

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2024-2025

# TÌM HIỂU NGÔN NGỮ XML VÀ XÂY DỰNG ỨNG DỤNG WEB QUẢN LÝ KẾT QUẢ HỌC TẬP SINH VIÊN

*Giáo viên hướng dẫn*  
ThS. Nguyễn Ngọc Đan Thanh

*Sinh viên thực hiện:*  
Họ tên: Ngô Huỳnh Quốc Khang  
Mã số sinh viên: 110122092  
Lớp: DA22TTD

*Trà Vinh, Tháng 01 Năm 2025*

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2024-2025

# **TÌM HIỂU NGÔN NGỮ XML VÀ XÂY DỰNG ỨNG DỤNG WEB QUẢN LÝ KẾT QUẢ HỌC TẬP SINH VIÊN**

*Giáo viên hướng dẫn*  
ThS. Nguyễn Ngọc Đan Thanh

*Sinh viên thực hiện:*  
Họ tên: Ngô Huỳnh Quốc Khang  
Mã số sinh viên: 110122092  
Lớp: DA22TTD

*Trà Vinh, Tháng 01 Năm 2025*

# Tìm hiểu ngôn ngữ XML và xây dựng ứng dụng web quản lý kết quả học tập sinh viên

### NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

[illegible]

Trà Vinh, ngày ... tháng ... năm 2024

**Giáo viên hướng dẫn**

(Ký và ghi rõ họ tên)

# Tìm hiểu ngôn ngữ XML và xây dựng ứng dụng web quản lý kết quả học tập sinh viên

## NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

Trà Vinh, ngày ... tháng ... năm 2025

## Thành viên hội đồng

(Ký và ghi rõ họ tên)

## **LỜI CẢM ƠN**

Đầu tiên, em xin gửi lời cảm ơn chân thành đến các thầy cô, những người đã hết mực chỉ bảo, hướng dẫn và sẵn sàng giải đáp thắc mắc, cung cấp những kiến thức hữu ích giúp em hoàn thành công việc trong suốt quá trình thực hiện báo cáo này.

Mặc dù đã cố gắng để thực hiện đề tài một cách nghiêm túc và tập trung trong suốt quá trình thu thập, tìm hiểu, phân tích và tổng hợp nhiều nguồn tài liệu khác nhau, song do sự hiểu biết còn chưa cao, trình độ chuyên môn còn hạn chế ở nhiều mặt và chưa có nhiều kỹ năng trong việc thực hiện các công việc trong chuyên ngành nên trong quá trình biên soạn đồ án không thể tránh khỏi các thiết sót. Do đó nếu như có những điểm trong báo cáo còn chưa hợp lý hoặc sai sót em rất mong được các thầy cô đóng góp thêm các ý kiến để có thể học hỏi và tiếp thu thêm các kiến thức còn thiếu sót.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

**Ngô Huỳnh Quốc Khang**

## MỤC LỤC

CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU.....	1
1.1. Giới thiệu đề tài.....	1
1.2. Mục đích nghiên cứu.....	1
1.3. Đối tượng nghiên cứu .....	1
1.4. Phạm vi nghiên cứu.....	1
1.5. Phương pháp nghiên cứu.....	1
CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT .....	2
2.1. Tổng quan thiết kế web.....	2
2.1.1 Web tĩnh.....	2
2.1.2 Web động .....	2
2.2. Kiến trúc ứng dụng web.....	3
2.2.1 Frontend .....	3
2.2.2 Backend.....	3
2.2.3 Database .....	3
2.3. Ngôn ngữ HTML .....	3
2.3.1 Tổng quan về ngôn ngữ HTML .....	3
2.3.2 Đặc điểm .....	4
2.3.3 Cấu trúc .....	4
2.3.4 Kiến thức cơ bản về HTML.....	5
2.3.4.1 Các phân tử tiêu đề(<h1 - h6>) .....	5
2.3.4.2 Phân tử hình ảnh(<img>) .....	6
2.3.4.3 Phân tử đoạn văn(<p>).....	7
2.3.4.4 Phân tử liên kết(<a>).....	7
2.3.4.5 Phân tử danh sách <ul> .....	8

2.3.4.6 Phần tử phân chia(<div>)	9
2.3.4.7 Phần tử biểu mẫu(<form>)	10
2.3.4.8 Phần tử bảng(<table>)	10
2.4. Ngôn ngữ CSS	12
2.4.1 Các kiểu Selector trong CSS	13
2.4.2 Cách chèn CSS	13
2.4.2.1 External CSS	13
2.4.2.2 Internal CSS	14
2.4.2.3 Inline CSS	16
2.5. Ngôn ngữ JavaScript	17
2.5.1 Tổng quan về JavaScript	17
2.5.2 Cách hoạt động và đặc điểm của JavaScript	17
2.5.3 Mối liên hệ giữa JavaScript, CSS và HTML	17
2.6. Ngôn ngữ XML	18
2.6.1 Đặc điểm ngôn ngữ XML	18
2.6.2 Cấu trúc một tài liệu XML	18
2.7. Ngôn ngữ PHP	21
2.7.1 Ứng dụng phổ biến của PHP	21
2.7.2 Kiến thức cơ bản về PHP	22
2.7.2.1 Mở và đóng PHP	22
2.7.2.2 Biến	22
2.7.2.3 Hằng số	22
2.7.2.4 Câu lệnh điều kiện	23
2.7.2.5 Vòng lặp	24
2.7.2.6 Hàm	25

2.7.2.7 Mảng .....	26
2.7.2.8 Xử lý chuỗi.....	26
2.8. Tương tác dữ liệu với PHP và XML .....	26
2.8.1 SimpleXML .....	26
2.8.2 XML DOM.....	27
2.9. Bootstrap framework.....	27
2.9.1 Tổng quan.....	27
2.9.2 Cài đặt .....	29
CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU .....	30
3.1. Mô tả bài toán.....	30
3.2. Phân tích thiết kế hệ thống .....	30
3.2.1 Đặc tả yêu cầu hệ thống .....	30
3.2.1.1 Yêu cầu chức năng .....	30
3.2.1.2 Yêu cầu phi chức năng.....	31
3.2.2 Tổ chức lưu trữ dữ liệu .....	31
3.2.2.1 Tập tin lop.xml .....	32
3.2.2.2 Tập tin sinhVien.xml.....	33
3.2.2.3 Tập tin giangvien.xml.....	34
3.2.2.4 Tập tin hocky.xml .....	35
3.2.2.5 Tập tin monhoc.xml .....	36
3.2.2.6 Tập tin phancong.xml .....	37
3.2.2.7 Tập tin ketquahoctap.xml.....	40
3.2.3 Thiết kế xử lý .....	41
3.2.3.1 Mô hình DFD mức ngữ cảnh .....	41
3.2.3.2 Mô hình DFD mức 1 .....	42



3.2.3.3 Mô hình DFD mức 2 .....	43
3.2.4 Thiết kế giao diện.....	44
3.2.4.1 Sơ đồ website .....	44
3.2.4.2 Giao diện đăng nhập .....	44
3.2.4.3 Giao diện trang chủ của sinh viên.....	45
3.2.4.4 Giao diện trang chủ của giảng viên.....	46
3.2.4.5 Giao diện trang quản trị viên.....	47
3.2.4.6 Giao diện trang quản lý lớp học .....	48
3.2.4.7 Giao diện trang quản lý sinh viên .....	48
3.2.4.8 Giao diện trang quản lý học kỳ .....	49
3.2.4.9 Giao diện trang quản lý giảng viên .....	50
3.2.4.10 Giao diện quản ký môn học .....	50
3.2.4.11 Giao diện phân công giảng dạy .....	51
3.2.4.12 Giao diện chức năng sửa thông tin.....	52
CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU .....	53
4.1. Dữ liệu thử nghiệm .....	53
4.2. Kết quả thực nghiệm .....	54
4.2.1 Chức năng đăng nhập.....	54
4.2.2 Quản lý sinh viên .....	55
4.2.3 Quản lý giảng viên .....	56
4.2.4 Quản lý môn học .....	56
4.2.5 Quản lý lớp học .....	56
4.2.6 Quản lý học kỳ .....	57
4.2.7 Phân công giảng dạy .....	57
4.2.8 Đăng ký môn học .....	57

4.2.9 Quản lý kết quả học tập của sinh viên .....	58
4.2.10 Trang chủ .....	58
CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	60
5.1. Kết luận .....	60
5.2. Hướng phát triển .....	60

## **DANH MỤC HÌNH ẢNH – BẢNG BIỂU**

Hình 2.1 Cấu trúc của một Selector .....	13
Hình 2.3 Cấu trúc cây của tài liệu XML .....	19
Hình 3.1 Sơ đồ hệ thống lưu trữ dưới dạng các tập tin XML .....	32
Hình 3.2 Mô hình DFD mức ngữ cảnh .....	42
Hình 3.3 Mô hình DFD mức 1 .....	42
Hình 3.4 Mô hình DFD mức 2 .....	43
Hình 3.5 Giao diện đăng nhập.....	45
Hình 3.6 Giao diện trang chủ của sinh viên .....	46
Hình 3.7 Giao diện trang chủ của giảng viên.....	47
Hình 3.8 Giao diện trang chủ của quản trị viên .....	47
Hình 3.9 Giao diện quản lý lớp học .....	48
Hình 3.10 Giao diện quản lý sinh viên.....	49
Hình 3.11 Giao diện quản lý học kỳ .....	49
Hình 3.12 Giao diện quản lý giảng viên .....	50
Hình 3.13 Giao diện quản lý môn học .....	51
Hình 3.14 Giao diện phân công giảng dạy .....	52
Hình 3.15 Giao diện chức năng sửa thông tin.....	52
Hình 4.1 Trang đăng nhập.....	55
Hình 4.2 Trang quản lý sinh viên.....	55
Hình 4.3 Trang quản lý giảng viên.....	56
Hình 4.4 Trang quản lý môn học .....	56
Hình 4.5 Trang quản lý lớp học .....	56
Hình 4.6 Trang quản lý học kỳ.....	57
Hình 4.7 Trang phân công giảng dạy .....	57

Hình 4.8 Trang đăng ký môn học.....	58
Hình 4.9 Trang quản lý kết quả học tập của sinh viên.....	58
Hình 4.10 Trang chủ .....	59
Hình 4.11 Trang hiển thị kết quả học tập của sinh viên.....	59
Bảng 3.1 Bảng dữ liệu tập tin lop.xml .....	32
Bảng 3.2 Bảng dữ liệu tập tin sinhVien.xml .....	33
Bảng 3.3 Bảng dữ liệu của tập tin giangvien.xml .....	34
Bảng 3.4 Bảng dữ liệu của tập tin hocky.xml .....	35
Bảng 3.5 Bảng dữ liệu của tập tin monhoc.xml.....	36
Bảng 3.6 Bảng dữ liệu của tập tin phancong.xml .....	38
Bảng 4.1 Bảng dữ liệu của trang quản lý sinh viên .....	53
Bảng 4.2 Bảng dữ liệu của trang quản lý giảng viên .....	53
Bảng 4.3 Bảng dữ liệu của trang quản lý lớp học .....	53
Bảng 4.4 Bảng dữ liệu của trang quản lý học kỳ .....	54
Bảng 4.5 Bảng dữ liệu của trang quản lý môn học .....	54
Bảng 4.6 Bảng dữ liệu của trang phân công giảng dạy.....	54

## CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU

### 1.1. Giới thiệu đề tài

Hiện nay, trước tình hình số lượng sinh viên trên cả nước đang ngày một gia tăng qua từng năm, đi kèm với đó là một khối lượng công việc khổng lồ đối với các Trường Đại học trong việc xử lý và lưu trữ các dữ liệu về điểm thi của sinh viên, đặc biệt là đối với quy mô của Trường Đại học Trà Vinh trong việc quản lý kết quả học tập của sinh viên.

### 1.2. Mục đích nghiên cứu

Vấn đề chính mà nguyên cứu này sẽ tập trung giải quyết là xây dựng một hệ thống website giúp cán bộ quản lý lưu trữ kết quả học tập của sinh viên và lập báo cáo thống kê kết quả học tập của sinh viên, đồng thời tích lũy kiến thức về cách thức triển khai một ứng dụng web với PHP và quản lý dữ liệu với XML.

### 1.3. Đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài được xác định bao gồm: Sinh viên, giảng viên, nghiệp vụ quản lý sinh viên của trường Đại học Trà Vinh, hệ thống website quản lý sinh viên của Trường Đại học trà vinh.

### 1.4. Phạm vi nghiên cứu

Phạm vi nghiên cứu của đề tài sẽ được giới hạn trong việc xây dựng một website quản lý kết quả học tập của sinh viên của trường Đại học Trà Vinh.

### 1.5. Phương pháp nghiên cứu

Tìm hiểu, tham khảo các tài liệu tổng quan về đặc điểm, cấu trúc và vai trò các ngôn ngữ HTML, CSS, JAVASCRIPT.

Tìm hiểu khái niệm, định nghĩa, chức năng cách thức hoạt động và sự khác nhau giữa 2 nhánh phát triển web **Front-end** và **Back-end**.

Tìm hiểu về cách sử dụng, chức năng và cấu trúc của một tài liệu XML. Cách lưu trữ, mã hóa và vận chuyển dữ liệu thông qua XML.

## **CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT**

### **2.1. Tổng quan thiết kế web**

Thiết kế web là quá trình tạo ra giao diện, cấu trúc và chức năng cho một website, làm cho website đẹp mắt, dễ thao tác và có tốc độ hiệu quả sao cho người dùng dễ dàng sử dụng và tìm thấy thông tin một cách nhanh chóng.

Thiết kế website được chia thành hai dạng là website dạng tĩnh và website dạng động. Tùy vào nhu cầu sử dụng và ngân sách của mỗi bên khác nhau mà sẽ chọn dạng web phù hợp.

#### **2.1.1 Web tĩnh**

Những trang web tĩnh sẽ sử dụng các đoạn mã HTML, JAVASCRIPT, CSS... Web tĩnh cung cấp cho người dùng tất cả những nội dung văn bản hay các video hình ảnh thu hút. Tuy nhiên, sử dụng web tĩnh chủ website sẽ không thể chỉnh sửa, update được bất cứ nội dung nào.

Ngay cả giao diện của website tĩnh cũng đã được cố định về bố cục, màu sắc. Nếu bạn không có những kiến thức chuyên sâu về lập trình và kinh nghiệm dày dặn về xử lý những thay đổi của web tĩnh thì việc chỉnh sửa loại hình web này gần như khó khả thi. Cách duy nhất bạn có thể cập nhật thông tin trên web tĩnh là xây dựng lại một trang web hoàn toàn mới.

#### **2.1.2 Web động**

Trái ngược hoàn toàn với web tĩnh, thiết kế website dạng động sử dụng những ngôn ngữ lập trình như JAVA, PHP, Python, Ruby. Web động cho phép người dùng có thể linh hoạt trong việc điều chỉnh và thay đổi nội dung trên website.

