and the site administrator) will be able to see them.') );
  $form['annotate']['submit'] = array(
    '#type' => 'submit',
    '#value' => t('Update') );
  return $form; }

```

Annotations

**Notes:**

Make your personal annotations about this content here. Only you (and the site administrator) will be able to see them.

Update

*Hình 2.11. Kết quả tạo form cho node.*

❖ *Chứa dữ liệu trong bảng cơ sở dữ liệu.*

Nếu như không sử dụng các bảng mà Drupal có sẵn thì có thể tạo mới bảng thông qua tập tin `.install`. Tập tin này cho phép tạo bảng vào trong cơ sở dữ liệu của Drupal và mọi thông tin của module `annotate` sẽ được lưu trữ ở bảng này.

Viết tập tin `annotate.install`

```
<?php
// $Id$

function annotate_install() {
  drupal_set_message(t('Beginning installation of annotate module.'));
  switch ($GLOBALS['db_type']) {
    case 'mysql':
    case 'mysqli':
      db_query("CREATE TABLE annotations (
uid int NOT NULL default 0,
nid int NOT NULL default 0,
note longtext NOT NULL,
timestamp int NOT NULL default 0, PRIMARY KEY (uid, nid)
)/*!40100 DEFAULT CHARACTER SET utf8 */;");
      $success = TRUE;
      break;
    case 'pgsql':
      db_query("CREATE TABLE annotations (
uid int NOT NULL DEFAULT 0,
nid int NOT NULL DEFAULT 0,
note text NOT NULL,
timestamp int NOT NULL DEFAULT 0, PRIMARY KEY (uid, nid)
```



```

);" );
$success = TRUE;
break;
default:
drupal_set_message(t('Unsupported database.)); }
if ($success) {
drupal_set_message(t('The module installed tables successfully.)); }
else {
drupal_set_message(t('The installation of the annotate module
was unsuccessful.),'error');
} }

```

❖ *Thêm mã nguồn vào trong tập tin annotate.module*

```

/* Lưu thông tin annotation vào trong cơ sở dữ liệu. */
function annotate_entry_form_submit($form_id, $form_values) {
global $user;
$nid = $form_values['nid'];
$note = $form_values['note'];
db_query("DELETE FROM {annotations} WHERE uid = %d and nid = %d", $user-
>uid, $nid);
db_query("INSERT INTO {annotations} (uid, nid, note, timestamp) VALUES (%d,
%d, '%s', %d)", $user->uid, $nid, $note, time());
drupal_set_message(t('Your annotation was saved.)); }

```

#### *VI.4 Các làm việc của Taxonomy*

Module taxonomy cho phép phân loại nội dung vào trong các kiểu (categories) và các kiểu con (subcategories).

#### ❖ Terms

Term là nhãn (label) được đặt cho node. Đôi khi term cũng được gọi là tag (thẻ), và thao tác ấn định term thành một đối tượng được gọi là tagging (đặt thẻ).

#### ❖ Vocabularies (Từ vựng)

Một từ vựng bao gồm tập hợp các term. Drupal cho phép kết hợp với một từ vựng một hoặc nhiều loại node.

#### ❖ Kiểu Taxonomy

Một số kiểu taxonomy: Kiểu đơn giản nhất là danh sách các term, và phức tạp nhất là những mối quan hệ đa phân cấp. Hơn nữa, các term có thể là các synonym của hoặc có quan hệ đến các term khác.

#### ❖ Hierarchical (Phân cấp)

Sự phân cấp được hình thành từ mỗi term có một mối quan hệ với một term khác, như bảng bên dưới:

*Bảng 2.3. Các term đơn giản trong một từ vựng.*

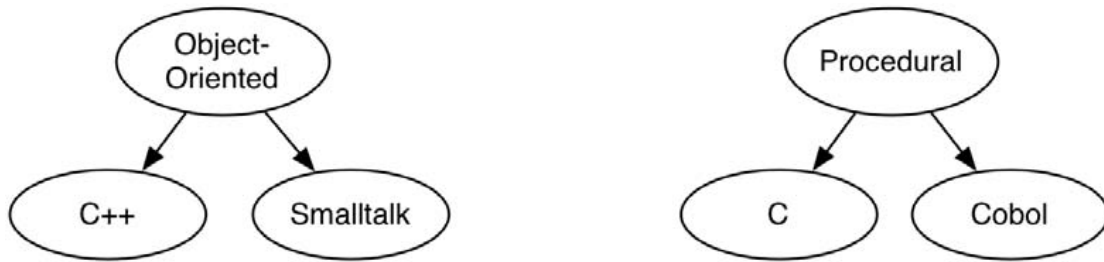
Term ID	Term Name
1	C
2	C++
3	Cobol

*Bảng 2.4. Các term phân cấp trong một từ vựng.*

Term ID	Term Name
1	Object-Oriented
2	C++
3	Smalltalk
4	Procedural
5	C

6	Cobol
---	-------

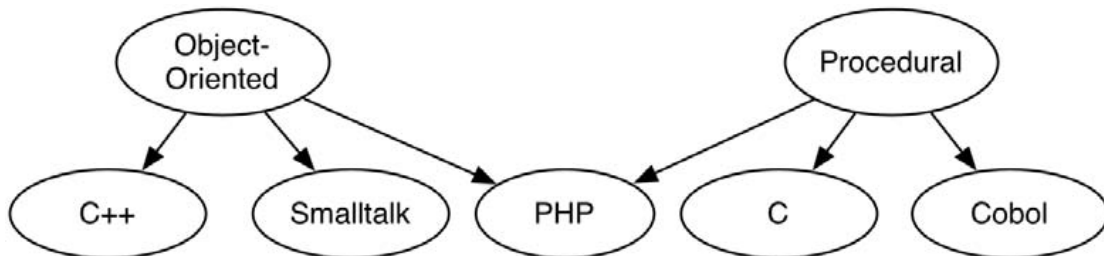
Hình 2.12. Chỉ ra mối quan hệ phân cấp tường minh. Trong đó, Object-Oriented và Procedural là term cha; C++, Smalltalk, C, Cobol là những term con.



Hình 2.13. Một từ vựng phân cấp có mối quan hệ cha con giữa các term.

❖ Multiple Hierarchical (Đa phân cấp)

Một từ vựng có thể có nhiều phân cấp. Có nghĩa, một term có thể có nhiều hơn một cha. Ví dụ, term PHP được thêm vào là term thêm con của Object-Oriented và Procedural vì ngôn ngữ PHP có thể được viết theo hướng thủ tục và hướng đối tượng, như ở Hình 2.14.



Hình 2.15. Trong một từ vựng đa phân cấp thì các term có thể có nhiều hơn một cha.

Do đó, trong giai đoạn phân tích trang web, cần xác các kiểu từ vựng cần thiết.

❖ Sử dụng AND và OR trong URL.

Cú pháp xây dựng taxonomy URL có hỗ trợ AND và OR bằng việc sử dụng ký tự dấu phẩy “,”, và dấu cộng “+”. Để node hiển thị nội dung có term ID 5 và 6, thì sử dụng URL như sau: <http://example.com/?q=taxonomy/term/5,6>

Sử dụng URL để chỉ các node mà được ấn định là term ID 1,2 hoặc 3:

<http://example.com/?q=taxonomy/term/1+2+3>

Sự kết hợp giữa AND và OR thì không được hỗ trợ ở taxonomy.module hiện hành.

❖ Xác định độ sâu của từ vựng phân cấp

Ở ví dụ trên, sử dụng URL: <http://example.com/?q=taxonomy/term/5>

có thể được viết thành <http://example.com/?q=taxonomy/term/5/0>

Trong đó số 0 là mức phân cấp hay độ sâu phân cấp, xem bảng 14-3.

**Bảng 2.5. Một từ vựng phân cấp**

Term ID	Term Name
1	Canada
2	British Columbia
3	Vancouver
4	Ontario
5	Toronto

Mức đầu tiên là nước Canada, có hai con là tỉnh British Columbia và Ontario.

Mỗi tỉnh có một con là lần lượt là Vancouver và Toronto.

Tất cả các node được gắn thẻ với Vancouver sẽ được chia sẻ theo URL sau:

<http://example.com?q=taxonomy/term/3> hoặc

<http://example.com?q=taxonomy/term/3/0>

Để hiển thị các node được gắn thẻ chỉ với British Columbia, sử dụng URL:

<http://example.com?q=taxonomy/term/2> hoặc với mức phân cấp:

<http://example.com?q=taxonomy/term/2/1>

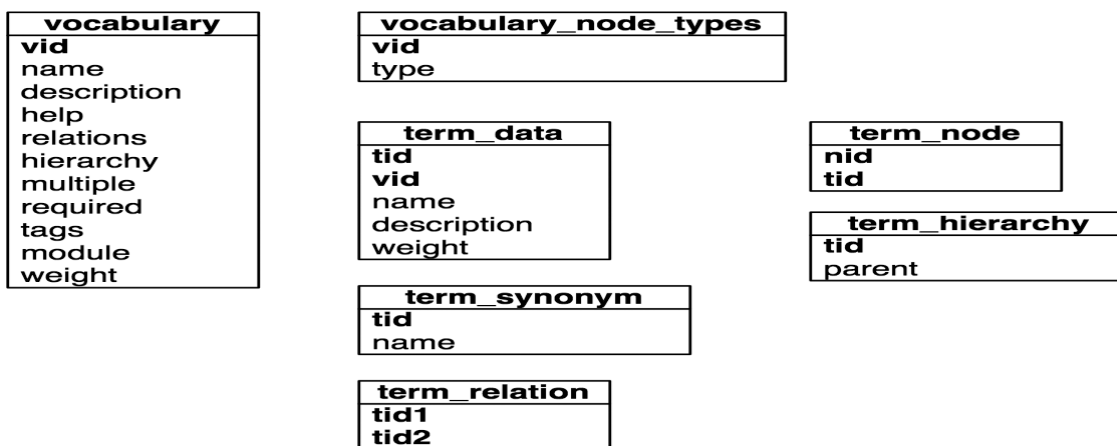
Tất cả các node được gắn thẻ với Canada hoặc với các tỉnh British Columbia, Ontario và thành phố Vancouver, Toronto của nó sẽ được hiển thị, sử dụng:

<http://example.com?q=taxonomy/term/1/all>

Ta có thể sử dụng module View để hỗ trợ cho từ vựng đa phân cấp (<http://drupal.org/project/views>).

❖ Lưu trữ Taxonomy

Để xây dựng một module Taxonomy riêng thì cần phải hiểu các taxonomy được lưu trữ như thế nào trong cơ sở dữ liệu. Đối với cơ sở dữ liệu không thuộc kiểu của Drupal, ta có thể tạo ra một flat taxonomy bằng cách thêm cột vào bảng dữ liệu, thông qua bảng cơ sở dữ liệu thông thường. Hình 14-4. Cho biết cấu trúc các bảng.



Hình 2.16. Các bảng của taxonomy của Drupal. Các khóa chính) được in đậm.

Các bảng dưới đây là hệ thống chứa taxonomy của Drupal:

- vocabulary: Chứa thông tin về từ vựng mà có thể được điều chỉnh thông qua giao diện các mục phân loại (Categories) của Drupal.
- vocabulary\_node\_types: Lưu các vốn từ được sử dụng với các kiểu node. Thuộc tính type là tên của kiểu node (kiểu blog) và phải phù hợp với cột type trong bảng node.
- term\_data: Chứa tên của term, cùng với những mô tả, thứ tự (weight) được tạo.
- term\_synonym: Các Synonym đối với các term ID tạo ra được chứa trong bảng này.
- term\_relation: Chứa các term ID của term mà được lựa chọn như là sự quan hệ khi định nghĩa một term.
- term\_hierarchy: Chứa term ID của một term và term ID cha của nó. Nếu term ở tại gốc (root) mà không có term cha thì ID của term cha là 0.

- term\_node: Được dùng để phân loại các term mà node được gắn thẻ với term tương ứng.

