**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**XÂY DỰNG WEBSITE HỖ TRỢ DẠY VÀ HỌC MÔN CƠ SỞ DỮ LIỆU**

**UTEHY.S.2024.006**

THUỘC NHÓM NGÀNH: CÔNG NGHỆ THÔNG TIN

**HƯNG YÊN – 2024**

**NHẬN XÉT**

**Nhận xét của giảng viên hướng dẫn:**

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**LỜI CAM ĐOAN**

Em xin cam đoan đồ án “Xây dựng website hỗ trợ dạy và học môn Cơ sở dữ liệu” là kết quả thực hiện của bản thân em dưới sự hướng dẫn của cô Trịnh Thị Nhị

Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong đồ án và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiện.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

*Hưng Yên, ngày … tháng … năm…..*

SINH VIÊN

. ..……………………………..

**LỜI CẢM ƠN**

**MỤC LỤC**

[DANH MỤC CÁC THUẬT NGỮ 7](#_heading=h.2et92p0)

[DANH MỤC CÁC BẢNG 8](#_heading=h.tyjcwt)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ 9](#_heading=h.3dy6vkm)

[CHƯƠNG 1:](#_heading=h.2u6wntf) TỔNG QUAN VỀ ĐỀ TÀI 10

[1.1. Lý do chọn đề tài 10](#_heading=h.19c6y18)

[1.2. Mục tiêu của đề tài 10](#_heading=h.3tbugp1)

[1.2.1 Mục tiêu tổng quát 10](#_heading=h.28h4qwu)

[1.2.2 Mục tiêu cụ thể 10](#_heading=h.nmf14n)

[1.3. Giới hạn và phạm vi của đề tài 10](#_heading=h.37m2jsg)

[1.3.1 Đối tượng nghiên cứu 10](#_heading=h.1mrcu09)

[1.3.2 Phạm vi nghiên cứu 11](#_heading=h.46r0co2)

[1.4. Nội dung thực hiện 11](#_heading=h.2lwamvv)

[1.5. Phương pháp tiếp cận 11](#_heading=h.111kx3o)

[CHƯƠNG 2:](#_heading=h.3l18frh) CƠ SỞ LÝ THUYẾT 12

[2.1. Quy trình phát triển phần mềm 12](#_heading=h.z337ya)

[2.2. Phương pháp phân tích thiết kế hướng đối tượng 12](#_heading=h.3j2qqm3)

[2.3. Giới thiệu các xu hướng lập trình web mới 12](#_heading=h.1y810tw)

[2.4. Tổng quan về Angular 12](#_heading=h.4i7ojhp)

[2.5. Lập trình Web API NET CORE 12](#_heading=h.2xcytpi)

[CHƯƠNG 3:](#_heading=h.1ci93xb) PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 13

[3.1 Đặc tả yêu cầu phần mềm 13](#_heading=h.206ipza)

[3.1.1 Các yêu cầu chức năng 13](#_heading=h.4k668n3)

[3.1.2 Biểu đồ lớp thực thể 13](#_heading=h.2zbgiuw)

[3.1.3 Các yêu cầu phi chức năng 13](#_heading=h.1egqt2p)

[3.2 Thiết kế hệ thống 13](#_heading=h.3ygebqi)

[3.2.1 Thiết kế cơ sở dữ liệu 13](#_heading=h.1pxezwc)

[3.2.2 Thiết kế lớp đối tượng 13](#_heading=h.2dlolyb)

[3.2.3 Thiết kế giao diện 13](#_heading=h.sqyw64)

[CHƯƠNG 4:](#_heading=h.147n2zr) TRIỂN KHAI WEBSITE 14

[4.1 Triển khai các chức năng cho phân hệ người dùng 14](#_heading=h.3cqmetx)

[4.1.1 Trang chủ (Ví dụ) 14](#_heading=h.1rvwp1q)