Ngoài ra, bạn cũng có thể thay đổi giao diện, tùy chỉnh đơn giản về bố cục, màu sắc. hiện nay website động được phần lớn cộng đồng các doanh nghiệp và cá nhân sử dụng nhiều hơn là web tĩnh.

## **2.2. Kiến trúc ứng dụng web**

Kiến trúc ứng dụng web là cách tổ chức và cấu trúc của một ứng dụng web, bao gồm các thành phần, mối quan hệ giữa chúng và cách chúng tương tác với nhau. Mục đích của kiến trúc này là để ứng dụng hoạt động hiệu quả, bảo mật và dễ bảo trì. Một ứng dụng web thường bao gồm ba phần chính bao gồm: Frontend, Backend, Database.

### **2.2.1 Frontend**

Là phần giao diện người dùng, nơi người dùng tương tác trực tiếp với website. Đây là những gì người dùng thấy và sử dụng, ví dụ như các trang web, biểu mẫu, nút bấm, hình ảnh, văn bản, và các hiệu ứng. HTML, CSS, JavaScript và các framework như React, Vue.js, Angular là các công nghệ được sử dụng để xây dựng giao diện bên phía Frontend.

### **2.2.2 Backend**

Là phần xử lý phía máy chủ, chịu trách nhiệm xử lý các yêu cầu từ người dùng và trả về dữ liệu cho Frontend. Nó bao gồm việc quản lý cơ sở dữ liệu, xử lý các tác vụ logic, và cung cấp các API để giao tiếp với frontend. Node.js, PHP, Python (Django, Flask), Ruby on Rails là các công nghệ dùng được sử dụng ở phía Backend.

### **2.2.3 Database**

Lưu trữ tất cả dữ liệu của ứng dụng như tài khoản người dùng, thông tin sản phẩm, lịch sử giao dịch Backend sẽ giao tiếp với cơ sở dữ liệu để lấy hoặc cập nhật thông tin khi người dùng yêu cầu MySQL, PostgreSQL, MongoDB, SQLite là các công nghệ được sử dụng để lưu trữ, xử lý và giao tiếp với Backend.

## **2.3. Ngôn ngữ HTML**

### **2.3.1 Tổng quan về ngôn ngữ HTML**

**HTML** (viết tắt của từ Hyper Text Markup Language , hay còn gọi là "Ngôn ngữ Đánh dấu Siêu văn bản") là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web trên World Wide Web. Nó có thể được trợ giúp bởi các công nghệ như CSS và các ngôn ngữ kịch bản giống như JavaScript [1].

### 2.3.2 Đặc điểm

HTML có cú pháp đơn giản, dễ hiểu, phù hợp cho người mới bắt đầu. HTML có thể được sử dụng trên bất kỳ hệ điều hành nào và hiển thị trên mọi trình duyệt web. Có thể kết hợp với CSS và JavaScript để tạo ra các trang web động.

### 2.3.3 Cấu trúc

Cấu trúc cơ bản của một tài liệu HTML:

```
<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <title>Trang Web Cơ Bản</title>

</head>

<body>

    Đây là một trang web cơ bản với HTML đơn giản.

</body>

</html>
```

Giải thích các phân tử:

**<!DOCTYPE html>**: Khai báo loại tài liệu và phiên bản HTML đang sử dụng.

**<html>**: Phần tử gốc bao quanh toàn bộ nội dung của tài liệu HTML.

**<head>**: Chứa các thông tin meta, tiêu đề trang, và các liên kết đến tài nguyên bên ngoài như CSS.

**<meta charset="UTF-8">**: Đặt mã hóa ký tự cho tài liệu.

**<title>**: Tiêu đề của trang web, hiển thị trên tab của trình duyệt.

**<body>**: Chứa nội dung chính của trang web, bao gồm văn bản, hình ảnh, liên kết, v.v



## 2.3.4 Kiến thức cơ bản về HTML

### 2.3.4.1 Các phân tử tiêu đề(<h1 - h6>)

Tiêu đề được dùng để chỉ ra tiêu đề hoặc đầu mục của các phần nội dung trên trang web. Có sáu cấp độ tiêu đề, từ <h1> đến <h6>, với <h1> là tiêu đề lớn nhất và <h6> là nhỏ nhất, Mỗi cấp có kích thước và độ quan trọng giảm dần, giúp xác định cấu trúc nội dung.

**<h1>**: Là cấp tiêu đề lớn nhất và quan trọng nhất, nên chỉ có một thẻ <h1> trên mỗi trang. Sử dụng cho tiêu đề chính hoặc chủ đề của trang web, ví dụ như tên bài viết, tên trang

**<h2>**: Được dùng cho các tiêu đề phụ của <h1>, như các mục nội dung lớn trong bài viết. Sử dụng <h2> để chia nhỏ và định hướng nội dung, đặc biệt là trong các bài viết dài.

**<h3>**: Thường là tiêu đề phụ của <h2>, dùng để chia nhỏ nội dung trong các phần <h2>. Thích hợp cho các tiêu đề nhỏ hơn hoặc tiêu đề của các đoạn giải thích chi tiết

**<h4>,<h5> và <h6>**: Ít được sử dụng hơn, thường dùng khi cần thêm nhiều lớp phân cấp trong các nội dung rất chi tiết. Các thẻ này chủ yếu là tiêu đề phụ của <h3> và thường có kích thước nhỏ, dùng cho các phần bổ sung hoặc tiêu đề nhỏ nhất.

Ví dụ về các phân tử tiêu đề:

```
<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <title>Trang Web Cơ Bản</title>

</head>

<body>
```

```
<h1>tiêu đề h1</h1>
<h2>tiêu đề h2</h2>
<h3>tiêu đề h3</h3>
<h4>tiêu đề h4</h4>
<h5>tiêu đề h5</h5>
<h5>tiêu đề h5</h5>
</body>
</html>
```

#### 2.3.4.2 Phần tử hình ảnh(<img>)

Phần tử <img> là một thẻ tự đóng (self-closing tag), nghĩa là không cần có thẻ đóng. Các thuộc tính của <img> bao gồm:

**src:** đường dẫn đến hình ảnh, có thể là đường dẫn tương đối (relative path) hoặc đường dẫn tuyệt đối (absolute path).

**alt:** văn bản thay thế mô tả nội dung của hình ảnh. Khi hình ảnh không thể hiển thị, thuộc tính alt sẽ cung cấp thông tin về hình ảnh đó.

Ví dụ về phần tử hình ảnh:

```
<!DOCTYPE html>
<html lang="vi">
<head>
  <meta charset="UTF-8">
  <title>Trang Web Cơ Bản</title>
</head>
<body>
  
</body>
```

```
</html>
```

#### 2.3.4.3 Phần tử đoạn văn(<p>)

Phần tử <p> dùng để tạo một đoạn văn bản trong tài liệu HTML. Mỗi đoạn văn bản mới thường được đặt trong một phần tử <p>. <p> là một phần tử khối (block element), do đó nó chiếm toàn bộ chiều rộng của khung chứa (container), và các đoạn văn được ngắt dòng tự động sau mỗi phần tử <p>.

Ví dụ về các phần tử đoạn văn:

```
<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <title>Trang Web Cơ Bản</title>

</head>

<body>

    <p>đây là một đoạn văn bản</p>

    <p>mỗi đoạn văn bản đặt trong phần tử p sẽ tự động
xuống dòng</p>

</body>

</html>
```

#### 2.3.4.4 Phần tử liên kết(<a>)

Phần tử <a> dùng để tạo các liên kết siêu văn bản (hyperlink) đến tài liệu khác hoặc vị trí khác trong cùng tài liệu.

Thuộc tính href: Là thuộc tính bắt buộc của <a> để xác định đường dẫn liên kết. Giá trị của href có thể là một URL đến một trang web bên ngoài, một tệp tài nguyên hoặc một đường dẫn nội bộ.

Ví dụ về phần tử liên kết có đường dẫn từ trang chủ sang trang đăng nhập:

```
<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <title>Trang chủ</title>

</head>

<body>

    <a href="./dangnhap.html">Liên kết tới trang đăng
nhập</a>

</body>

</html>
```

#### 2.3.4.5 Phần tử danh sách <ul>

Phần tử <ul> được dùng để liệt kê ra các văn bản theo kiểu danh sách bằng các ký hiệu như dấu chấm hoặc hình tròn để liệt kê các mục không có thứ tự đặc biệt.

Phần tử <ul> chứa các phần tử <li> dùng để liệt kê từng mục trong danh sách, mặc định sẽ hiển thị với ký hiệu dấu chấm.

Ví dụ về cấu trúc của một phần tử <ul> chứa các thẻ <li> dùng để liệt kê một danh sách:

```
<ul>

    <li>nội dung thứ nhất</li>

    <li>nội dung thứ hai</li>

    <li>nội dung thứ ba</li>

</ul>
```

#### 2.3.4.6 Phần tử phân chia(<div>)

Phần tử <div> dùng để chia trang thành các vùng riêng biệt, chứa các phần tử khác như văn bản, hình ảnh, liên kết, hoặc các phần tử HTML khác. Phần tử <div> không có kiểu dáng hoặc định dạng mặc định và thường chỉ có vai trò cấu trúc. Để có phong cách cụ thể, cần kết hợp với CSS. Khi được áp dụng CSS, <div> sẽ hoạt động như một hộp chứa cho các phần tử bên trong và có thể được định dạng về kích thước, màu sắc, bố cục, và khoảng cách.

Các phần tử <div> thường dựa vào 2 thuộc tính để có thể áp dụng CSS vào việc định dạng đó là thuộc tính id và thuộc tính class.

**Id:** Xác định một mã định danh duy nhất cho một <div> để truy cập và áp dụng CSS hoặc JavaScript.

**Class:** Cho phép sử dụng CSS để định dạng nhiều phần tử <div> có cùng kiểu dáng.

Ví dụ về phần tử <div>:

```
<!DOCTYPE html>

<html lang="vi">

<head>

    <meta charset="UTF-8">

    <title>Trang chủ</title>

</head>

<body>

    <div id="header">đầu trang</div>

    <div id="nav">

        <a class="link">liên kết 1</a>

        <a class="link">liên kết 2</a>

        <a class="link">liên kết 3</a>
```

```
</div>

<div id="content">nội dung</div>

<div id="footer">cuối trang</div>

</body>

</html>
```

#### 2.3.4.7 Phần tử biểu mẫu(<form>)

Phần tử <form> là phần tử HTML chứa các trường nhập liệu và các nút hành động để người dùng có thể điền thông tin và gửi dữ liệu. Form có thể gửi dữ liệu đến máy chủ bằng các phương thức HTTP như GET và POST. Phần tử <form> có các thuộc tính quan trọng như:

**action:** Xác định URL mà dữ liệu sẽ được gửi đến khi người dùng nhấn gửi.

**method:** Xác định phương thức gửi dữ liệu (GET hoặc POST).

Ví dụ về phần tử <form> yêu cầu người dùng nhập vào thông tin tên và tuổi:

```
<form>

    <label for="name">tên</label>

    <input type="text" name="name">

    <label for="age">tuổi</label>

    <input type="number" name="age">

    <input type="submit" value="gửi">

</form>
```

#### 2.3.4.8 Phần tử bảng(<table>)

Phần tử bảng được sử dụng để hiển thị dữ liệu dưới dạng bảng. Bảng trong HTML được cấu thành từ nhiều phần tử khác nhau, mỗi phần tử đóng vai trò riêng trong cấu trúc của bảng. Các phần tử con trong bảng HTML bao gồm:

Phần tử **<thead>**: Định nghĩa phần đầu bảng, cụ thể là phần chứa các tiêu đề của từng cột dữ liệu trong bảng.

Phần tử **<tbody>**: Định nghĩa phần thân bảng, là phần chứa toàn bộ dữ liệu của bảng HTML.

Phần tử **<tr>**: Định nghĩa một dòng dữ liệu trong bảng HTML.

Phần tử **<th>**: Định nghĩa tiêu đề của một cột trong bảng HTML.

Phần tử **<td>**: Định nghĩa dữ liệu của một cột trong bảng HTML.

Ví dụ về một bảng dữ liệu về thông tin cá nhân của học sinh:

```
<table>
  <thead>
    <tr>
      <th>mã số</th>
      <th>họ tên</th>
      <th>lớp học</th>
      <th>năm sinh</th>
      <th>nơi sinh</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>001</td>
      <td>Nguyễn Văn A</td>
      <td>7</td>
      <td>2005</td>
      <td>Trà Vinh</td>
    </tr>
  </tbody>
</table>
```

```
</tr>

<tr>

    <td>002</td>

    <td>Nguyễn Văn B</td>

    <td>8</td>

    <td>2004</td>

    <td>Cần Thơ</td>

</tr>

</thead>

</table>
```

## 2.4. Ngôn ngữ CSS

CSS là viết tắt của cụm từ Cascading Style Sheets, đây là ngôn ngữ dùng để tìm và định dạng lại các phần tử được tạo ra bởi ngôn ngữ HTML. Để nói một cách ngắn gọn, CSS là ngôn ngữ dùng để tạo nên phong cách cho trang web.

Cú pháp của CSS gồm các khối khai báo (declaration block) và các bộ chọn (selector). Trong đó bộ chọn trỏ đến phần tử HTML được tạo kiểu, khối khai báo chứa tối thiểu 1 khai báo, ngăn cách nhau bởi dấu chấm phẩy. Mỗi khai báo gồm một tên thuộc tính và một giá trị, phân tách nhau bởi dấu hai chấm. Các khối khai báo nằm giữa dấu ngoặc nhọn [2].

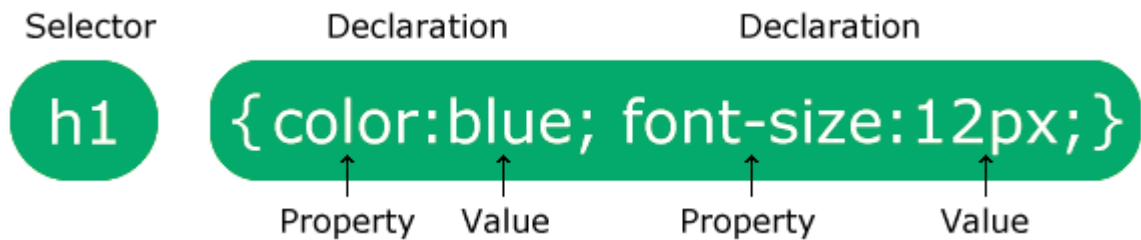
Một quy tắc CSS bao gồm một **Selector** và một **khối khai báo(Declaration)**.

Selector trỏ đến phần tử HTML mà bạn muốn định dạng. Khối khai báo chứa một hoặc nhiều khai báo được phân tách bằng dấu chấm phẩy.

Mỗi khai báo bao gồm một tên thuộc tính CSS và một giá trị, được phân tách bằng dấu hai chấm.

Nhiều khai báo CSS được phân tách bằng dấu chấm phẩy và các khối khai báo được bao quanh bằng dấu ngoặc nhọn.