## CHƯƠNG III: MÔ TẢ HỆ THỐNG WEBSITE

### I. Phân tích.

#### I.1. Mục đích.

Xây dựng trang web khoa công nghệ thông tin sử dụng mã nguồn mở Drupal. Với các nhiệm vụ:

- ❖ Thiết kế giao diện thân thiện.
- ❖ Đảm bảo các chức năng hiện có.
- ❖ Bổ sung một số chức năng cần thiết.

#### I.2. Một số yêu cầu từ người dùng.

- ❖ Giao diện rõ ràng, đơn giản và đầy đủ.
- ❖ Riêng trang chủ: phải thiết kế sao cho thể hiện được bốn đối tượng (cả về nội dung và hình thức):
  - Sinh viên tương lai.
  - Sinh viên hiện tại.
  - Giáo viên.
  - Cựu sinh viên.
- ❖ Phân quyền cho các vai trò: người quản trị, thư ký, giảng viên, user, người quản lý câu lạc bộ: phần mềm, hệ thống thông tin, mạng máy tính, người quản lý diễn đàn.
- ❖ Đảm bảo các chức năng hiện có.

#### I.3. Mô tả bài toán

Khoa Công Nghệ Thông Tin muốn tạo sự tiện lợi, thân thiện cho người dùng, an toàn cho người quản trị hệ thống website.

Văn phòng khoa phụ trách việc cập nhật thông báo, thời khoá biểu, lịch giảng dạy, điểm thi, biểu mẫu, các hoạt động trong khoa và có thể thay đổi tài khoản được cấp.

Đối với sinh viên, khoa đào tạo bốn hệ: đại học, cao đẳng, văn bằng II, hoàn chỉnh đại học được cung cấp các thông tin như:

Hàng năm vào đầu mỗi kỳ học, sinh viên có thể xem lịch học.

Trong suốt quá trình học, sinh viên có thể xem thông báo, tin tức, biểu mẫu và các hoạt động trong khoa. Đặc biệt sinh viên có thể download bài giảng của môn học cũng như tài liệu tham khảo để phục vụ cho việc học và nghiên cứu của mình.

Sau mỗi kỳ thi, sinh viên có thể xem điểm thi và thành tích học tập. Với những sinh viên có thành tích tốt, nổi bật thì sẽ được lưu danh trong bảng vàng. Sinh viên có thể dõi theo các hoạt động nghiên cứu trong khoa như: luận văn tốt nghiệp, các đồ án môn học, nghiên cứu khoa học.

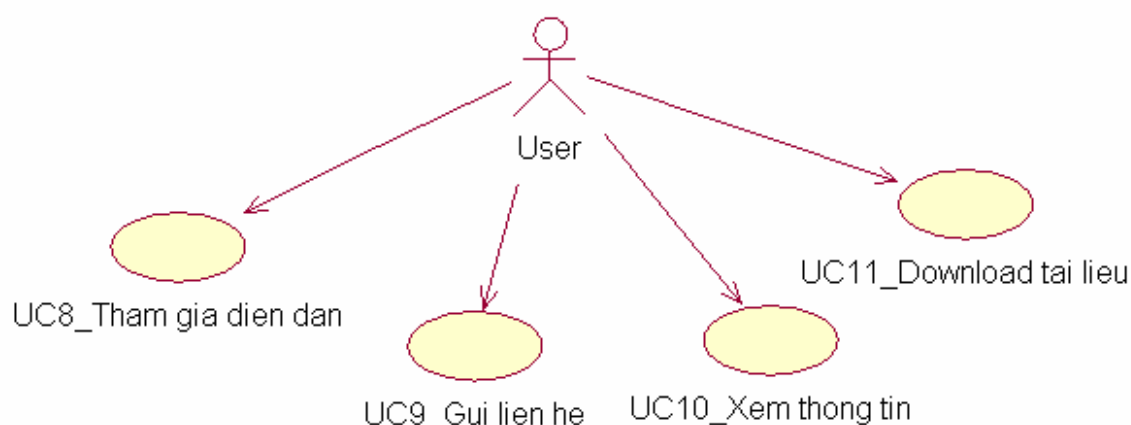
Ngoài ra, còn được cung cấp các thông tin về các hội thảo, hoạt động câu lạc bộ và tham gia diễn đàn thảo luận.

Giảng viên được xem thông báo, biểu mẫu, lịch giảng dạy, upload bài giảng và thay đổi tài khoản của mình.

Người quản trị có thể quản lý tài khoản người dùng và thao tác trên toàn bộ hệ thống website.

#### *I.4. Mô tả usecase*

##### *I.4.1. User*



*Hình 3.1. Usecase Diagram của User.*

Thông tin ở đây là:

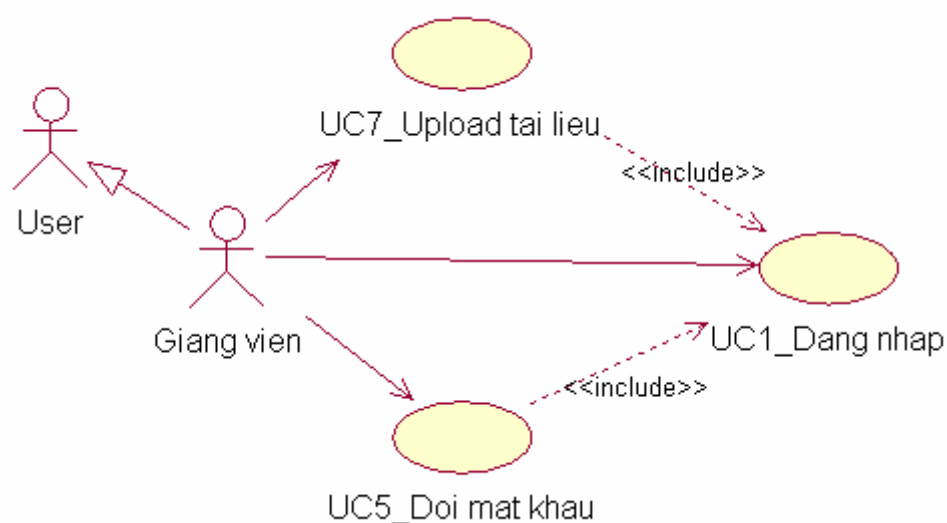


- Thông báo.
- Tin tức.
- Bảng điểm.
- Việc làm mới.
- Thời khóa biểu.
- Biểu mẫu.
- Lịch giảng dạy.
- Danh sách giảng viên.
- Danh sách cựu sinh viên.
- Câu lạc bộ.
- Và các trang khác trên web.

STT	User_case	Mô tả
1	UC10_Xem thông tin	Xem tất cả các thông tin trên trang web.
2	UC8_Tham gia diễn đàn.	User có thể vào diễn đàn đọc bài hoặc gửi bài viết tham gia thảo luận.
3	UC9_gui lien he	Gửi mail liên hệ đến mail server của khoa.
4	UC11_download tai lieu	Download tài liệu trên web về máy mình.

*Bảng 3.1 Mô tả Usecase của User.*

#### I.4.2. Giảng viên



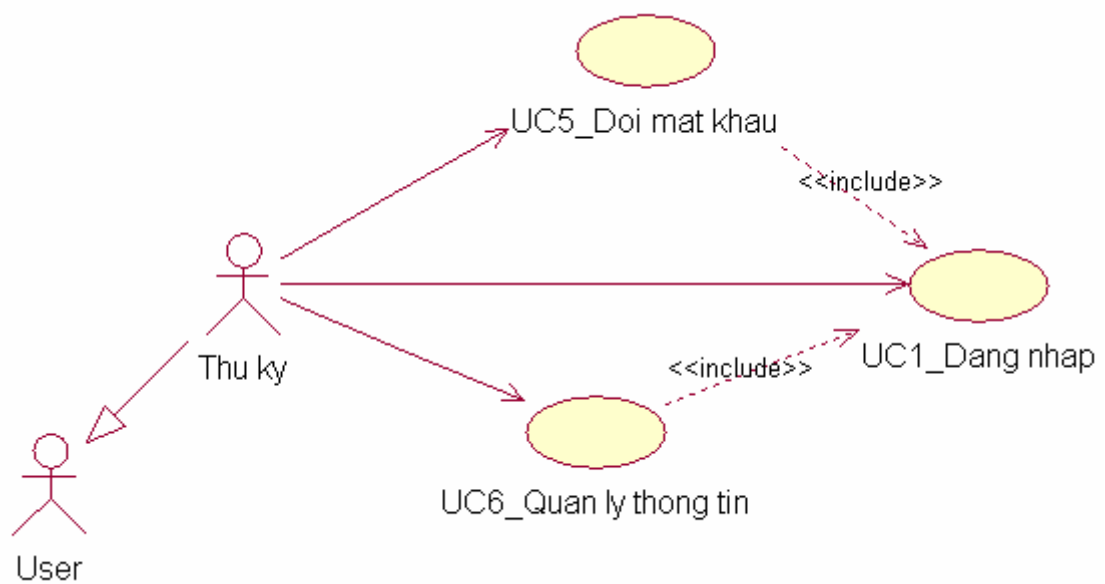
Hình 3.2. Usecase Diagram của Giảng viên.

STT	User_case	Mô tả
1	UC1_Dang nhap	Giảng viên đăng nhập theo tài khoản riêng để thực hiện chức năng Upload tài liệu và Đổi mật khẩu.
2	UC5_Doi mat khau	Giảng viên có quyền đổi mật khẩu khác nếu muốn.
3	UC7_Upload tai lieu	Giảng viên sẽ đưa bài giảng môn học hoặc các tài liệu khác lên web.

Bảng 3.2 Mô tả Usecase của Giảng viên.

Ngoài các chức năng giống như một User ra thì Giảng viên còn thêm 2 chức năng trên.

#### I.4.3. Thư ký



*Hình 3.3 Usecase Diagram của Thu ký.*

Thông tin ở đây bao gồm:

- Thông báo:
  - Đại học.
  - Cao đẳng.
  - Văn bằng II.
  - Hoàn chỉnh đại học.
- Bảng điểm.
  - Đại học.
  - Cao đẳng.
  - Văn bằng II.
  - Hoàn chỉnh đại học.
- Việc làm mới.
- Thời khóa biểu
  - Đại học.
  - Cao đẳng.

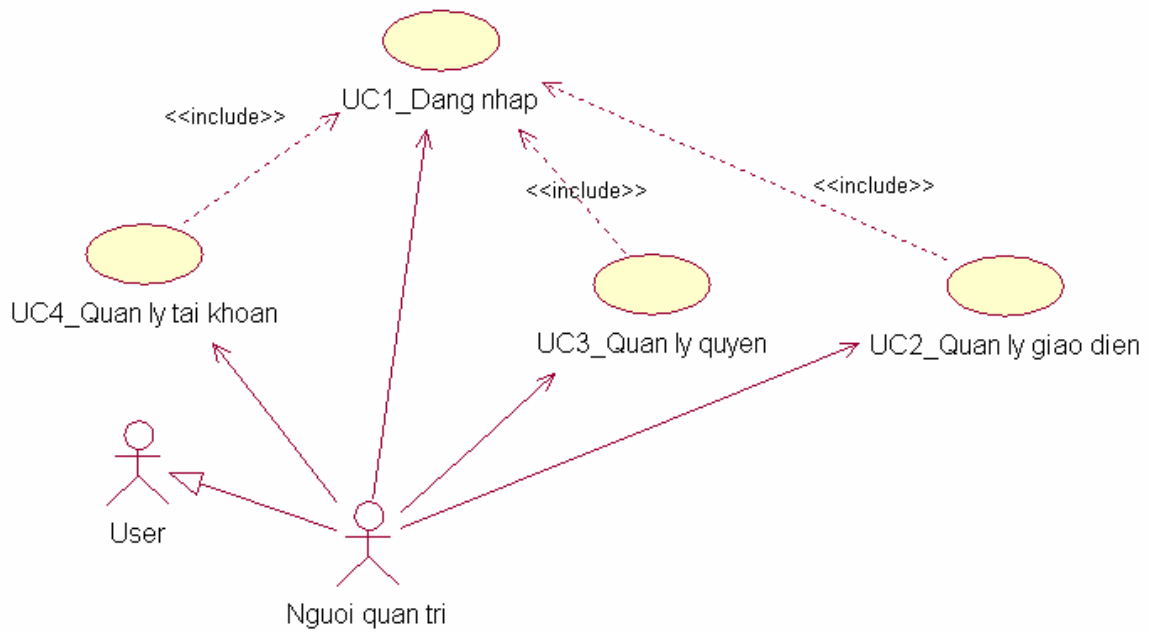
- Văn bằng II.