[4.1.2 Trang Xem Sản phẩm (Ví dụ) 15](#_heading=h.4bvk7pj)

[4.1.3 Trang Quản lý giỏ (Ví dụ) 15](#_heading=h.2r0uhxc)

[4.2 Triển khai các chức năng cho phân hệ quản trị nội dung (nếu có) 15](#_heading=h.1664s55)

[4.3 Kiểm thử và triển khai ứng dụng 15](#_heading=h.3q5sasy)

[4.3.1 Kiểm thử 15](#_heading=h.25b2l0r)

[4.3.2 Đóng gói ứng dụng 15](#_heading=h.kgcv8k)

[4.3.3 Triển khai ứng dụng 15](#_heading=h.34g0dwd)

[KẾT LUẬN 16](#_heading=h.1jlao46)

[TÀI LIỆU THAM KHẢO 17](#_heading=h.43ky6rz)

**DANH MỤC CÁC THUẬT NGỮ**

| **STT** | **Từ viết tắt** | **Cụm từ tiếng anh** | **Diễn giải** |
| --- | --- | --- | --- |
| 1 | MVC | Model View Control | Mẫu thiết kế MVC |
| ………. | ………. | ………. | ………. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**DANH MỤC CÁC BẢNG**

[Bảng 31: Thiết kế bảng Skins lưu trữ các loại khung giao diện](#_heading=h.2iq8gzs) 15

[Bảng 32: Thiết kế bảng lưu trữ các Module được thiết kế](#_heading=h.xvir7l) 15

**DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ**

[Hình 31: Mô hình kiến trúc tổng thể hệ thống SEVN Framework](#_heading=h.3hv69ve) 14

[Hình 32: Mô hình kiến trúc chi tiết của hệ thống SEVN Framework](#_heading=h.1x0gk37) 14

[Hình 33: Mô hình lưu trữ cấu trúc website trên Database](#_heading=h.4h042r0) 14

1. **TỔNG QUAN VỀ ĐỀ TÀI**

**1.1. Lý do chọn đề tài**

Công nghệ thông tin là tập hợp các phương pháp khoa học và các công cụ kỹ

thuật hiện đại, chủ yếu là kỹ thuật máy tính và viễn thơng nhằm tổ chức khai thác vàsử dụng có hiệu quả các nguồn tài nguyên thông tin phong phú, tiềm năng trong mọi lĩnh vực hoạt động của con người và xã hội.

Ngày nay, công nghệ thông tin phát triển nhanh chóng và ứng dụng vào tất cả các lĩnh vực, có thể nói công nghệ thông tin trở thành thước đo để đánh giá sự phát triển của xã hội hiện đại – nơi mà con người đang từ bỏ cách làm việc thủ công, tiến đến tin học hóa trong tất cả các lĩnh vực, để công việc thực hiện có hiệu quả hơn, tiết kiệm thời gian và nhân lực hơn.

Đề tài "**Xây dựng website hỗ trợ dạy và học môn Cơ sở dữ liệu**" được lựa chọn với những lý do cân nhắc kỹ lưỡng, nhằm đáp ứng nhu cầu ngày càng cao về kiến thức Cơ sở dữ liệu trong cộng đồng học sinh và sinh viên. Trong bối cảnh môi trường học tập ngày càng chuyển đổi sang hình thức trực tuyến, việc xây dựng một nền tảng trực tuyến sẽ mang lại nhiều lợi ích đối với cả giáo viên và học sinh.

Website này sẽ không chỉ giúp phổ cập kiến thức một cách hiệu quả mà còn tạo ra một môi trường học tập tương tác, thú vị, và linh hoạt. Việc tích hợp các công cụ như diễn đàn, thảo luận trực tuyến sẽ khuyến khích sự giao tiếp giữa giáo viên và sinh viên, cũng như giữa những sinh viên có cùng sở thích và mục tiêu học tập.