**Hình 2.1 Cấu trúc của một Selector**

#### **2.4.1 Các kiểu Selector trong CSS**

Selector trong CSS được sử dụng để "tìm" (hoặc chọn) các phần tử HTML mà bạn muốn định kiểu. Chúng ta có thể chia Selector trong CSS thành năm loại:

Selector Simples (chọn các phần tử dựa trên tên, id, lớp)

Combinator selectors (chọn các phần tử dựa trên mối quan hệ cụ thể giữa chúng).

Pseudo-class selectors (chọn các phần tử dựa trên trạng thái nhất định).

Pseudo-elements selectors (chọn và định kiểu một phần của phần tử).

Attribute selectors (chọn các phần tử dựa trên thuộc tính hoặc giá trị thuộc tính).

Trong phần này chúng ta sẽ chỉ giải thích Selector cơ bản nhất là Selector Simples.

#### **2.4.2 Cách chèn CSS**

Khi trình duyệt đọc một bảng định kiểu(style sheet) chứa CSS, nó sẽ định dạng tài liệu HTML theo thông tin trong bảng định kiểu đó. Có ba cách chèn bảng định kiểu vào tài liệu HTML: External CSS, internal CSS, inline CSS.

##### **2.4.2.1 External CSS**

Với một external css, bạn có thể thay đổi giao diện của toàn bộ trang web chỉ bằng cách thay đổi một file.

Mỗi trang HTML phải bao gồm một tham chiếu đến tệp bảng định kiểu bên ngoài bên trong phần tử <link>, bên trong phần đầu.

```
body {
```

```
font-family: Arial, sans-serif;

background-color: #f0f0f0;

color: #333;

margin: 0;

padding: 0;

}

h1 {

    color: #4CAF50;

    text-align: center;

}

p {

    font-size: 18px;

    line-height: 1.6;

    text-align: center;

}
```

#### 2.4.2.2 Internal CSS

Có thể sử dụng internal css nếu trang HTML duy nhất chỉ có một kiểu CSS. internal css được định nghĩa bên trong phần tử <style>, bên trong phần đầu.

Các internal css được định nghĩa trong phần tử <style>, bên trong phần <head> của trang HTML:

```
<!DOCTYPE html>

<html lang="vi">

<head>
```

```
<meta charset="UTF-8">

<meta name="viewport" content="width=device-width,
initial-scale=1.0">

<title>Trang Web với Internal CSS</title>

<style>

    body {

        font-family: Arial, sans-serif;

        background-color: #f0f0f0;

        color: #333;

        margin: 0;

        padding: 0;

    }

    h1 {

        color: #4CAF50;

        text-align: center;

    }

    /* Định dạng cho thẻ p

    p {

        font-size: 18px;

        line-height: 1.6;

        text-align: center;
```

```
        }  
    </style>  
  
</head>  
  
<body>  
  
    <h1>Chào mừng đến với trang web của tôi</h1>  
  
    <p>Đây là một trang web sử dụng Internal CSS để định  
dạng giao diện.</p>  
  
  
</body>  
  
</html>
```

#### 2.4.2.3 Inline CSS

Có thể sử dụng inline css để áp dụng kiểu duy nhất cho một phần tử HTML duy nhất. Để sử dụng inline css, hãy thêm thuộc tính style vào phần tử HTML có liên quan. Thuộc tính style có thể chứa bất kỳ thuộc tính CSS nào.

```
<!DOCTYPE html>  
  
<html lang="vi">  
  
  <head>  
  
    <meta charset="UTF-8">  
  
    <meta name="viewport" content="width=device-width,  
initial-scale=1.0">  
  
    <title>Ví Dụ Inline CSS</title>  
  
  </head>  
  
  <body>
```

```
<h1 style="color: red; text-align: center;">chào</h1>  
</body>  
</html>
```

## 2.5. Ngôn ngữ JavaScript

### 2.5.1 Tổng quan về JavaScript

Là một ngôn ngữ lập trình được cho các trang web (phía người dùng) cũng như phía máy chủ (với Nodejs). Trên trình duyệt, rất nhiều trang web sử dụng JavaScript để thiết kế trang web động và một số hiệu ứng hình ảnh thông qua DOM. JavaScript còn được dùng để thực hiện một số tác vụ không thể thực hiện được với chỉ HTML như kiểm tra thông tin nhập vào, tự động thay đổi hình ảnh [3].

### 2.5.2 Cách hoạt động và đặc điểm của JavaScript

JavaScript chạy trên trình duyệt web, nơi mà nó có thể tương tác với các phần tử HTML và CSS. Trình duyệt sẽ biên dịch mã JavaScript ngay lập tức khi nó gặp đoạn mã trong tài liệu HTML và thực thi nó.

Hỗ trợ các khái niệm như đối tượng và kế thừa. Không yêu cầu khai báo kiểu dữ liệu trước khi sử dụng. Cho phép phát triển ứng dụng web mà không cần cài đặt phần mềm bổ sung.

Cú pháp dễ học và dễ hiểu, tương tự như cú pháp của các ngôn ngữ lập trình khác như C hay Java.

### 2.5.3 Môi liên hệ giữa JavaScript, CSS và HTML

**HTML (HyperText Markup Language):** Cung cấp cấu trúc cho trang web. Nó xác định các phần tử như tiêu đề, đoạn văn, hình ảnh, v.v.

**CSS (Cascading Style Sheets):** Được sử dụng để định kiểu và trang trí các phần tử HTML. Nó quyết định giao diện và bố cục của trang.

**JavaScript:** Tạo ra sự tương tác cho các phần tử HTML và có thể thay đổi kiểu dáng do CSS xác định. Cả ba công nghệ này hoạt động cùng nhau để xây dựng trang web hoàn chỉnh.

## 2.6. Ngôn ngữ XML

XML là một công cụ độc lập với phần mềm và phần cứng dùng để lưu trữ và vận chuyển dữ liệu. XML là viết tắt của eXtensible Markup Language.

XML là ngôn ngữ đánh dấu giống như HTML, được thiết kế để lưu trữ và vận chuyển dữ liệu.

XML được thiết kế để tự mô tả và là Khuyến nghị của W3C.

XML được sử dụng trong nhiều khía cạnh của phát triển web.

XML thường được sử dụng để tách dữ liệu khỏi phần trình bày.

XML đơn giản hóa mọi thứ:

XML đơn giản hóa việc chia sẻ dữ liệu.

XML đơn giản hóa việc vận chuyển dữ liệu.

XML đơn giản hóa việc thay đổi nền tảng.

XML đơn giản hóa tính khả dụng của dữ liệu.

### 2.6.1 Đặc điểm ngôn ngữ XML

XML không mang bất kỳ thông tin nào về cách hiển thị. một dữ liệu XML có thể được sử dụng trong nhiều tình huống trình bày khác nhau. Do đó, với XML, có sự tách biệt hoàn toàn giữa dữ liệu và trình bày.

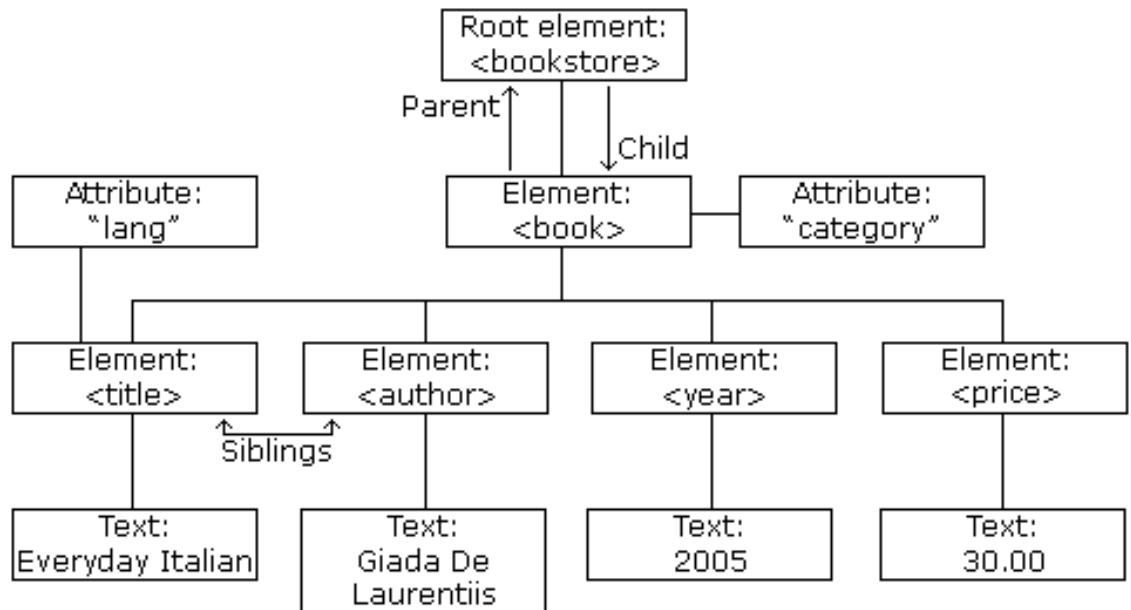
Trong nhiều ứng dụng HTML, XML được sử dụng để lưu trữ hoặc vận chuyển dữ liệu, trong khi HTML được sử dụng để định dạng và hiển thị cùng một dữ liệu.

Các thẻ XML không được xác định trước. Với XML, tác giả phải xác định cả thẻ và cấu trúc tài liệu.

Hầu hết các ứng dụng XML sẽ hoạt động như mong đợi ngay cả khi dữ liệu mới được thêm vào (hoặc xóa).

### 2.6.2 Cấu trúc một tài liệu XML

Các tài liệu XML tạo thành một cấu trúc cây bắt đầu từ root và phân nhánh đến các leaves.



**Hình 2.2 Cấu trúc cây của tài liệu XML**

Các thuật ngữ parent, child và anh sibling được sử dụng để mô tả mối quan hệ giữa các thành phần.

Parent có thể có nhiều nhánh child nhưng tất cả các nhánh child chỉ có một gốc đó là parent. sibling là những nhánh child cùng cấp độ.

Tất cả các thành phần trong cấu trúc cây tài liệu XML có thể có nội dung văn bản và thuộc tính.

Ví dụ về một tài liệu XML quản lý thông tin cá nhân của các sinh viên trường đại học:

```
<?xml version="1.0" encoding="UTF-8"?>
<dssv>
  <sinhvien>
    <masv>110000002</masv>
    <tensv>Nguyễn Thanh Hà</tensv>
    <matkhau>213</matkhau>
    <gioitinh>Nữ</gioitinh>
    <namsinh>2001</namsinh>
```

```
<noisinh>Cà Mau</noisinh>
<nganh>Sư Phạm Mầm Non</nganh>
</sinhvien>
<sinhvien>
  <masv>110000009</masv>
  <tensv>Lê Văn Khang</tensv>
  <matkhau>123</matkhau>
  <gioitinh>Nam</gioitinh>
  <namsinh>2004</namsinh>
  <noisinh>Trà Vinh</noisinh>
  <nganh>CNTT</nganh>
</sinhvien>
<sinhvien>
  <masv>110000001</masv>
  <tensv>Lê Thị Xuân</tensv>
  <matkhau>124</matkhau>
  <gioitinh>Nữ</gioitinh>
  <namsinh>1999</namsinh>
  <noisinh>Trà Vinh</noisinh>
  <nganh>Công Nghệ Thông Tin</nganh>
</sinhvien>
<sinhvien>
  <masv>110000000</masv>
  <tensv>Ngô Văn Khoa</tensv>
```



```
<matkhau>123</matkhau>

<gioitinh>Nam</gioitinh>

<namsinh>2000</namsinh>

<noisinh>Hà Nội</noisinh>

<nganh>Công Nghệ Thông Tin</nganh>

</sinhvien>

</dssv>
```

Tài liệu XML phải chứa một phần tử gốc là phần tử cha của tất cả các phần tử khác. Trong ví dụ hình 2.7 trên thì <library> là phần tử gốc:

Dòng <?xml version="1.0" encoding="UTF-8"?> được gọi là prolog XML.

Tất cả các phần tử XML phải có thẻ đóng.

Thẻ XML phân biệt chữ hoa chữ thường.

Nếu bạn đặt một ký tự như "<" bên trong một phần tử XML, nó sẽ tạo ra lỗi vì trình phân tích cú pháp hiểu đó là phần bắt đầu của một phần tử mới. Để tránh lỗi này, hãy thay thế ký tự "<" bằng tham chiếu thực thể &lt; [4].

## 2.7. Ngôn ngữ PHP

PHP là từ viết tắt của PHP: Hypertext Preprocessor là ngôn ngữ kịch bản mã nguồn mở được sử dụng phổ biến, các đoạn mã PHP được thực thi trên server(máy chủ) PHP có thể tải xuống và sử dụng miễn phí [4].

### 2.7.1 Ứng dụng phổ biến của PHP

PHP có thể tạo nội dung trang web động.

PHP có thể tạo, mở, đọc, ghi, xóa và đóng các tệp trên máy chủ.

PHP có thể thu thập dữ liệu biểu mẫu.

PHP có thể gửi và nhận cookie.

PHP có thể thêm, xóa, sửa đổi dữ liệu trong cơ sở dữ liệu của bạn.

PHP có thể được sử dụng để kiểm soát quyền truy cập của người dùng.

PHP có thể mã hóa dữ liệu.

Với PHP, bạn không bị giới hạn ở đầu ra HTML. Đầu ra có thể là dạng hình ảnh hoặc tệp PDF. Bạn cũng có thể để đầu ra bất kỳ văn bản nào, chẳng hạn như XHTML và XML.