  - Hoàn chỉnh đại học.
- Biểu mẫu
- Sinh viên.
  - Giảng viên.
- Lịch giảng dạy.
- Danh sách giảng viên.
- Danh sách cựu sinh viên.

STT	User_case	Mô tả
1	UC1_Dang nhap	Thư ký đăng nhập theo tài khoản của thư ký.
2	UC5_Doi mat khau	Thư ký có quyền đổi mật khẩu khác nếu muốn.
3	UC7_Quan ly thong tin	Thư ký có quyền thêm, xoá, sửa thông tin, dữ liệu.

*Bảng 3.3. Mô tả Usecase của Thư ký.*

Ngoài các chức năng giống như một User ra thì Thư ký còn thêm các chức năng trên.

#### I.4.4. Người quản trị.



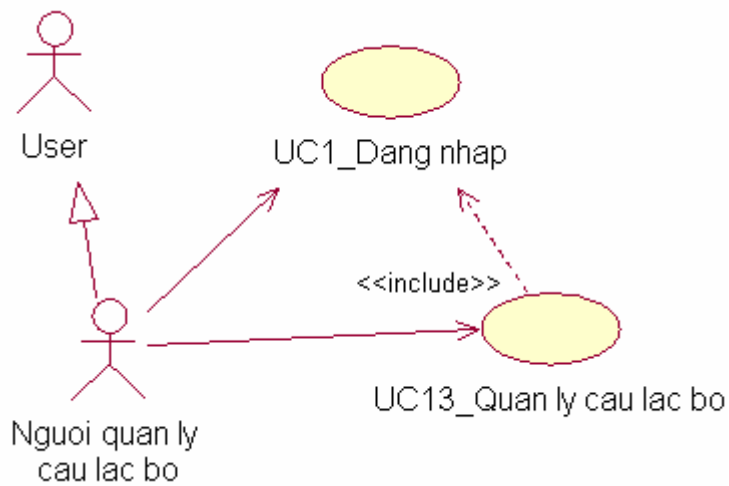
Hình 3.4. Usecase Diagram của Người quản trị.

STT	User_case	Mô tả
1	UC1_Dang nhap	Người quản trị phải đăng nhập tài khoản riêng.
2	UC2_Quan ly giao dien	Thêm, xoá, sửa menu, block, theme.
3	UC3_Quan ly quyen	Thêm, xoá, sửa vai trò, quyền hạn sử dụng.
4	UC4_Quan ly tai khoan	Có quyền thêm, xoá, thay đổi tài khoản người dùng.

Bảng 3.4. Mô tả Usecase của Người quản trị.

Ngoài các chức năng giống như một User thì Người quản trị còn thêm các chức năng trên.

#### I.4.5. Câu lạc bộ.

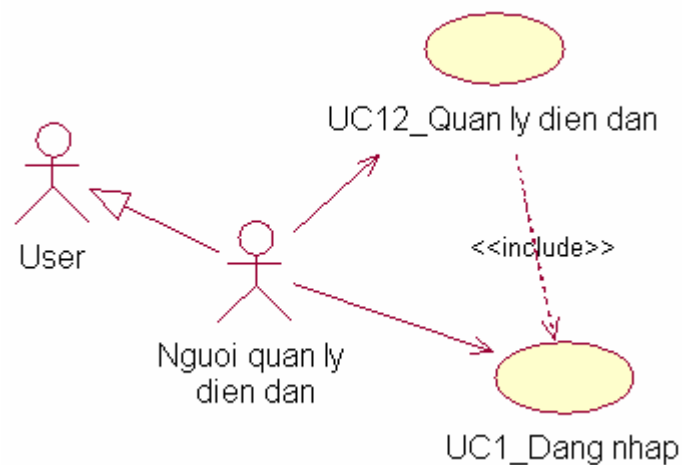


Hình 3.5. Usecase Diagram của Người quản lý câu lạc bộ.

STT	User_case	Mô tả
1	UC1_Dang nhap	Người quản lý câu lạc bộ đăng nhập theo tài khoản riêng.
2	UC13_Quan ly cau lac bo	Thêm, xoá, sửa bài viết, lịch sinh hoạt của câu lạc bộ.

Bảng 3.5. Mô tả Usecase của Người quản lý câu lạc bộ.

#### I.4.6. Diễn đàn.



Hình 3.6. Usecase Diagram của Người quản lý diễn đàn.

STT	User_case	Mô tả
-----	-----------	-------

1	UC1_Dang nhap	Người quản lý diễn đàn đăng nhập theo tài khoản riêng.
2	UC12_Quan ly dien dan	Xem gửi bài tham gia diễn đàn và tạo chủ đề thảo luận trong diễn đàn.

*Bảng 3.6. Mô tả Usecase của Người quản lý diễn đàn.*

## II. Đặc tả Usecase

### II.1. Đăng nhập

Usecase	- Đăng nhập (ID: UC1)
Mô tả	- Người dùng đăng nhập vào hệ thống theo tài khoản cá nhân đã được cấp để làm các chức năng riêng của mình.
Người dùng	- Người quản trị, giảng viên, thư ký, người quản lý diễn đàn, người quản lý câu lạc bộ.
Sự kiện chính	- Người dùng nhập vào các trường sau: tên đăng nhập và mật khẩu. Sau đó nhấn nút Đăng nhập.
Sự kiện phụ	- Thông báo lỗi sẽ hiển thị nếu bạn nhập sai.
Điều kiện cuối	- Hiển thị các chức năng có thể thực hiện được của người dùng.

*Bảng 3.7. Đặc tả Usecase Đăng nhập.*

### II.2. Quản lý câu lạc bộ

Usecase	- Quản lý câu lạc bộ (ID:UC3).
Mô tả	- Thêm, xoá, sửa bài giới thiệu, lịch sinh hoạt trên trang câu lạc bộ phần mềm, câu lạc bộ mạng, câu lạc bộ hệ thống thông tin.
Người dùng	- Người quản lý câu lạc bộ.
Điều kiện đầu	- Người dùng phải đăng nhập.
Sự kiện chính	- Thêm: Người dùng vào đường dẫn trang chủ → tạo nội dung sau đó nhập vào các trường sau: tiêu đề, chọn menu, nhập nội

	<p>dung, đính tập đính kèm (nếu có). Sau đó bấm nút “Gửi”.</p> <ul style="list-style-type: none"> <li>- Xoá: Người dùng chọn thông tin cần xoá rồi nhấn nút “xoá”.</li> <li>- Sửa: Người dùng chọn thông tin cần sửa, nhấn nút “sửa” rồi sửa thông tin theo ý muốn.</li> </ul>
Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập, thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	<ul style="list-style-type: none"> <li>- Thêm: Dòng chữ thêm tin thành công sẽ hiển thị.</li> <li>- Xoá: Dòng chữ xoá tin thành công sẽ hiển thị.</li> <li>- Sửa: Dòng chữ sửa tin thành công sẽ hiển thị.</li> </ul>

*Bảng 3.7. Đặc tả Usecase Quản lý câu lạc bộ.*

### *II.3. Quản lý diễn đàn*

Usecase	- Quản lý diễn đàn (ID:UC4).
Mô tả	- Thêm, sửa, xoá diễn đàn. Ngoài ra còn thêm, xoá, sửa chủ đề của diễn đàn
Người dùng	- Người quản lý diễn đàn.
Điều kiện đầu	- Người dùng phải đăng nhập.
Sự kiện chính	<ul style="list-style-type: none"> <li>- Thêm diễn đàn: vào chức năng thêm diễn đàn, nhập tên diễn đàn rồi nhấn nút “thêm”</li> <li>- Sửa diễn đàn: chọn diễn đàn đã tạo, rồi nhấn nút sửa. Nhập tên diễn đàn mới rồi nhấn nút lưu.</li> <li>- Xóa diễn đàn: chọn diễn đàn đã tạo, rồi nhấn nút xóa.</li> </ul>
Sự kiện phụ	- Người dùng bỏ trống những thông tin bắt buộc thì thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	- Thông báo thao tác thành công.



Bảng 3.8. Đặc tả Usecase Quản lý diễn đàn.

#### II.4. Đổi mật khẩu

Usecase	- Đổi mật khẩu (ID:UC5).
Mô tả	- Thay đổi tài khoản cá nhân của người dùng tùy theo ý muốn.
Người dùng	- Giảng viên và thư ký.
Điều kiện đầu	- Người dùng phải đăng nhập vào hệ thống.
Sự kiện chính	- Đổi tài khoản: Người dùng vào tài khoản cá nhân, nhấn nút “sửa” rồi sửa các thông tin theo ý muốn. Cuối cùng nhấn nút “Gửi”.
Sự kiện phụ	- Nếu người dùng thay đổi cả tên đăng nhập thì lần sau phải đăng nhập với tài khoản mới, nếu đăng nhập theo tài khoản cũ thì thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	- Dòng chữ thay đổi thành công được hiển thị.

Bảng 3.9. Đặc tả Usecase Đổi mật khẩu.

#### II.5. Upload tài liệu

Usecase	- Upload tài liệu (ID:UC6)
Mô tả	- Thêm, xoá file tài liệu trên web.
Người dùng	- Giảng viên.
Điều kiện đầu	- Giảng viên phải đăng nhập.
Sự kiện chính	- Người dùng vào đường dẫn trang chủ → tạo nội dung, rồi chọn upload bài giảng, sau đó nhập vào các trường sau: tiêu đề, chọn menu, nhập nội dung, chọn tập tin đính kèm rồi nhấn nút “đính kèm”, cuối cùng nhấn nút “Gửi”.  - Xoá: Người dùng vào menu download bài giảng của SVHT chọn thông tin có file đính kèm muốn xoá, sau đó chọn file để

	<p>xoá rồi nhấn nút “Gửi”.</p> <ul style="list-style-type: none"> <li>- Sửa: Người dùng vào menu download bài giảng của SVHT chọn thông tin có file đính kèm muốn sửa, sau đó sửa file theo ý muốn.</li> </ul>
Sự kiện phụ	<ul style="list-style-type: none"> <li>- Người dùng bỏ trống các trường bắt buộc nhập tải file có dung lượng lớn hơn cho phép thì thông báo lỗi sẽ xuất hiện.</li> </ul>
Điều kiện cuối	<ul style="list-style-type: none"> <li>- Thêm: Dòng chữ upload thành công sẽ hiển thị.</li> <li>- Xoá: Dòng chữ xoá upload thành công sẽ hiển thị.</li> <li>- Sửa: Dòng chữ sửa upload thành công sẽ hiển thị.</li> </ul>

*Bảng 3.10. Đặc tả Usecase Upload tài liệu.*

## II.6. Quản lý thông tin

Usecase	- Quản lý thông tin(ID:UC7)
Mô tả	<ul style="list-style-type: none"> <li>- Thêm, xoá, sửa các thông tin của khoa như: thông báo cho: đại học, cao đẳng, văn bằng II, hoàn chỉnh đại học, cựu sinh viên, giảng viên; tin tức; bảng điểm cho các hệ đào tạo: đại học, cao đẳng, văn bằng II, hoàn chỉnh đại học; thời khoá biểu cho các hệ đào tạo như trên; biểu mẫu cho sinh viên và cho giảng viên; việc làm mới; lịch giảng dạy; danh sách giảng viên; danh sách cựu sinh viên; các bài viết về khoa, về các đối tác; các hoạt động nghiên cứu và sinh hoạt của sinh viên;</li> </ul>
Người dùng	- Thư ký.
Điều kiện đầu	- Thư ký phải đăng nhập vào hệ thống.