## 2.7.2 Kiến thức cơ bản về PHP

### 2.7.2.1 Mở và đóng PHP

Mọi mã PHP được viết theo cấu trúc **<?php ?>**. Trong đó echo là câu lệnh in ra màn hình. Ví dụ cách đóng và mở PHP:

```
<?php  
    echo "Hello world";  
?>
```

### 2.7.2.2 Biến

Biến trong PHP được khai báo bằng ký hiệu \$, không cần chỉ định kiểu dữ liệu. Ví dụ về một biến trong PHP:

```
<?php  
$name = "Nguyễn Văn T";  
?>
```

### 2.7.2.3 Hằng số

Hằng số được khai báo bằng hàm **define** hoặc từ khóa **const**.

```
<?php  
    define("sanPham", "tivi");  
    const MaSanPham = 001;  
?>
```

#### 2.7.2.4 Câu lệnh điều kiện

**Câu lệnh if:** Câu lệnh điều kiện **if** dùng để kiểm tra điều kiện của mệnh đề mà người dùng đặt ra nếu mệnh đề đúng thì thực thi câu lệnh ngược lại xe không thực thi câu lệnh.

```
<?php
$age = 20; // Tuổi của người

if ($age >= 18) {
    echo "Bạn đủ tuổi để lái xe.";
}
?>
```

**Câu lệnh switch-case:** dùng để kiểm tra nhiều giá trị:

```
<?php
$day = 3; // Giả sử ngày là 3

switch ($day) {
    case 1:
        echo "Chủ nhật";
        break;
    case 2:
        echo "Thứ hai";
        break;
    case 3:
        echo "Thứ ba";
        break;
}
```

```
case 4:
    echo "Thứ tư";
    break;

case 5:
    echo "Thứ năm";
    break;

case 6:
    echo "Thứ sáu";
    break;

case 7:
    echo "Thứ bảy";
    break;

default:
    echo "Ngày không hợp lệ";
}

?>
```

#### 2.7.2.5 Vòng lặp

**For loop:** Lặp với số lần xác định.

```
<?php
    for ($i = 1; $i <= 5; $i++) {
        echo "Sản phẩm phụ kiện điện thoại #<i>$i</i><br>";
    }?>
```

**While loop:** Lặp khi điều kiện đúng.

```
<?php
```

```
$SoLuong = 3;

while ($SoLuong > 0) {

    echo "Còn $SoLuong sản phẩm trong kho.<br>";

    $SoLuong--;

}

?>
```

**Do-while loop:** Lặp ít nhất một lần trước khi kiểm tra điều kiện

```
<?php

$SoLuong = 0;

do {

    echo "Hiển thị sản phẩm ngay cả khi hết hàng.<br>";

} while ($SoLuong > 0);

?>
```

**Foreach loop:** Duyệt qua mảng

```
<?php

$SanPham = ["Ốp lưng", "Tai nghe", "Cáp sạc", "Pin dự
phòng"];

foreach ($SanPham as $item) {

    echo "Sản phẩm: $item<br>";

}

?>
```

#### 2.7.2.6 Hàm

Hàm được định nghĩa bằng từ khóa **function**.

```
<?php
```

```
function sum($a, $b) {  
    return $a + $b;  
}  
  
echo sum(5, 4);  
  
?>
```

#### 2.7.2.7 Mảng

Mảng có thể chứa nhiều giá trị khác nhau cùng lúc:

```
<?php  
  
$SanPham = [1, 2, 3, 4];  
  
echo $SanPham[2];  
  
?>
```

#### 2.7.2.8 Xử lý chuỗi

Kết hợp 2 chuỗi lại với nhau bằng dấu chấm:

```
<?php  
  
$ThuongHieu= "Phụ kiện XYZ";  
  
$SanPham= "Ốp lưng silicon";  
  
echo $ ThuongHieu. " giới thiệu sản phẩm: " . $ SanPham;  
  
?>
```

### 2.8. Tương tác dữ liệu với PHP và XML

Ngôn ngữ PHP khi tương tác với ngôn ngữ cấu trúc XML để thao tác với dữ liệu được tiếp cận theo 2 hướng chính đó là SimpleXML và XML DOM.

#### 2.8.1 SimpleXML

SimpleXML là một API đơn giản và dễ sử dụng, được thiết kế thao tác với PHP để truy xuất dữ liệu trong cấu trúc XML. SimpleXML cho phép người dùng thao tác với cấu trúc XML như là một đối tượng trong ngôn ngữ PHP [4, 5].

## 2.8.2 XML DOM

XML DOM là một API với độ phức tạp cao hơn so với SimpleXML. XML DOM cung cấp các tính năng để PHP thao tác với cây DOM của XML giống với JavaScript làm với cây DOM của HTML [4, 5].

## 2.9. Bootstrap framework

### 2.9.1 Tổng quan

Bootstrap cung cấp các thành phần giao diện sẵn có (như các nút, form, bảng, modal, v.v.) và các tiện ích CSS, giúp việc phát triển giao diện web trở nên nhanh chóng và hiệu quả. Nó được thiết kế với mục tiêu giúp các nhà phát triển xây dựng các trang web đáp ứng (responsive) và có tính tương thích cao trên các trình duyệt khác nhau mà không cần phải viết quá nhiều mã CSS [5].

### 2.9.2 Kiến thức cơ bản về Bootstrap

Trong bootstrap có hệ thống lưới (Grid system) giúp bố trí các phần tử một cách linh hoạt trên các kích thước màn hình khác nhau. Hệ thống lưới trong Bootstrap hoạt động dựa trên:

Hàng (rows): Sử dụng class `.row` để nhóm các cột lại với nhau.

Cột (columns): Chia bố cục thành tối đa 12 cột mỗi hàng, sử dụng class như `.col`, `.col-4`, `.col-md-6`...

Ví dụ về cấu trúc lưới của Bootstrap:

```
<div class="container">
  <div class="row">
    <div class="col-3">Cột 1</div>
    <div class="col-3">Cột 2</div>
    <div class="col-3">Cột 3</div>
    <div class="col-3">Cột 4</div>
  </div>
```

```
</div>
```

Breakpoints: Cung cấp các điểm ngắt cho thiết bị (xs, sm, md, lg, xl, xxl) để điều chỉnh giao diện theo kích thước màn hình. Các breakpoints giúp xác định cách hiển thị trên từng kích thước màn hình:

**.col-sm-\***: Áp dụng cho màn hình nhỏ ( $\geq 576\text{px}$ )

**.col-md-\***: Áp dụng cho màn hình vừa ( $\geq 768\text{px}$ )

**.col-lg-\***: Áp dụng cho màn hình lớn ( $\geq 992\text{px}$ )

**.col-xl-\***: Áp dụng cho màn hình rất lớn ( $\geq 1200\text{px}$ )

**.col-xxl-\***: Áp dụng cho màn hình cực lớn ( $\geq 1400\text{px}$ )

```
<div class="container">
  <div class="row">
    <div class="col-sm-6 col-md-4 col-lg-3">Cột 1</div>
    <div class="col-sm-6 col-md-4 col-lg-3">Cột 2</div>
    <div class="col-sm-6 col-md-4 col-lg-3">Cột 3</div>
    <div class="col-sm-6 col-md-4 col-lg-3">Cột 4</div>
  </div> </div>
```

Hệ thống lưới hỗ trợ thêm khoảng cách giữa các cột hoặc hàng bằng class: Padding/margin: Sử dụng class như .px-\*, .py-\*, .mx-\*, .my-\*

```
<div class="container">
  <div class="row">
    <div class="col-6 my-2">Cột 1</div>
    <div class="col-6 my-2">Cột 2</div>
  </div>
</div>
```



### 2.9.3 Cài đặt

Thêm các liên kết tới các tệp CSS và JavaScript của Bootstrap vào trong phần `<head>` và trước thẻ `</body>` của tài liệu HTML.

```
<!-- Liên kết Bootstrap CSS -->

<link
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/
bootstrap.min.css" rel="stylesheet">


<!-- Liên kết jQuery -->

<script          src="https://code.jquery.com/jquery-
3.5.1.slim.min.js"></script>


<!-- Liên kết Bootstrap JS -->

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bo
otstrap.min.js"></script>
```

## CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1. Mô tả bài toán

Nhằm đáp ứng nhu cầu quản lý điểm số của sinh viên các lớp trong quá trình giảng dạy, tôi quyết định xây dựng một hệ thống website lưu trữ kết quả học tập của sinh viên, để làm được điều đó thì mục tiêu của tôi sẽ là bắt đầu tìm hiểu sâu vào nguyên cứu về khả năng lưu trữ của ngôn ngữ XML.

Hệ thống website sẽ bao gồm các trang như: Trang đăng nhập, trang chủ, trang đăng ký môn dành cho sinh viên, trang phân công dành cho giảng viên, trang quản trị viên và các trang quản lý môn học, lớp học, học kỳ, sinh viên, giảng viên, phân công lịch giảng dạy, sửa thông tin.

các thông tin cá nhân của từng sinh viên là: Mã sinh viên, tên sinh viên, mật khẩu, năm sinh, nơi sinh, ngành. Các sinh viên có thể chọn môn học để đăng ký, thông tin của môn học gồm: Mã môn học, tên môn học, số tín chỉ lý thuyết, số tín chỉ thực hành. Sau khi sinh viên đăng ký môn học thì sẽ có thêm các thông tin về kết quả học tập gồm: Mã sinh viên, tên sinh viên, mã môn học, mã học kỳ, điểm lần 1, điểm lần 2.

Mỗi môn học mà sinh viên đăng ký sẽ thuộc về một học kỳ với các thông tin cụ thể: mã học kỳ, năm học. Mỗi sinh viên sẽ thuộc về một lớp học cụ thể với các thông tin gồm: Mã lớp học, tên lớp học.

Ngoài quản lý thông tin cá nhân và kết quả học tập của sinh viên thì nhà trường cũng cần phải quản lý các thông tin của giảng viên gồm: Mã giảng viên, tên giảng viên, số điện thoại, email.

### 3.2. Phân tích thiết kế hệ thống

#### 3.2.1 Đặc tả yêu cầu hệ thống

##### 3.2.1.1 Yêu cầu chức năng

**Hiển thị thông tin danh sách điểm môn học của sinh viên:** Hiển thị được các thông tin như môn học, lớp học, năm học, lớp học và giảng viên giảng dạy của từng danh sách điểm sinh viên.

**Xuất danh sách điểm:** Xuất ra danh sách bảng điểm của môn học mà giảng viên đang giảng dạy khi giảng viên đã hoàn thành việc chấm điểm cho toàn bộ sinh viên đăng ký môn học. Sau khi giảng viên xuất danh sách điểm, danh sách sẽ hiển thị trên trang chủ để sinh viên có thể vào xem điểm.

**Chức năng thêm thông tin:** Thêm vào các thông tin mới như sinh viên, giảng viên, môn học, học kỳ và lớp học để quản trị viên có thể phân công việc giảng dạy.

**Chức năng xóa thông tin:** Xóa các thông tin như sinh viên, giảng viên, môn học, học kỳ và lớp học.

**Chức năng sửa thông tin:** Sửa các thông tin như sinh viên, giảng viên, môn học, học kỳ và lớp học.

**Chức năng tìm kiếm thông tin:** Tìm kiếm các thông tin như tên của sinh viên hoặc giảng viên, tên môn học.

**Chức năng đăng ký môn học:** Sinh viên đăng ký các môn học dựa theo mong muốn của bản thân. Sau khi đăng ký môn học, sinh viên sẽ được đưa vào lớp học của giảng viên giảng dạy môn học đó.

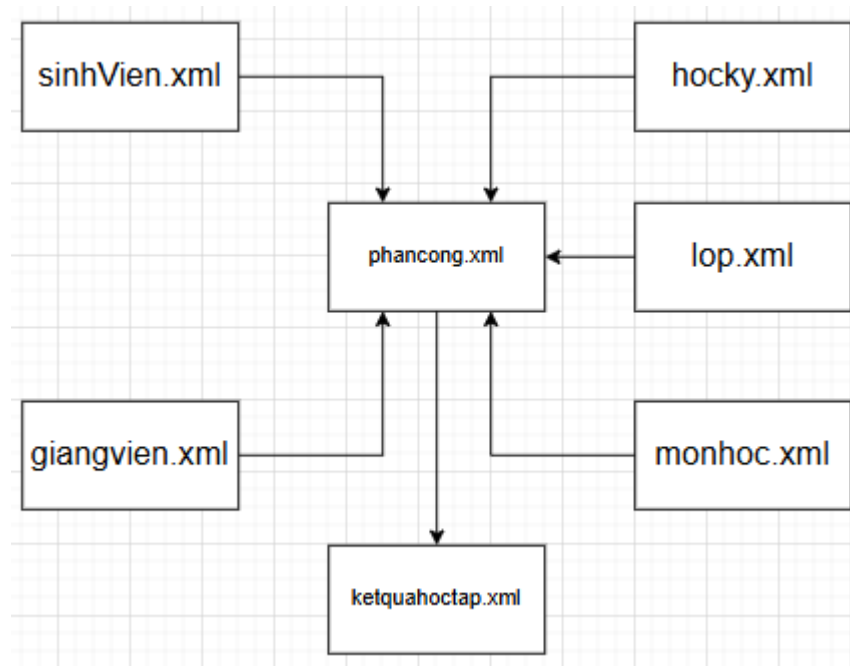
### 3.2.1.2 Yêu cầu phi chức năng

Giao diện đơn giản, đẹp mắt, thân thiện và dễ dàng sử dụng với 3 người dùng chính của trang web là sinh viên, giảng viên và quản trị viên. Phong chữ ưa nhìn và kích thước phù hợp với từng bố cục của website. Có thể ẩn các chức năng của quản trị viên khi tài khoản đăng nhập là tài khoản của sinh viên hoặc giảng viên. Chặn và điều hướng sinh viên và giảng viên tránh xa các trang của quản trị viên khi sinh viên hoặc giảng viên cố gắng truy cập để bảo mật thông tin của website.

### 3.2.2 Tổ chức lưu trữ dữ liệu

Dữ liệu lưu trữ dưới dạng tập tin .xml và có cấu trúc gồm một phần tử chính và các phần tử con chứa các thuộc tính.

Hệ thống lưu trữ dữ liệu dạng tập tin XML bao gồm các tệp XML:



**Hình 3.1** Sơ đồ hệ thống lưu trữ dưới dạng các tập tin XML

### 3.2.2.1 Tập tin lop.xml

Chứa thông tin về các lớp học, có phần tử gốc là **<dslop>**, trong đó có nhiều phần tử con là **<lop>** đại diện cho 1 lớp học cụ thể. Mỗi phần tử **<lop>** chứa 2 thẻ con là **<malop>** đại diện cho mã lớp học và **<tenlop>** đại diện cho tên lớp học.