Sự kiện chính	<ul style="list-style-type: none"> <li>- Thêm thông tin: Thư ký vào trang tạo nội dung và chọn đối tượng để thêm thông tin, sau đó nhập các trường sau: Tiêu đề, chọn menu, soạn nội dung, chọn file đính kèm (nếu có). Sau đó</li> </ul>

	<p>nhấn nút “Gửi”.</p> <ul style="list-style-type: none"> <li>- Xoá thông tin: Thư ký chọn thông tin rồi nhấn nút “xoá”.</li> <li>- Sửa thông tin: Thư ký chọn thông tin muốn sửa, nhấn nút sửa rồi sửa thông tin theo ý muốn.</li> </ul>
Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập, thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	<ul style="list-style-type: none"> <li>- Thêm: Dòng chữ thêm tin thành công sẽ hiển thị.</li> <li>- Xoá: Dòng chữ xoá tin thành công sẽ hiển thị.</li> <li>- Sửa: Dòng chữ sửa tin thành công sẽ hiển thị.</li> </ul>

*Bảng 3.11. Đặc tả Usecase Quản lý thông tin.*

## *II.7. Tham gia diễn đàn.*

Usecase	- Tham gia diễn đàn (ID:UC8).
Mô tả	- Xem gửi bài tham gia diễn đàn và tạo chủ đề thảo luận trong diễn đàn.
Người dùng	- Tất cả các loại người dùng và thành viên diễn đàn.
Điều kiện đầu	-Thành viên phải đăng nhập
Sự kiện chính	<ul style="list-style-type: none"> <li>- Người dùng vào xem thông tin diễn đàn.</li> <li>- Thêm chủ đề: chọn diễn đàn, chọn chức năng thêm, điền thông tin thêm chủ đề, rồi nhấn nút tạo.</li> <li>- Sửa chủ đề: chọn chủ đề đã tạo, rồi nhấn nút sửa, điền thông tin sửa diễn đàn và nhấn nút lưu.</li> <li>- Xóa chủ đề: Chọn chủ đề đã tạo, rồi nhấn nút xóa.</li> </ul>
Sự kiện phụ	- Trước khi xóa chủ đề sẽ hỏi người dùng có muốn xóa.
Điều kiện cuối	- Hiển thị thông tin thao tác thành công.

*Bảng 3.12. Đặc tả Usecase Tham gia diễn đàn.*

*II.8. Gửi liên hệ*

Usecase	- Gửi liên hệ (ID:UC9).
Mô tả	- Gửi thông tin liên hệ địa chỉ mặc định.
Người dùng	- Tất cả các loại người dùng.
Sự kiện chính	- Người dùng phải vào trang liên hệ, nhấp vào link địa chỉ thì xuất hiện giao diện liên hệ. Sau đó người dùng nhập các trường dữ liệu và nhấn nút “send”.
Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập, thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	- Thông tin sẽ được gửi vào hộp thư của người

*Bảng 3.13 Đặc tả Usecase Gửi liên hệ.*

*II.9. Xem thông tin*

Usecase	- Xem thông tin (ID:UC10)
Mô tả	- Xem tất cả các thông tin trên website khoa bao gồm: thông báo cho các hệ đào tạo và giảng viên, sinh viên, tin tức, thời khoá biểu, bảng điểm, bảng điểm, lịch giảng dạy, các bài viết về khoa, các hoạt động của khoa và sinh viên, các liên kết ngoài khác.
Người dùng	- Tất cả các loại người dùng.
Điều kiện đầu	- Truy cập vào cổng thông tin.
Sự kiện chính	- Người dùng chọn thông tin cần xem, nhấp vào liên kết
Điều kiện cuối	- Thông tin sẽ được hiển thị.

*Bảng 3.14. Đặc tả Usecase Xem thông tin.*

### II.10. Download tài liệu

Usecase	- Download tài liệu (ID:UC11).
Mô tả	- Download tài liệu trên web về máy mình.
Người dùng	- Tất cả các loại người dùng.
Điều kiện đầu	- Người dùng phải vào trang web.
Sự kiện chính	- Người dùng vào trang của sinh viên hiện tại chọn file muốn download, sau đó nhấp vào file.
Điều kiện cuối	- Trở về giao diện ban đầu.

Bảng 3.15 Đặc tả Usecase Download tài liệu.

### II.11. Quản lý tài khoản

Usecase	- Quản lý tài khoản (ID:UC12).
Mô tả	- Đổi mật khẩu, Thêm mới tài khoản, xoá tài khoản người dùng.
Người dùng	- Người quản trị.
Điều kiện đầu	- Người dùng phải đăng nhập vào hệ thống.
Sự kiện chính	<p>- Đổi mật khẩu: Người dùng vào tài khoản cá nhân, rồi nhấp vào nút “sửa”, sau đó nhập vào những trường muốn thay đổi. Cuối cùng nhấp vào nút “Gửi”.</p> <p>- Thêm tài khoản: Người dùng vào trang người dùng trong quản lý hệ thống, nhập các trường: tên đăng nhập, địa chỉ thư điện tử, mật khẩu, xác nhận lại mật khẩu, chọn trạng thái và vai trò. Cuối cùng nhấn nút “đăng ký”.</p> <p>- Sửa tài khoản: Người dùng vào trang người dùng trong quản lý hệ thống, chọn tài khoản muốn thay đổi để nhập các trường</p>

	<p>muốn thay đổi, rồi nhấn nút “Gửi”.</p> <p>- Xoá tài khoản: Người dùng vào trang người dùng trong quản lý hệ thống, chọn tài khoản để xoá.</p>
Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập hoặc nhập địa chỉ mail không đúng định dạng thì thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	<p>- Thêm: Dòng chữ thêm tài khoản thành công sẽ hiển thị.</p> <p>- Xoá: Dòng chữ xoá tài khoản thành công sẽ hiển thị.</p> <p>- Sửa: Dòng chữ sửa tài khoản thành công sẽ hiển thị.</p>

*Bảng 3.16. Đặc tả Usecase Quản lý tài khoản.*

## II.12. Quản lý quyền

Usecase	- Quản lý quyền (ID:UC13).
Mô tả	- Thêm, xoá, sửa vai trò, quyền hạn sử dụng.
Người dùng	- Người quản trị.
Điều kiện đầu	- Người dùng phải đăng nhập vào hệ thống.
Sự kiện chính	<p>- Thêm: Người dùng vào trang vai trò trong quản trị hệ thống và nhập vào field tên rồi nhấn nút “thêm vai trò”.</p> <p>- Xoá: Trong trang vai trò người dùng chọn vai trò rồi nhấn vào mục chỉnh sửa vai trò rồi nhấn nút “xoá vai trò”.</p> <p>- Sửa: Trong trang vai trò,</p> <ul style="list-style-type: none"> <li>• Sửa quyền hạn: Người dùng vào mục chỉnh sửa quyền hạn tương ứng với vai trò muốn sửa, chọn những chức năng cho vai trò. Sau đó nhấn nút “lưu quy tắc”.</li> <li>• Sửa vai trò: Người dùng vào mục chỉnh sửa vai trò tương ứng với vai trò muốn sửa, nhập vào tên vai trò sửa theo ý muốn. Sau đó nhấn nút “lưu vai trò”.</li> </ul>

Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập như tên, tên vai trò, thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	- Thêm: Dòng chữ thêm vai trò thành công sẽ hiển thị. - Xoá: Dòng chữ xoá vai trò thành công sẽ hiển thị. - Sửa: Dòng chữ sửa vai trò thành công sẽ hiển thị.

*Bảng 3.17. Đặc tả Usecase Quản lý quyền.*

### *II.13. Quản lý giao diện*

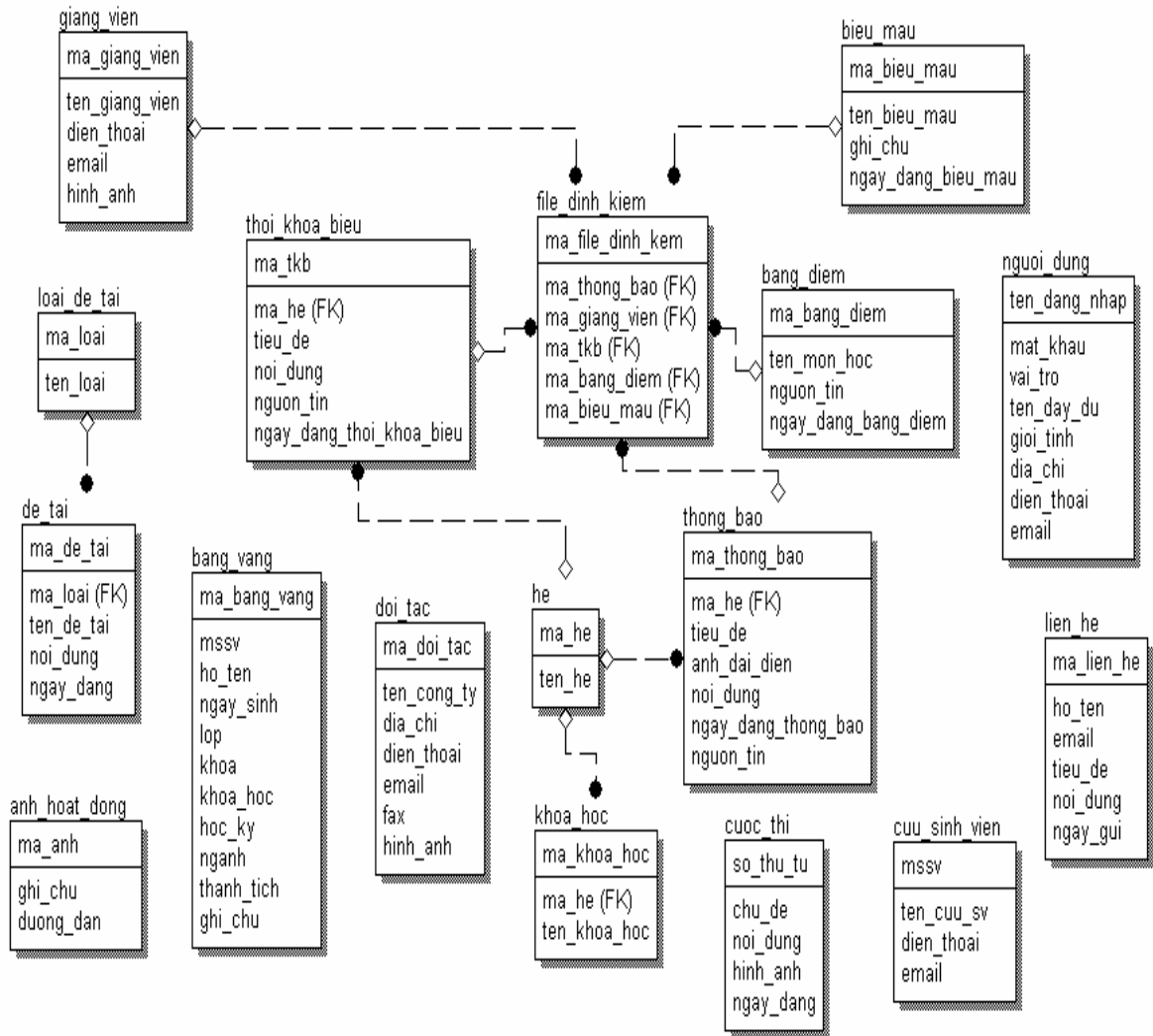
Usecase	- Quản lý giao diện (ID:UC14).
Mô tả	- Thêm, xoá, sửa menu, block, theme.
Người dùng	- Người quản trị.
Điều kiện đầu	- Người quản trị phải đăng nhập vào hệ thống.
Sự kiện chính	- Thêm: <ul style="list-style-type: none"> <li>• Menu: Người dùng vào trang thêm mục liên kết, nhập các trường sau: tiêu đề, mô tả, đường dẫn, chọn mục liên kết cấp cao hơn, chọn thứ tự. Cuối cùng nhấn nút “Gửi”.</li> <li>• Block: Người dùng vào trang thêm block, nhập các trường sau: mô tả block, nội dung block. Rồi nhấn nút “Lưu lại”.</li> <li>• Theme: Copy mô-đun theme mới bỏ vào thư mục mô-đun của Drupal.</li> </ul> - Xoá: <ul style="list-style-type: none"> <li>• Menu: Người dùng vào trang menu, chọn menu cần xoá rồi nhấn nút “xoá”.</li> </ul>

	<ul style="list-style-type: none"> <li>• Block: Vào trang Block, chọn block tương ứng rồi nhấn nút “xoá”.</li> <li>• Theme: Xoá mô-đun theme ra khỏi thư mục mô-đun của Drupal.</li> </ul> <p>- Sửa:</p> <ul style="list-style-type: none"> <li>• Menu: Người dùng vào trang chỉnh sửa mục liên kết, chỉnh sửa các thông tin theo ý muốn. Rồi nhấn nút “Gửi”.</li> <li>• Block: Vào trang Block, chọn cấu hình của Block tương ứng và chỉnh sửa các trường theo ý muốn. Rồi nhấn nút “Lưu lại”.</li> <li>• Theme: Vào trang Giao diện, chọn theme muốn thay đổi chọn “bật”, “Mặc định” rồi nhấn nút “Lưu cấu hình”.</li> </ul>
Sự kiện phụ	- Người dùng bỏ trống các trường bắt buộc nhập thì thông báo lỗi sẽ xuất hiện.
Điều kiện cuối	<p>- Thêm: Dòng chữ thêm menu, block thành công sẽ hiển thị.</p> <p>- Xoá: Dòng chữ xoá menu, block thành công sẽ hiển thị.</p> <p>- Sửa: Dòng chữ sửa menu, block, theme thành công sẽ hiển thị.</p>

*Bảng 3.18. Đặc tả Usecase Quản lý giao diện.*



Như vậy để quản lý được các thông tin trên chúng ta cần có một cơ sở dữ liệu như sau:



Hình 3.7 Mô hình ERD của khoa.

Dưới đây là danh sách các bảng tương ứng:

Bảng 3.19. Bảng de\_tai

Field	Type	Null	Description
ma_de_tai	bigint(20)	No	Mã đề tài
ten_de_tai	Mediumtext	Yes	Tên đề tài
noi_dung	Text	Yes	Nội dung
ngay_dang	Date	Yes	Ngày đăng
ma_loai	bigint(20)	Yes	Mã loại

*Bảng 3.20. Bảng giảng\_vien*

Field	Type	Null	Description
ma_giang_vien	bigint(11)	No	Mã giảng viên
ten_giang_vien	varchar(30)	Yes	Họ tên giảng viên
dien_thoai	varchar(20)	Yes	Điện thoại
Email	varchar(50)	Yes	Email
hinh_anh	varchar(255)	Yes	Đường dẫn ảnh
ma_bai_tap	bigint(20)	Yes	Mã bài tập

*Bảng 3.21. Bảng anh\_hoat\_dong*

Field	Type	Null	Description
ma_anh	bigint(20)	No	Mã ảnh
ghi_chu	Text	Yes	Ghi chú
duong_dan	Mediumtext	No	Đường dẫn

*Bảng 3.22. Bảng lien\_he*

Field	Type	Null	Description
ma_lien_he	bigint(20)	No	Mã liên hệ
ho_ten	varchar(30)	Yes	Họ tên
email	varchar(50)	Yes	Email
tieu_de	varchar(255)	Yes	Tiêu đề
noi_dung	Text	Yes	Nội dung
ngay_gui	Date	Yes	Ngày gửi

*Bảng 3.23. Bảng bieu\_mau*

Field	Type	Null	Description
-------	------	------	-------------

ma_bieu_mau	bigint(11)	No	Mã biểu mẫu
ten_bieu_mau	varchar(255)	Yes	Tên biểu mẫu
ghi_chu	Text	Yes	Ghi chú
ngay_dang	Date	Yes	ngày đăng

*Bảng 3.24. Bảng thong\_bao*

Field	Type	Null	Description
ma_thong_bao	bigint(11)	No	Mã thông báo
ma_he	bigint(11)	Yes	Mã hệ
anh_dai_dien	varchar(255)	Yes	Đường dẫn ảnh
noi_dung	Text	Yes	Nội dung
ngay_dang_thong_bao	Date	Yes	Ngày đăng thông báo
nguồn_tin	varchar(30)	Yes	Nguồn tin

*Bảng 3.25. Bảng thoi\_khoa\_bieu*

Field	Type	Null	Description
ma_tkb	bigint(11)	No	Mã thời khóa biểu
tieu_de	varchar(255)	Yes	Tiêu đề
noi_dung	Text	Yes	Nội dung
nguoit_tin	varchar(30)	Yes	Nguồn tin
ngay_dang	Date	Yes	Ngày đăng
ma_he	bigint(11)	Yes	Mã hệ
ma_khoa_hoc	bigint(11)	Yes	Mã khóa học

*Bảng 3.26. Bảng he*

Field	Type	Null	Description
ma_he	bigint(11)	No	Mã hệ
ten_he	varchar(30)	Yes	Tên hệ

*Bảng 3.27. Bảng bang\_diem*

Field	Type	Null	Description
ma_bang_diem	bigint(11)	No	Mã bảng điểm
ten_mon_hoc	varchar(50)	Yes	Tên môn học
nguồn_tin	varchar(30)	Yes	Nguồn tin
ngay_dang	Date	Yes	Ngày đăng
ma_he	bigint(11)	Yes	Mã hệ
ma_khoa_hoc	bigint(11)	Yes	Mã khoa học

*Bảng 3.28. Bảng khoa\_hoc*

Field	Type	Null	Description
ma_khoa_hoc	bigint(11)	No	Mã khoa học
ten_khoa_hoc	varchar(20)	Yes	Tên khoa học

*Bảng 3.29. Bảng cuu\_sinh\_vien*

Field	Type	Null	Description
so_thu_tu	bigint(11)	No	Số thứ tự
ten_cuu_sv	varchar(30)	Yes	Tên họ cựu sinh viên
dien_thoai	varchar(20)	Yes	Điện thoại
email	varchar(50)	Yes	Email

*Bảng 3.30. Bảng bang\_vang*

Field	Type	Null	Description
ma_bang_vang	bigint(11)	No	Mã bảng vàng
mssv	varchar(20)	Yes	Mã số sinh viên
ho_ten	varchar(30)	Yes	Họ tên sinh viên
ngay_sinh	Date	Yes	Ngày sinh
lop	varchar(11)	Yes	Tên lớp

Khoa	varchar(30)	Yes	Tên khoa
khoa_hoc	varchar(11)	Yes	Khóa học
hoc_ky	varchar(11)	Yes	Học kỳ
nganh	varchar(50)	Yes	Ngành
thanh_tich	varchar(255)	Yes	thành tích
ghi_chu	Text	Yes	Ghi chú

*Bảng 3.31. Bảng người\_dung*

Field	Type	Null	Description
ten_dang_nhap	varchar(30)	No	Tên đăng nhập
mat_khau	Text	Yes	Mật khẩu
vai_tro	tinyint(3)	Yes	Vai trò
ten_day_du	varchar(30)	Yes	Họ tên
gioi_tinh	varchar(4)	Yes	Giới tính
dia_chi	varchar(255)	Yes	Địa chỉ
dien_thoai	varchar(20)	Yes	Điện thoại
email	varchar(50)	Yes	Email

*Bảng 3.32 Bảng cuộc\_thi*

Field	Type	Null	Description
so_thu_tu	bigint(11)	No	Số thứ tự
chu_de	varchar(255)	Yes	Chủ đề
noi_dung	Text	Yes	Nội dung
hinh_anh	varchar(255)	Yes	Đường dẫn ảnh
ngay_dang	Date	Yes	Ngày đăng

*Bảng 3.33 Bảng file\_dinh\_kem*

Field	Type	Null	Description
ma_file_dinh_kem	bigint(11)	No	Mã file đính kèm

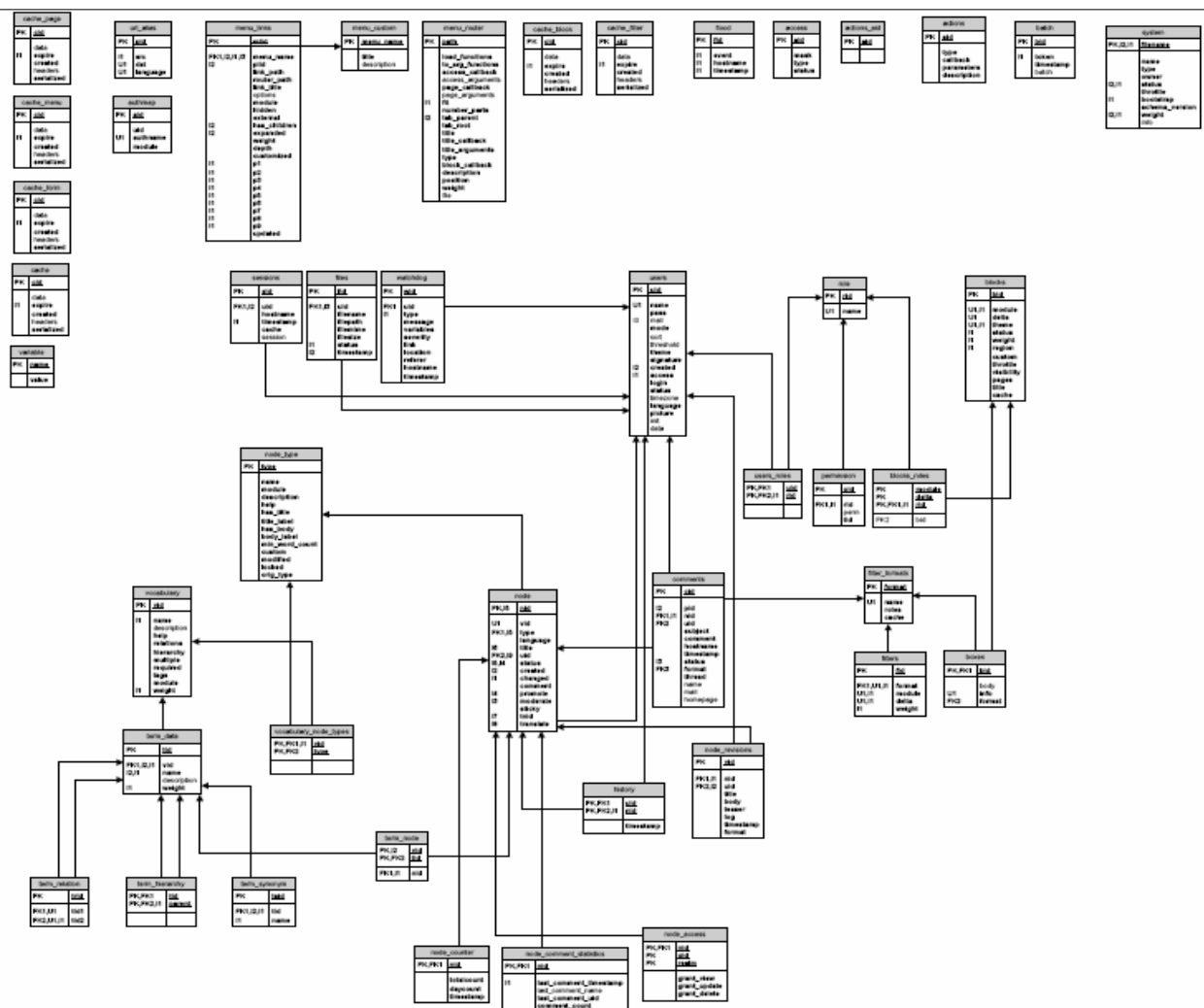
ma_bang_diem	bigint(11)	Yes	Mã bảng điểm
ma_tin_tuc	bigint(11)	Yes	Mã tin tức
ma_tkb	bigint(11)	Yes	Mã thời khóa biểu
ma_bieu_mau	bigint(11)	Yes	Mã biểu mẫu
ma_giang_vien	bigint(11)	Yes	Mã giảng viên

*Bảng 3.34. Bảng doi\_tac*

Field	Type	Null	Description
ma_doi_tac	bigint(11)	No	Mã đối tác
ten_cong_ty	varchar(255)	Yes	Tên công ty
dia_chi	varchar(255)	Yes	Địa chỉ
dien_thoai	varchar(20)	Yes	Điện thoại
email	varchar(50)	Yes	Email
fax	varchar(20)	Yes	Fax
hinh_anh	varchar(255)	Yes	Đường dẫn ảnh

*Bảng 3.35. Bảng loai\_de\_tai*

Field	Type	Null	Description
ma_loai	bigint(11)	No	Mã loại đề tài
ten_loai	varchar(255)	Yes	Tên loại đề tài



Hình 3.8 Các bảng cơ sở dữ liệu cơ bản của Drupal.