**Bảng 3.1** Bảng dữ liệu tập tin lop.xml

STT	Thẻ XML	Diễn giải	Ghi chú
1	<dslop>	Danh sách các lớp học	Phần tử gốc
2	<lop>	Lớp học	Lưu trữ thông tin một lớp học
3	<malop>	Mã lớp học	Mã định danh của lớp học
4	<tenlop>	Tên lớp học	

Ví dụ: Cấu trúc tài liệu XML lưu trữ thông tin của một lớp học cụ thể gồm mã lớp DA22TTD và tên lớp là Đại học Công nghệ Thông tin khóa 2022.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dslop>
```

```
<lop>
```

```
<malop>DA22TTD</malop>

<tenlop>Công nghệ Thông tin</tenlop>

</lop>

</dslop>
```

### 3.2.2.2 Tập tin sinhVien.xml

Tập tin có phần tử gốc là **<dssv>**, trong đó có nhiều phần tử con là **<sinhvien>** đại diện cho 1 sinh viên cụ thể. Mỗi phần tử **<sinhvien>** gồm các thẻ con là **<masv>**, **<tensv>**, **<matkhau>**, **<gioitinh>**, **<namsinh>**, **<noisinh>**, **<nganh>**.

**Bảng 3.2** Bảng dữ liệu tập tin sinhVien.xml

<i>STT</i>	<i>Nội dung</i>	<i>Diễn giải</i>	<i>Ghi chú</i>
1	<b>&lt;dssv&gt;</b>	Danh sách các sinh viên	Phần tử gốc chứa thông tin các sinh viên
2	<b>&lt;sinhvien&gt;</b>	Sinh viên	Lưu trữ thông tin sinh viên
3	<b>&lt;masv&gt;</b>	Mã sinh viên	Mã định danh của sinh viên
4	<b>&lt;tensv&gt;</b>	Tên sinh viên	
5	<b>&lt;matkhau&gt;</b>	Mật khẩu	
6	<b>&lt;gioitinh&gt;</b>	Giới tính	
7	<b>&lt;namsinh&gt;</b>	Năm sinh	
8	<b>&lt;noisinh&gt;</b>	Nơi sinh	
9	<b>&lt;nganh&gt;</b>	Ngành	

Ví dụ tập sinh thông tin của sinh viên:

```
<?xml version="1.0" encoding="UTF-8"?>

<dssv>

  <sinhvien>

    <masv>110000002</masv>
```

```

<tensv>Lê Thanh Hà</tensv>

<matkhau>213</matkhau>

<gioitinh>Nữ</gioitinh>

<namsinh>2001</namsinh>

<noisinh>Cà Mau</noisinh>

<nganh> Sư Phạm Mầm Non</nganh>

</sinhvien>

</dssv>

```

### 3.2.2.3 Tập sinh giangvien.xml

Tập tin có phần tử gốc là **<dsgiangvien>**, trong đó có nhiều phần tử con là **<giangvien>** đại diện cho 1 giảng viên cụ thể. Mỗi phần tử **<giangvien>** gồm các thẻ con là **<magv>**, **<tengv>**, **<matkhau>**, **<gioitinh>**, **<namsinh>**, **<noisinh>**, **<sdt>**.

**Bảng 3.3 Bảng dữ liệu của tập tin giangvien.xml**

STT	Nội dung	Diễn giải	Ghi chú
1	<dsgiangvien>	Danh sách các giảng viên	Phần tử gốc chứa thông tin các giảng viên
2	<giangvien>	Giảng viên	Lưu trữ thông tin giảng viên
3	<magv>	Mã giảng viên	Mã định danh của giảng viên
4	<tengv>	Tên giảng viên	
5	<matkhau>	Mật khẩu	
6	<gioitinh>	Giới tính	
7	<namsinh>	Năm sinh	
8	<noisinh>	Nơi sinh	
9	<sdt>	Số điện thoại của giảng viên	

Ví dụ thông tin của giảng viên:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dsgiangvien>
```

```
<giangvien>
```

```
<magv>220000001</magv>
```

```
<tengv>Lê Thanh Khang</tengv>
```

```
<matkhau>090</matkhau>
```

```
<gioitinh>Nam</gioitinh>
```

```
<namsinh>2004</namsinh>
```

```
<noisinh>Trà Vinh</noisinh>
```

```
<sdt>1234567771</sdt>
```

```
</giangvien> </dsgiangvien>
```

### 3.2.2.4 Tập tin hocky.xml

Tập tin hocky.xml có thể gốc là **<dshocky>**, trong đó chứa các thẻ con là **<hocky>** gồm các thẻ con là **<maHK>**, **<tenHK>**, **<namHoc>**.

**Bảng 3.4** Bảng dữ liệu của tập tin hocky.xml

STT	Nội dung	Diễn giải	Ghi chú
1	<dshocky>	Danh sách các học kỳ	Phần tử gốc chứa thông tin các học kỳ
2	<hocky>	Học kỳ	Lưu trữ thông tin một học kỳ
3	<maHK>	Mã học kỳ	Mã định danh của học kỳ
4	<tenHK>	Tên học kỳ	
5	<namHoc>	Năm học	

Ví dụ thông tin của học kỳ:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dshocky>
    <hocky>
        <maHK>112</maHK>
        <tenHK>Học Kỳ 2</tenHK>
        <namHoc>2024-2025</namHoc>
    </hocky>
    <hocky>
        <maHK>111</maHK>
        <tenHK>Học Kỳ 1</tenHK>
        <namHoc>2022-2023</namHoc>
    </hocky>
</dshocky>
```

### 3.2.2.5 Tập tin monhoc.xml

Tập tin monhoc.xml có thể gốc là **<mon>**, trong đó chứa các thẻ con là **<monHoc>** XMLXML gồm các thẻ con là **<maMonHoc>**, **<tenMonHoc>**, **<soTCLT>**, **<soTCTH>**.

**Bảng 3.5 Bảng dữ liệu của tập tin monhoc.xml**

STT	Nội dung	Diễn giải	Ghi chú
1	<mon>	Danh sách các môn học	Phần tử gốc chứa thông tin các môn học
2	<monHoc>	Chỉ một môn học cụ thể	Lưu trữ thông tin một môn học
3	<maMonHoc>	Mã môn học	Mã định danh của môn học
4	<tenMonHoc>	Tên môn học	
5	<soTCLT>	Số tín chỉ lý thuyết	



6	<soTCTH>	Số tín chỉ thực hành	
---	----------	----------------------	--

Ví dụ thông tin môn học:

```
<?xml version="1.0" encoding="UTF-8"?>

<mon>

    <monHoc>

        <maMonHoc>112</maMonHoc>

        <tenMonHoc>Hệ Điều Hành</tenMonHoc>

        <soTCLT>1</soTCLT>

        <soTCTH>2</soTCTH>

    </monHoc>

</mon>
```

### 3.2.2.6 Tập tin phancong.xml

Tập tin có phần tử gốc là **<dsphancong>**, trong đó có nhiều phần tử con là **<phancong>**. Mỗi phần tử **<phancong>** gồm các thẻ con là **<maphancong>**, **<hocky>**, **<monhoc>**, **<namhoc>**, **<giangvien>**, **<lop>**, **<dssinhvien>**.

Trong thẻ **<dssinhvien>** cũng lại bao gồm nhiều thẻ con **<sinhvien>** đại diện cho thông tin về điểm số của một sinh viên cụ thể. Trong thẻ sinh viên gồm có các thẻ như **<masv>**, **<tensv>**, **<diemlan1>**, **<diemlan2>**.

Trong thẻ **<diemlan1>** lại có các thẻ con chứa điểm quá trình, điểm kiểm tra của sinh viên là **<diemqt1>**, **<diemqt2>**, **<diemqt3>**, **<diemkt1>**, **<diemkt2>** và **<diemthi1>**. Trong thẻ **<diemlan2>** thì có 1 thẻ con là **<diemthi2>**.

**Bảng 3.6 Bảng dữ liệu của tập tin phancong.xml**

<i>STT</i>	<i>Nội dung</i>	<i>Diễn giải</i>	<i>Ghi chú</i>
1	<dsphancong>	Danh sách các phân công	Phần tử gốc chứa thông tin các phân công
2	<phancong>	Chỉ một phân công cụ thể	Lưu trữ thông tin một phân công
3	<hocky>	Tên học kỳ	
4	<monhoc>	Tên môn học	
5	<namhoc>	Năm học	
6	<giangvien>	Giảng viên	
7	<lop>	Lớp học	
8	<dssinhvien>	Danh sách các sinh viên	Danh sách các sinh viên có đăng ký môn học của giảng viên bất kỳ, khi đăng ký giảng viên sẽ thấy tên sinh viên trong danh sách sinh viên cần chấm điểm.
9	<sinhvien>	Sinh viên	
10	<masv>	Mã sinh viên	
11	<tensv>	Tên sinh viên	
12	<diemlan1>	Điểm lần 1	Điểm lần một sẽ được tính toán dựa vào các cột điểm quá trình, điểm kiểm tra và điểm thi lần 1 trong nó để cho ra kết quả trung bình
13	<diemlan2>	Điểm lần 2	Điểm lần dựa trên điểm lần 1 cộng thêm điểm thi lần 2 để cho ra kết quả trung bình
14	<diemqt1>	Điểm quá trình lần 1	
15	<diemqt2>	Điểm quá trình lần 2	
16	<diemqt3>	Điểm quá trình lần 3	
17	<diemkt1>	Điểm kiểm tra lần 1	

18	<diemkt2>	Điểm kiểm tra lần 2	
19	<diemthi1>	Điểm thi lần 1	
20	<diemthi2>	Điểm thi lần 2	

Ví dụ về thông tin phân công:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<dsphancong>
```

```
  <phancong>
```

```
    <maphancong>001</maphancong>
```

```
    <monhoc>Hệ Điều Hành</monhoc>
```

```
    <hocky>Học Kỳ 1</hocky>
```

```
    <namhoc>2024-2025</namhoc>
```

```
    <giangvien>Lê Thanh Khang</giangvien>
```

```
    <lop>DA22TTZ</lop>
```

```
    <dssinhvien>
```

```
      <sinhvien>
```

```
        <masv>110000001</masv>
```

```
        <tensv>Lê Thị Xuân</tensv>
```

```
        <diemlan1>
```

```
          <diemqt1>8</diemqt1>
```

```
          <diemqt2>8</diemqt2>
```

```
          <diemqt3>6</diemqt3>
```

```
          <diemkt1>7</diemkt1>
```

```
          <diemkt2>8</diemkt2>
```

```
<diemthi1>9</diemthi1>

<diemthi2/>

</diemlan1>

<diemlan2/>

</sinhvien>

</dssinhvien>

</phancong>

</dsphancong>
```

### 3.2.2.7 Tập tin ketquahoctap.xml

Sau khi giảng viên đã chấm điểm cho toàn bộ các sinh viên có trong lớp học và bắt đầu tiến hành xuất danh sách bảng điểm sinh viên lên trên trang chủ thì toàn bộ các thông tin của thẻ **<phancong>** trước đó sẽ được sao chép và chuyển sang cho tệp ketquahoctap.xml. Thẻ **<phancong>** khi chuyển sang tệp ketquahoctap.xml sẽ được chứa trong thư mục gốc là **<dsketquahoctap>**. Ví dụ:

```
<?xml version="1.0" encoding="UTF-8"?>

<dsketquahoctap>

  <phancong>

    <maphancong>001</maphancong>

    <monhoc>Hệ Điều Hành</monhoc>

    <hocky>Học Kỳ 1</hocky>

    <namhoc>2024-2025</namhoc>

    <giangvien>Lê Thanh Khang</giangvien>

    <lop>DA22TTZ</lop>

    <dssinhvien>
```

```
<sinhvien>

    <masv>110000001</masv>

    <tensv>Lê Thị Xuân</tensv>

    <diemlan1>

        <diemqt1>8</diemqt1>

        <diemqt2>8</diemqt2>

        <diemqt3>6</diemqt3>

        <diemkt1>7</diemkt1>

        <diemkt2>8</diemkt2>

        <diemthi1>9</diemthi1>

        <diemthi2/>

    </diemlan1>

    <diemlan2/>

</sinhvien>

</dssinhvien>

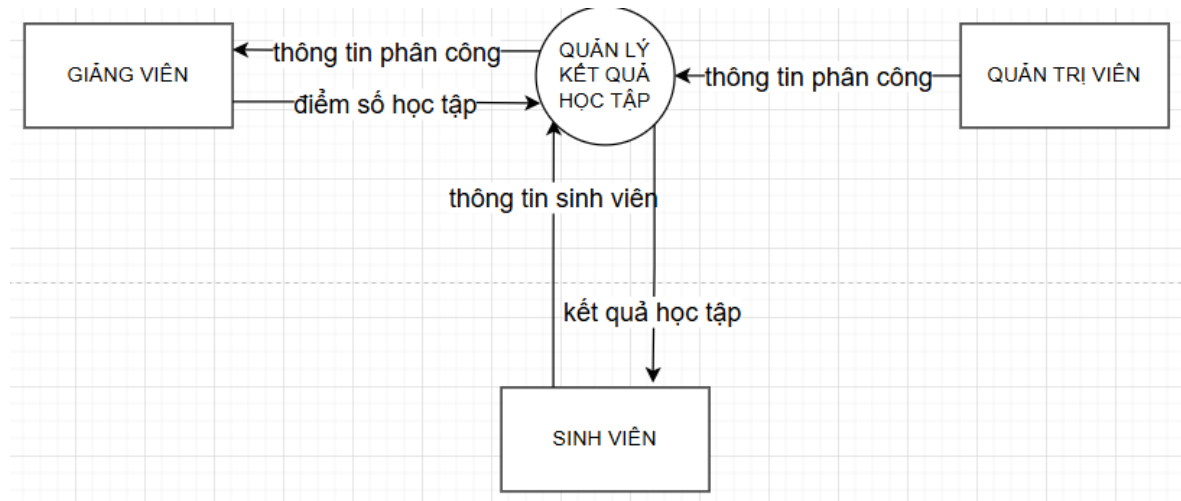
</phancong>

</dsketquahoctap>
```

### 3.2.3 Thiết kế xử lý

#### 3.2.3.1 Mô hình DFD mức ngữ cảnh

Mô hình DFD mức ngữ cảnh bao gồm có 3 thực thể là: **Sinh viên**, **giảng viên** và **quản trị viên**, luồng dữ liệu của 3 thực thể được xử lý thông qua một quá trình duy nhất là **quản lý kết quả học tập**. Đây cũng là mô hình có mức cao nhất trong mô hình DFD.

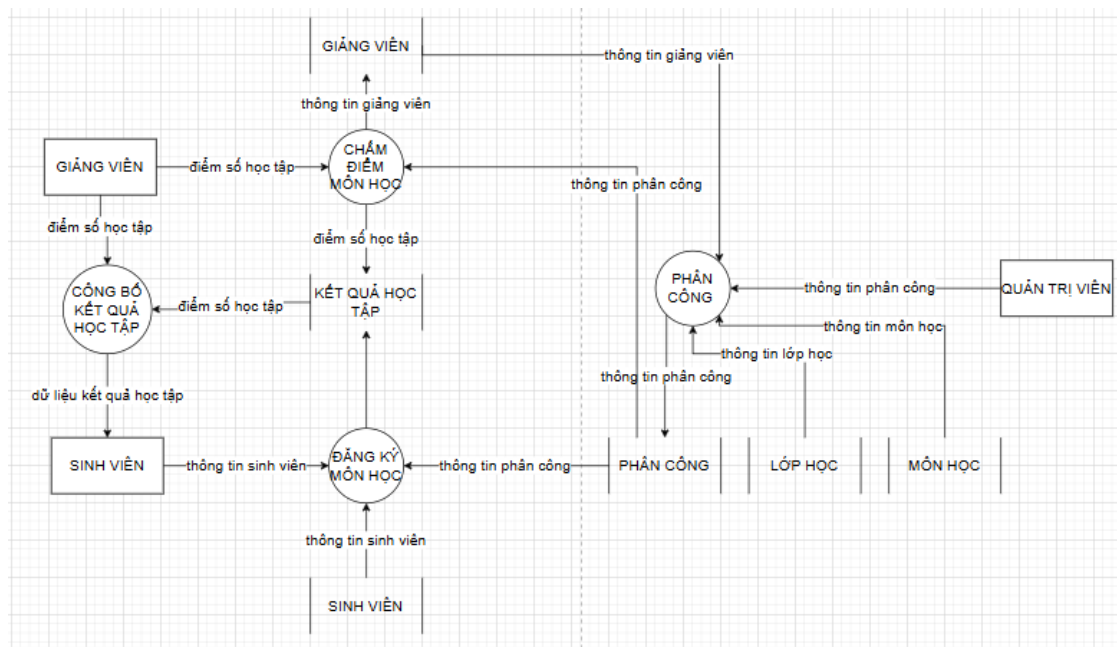


Hình 3.2 Mô hình DFD mức ngữ cảnh

### 3.2.3.2 Mô hình DFD mức 1

Mô hình DFD mức 1 bắt đầu đi sâu vào chi tiết hơn các cấu trúc con của mô hình, cụ thể là từ một quá trình duy nhất là **quản lý kết quả học tập** ở mức ngữ cảnh thì trong mô hình DFD mức một sẽ chia ra thành các phần nhỏ hơn là **đăng ký môn học**, **phân công**, **chấm điểm môn học** và **công bố kết quả**.