Theo như cách tổ chức lưu trữ của Drupal (đã được trình bày ở chương II) và mỗi quan hệ giữa các bảng của Drupal. Cho nên, ta có thể lưu trữ tất cả các nội dung

trang web khoa gồm: de\_tai, giang\_vien, anh\_hoat\_dong, lien\_he, bieu\_mau, thoi\_khoa\_bieu, he, khoa\_hoc, cuu\_sinh\_vien, bang\_vang, nguoi\_dung, cuoc\_thi, file\_dinh\_kem, doi\_tac.

- Các nội dung sẽ được lưu trữ trong hai bảng chính là node và node\_revisions.

node		node_revisions	
PK,I5	<u>nid</u>	PK	<u>vid</u>
U1	vid	FK1,I1	nid
FK1,I5	type	FK2,I2	uid
	language		title
I6	title		body
FK2,I9	uid		teaser
I5,I4	status		log
I2	created		timestamp
I1	changed		format
	comment		
I4	promote		
I3	moderate		
	sticky		
I7	tnid		
I8	translate		

Hình 3.9. Bảng node và node\_revisions trong cơ sở dữ liệu của Drupal.

- Drupal còn hỗ trợ thêm các bảng:

- node\_type: Cho phép lưu trữ các kiểu nội dung khác nhau.
- users: Cho phép lưu trữ thông tin tài khoản của người dùng.
- role: Cho phép lưu trữ các vai trò của người dùng.
- files: Cho phép lưu trữ các thông tin về file được upload.



node_type	
PK	<u>type</u>
	name module description help has_title title_label has_body body_label min_word_count custom modified locked orig_type

users	
PK	<u>uid</u>
U1	name
I3	pass
	mail
	mode
	sort
	threshold
	theme
I2	signature
I1	created
	access
	login
	status
	timezone
	language
	picture
	init
	data

role	
PK	<u>rid</u>
U1	name

files	
PK	<u>fid</u>
FK1,I3	uid
	filename
	filepath
	filemime
	filesize
I1	status
I2	timestamp

Hình 3.10. Các bảng trong cơ sở dữ liệu Drupal.

Kết luận: Drupal hoàn toàn có khả năng lưu trữ cơ sở dữ liệu và đáp ứng nhu cầu xây dựng thông tin website khoa.

# CHƯƠNG IV: ỨNG DỤNG DRUPAL – XÂY DỰNG WEBSITE

## KHOA CÔNG NGHỆ THÔNG TIN

### I. Cài đặt

#### *1.1 Yêu cầu phần mềm*

- Máy chủ web: WampServer2.0.
- Phần mềm quản trị nội dung Drupal: drupal 5.14.
- Bản việt hóa cho Drupal: 5.x – 1.0 tại trang <http://drupal.org>.
- Trình soạn thảo PHP như Macromedia Dreamweaver 8.

#### *1.2 Tiến hành cài đặt*

- Cài đặt Wamp: Chạy tập tin WampServer2.0c.exe.
- Giải nén tập tin drupal-5.14.tar.gz. Chép tất cả các tập tin trong drupal-5.14 vào trong thư mục C:\wamp\www.

Dùng trình duyệt web truy cập vào địa chỉ [http://localhost/tên\\_thư\\_mục\\_chứa\\_web](http://localhost/tên_thư_mục_chứa_web) để kích hoạt chương trình cài đặt tự động. Hoặc nhấp vào biểu tượng Wamp ở góc dưới màn hình chọn localhost. Gồm các thông số quan trọng sau:

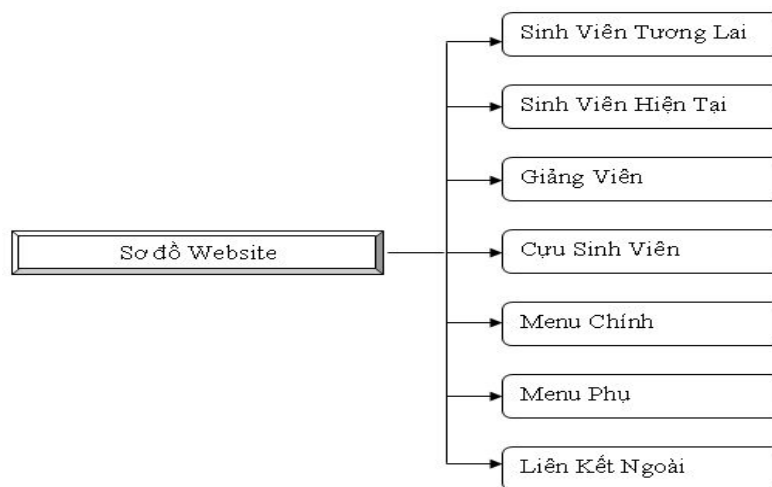
- Database type gồm có mysql và mysqli.
- Database name là tên cơ sở dữ liệu kết nối.
- Database username: tài khoản người dùng trong cơ sở dữ liệu (thường là root).
- Database password là mật khẩu tương ứng của username.
- Database host là tên miền Server chứa cơ sở dữ liệu.
- Site name là tiêu đề của trang web đi kèm với logo.
- Site e-mail address là địa chỉ email.
- Username là tên đăng nhập có quyền hạn là admin.
- Password là mật khẩu của admin.

- E-mail address là địa chỉ email.
- Việt hóa Drupal. Đăng nhập vào tài khoản admin. Bật module Locale tại `admin/build/modules` sau đó tiến hành như sau:
  - Vào `admin/build/module/TransLate Interface` và chọn *Import*.
  - Chọn *vietnamese* trong *import into*, chỉ đường dẫn đến gói chứa tập tin tiếng việt download về ở trên.
  - Vào `admin/configuration/languages` đánh dấu vào mục *Default* là *vietnamese*.

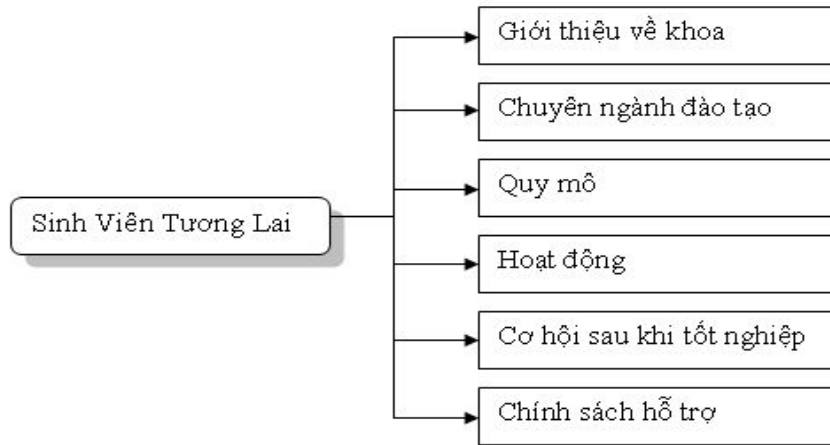
## II. Các sơ đồ

Căn cứ vào những phân tích và đặc tả ở chương III, ta có sơ đồ tổng quát của toàn bộ ứng dụng như sau:

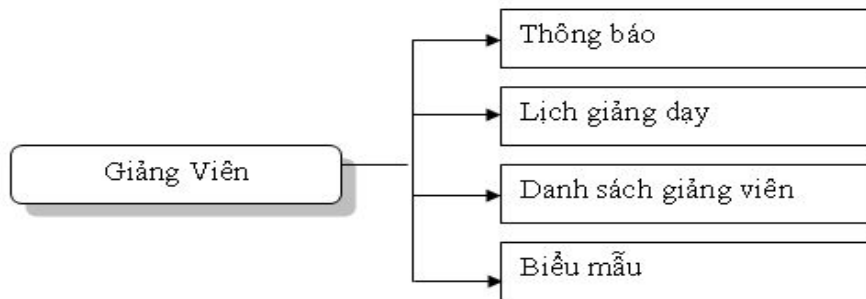
### II.1 Sơ đồ Website



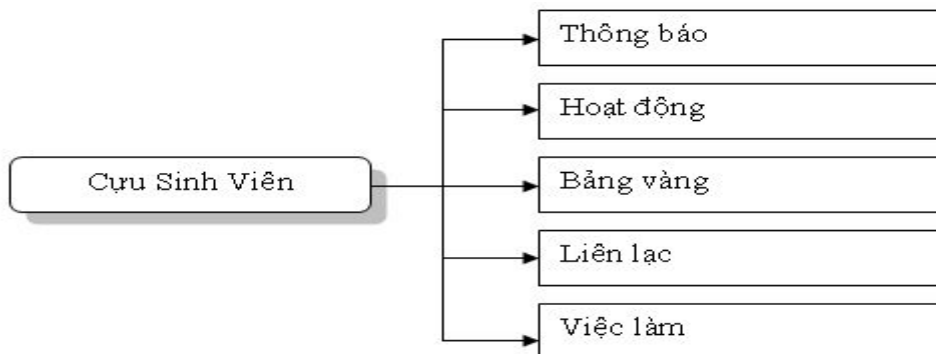
Hình 4.1. Sơ đồ mô tả chức năng tổng quát.



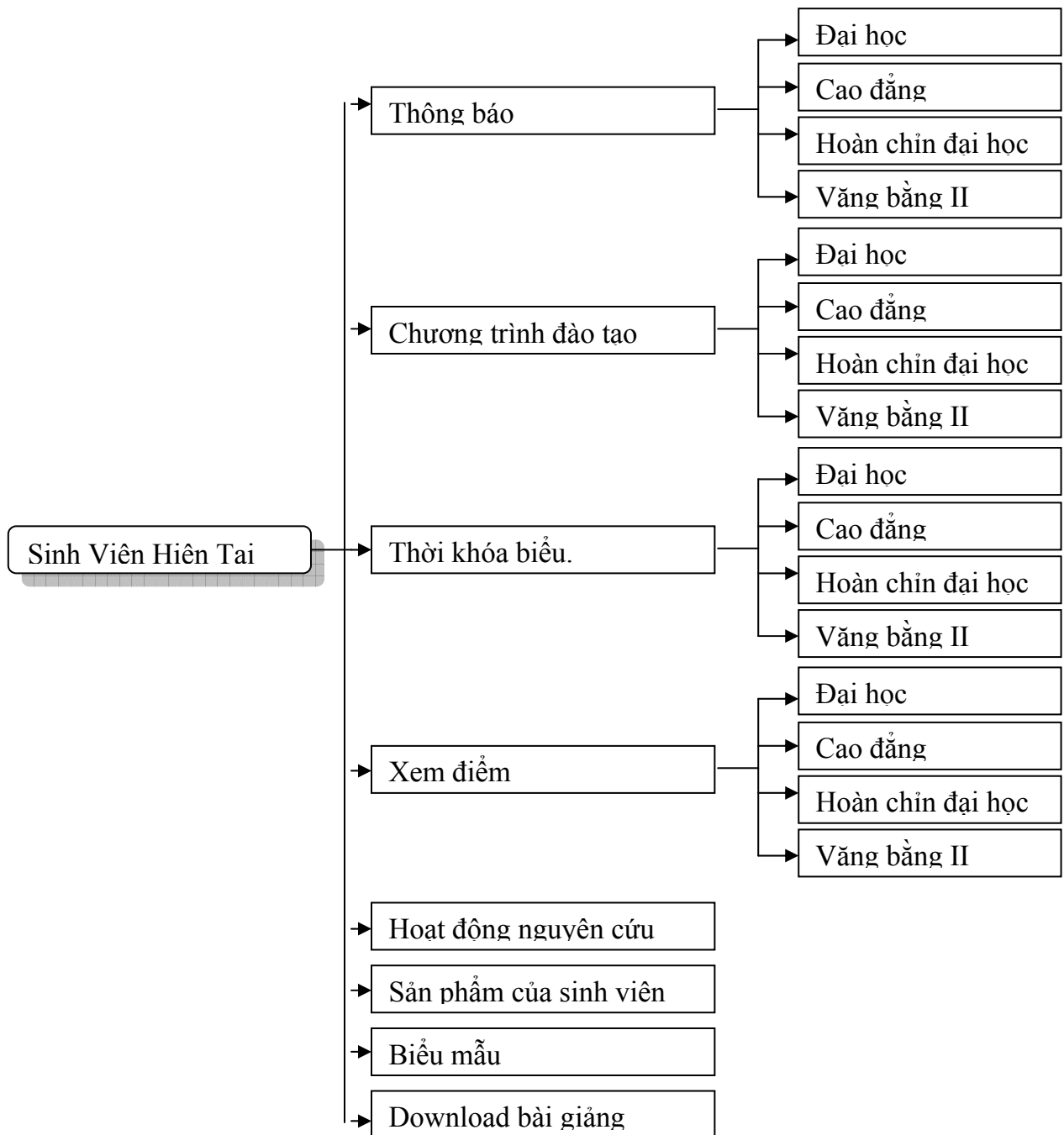
Hình 4.2. Sơ đồ của mô tả chức năng của Sinh Viên Tương Lai.



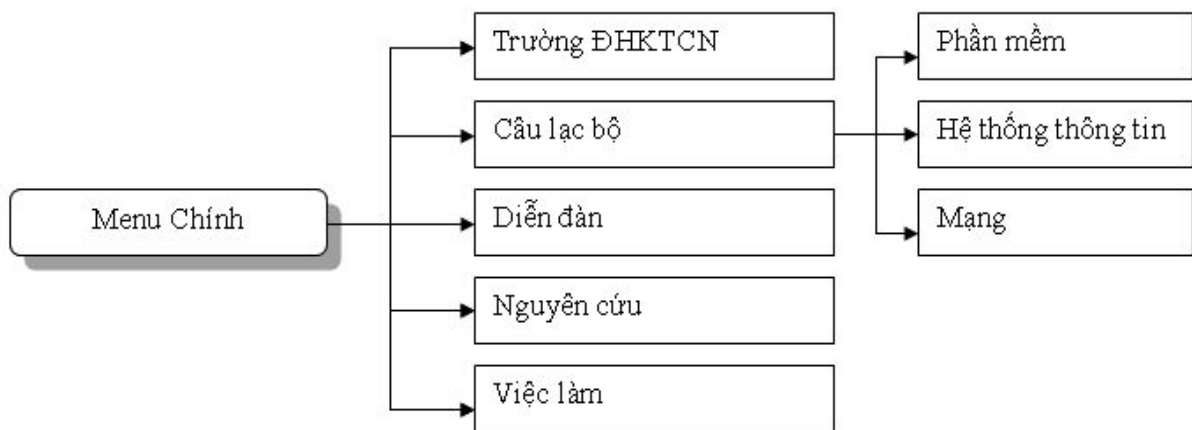
Hình 4.4. Sơ đồ mô tả chức năng của Giảng Viên.



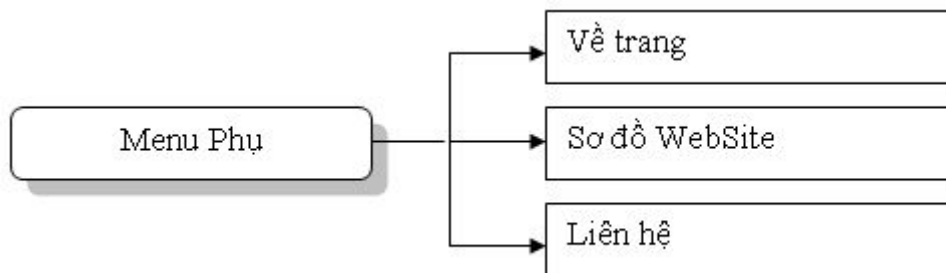
Hình 4.5. Sơ đồ mô tả chức năng của Cựu Sinh Viên.



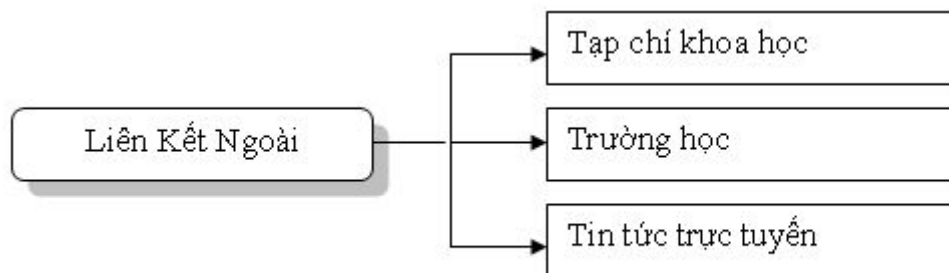
Hình 4.3 Sơ đồ mô tả chức năng của Sinh Viên Hiện Tại.



Hình 4.6. Sơ đồ mô tả chức năng Menu Chính.

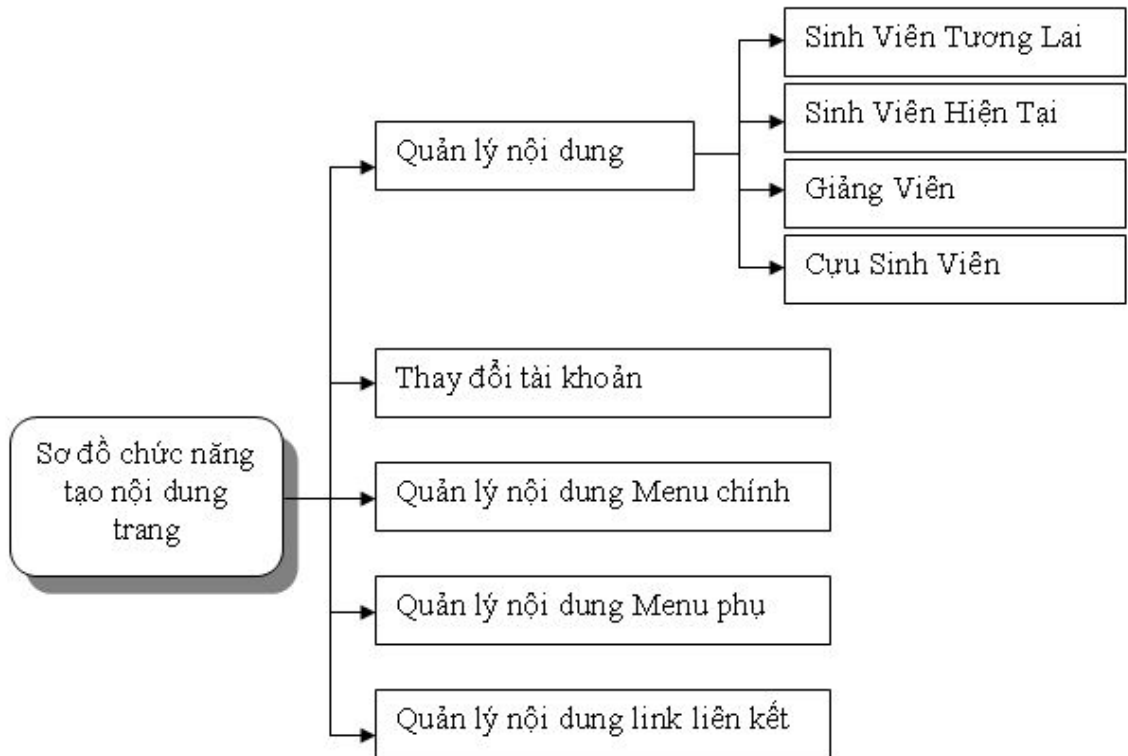


Hình 4.7. Sơ đồ mô tả chức năng Menu Phụ.



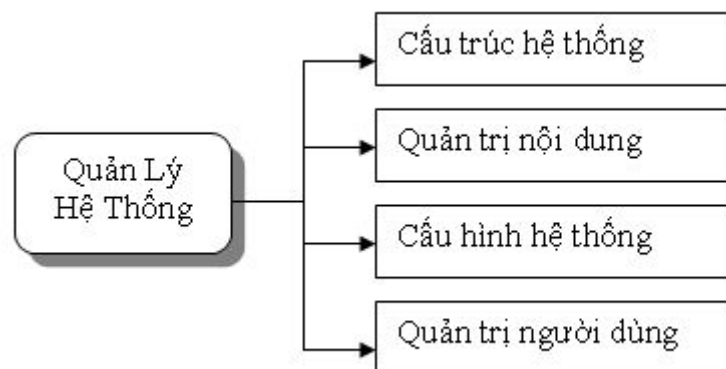
Hình 4.8. Sơ đồ mô tả chức năng Liên Kết Ngoài.

## II.2 Sơ đồ chức năng tạo nội dung trang



Hình 4.9. Sơ đồ chức năng tạo chức năng trang.

- Sơ đồ của người quản trị (admin)



Hình 4.10. Sơ đồ của người quản trị.

## III. Xây dựng

### III.1 Tạo giao diện theo phong cách riêng

Những giao diện chính:

### III.1.1 Trang chủ



**KHOA CÔNG NGHỆ THÔNG TIN**  
**ĐẠI HỌC KỸ THUẬT CÔNG NGHỆ TP.HCM**

Về trang | Sơ đồ website | Liên hệ

#### GIỚI THIỆU

Khoa công nghệ thông tin có chức năng đào tạo kỹ sư công nghệ thông tin có phẩm chất đạo đức và kỹ năng giao tiếp tốt, thông thạo ngoại ngữ, tinh thần làm việc tập thể cao, có năng lực tự tạo việc làm cho mình và cho người khác.



#### Sinh viên tương lai

- › Giới thiệu Khoa
- › Quy mô của khoa
- › Hoạt động của khoa



#### Sinh viên hiện tại

- › Thông báo
- › Chương trình đào tạo
- › Thời khóa biểu



#### Giảng viên

- › Thông báo
- › Lịch giảng
- › Danh sách giảng viên



#### Cựu sinh viên

- › Thông báo
- › Hoạt động
- › Bảng vàng

---

Khoa Công Nghệ Thông Tin - Đại Học Kỹ Thuật Công Nghệ - [Đăng nhập](#)  
Địa chỉ : 144/24 Điện Biên Phủ - Phường 25 - Quận Bình Thạnh - TP.HCM  
Điện thoại : 84-08-5120791 - Fax : 84-08-5120786  
E-mail : admin@it.hutech.edu.vn

*Hình 4.11. Giao diện trang chủ.*



### III.1.2 Sinh viên tương lai



**KHOA CÔNG NGHỆ THÔNG TIN**  
ĐẠI HỌC KỸ THUẬT CÔNG NGHỆ TP.HCM

Về trang | Sơ đồ website | Liên hệ

<b>TRANG CHỦ</b>	<b>TRƯỜNG ĐHKTCN</b>	<b>SINH VIÊN TƯƠNG LAI</b>	<b>SINH VIÊN HIỆN TẠI</b>	<b>GIẢNG VIÊN</b>	<b>CỤM SINH VIÊN</b>
------------------	----------------------	----------------------------	---------------------------	-------------------	----------------------

Trang chủ

Sinh viên tương lai

Giới thiệu khoa công nghệ thông tin

- Giới thiệu khoa**
- Chuyên ngành đào tạo
- Quy mô của khoa
- Hoạt động của khoa
- Cơ hội sau khi tốt nghiệp
- Chính sách hỗ trợ

Khoa CNTT có chức năng đào tạo Kỹ sư Công nghệ Thông tin.