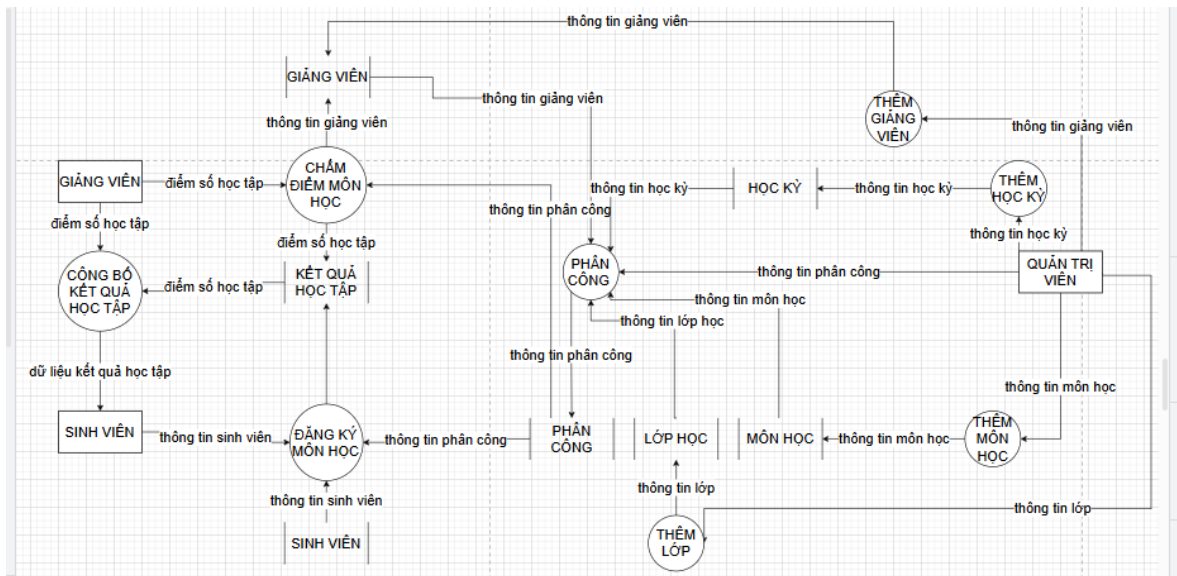
Ngoài ra mô hình cũng xuất hiện thêm các kho chứa dữ liệu gồm: **Giảng viên**, **sinh viên**, **lớp học**, **môn học**, **phân công** và **kết quả học tập**.



Hình 3.3 Mô hình DFD mức 1

### 3.2.3.3 Mô hình DFD mức 2

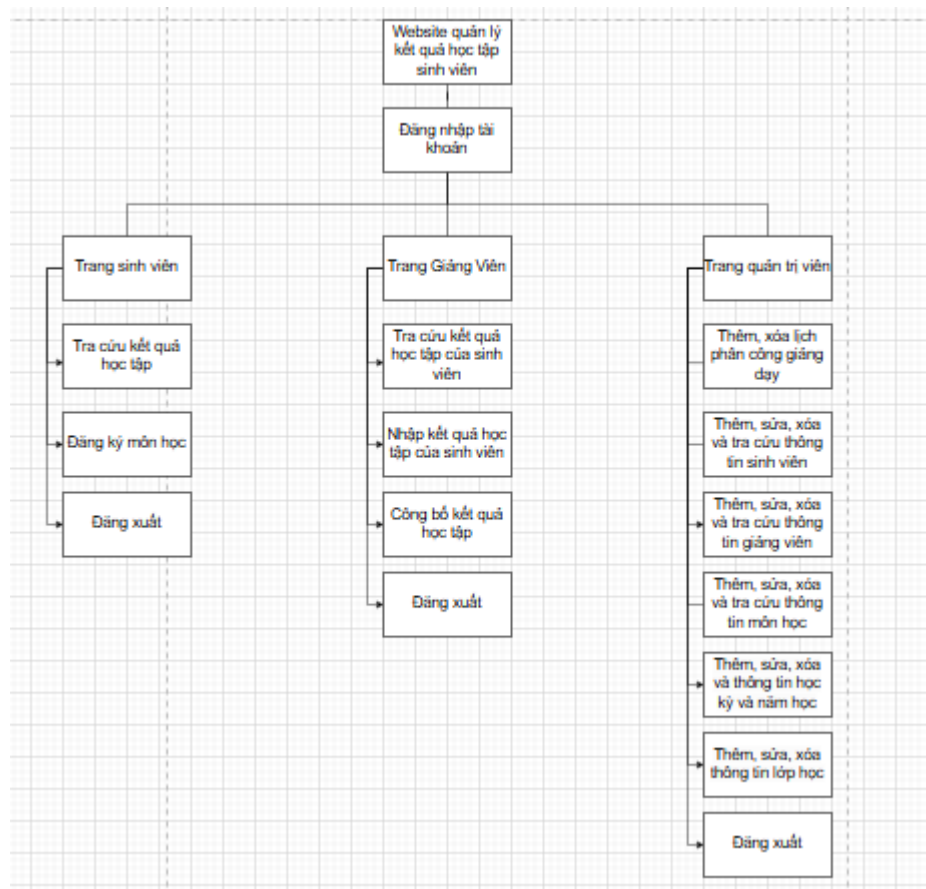
Mô hình DFD mức 2 là mô hình mô tả chi tiết nhất về hướng duy chuyển của luồng dữ liệu trong hệ thống. Trong mức này thì các quá trình còn lại của hệ thống sẽ điều được chia tách và hiển thị ra một cách cụ thể, làm cho mô hình có cái nhìn toàn diện và rõ ràng nhất.



Hình 3.4 Mô hình DFD mức 2

### 3.2.4 Thiết kế giao diện

#### 3.2.4.1 Sơ đồ website



Hình 3.5 Sơ đồ website

#### 3.2.4.2 Giao diện đăng nhập

Khi người dùng vừa vào website thì hệ thống sẽ kiểm tra xem người dùng đã đăng nhập tài khoản trước đó hay chưa, nếu như chưa đăng nhập thì hệ thống sẽ điều hướng người dùng tới trang đăng nhập và sẽ chặn không cho người dùng truy cập vào các trang khác khi chưa đăng nhập tài khoản.



LOGO	TÊN NGƯỜI DÙNG	ICON NGƯỜI DÙNG
ĐĂNG NHẬP		
ĐĂNG NHẬP TÀI KHOẢN		
<input type="text" value="MÃ SỐ"/>		
<input type="password" value="MẬT KHẨU"/>		
<input type="button" value="ĐĂNG NHẬP"/>		

**Hình 3.6 Giao diện đăng nhập**

#### 3.2.4.3 Giao diện trang chủ của sinh viên

Giao diện có tác dụng giúp cho các sinh viên có thể lựa chọn đăng ký các môn học và tra cứu điểm số các môn đã đăng ký trước đó.

Giao diện khi sinh viên vừa hoàn tất việc đăng nhập sẽ là giao diện trang chủ dành cho sinh viên, bên trong trang chủ sẽ bao gồm các phần như:

**Phần đầu trang:** chứa các thông tin giới thiệu như logo của website, icon hình đại diện của sinh viên và họ tên của sinh viên.

**Phần thanh điều hướng:** chứa các đường dẫn liên kết tới các trang khác của website gồm trang chủ, trang đăng ký môn học và trang đăng xuất.

**Phần nội dung:** Chứa toàn bộ nội dung chính của trang web, có chức năng hiển thị danh sách điểm của các môn học của từng học kỳ và giáo viên giảng dạy môn học đó kèm theo thời gian mà danh sách được công bố, trong đó sẽ chia ra làm 2 phần nhỏ hơn đó là phần hiển thị danh sách điểm mới nhất và phần hiển thị toàn bộ các danh sách điểm của website.



**Hình 3.7** Giao diện trang chủ của sinh viên

#### 3.2.4.4 Giao diện trang chủ của giảng viên

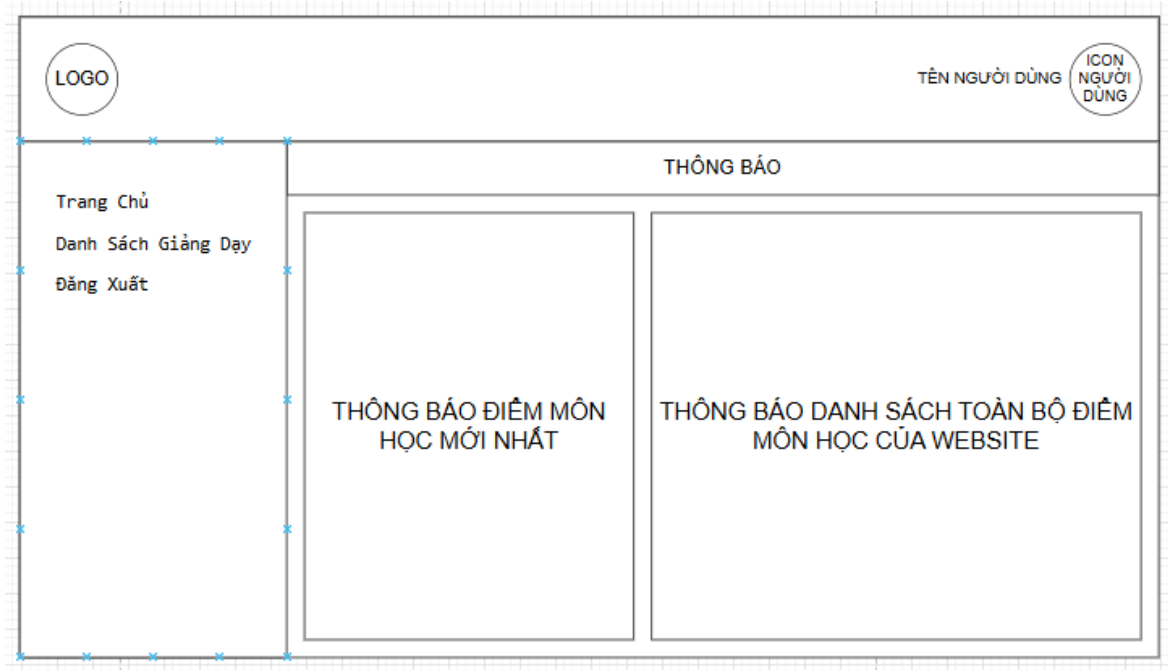
Giao diện giúp cho các giảng viên tra cứu các môn học, lớp học được nhà trường phân công giảng dạy và chấm điểm cho từng sinh viên trong lớp học mà họ được phân công trước đó.

Giao diện khi giảng viên vừa hoàn tất việc đăng nhập sẽ là giao diện trang chủ dành cho giảng viên, bên trong trang chủ sẽ bao gồm các phần như:

**Phần đầu trang:** chứa các thông tin giới thiệu như logo của website, icon hình đại diện của giảng viên và họ tên của giảng viên.

**Phần thanh điều hướng:** chứa các đường dẫn liên kết tới các trang khác của website gồm trang chủ, trang danh sách giảng dạy và trang đăng xuất.

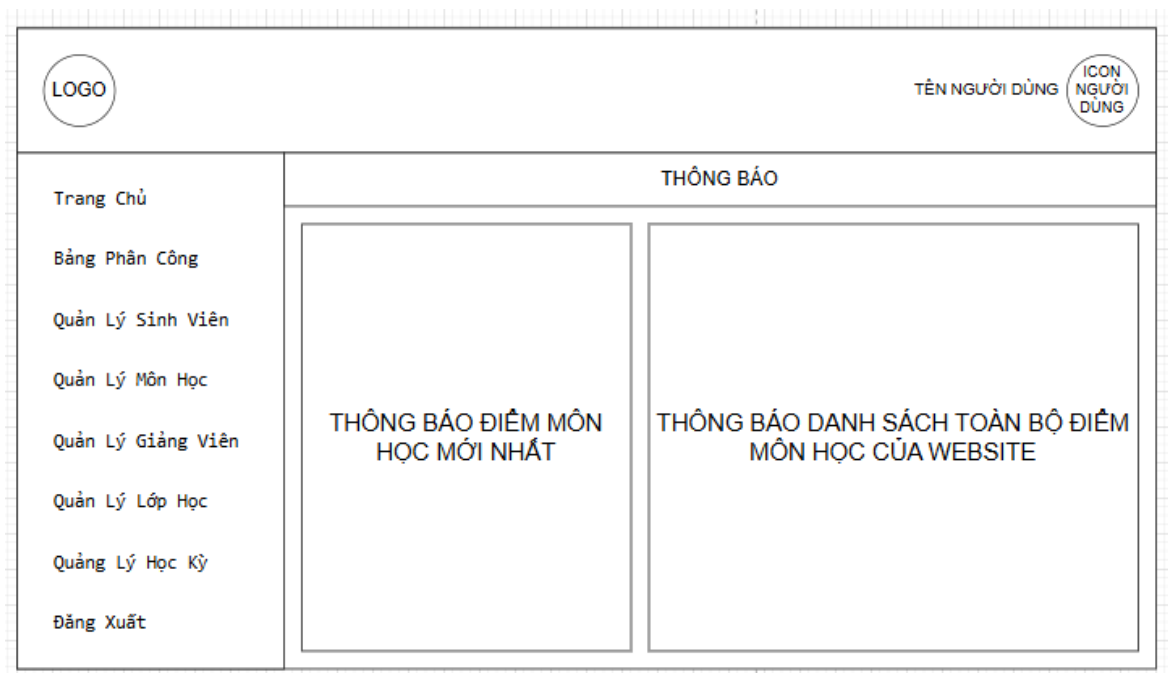
**Phần nội dung:** Chứa toàn bộ nội dung chính của trang web, có chức năng hiển thị danh sách điểm của các môn học của từng học kỳ và giáo viên giảng dạy môn học đó kèm theo thời gian mà danh sách được công bố, trong đó sẽ chia ra làm 2 phần nhỏ hơn đó là phần hiển thị danh sách điểm mới nhất và phần hiển thị toàn bộ các danh sách điểm của website.



**Hình 3.8 Giao diện trang chủ của giảng viên**

#### 3.2.4.5 Giao diện trang quản trị viên

Giao diện giúp cho các quản trị viên quản lý toàn bộ dữ liệu môn học, lớp học, sinh viên và giảng viên đồng thời quản trị viên cũng là người sẽ trực tiếp phân công lịch giảng dạy cho các giảng viên cũng như thêm vào các lớp học, môn học cho sinh viên có thể lựa chọn đăng ký.



**Hình 3.9 Giao diện trang chủ của quản trị viên**

#### 3.2.4.6 Giao diện trang quản lý lớp học

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện quản lý lớp học có một bảng lớn với các cột như mã lớp học, tên lớp và cột chức năng sửa, xóa thông tin lớp học kèm theo với đó là một nút bấm thêm lớp học, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu nhập các thông tin cần thiết để thêm một lớp học mới.

QUẢN LÝ LỚP HỌC		
<div>THÊM LỚP</div>		
ID	TÊN LỚP	CHỨC NĂNG
		SỬA, XÓA
		SỬA, XÓA
		SỬA, XÓA
		SỬA, XÓA
		SỬA, XÓA

**Hình 3.10** Giao diện quản lý lớp học

#### 3.2.4.7 Giao diện trang quản lý sinh viên

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện quản lý sinh viên có một bảng lớn với các cột như mã sinh viên, tên sinh viên, mật khẩu, giới tính, năm sinh, nơi sinh, ngành và cột chức năng sửa, xóa thông tin sinh viên kèm theo với đó là một nút bấm thêm sinh viên, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu nhập các thông tin cần thiết để thêm một sinh viên mới.

LOGO

TÊN NGƯỜI DÙNG

ICON NGƯỜI DÙNG

Trang Chủ

Bảng Phân Công

Quản Lý Sinh Viên

Quản Lý Môn Học

Quản Lý Giảng Viên

Quản Lý Lớp Học

Quản Lý Học Kỳ

Đăng Xuất

QUẢN LÝ SINH VIÊN

THÊM SINH VIÊN

TÌM KIẾM TÊN SINH VIÊN

ID	HỌ TÊN	MẬT KHẨU	GIỚI TÍNH	NĂM SINH	NƠI SINH	NGÀNH	CHỨC NĂNG
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA

**Hình 3.11** Giao diện quản lý sinh viên

#### 3.2.4.8 Giao diện trang quản lý học kỳ

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện quản lý học kỳ có một bảng lớn với các cột như mã học kỳ, tên học kỳ, năm học và cột chức năng sửa, xóa thông tin sinh viên kèm theo với đó là một nút bấm thêm học kỳ, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu nhập các thông tin cần thiết để thêm một học kỳ mới.

LOGO

TÊN NGƯỜI DÙNG

ICON NGƯỜI DÙNG

Trang Chủ

Bảng Phân Công

Quản Lý Sinh Viên

Quản Lý Môn Học

Quản Lý Giảng Viên

Quản Lý Lớp Học

Quản Lý Học Kỳ

Đăng Xuất

QUẢN LÝ HỌC KỲ

THÊM HỌC KỲ

ID	HỌC KỲ	NĂM HỌC	CHỨC NĂNG
			SỬA, XÓA
			SỬA, XÓA
			SỬA, XÓA
			SỬA, XÓA
			SỬA, XÓA

**Hình 3.12** Giao diện quản lý học kỳ

### 3.2.4.9 Giao diện trang quản lý giảng viên

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện quản lý giảng viên có một bảng lớn với các cột như mã giảng viên, tên giảng viên, mật khẩu, giới tính, năm sinh, nơi sinh, số điện thoại và cột chức năng sửa, xóa thông tin giảng viên kèm theo với đó là một nút bấm thêm giảng viên, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu nhập các thông tin cần thiết để thêm một giảng viên mới.

QUẢN LÝ GIẢNG VIÊN							
THÊM GIẢNG VIÊN							TÌM KIẾM TÊN GIẢNG VIÊN
ID	HỌ TÊN	MẬT KHẨU	GIỚI TÍNH	NĂM SINH	NƠI SINH	SỐ ĐIỆN THOẠI	CHỨC NĂNG
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA
							SỬA, XÓA

**Hình 3.13** Giao diện quản lý giảng viên

### 3.2.4.10 Giao diện quản lý môn học

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện quản lý môn học có một bảng lớn với các cột như mã môn học, tên môn học, số tín chỉ thực hành, số tín chỉ lý thuyết và cột chức năng sửa, xóa thông tin môn học kèm theo với đó là một nút bấm thêm môn học, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu nhập các thông tin cần thiết để thêm một môn học mới.

LOGO

TÊN NGƯỜI DÙNG

ICON NGƯỜI DÙNG

QUẢN LÝ MÔN HỌC

THÊM MÔN HỌC

TÌM KIẾM TÊN MÔN HỌC

ID	MÔN HỌC	TÍN CHỈ THỰC HÀNH	TÍN CHỈ LÝ THUYẾT	CHỨC NĂNG
				SỬA, XÓA
				SỬA, XÓA
				SỬA, XÓA
				SỬA, XÓA
				SỬA, XÓA

Trang Chủ

Bảng Phân Công

Quản Lý Sinh Viên

Quản Lý Môn Học

Quản Lý Giảng Viên

Quản Lý Lớp Học

Quản Lý Học Kỳ

Đăng Xuất

**Hình 3.14 Giao diện quản lý môn học**

#### 3.2.4.11 Giao diện phân công giảng dạy

Là giao diện yêu cầu phải đăng nhập tài khoản quản trị viên để có thể truy cập được. Phần nội dung giao diện phân công giảng dạy có một bảng lớn với các cột như mã phân công, học kỳ, năm học, môn học, giảng viên, lớp và cột chức năng xóa thông tin kèm theo với đó là một nút bấm thêm phân công, khi bấm vào sẽ hiển thị ra biểu mẫu yêu cầu chọn các thông tin từ các bảng quản lý học, môn học, giảng viên và lớp học để thêm một dòng phân công mới.

ID	HỌC KỲ	NĂM HỌC	MÔN HỌC	GIẢNG VIÊN	LỚP	CHỨC NĂNG
						XÓA
						XÓA
						XÓA
						XÓA
						XÓA

**Hình 3.15** Giao diện phân công giảng dạy

#### 3.2.4.12 Giao diện chức năng sửa thông tin

Trang giao diện chức năng sửa thông tin được hệ thống điều hướng tới khi người dùng bấm chọn vào nút sửa thông tin trên cột chức năng của một bảng bất kỳ trong hệ thống quản lý.

**ĐIỀN THÔNG TIN**

**LƯU**

**Hình 3.16** Giao diện chức năng sửa thông tin



## CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU

### 4.1. Dữ liệu thử nghiệm

Bảng dữ liệu về thông tin cá nhân của sinh viên.

**Bảng 4.1 Bảng dữ liệu của trang quản lý sinh viên**

<i>ID</i>	<i>Họ Tên</i>	<i>Mật Khẩu</i>	<i>Giới Tính</i>	<i>Năm Sinh</i>	<i>Nơi Sinh</i>	<i>Ngành</i>
110122028	Liễu Kiện An	123	Nam	23/05/2004	Trà Vinh	Công Nghệ Thông Tin
110122056	Hồ Nguyễn Quốc Dũng	123	Nam	19/08/2004	Trà Vinh	Công Nghệ Thông Tin
110122058	Đào Công Duy	123	Nam	17/06/2004	Trà Vinh	Công Nghệ Thông Tin
110122059	Huỳnh Khánh Duy	123	Nam	29/12/2004	Trà Vinh	Công Nghệ Thông Tin
110122060	Lê Hà Duy	123	Nam	19/08/2004	Trà Vinh	Công Nghệ Thông Tin
110122061	Nguyễn Lê Duy	123	Nam	26/06/2003	Trà Vinh	Công Nghệ Thông Tin
110122062	Nguyễn Thanh Duy	123	Nam	26/02/2004	Trà Vinh	Công Nghệ Thông Tin
110122064	Trương Mỹ Duyên	123	Nữ	18/06/2004	Trà Vinh	Công Nghệ Thông Tin
110122054	Trần Lâm Phú Đức	123	Nam	18/01/2004	Trà Vinh	Công Nghệ Thông Tin
110122055	Trần Minh Đức	123	Nam	03/01/2004	Trà Vinh	Công Nghệ Thông Tin

Bảng dữ liệu về thông tin giảng viên.

**Bảng 4.2 Bảng dữ liệu của trang quản lý giảng viên**

<i>ID</i>	<i>Họ và Tên</i>	<i>Mật Khẩu</i>	<i>Giới Tính</i>	<i>Năm Sinh</i>	<i>Nơi Sinh</i>	<i>Số Điện Thoại</i>
220000000	Trần Văn M	123	Nam	1988	Cần Thơ	0789444111
220000001	Lê Thị V	123	Nữ	1986	Cà Mau	1234567811

Bảng dữ liệu về thông tin lớp học.

**Bảng 4.3 Bảng dữ liệu của trang quản lý lớp học**

<i>ID</i>	<i>Tên lớp</i>
DA22TTD	Công Nghệ Thông Tin

Bảng dữ liệu về thông tin học kỳ.

**Bảng 4.4 Bảng dữ liệu của trang quản lý học kỳ**

<i>ID</i>	<i>Học Kỳ</i>	<i>Năm Học</i>
001	Học Kỳ 1	2024 - 2025

Bảng dữ liệu về thông tin Môn Học.

**Bảng 4.5 Bảng dữ liệu của trang quản lý môn học**

<i>ID</i>	<i>Môn Học</i>	<i>Tín Chỉ Lý Thuyết</i>	<i>Tín Chỉ Thực Hành</i>
003	Thống Kê và Phân Tích Dữ Liệu	2	1
002	Phân Tích và Thiết Kế Hệ Thống Thông Tin	2	1
001	Đồ Án Cơ Sở Ngành	3	2

Bảng dữ liệu về thông tin phân công giảng dạy.

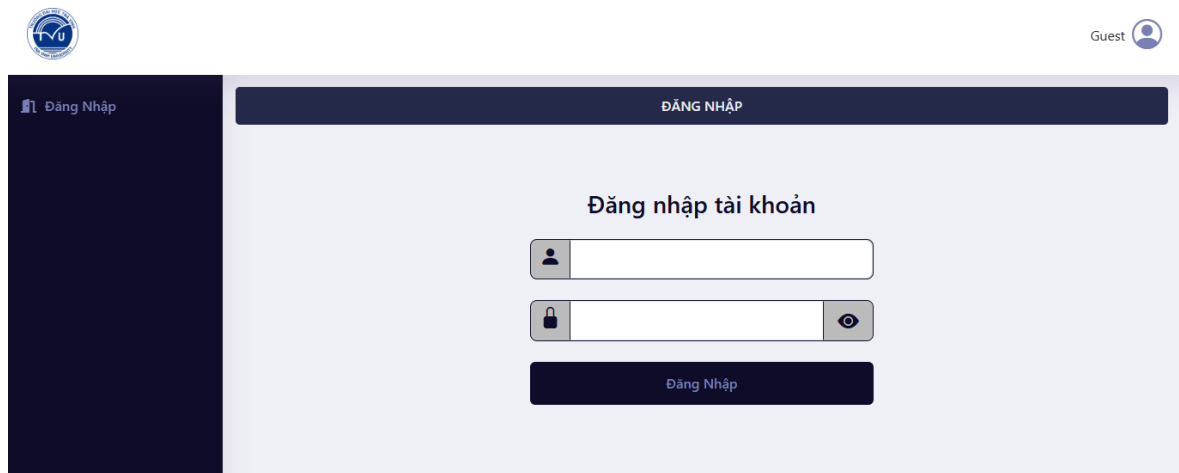
**Bảng 4.6 Bảng dữ liệu của trang phân công giảng dạy**

<i>ID</i>	<i>Học Kỳ</i>	<i>Môn Học</i>	<i>Giảng Viên</i>	<i>Lớp</i>
001	Học Kỳ 1 năm học 2024 - 2025	Phân Tích và Thiết Kế Hệ Thống Thông Tin	Trần Văn M	Công Nghệ Thông Tin

## 4.2. Kết quả thực nghiệm

### 4.2.1 Chức năng đăng nhập

Bất kỳ người dùng nào khi vừa truy cập vào website đều phải bắt buộc đăng nhập tài khoản để có thể thực hiện các tác vụ khác của trang web. Trang đăng nhập sẽ yêu cầu người dùng nhập vào 2 trường dữ liệu là mã số người dùng và mật khẩu, sau khi hệ thống xác nhận rằng dữ liệu vừa nhập vào hợp lệ thì người dùng sẽ được chuyển sang trang chứa các tác vụ khác.



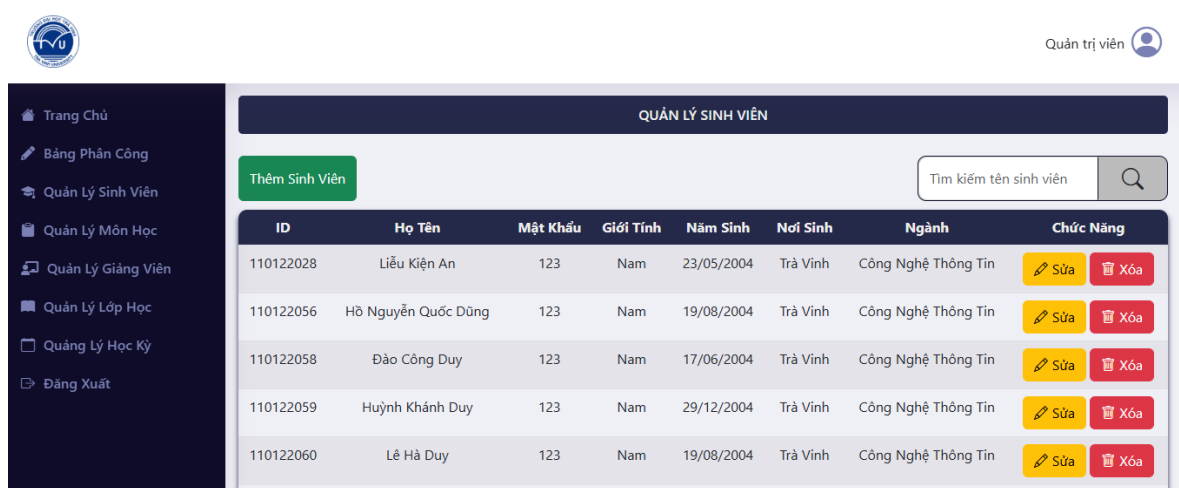
**Hình 4.1 Trang đăng nhập**

Ngoài ra tùy thuộc vào mã số mà người dùng cung cấp, hệ thống có thể xác định được chức vụ của người. Có 3 chức vụ chính trong hệ thống là quản trị viên, giảng viên và sinh viên.