Có phẩm chất đạo đức và kỹ năng giao tiếp tốt, thông thạo Anh ngữ, tinh thần làm việc tập thể cao, có năng lực tự tạo việc làm cho mình và cho những người khác.

Có kiến thức CNTT vững vàng, kỹ năng thực hành cao, và biết vận dụng các phương pháp khoa học hiện đại có thể nhanh chóng và tham gia nghiên cứu những kỹ thuật và công nghệ mới nhất về CNTT đang phát triển; điều khiển các cơ sở tin học, các trung tâm tính toán.

Trang bị các kỹ năng tổ chức thiết kế các phần mềm cho nhu cầu khoa học, quản lý, kinh tế, xã hội hay các phần mềm ứng dụng trong công nghệ điều khiển các thiết bị kỹ thuật, các hệ thống mạng máy tính viễn thông, cũng như xây dựng các môi trường phát triển phần mềm.

Liên kết ngoài

- Tạp chí khoa học
- Trường học
- Tin tức trực tuyến

[Giới thiệu khoa](#)

Đăng nhập

Tên đăng nhập: \*

Mật khẩu: \*

Quên mật khẩu

Khoa Công Nghệ Thông Tin - Đại Học Kỹ Thuật Công Nghệ  
Địa chỉ : 144/24 Điện Biên Phủ - Phường 25 - Quận Bình Thạnh - TP.HCM  
Điện thoại : 84-08-5120791 - Fax : 84-08-5120786  
E-mail : admin@it.hutech.edu.vn

Hình 4.12. Giao diện trang Sinh Viên Tương Lai.

### III.1.3 Sinh viên hiện tại



**KHOA CÔNG NGHỆ THÔNG TIN**  
ĐẠI HỌC KỸ THUẬT CÔNG NGHỆ TP.HCM

Về trang | Sơ đồ website | Liên hệ

TRANG CHỦ	TRƯỜNG ĐHKTCN	SINH VIÊN TƯƠNG LAI	SINH VIÊN HIỆN TẠI	GIẢNG VIÊN	CỰU SINH VIÊN
-----------	---------------	---------------------	--------------------	------------	---------------

Trang chủ > Thông báo

Sinh viên hiện tại

▼ Thông báo

- **Đại học**
- Cao đẳng
- Văn bằng II
- Hoàn chỉnh đại học
- ▷ Chương trình đào tạo
- ▷ Thời khóa biểu
- ▷ Xem điểm
- Hoạt động nghiên cứu
- Sản phẩm của sinh viên
- Biểu mẫu
- Download bài giảng

Liên kết ngoài

- Tạp chí khoa học
- Trường học
- Tin tức trực tuyến

Đăng nhập

Tên đăng nhập: \*

Mật khẩu: \*

- Quên mật khẩu

Đại học

Thời khóa biểu lớp 07HTHH2 - từ 05/01/09

Thứ 2 : Khai khoáng dữ liệu  
Thứ 3 và thứ 7 : Thực hành cấu trúc máy tính , P. máy lâu 3.  
Tối thứ 4 : Cấu trúc máy tính  
Tối thứ 5 : Thương mại điện tử, P.201  
Tối thứ 6 : Quản trị dự án, P.201

Đại học

Thi môn vi xử lý - lớp 07HTHM2

Thi ngày 03/01/09, nền thứ 6 , 02/01/09 học bình thường.

Đại học

Bảo vệ đồ án cơ sở do Thầy Lê Mạnh Hải hướng dẫn

Các bạn K05 làm đồ án cơ sở do Thầy Lê Mạnh Hải hướng dẫn báo cáo vào sáng thứ 4 ngày 7/1/2009. Bạn nào không thu xếp được vào buổi trên liên hệ với thầy để chọn ngày khác-dự kiến chiều thứ 6 ngày 9/1/2009.

Đại học

Việc nộp luận văn lớp 04DTH - Chính quy

Ngày nộp luận văn : 05/01/09. Chỉ nộp trong ngày này, nếu trễ khoa không chịu trách nhiệm.  
Sinh viên làm theo mẫu trên mục qui định biểu mẫu- trang web khoa.  
1 đề tài làm 3 quyển : 1 quyển gửi giáo viên hướng dẫn, 1 quyển gửi giáo viên phản biện,  
1 quyển nộp VPK để cho các thầy trong hội đồng bảo vệ tham khảo, sau ngày 15/01/09 lên khoa nhận về.  
Ngày bảo vệ : 15/01/09

Đại học



Khoa Công Nghệ Thông Tin - Đại Học Kỹ Thuật Công Nghệ  
Địa chỉ : 144/24 Điện Biên Phủ - Phường 25 - Quận Bình Thạnh - TP.HCM  
Điện thoại : 84-08-5120791 - Fax : 84-08-5120786  
E-mail : admin@it.hutech.edu.vn

Hình 4.13. Giao diện trang Sinh Viên Hiện Tại.

### III.1.4 Giảng viên

**KHOA CÔNG NGHỆ THÔNG TIN**  
**ĐẠI HỌC KỸ THUẬT CÔNG NGHỆ TP.HCM**

Về trang | Sơ đồ website | Liên hệ

**TRANG CHỦ** | **TRƯỜNG ĐHKTCN** | **SINH VIÊN TƯƠNG LAI** | **SINH VIÊN HIỆN TẠI** | **GIẢNG VIÊN** | **CỰU SINH VIÊN**

Trang chủ  
Giảng viên  
Danh sách giảng viên

- Thông báo
- Lịch giảng
- **Danh sách giảng viên**
- Biểu mẫu

Liên kết ngoài

- Tạp chí khoa học
- Trường học
- Tin tức trực tuyến

Đăng nhập

**Tên đăng nhập: \***

**Mật khẩu: \***

Quên mật khẩu

---

Khoa Công Nghệ Thông Tin - Đại Học Kỹ Thuật Công Nghệ  
Địa chỉ : 144/24 Điện Biên Phủ - Phường 25 - Quận Bình Thạnh - TP.HCM  
Điện thoại : 84-08-5120791 - Fax : 84-08-5120786  
E-mail : admin@it.hutech.edu.vn

Hình 4.14. Giao diện trang Giảng Viên.

### III.1.5 Cựu sinh viên



TRANG CHỦ	TRƯỜNG ĐHKTCN	SINH VIÊN TƯƠNG LAI	SINH VIÊN HIỆN TẠI	GIẢNG VIÊN	CỰU SINH VIÊN
Trang chủ					
Cựu sinh viên			Việc làm		
<input type="radio"/> Thông báo			Tuyển lập trình viên Java		
<input type="radio"/> Hoạt động			Chủ Nhật, 01/04/2009 - 18:20 — hoang		
<input type="radio"/> Bảng vàng				THÔNG TIN NHÀ TUYỂN DỤNG	
<input type="radio"/> Liên lạc					
<input checked="" type="radio"/> Việc làm					
Liên kết ngoài					
<input type="radio"/> Tạp chí khoa học			Tỉnh / Thành:	Hà Nội	
<input type="radio"/> Trường học			Địa chỉ:	Tầng 8, Khách sạn Thể thao, Làng Sinh viên Hacinco, Thanh Xuân, Hà Nội	
<input type="radio"/> Tin tức trực tuyến			Loại hình:	Doanh nghiệp tư nhân (TNHH, CP)	
			Thuộc về:	Tư nhân	
			Số nhân viên:	100 - 500	
Đăng nhập			Giới thiệu công ty:	Công ty cổ phần Công nghệ Tinh Vân là một công ty tập trung vào việc phát triển các giải pháp phần mềm lớn và chuyên nghiệp, đã đạt nhiều giải thưởng có uy tín như: cúp vàng CNTT truyền thông, giải Sao Khuê ... Qua hơn 10 năm hình thành và phát triển, Tinh Vân đã khẳng định được vị thế của mình với hệ thống khách hàng là các Bộ, ngành, địa phương, các tổng công ty, các trường Đại học lớn trên toàn quốc. Bên cạnh đó, chúng tôi luôn có ý thức xây dựng một môi trường làm việc chuyên nghiệp, điều kiện vật chất khang trang và văn hóa đặc sắc. Đội ngũ cán bộ của Tinh Vân được xây dựng bài bản, luôn có những ưu đãi xứng đáng, có nhiều cơ hội thăng tiến, được tạo điều kiện nâng cao chuyên môn và luôn được khuyến khích tìm tòi, sáng tạo trong công việc ...	
Tên đăng nhập: *			Website:	www.tinhvan.com, www.xalo.vn	
<input type="text"/>			Tên người liên hệ:	Ms. Trần Thị Thu Hà	
Mật khẩu: *			Số Điện thoại:	5589970-121	
<input type="text"/>			Fax:		
<input type="button" value="Đăng nhập"/>			Email:		
<input type="radio"/> Quên mật khẩu					Việc làm

Hình 4.15. Giao diện trang Cựu Sinh Viên.

### III.2 Module thêm vào

- FCKEditor: Giúp cho thao tác nhập liệu dễ dàng, nhanh chóng, tương tự như đang thao tác trên môi trường word của windows mà không đòi hỏi phải có kiến thức về HTML.
- IMCE: Giúp cho việc upload tập tin và liên kết đến tập tin hình ảnh trên Server.

### III.3 Danh sách người dùng

Tên đăng nhập	Mật khẩu	Vai trò
QuanLyDD	123456789	Admin_Forum
VanThienHoang	123456789	GiangVien
CoThuy	123456789	ThuKy
QuanLyAdmin	123456789	Admin
QuanLyCLBM	123456789	Admin_CLB_M
QuanLyCLBHTTT	123456789	Admin_CLB_HTTT
QuanLyCLBPM	123456789	Admin_CLB_PM

Bảng 4.1 Danh sách địa chỉ người dùng.

## IV. Tổng kết

- *Kết quả đạt được*
  - Hiểu được lợi ích của phần nguồn phần mềm mở.
  - Những quy định của một hệ quản trị nội dung.
  - Hiểu được tổng quan của Drupal.
  - Phát triển và mở rộng Drupal.
  - Phân tích và thiết kế hệ thống tin của khoa.
  - Ứng dụng Drupal vào Webstie khoa công nghệ thông tin.
- Chức năng chưa hoàn thành
  - Diễn đàn vẫn chưa hoàn thành. Sẽ làm tiếp cho đến ngày bảo vệ.

## Tài liệu tham khảo

*Từ các trang web:*

<http://groups.drupal.org/node/10683>: Hướng dẫn xây dựng Drupal.

<http://store.lullabot.com/products/understanding-drupal>: Hiểu biết về Drupal.

<http://vnkit.com/pa/web/content.php?cid=174>: Cung cấp Modules, Themes.

<http://groups.drupal.org/taxonomy/term/2440?page=1>: Một số vấn đề phát triển Drupal.

[http://dev.vnchannel.net/vi/drupal\\_installation\\_instruction\\_notes](http://dev.vnchannel.net/vi/drupal_installation_instruction_notes): Cách cài đặt module, theme.

<http://www.drupaldude.com/what-node-drupal>: Video về Drupal.

<http://dnews.liva.com.vn/article/xay-dung-he-thong-menu-tren-drupal>: Xây dựng hệ thống menu.  
<http://api.drupal.org/api/group/menu/7>: API của Drupal.

<http://drupal.org/node/313355>: Drupal có thể làm được mọi thứ.

<http://tips.webdesign10.com/drupal-taxonomy-tutorial>: Cách làm việc của taxonomy.

<http://api.drupal.org/api/group/hooks/6>: Vai trò của hook trong Drupal Core.

<http://www.cmsmatrix.org/matrix/cms-matrix>: So sách các hệ CMS.

<http://custompcsoftware.com/blog/2006/11/20/3/>: So sánh Drupal và Joomla.

<http://codeprovn.com/forums/viewtopic.php?f=36&t=888>: Cách tạo kiểu nội dung.

<http://drupal.org/writing-secure-code>: Tính an toàn của Drupal.

<http://drupal.org/node/101723>: Cách làm việc của CCK.

Sách tham khảo:

[1] John K. VanDyk và Matt Westgate.(2007).Pro Drupal Development.