Sau khi xác định được chức vụ người dùng, hệ thống sẽ dựa vào đó để phân quyền cho từng người dùng khác nhau dựa vào chức vụ của họ.

#### 4.2.2 Quản lý sinh viên

Thực hiện các chức năng thêm, sửa, xóa và tìm kiếm thông tin sinh viên.



ID	Họ Tên	Mật Khẩu	Giới Tính	Năm Sinh	Nơi Sinh	Ngành	Chức Năng
110122028	Liều Kiện An	123	Nam	23/05/2004	Trà Vinh	Công Nghệ Thông Tin	<a href="#">Sửa</a> <a href="#">Xóa</a>
110122056	Hồ Nguyễn Quốc Dũng	123	Nam	19/08/2004	Trà Vinh	Công Nghệ Thông Tin	<a href="#">Sửa</a> <a href="#">Xóa</a>
110122058	Đào Công Duy	123	Nam	17/06/2004	Trà Vinh	Công Nghệ Thông Tin	<a href="#">Sửa</a> <a href="#">Xóa</a>
110122059	Huỳnh Khánh Duy	123	Nam	29/12/2004	Trà Vinh	Công Nghệ Thông Tin	<a href="#">Sửa</a> <a href="#">Xóa</a>
110122060	Lê Hà Duy	123	Nam	19/08/2004	Trà Vinh	Công Nghệ Thông Tin	<a href="#">Sửa</a> <a href="#">Xóa</a>

**Hình 4.2 Trang quản lý sinh viên**

### 4.2.3 Quản lý giảng viên

Thực hiện các chức năng thêm, sửa, xóa và tìm kiếm thông tin giảng viên.



Hình 4.3 Trang quản lý giảng viên

### 4.2.4 Quản lý môn học

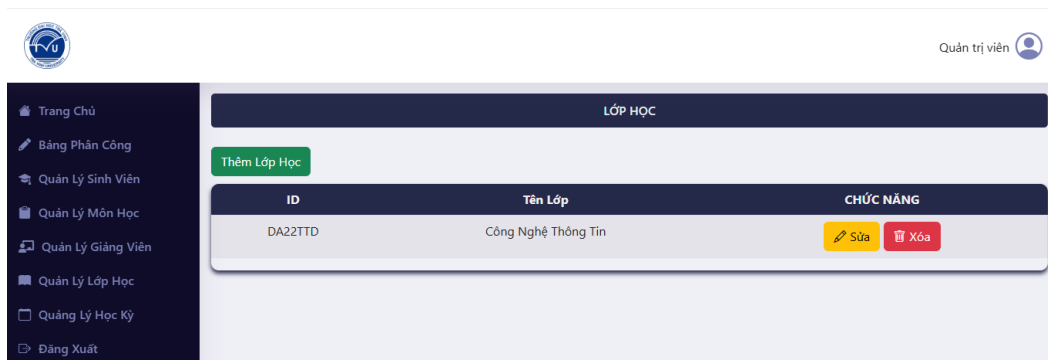
Thực hiện các chức năng thêm, sửa, xóa và tìm kiếm thông tin môn học.



Hình 4.4 Trang quản lý môn học

### 4.2.5 Quản lý lớp học

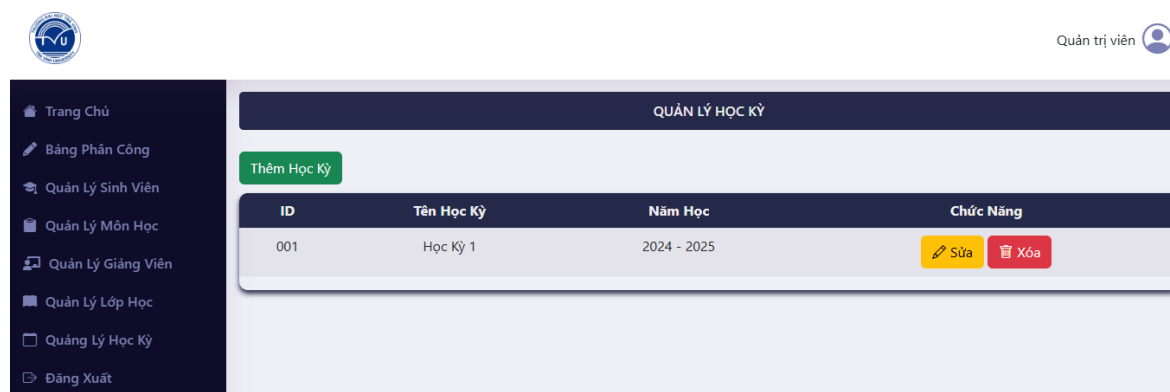
Thực hiện các chức năng thêm, sửa, xóa thông tin lớp học.



Hình 4.5 Trang quản lý lớp học

## 4.2.6 Quản lý học kỳ

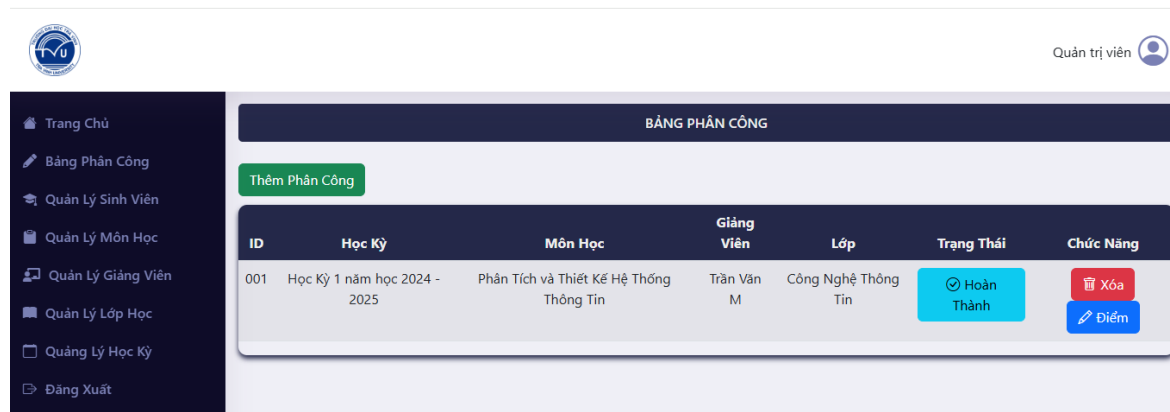
Thực hiện các chức năng thêm, sửa, xóa thông tin học kỳ.



Hình 4.6 Trang quản lý học kỳ

## 4.2.7 Phân công giảng dạy

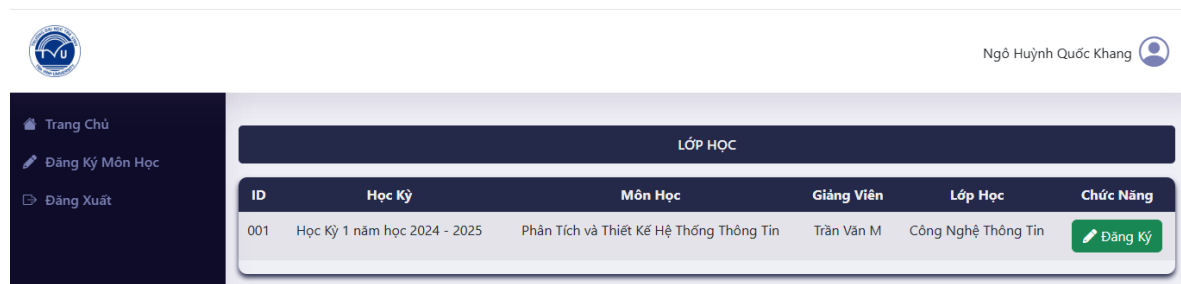
Đây là trang tổng hợp các dữ liệu của các bảng học kỳ, môn học, giảng viên, lớp học đã có trước đó để tạo thành một bảng mới giúp cho quản trị viên có thể quản lý các thông tin về việc giảng dạy các môn học của từng lớp, từng học kỳ và từng giáo viên.



Hình 4.7 Trang phân công giảng dạy

## 4.2.8 Đăng ký môn học

Sau khi sinh viên đăng nhập được vào trang web, hệ thống sẽ hiển thị ra danh sách các môn học được quản trị viên phân công trước đó để sinh viên có thể lựa chọn các môn học cần đăng ký.



**Hình 4.8 Trang đăng ký môn học**

#### 4.2.9 Quản lý kết quả học tập của sinh viên

Cho phép giảng viên thực hiện các tác vụ quản lý như chấm điểm, sửa điểm của các sinh viên đã đăng ký môn học của giảng viên đó.

Giảng viên sau khi hoàn thành việc quản lý điểm của từng sinh viên thì có thể xem xét đến trường hợp thực hiện chức năng công bố kết quả của sinh viên.

Kết quả khi công bố sẽ được hiển thị ở trang chủ.

ID	Tên Sinh Viên	Điểm Lần 1	Điểm Lần 2	Chức Năng
110122028	Liều Kiên An	7.8	không thi lại	<a href="#">Nhập Điểm</a>
110122056	Hồ Nguyễn Quốc Dũng	6.3	không thi lại	<a href="#">Nhập Điểm</a>
110122058	Đào Công Duy	8.6	không thi lại	<a href="#">Nhập Điểm</a>
110122059	Huỳnh Khánh Duy	7.5	không thi lại	<a href="#">Nhập Điểm</a>
110122060	Lê Hà Duy	5.8	không thi lại	<a href="#">Nhập Điểm</a>
110122061	Nguyễn Lê Duy	5.5	không thi lại	<a href="#">Nhập Điểm</a>
110122062	Nguyễn Thanh Duy	3.9	4.3	<a href="#">Nhập Điểm</a>
110122064	Trương Mỹ Duyên	6.2	không thi lại	<a href="#">Nhập Điểm</a>
110122054	Trần Lâm Phú Đức	3.3	4.3	<a href="#">Nhập Điểm</a>
110122055	Trần Minh Đức	6.3	không thi lại	<a href="#">Nhập Điểm</a>

**Hình 4.9 Trang quản lý kết quả học tập của sinh viên**

#### 4.2.10 Trang chủ

Trang chủ là trang hiển thị ra các thông báo về kết quả học tập được các giảng viên công bố sau khi đã hoàn thành việc chấm điểm cho sinh viên.

## Tìm hiểu ngôn ngữ XML và xây dựng ứng dụng web quản lý kết quả học tập sinh viên

Trang Chủ

Bảng Phân Công

Quản Lý Sinh Viên

Quản Lý Môn Học

Quản Lý Giảng Viên

Quản Lý Lớp Học

Quản Lý Học Kỳ

Đăng Xuất

THÔNG BÁO

Điểm môn Phân Tích và Thiết Kế Hệ Thống Thông Tin Lớp Công Nghệ Thông Tin của giảng viên Trần Văn M năm học Học Kỳ 1 năm học 2024 - 2025

2025-01-05

Hình 4.10 Trang chủ

Trang Chủ

Bảng Phân Công

Quản Lý Sinh Viên

Quản Lý Môn Học

Quản Lý Giảng Viên

Quản Lý Lớp Học

Quản Lý Học Kỳ

Đăng Xuất

Kết Quả Học Tập

ID	Tên Sinh Viên	Điểm Lần 1	Điểm Lần 2
110122028	Liều Kiện An	7.8	Không Thi Lại
110122056	Hồ Nguyễn Quốc Dũng	6.3	Không Thi Lại
110122058	Đào Công Duy	8.6	Không Thi Lại
110122059	Huỳnh Khánh Duy	7.5	Không Thi Lại
110122060	Lê Hà Duy	5.8	Không Thi Lại
110122061	Nguyễn Lê Duy	5.5	Không Thi Lại
110122062	Nguyễn Thanh Duy	3.9	4.3
110122064	Trương Mỹ Duyên	6.2	Không Thi Lại
110122054	Trần Lâm Phú Đức	3.3	4.3

Hình 4.11 Trang hiển thị kết quả học tập của sinh viên

## **CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **5.1. Kết luận**

Sau khi hoàn thành xây dựng hệ thống website quản lý kết quả học tập của sinh viên, tôi đã đạt được một số kết quả sau:

(1) Có được một giao diện và hệ thống website với các chức năng quản lý cơ bản như thêm, sửa, xóa và tìm kiếm.

(2) củng cố các kiến thức về HTML, CSS, JavaScript đồng thời học hỏi và tiếp thu các kiến thức mới về PHP, Bootstrap.

(3) Hiểu được cấu trúc cơ bản của một tập tin XML và cách thức xử lý các dữ liệu trong tập tin XML bằng SimpleXML và XML DOM thông qua PHP. Tổ chức quản lý, sắp xếp các tập tin XML để lưu trữ dữ liệu.

### **5.2. Hướng phát triển**

Cải thiện thêm hệ thống quản lý điểm như thêm vào chức năng tìm kiếm một sinh viên cụ thể trong danh sách, giúp cho giảng viên có thể nhanh chóng xác định sinh viên cần nhập điểm. Ngoài ra cần cải thiện thêm tính năng chọn nhiều sinh viên cùng lúc để giảng viên có thể nhập điểm cho nhiều sinh viên thay vì một sinh viên.

Tìm hiểu và phát triển thêm chức năng thông báo mỗi khi có kết quả học tập mới được công bố giúp cho sinh viên nhận được tin tức một cách sớm nhất, cụ thể hơn thì sẽ thêm vào một biểu tượng thông báo sẽ hoạt động theo thời gian thực, để khi có kết quả mới nhất đến từ hệ thống thì biểu tượng sẽ hiển thị thông báo đó đến sinh viên.

Gia tăng khả năng bảo mật cho hệ thống, cụ thể là trong các quá trình truy suất dữ liệu và đăng nhập tài khoản.



## TÀI LIỆU THAM KHẢO

- [1] W3Schools, "HTML Tutorial," [Online]. Available: <https://www.w3schools.com/html/>. [Accessed 15 11 2024].
- [2] W3Schools, "CSS Tutorial," [Online]. Available: <https://www.w3schools.com/css/>. [Accessed 15 11 2024].
- [3] W3Schools, "JavaScript Tutorial," [Online]. Available: <https://www.w3schools.com/js/>. [Accessed 15 11 2024].
- [4] W3Schools, "PHP Tutorial," [Online]. Available: <https://www.w3schools.com/php/>. [Accessed 15 11 2024].
- [5] W3Schools, "XML Tutorial," [Online]. Available: <https://www.w3schools.com/xml/>. [Accessed 15 11 2024].
- [6] W3Schools, "Bootstrap 5 Tutorial," [Online]. Available: <https://www.w3schools.com/bootstrap5/>. [Accessed 15 11 2024].
- [7] L. Q. Khải, “Tìm hiểu mô hình MVC dành cho người mới bắt đầu: Cấu trúc và ví dụ,” [Trực tuyến]. Available: <https://viblo.asia/p/tim-hieu-mo-hinh-mvc-danh-cho-nguoi-moi-bat-dau-cau-truc-va-vi-du-V3m5WLDyKO7>. [Đã truy cập 17 11 2024].