Quản lý tài liệu sẽ trở nên dễ dàng hơn, giúp sinh viên tiếp cận thông tin nhanh chóng và có thể thực hành thêm thông qua bài tập và dự án mẫu được cung cấp trên nền tảng. Sự linh hoạt của website cho phép sinh viên tự quản lý học tập theo tốc độ cá nhân, đồng thời theo dõi tiến trình học tập một cách đơn giản và hiệu quả.

Cuối cùng, việc xây dựng một giao diện thân thiện, hỗ trợ nhiều ngôn ngữ và đặt ưu tiên cao về an toàn thông tin sẽ tạo điều kiện thuận lợi nhất cho quá trình học tập, đồng thời đảm bảo tính bảo mật và quyền riêng tư của tất cả người sử dụng trên nền tảng này. Đề tài này không chỉ đáp ứng nhu cầu hiện tại mà còn hướng tới tương lai của giáo dục 4.0, nơi công nghệ thông tin đóng vai trò quan trọng trong việc nâng cao chất lượng giáo dục.

**1.2. Mục tiêu của đề tài**

***1.2.1 Mục tiêu tổng quát***

Mục tiêu tổng quát của đề tài là xây dựng một website hỗ trợ dạy và học môn Cơ sở dữ liệu nhằm nâng cao chất lượng giáo dục và học tập trong lĩnh vực này. Đặc biệt, chúng tôi hướng đến việc cung cấp một nền tảng linh hoạt, tương tác và hiệu quả, kết nối sinh viên và giáo viên để họ có thể chia sẻ kiến thức, thảo luận về các vấn đề liên quan đến Cơ sở dữ liệu, và áp dụng kiến thức vào thực tế.

***1.2.2 Mục tiêu cụ thể***

Hướng đến việc xây dựng một môi trường học tập đa chiều và tích cực, đề tài đặt ra những mục tiêu cụ thể sau:

* Tạo Môi Trường Học Tập Tương Tác:Phát triển các tính năng tương tác như diễn đàn, thảo luận trực tuyến để khuyến khích sự giao tiếp và chia sẻ kiến thức giữa sinh viên và giáo viên.
* Quản Lý Tài Liệu Hiệu Quả: Xây dựng hệ thống quản lý tài liệu thông minh, giúp sinh viên và giáo viên dễ dàng truy cập, chia sẻ và quản lý tài liệ
* Thực Hành Thêm và Dự Án Mẫu: Cung cấp bài tập thực hành và dự án mẫu để sinh viên có thể áp dụng kiến thức vào các tình huống thực tế, nâng cao khả năng ứng dụng.
* Theo Dõi Tiến Trình Học Tập: Tích hợp công cụ theo dõi tiến trình học tập để giáo viên và sinh viên có thể đánh giá hiệu suất và tiến bộ học tập một cách chi tiết.

Mục tiêu cụ thể này sẽ giúp đảm bảo rằng website không chỉ đáp ứng các yêu cầu cơ bản về học tập mà còn đáp ứng các yêu cầu và mong muốn của cộng đồng học Cơ sở dữ liệu, tạo ra một môi trường học tập đầy đủ và tích cực.

**1.3. Giới hạn và phạm vi của đề tài**

***1.3.1 Đối tượng nghiên cứu***

* Đối tượng nghiên cứu: Hệ thống website hỗ trợ dạy và học môn Cơ sở dữ liệu.
* Khách thể nghiên cứu: Sinh viên, giáo viên, quản trị viên hệ thống,....

***1.3.2 Phạm vi nghiên cứu***

* Phạm vi không gian: Khoa CNTT trường Đại học SPKT Hưng Yên.
* Phạm vi thời gian: Các thông tin và số liệu được thu thập trong suốt quá trình nghiên cứu thực hiện đồ án ( Từ tháng 9 năm 2023 đến tháng 5 năm 2024)
* Ý nghĩa khoa học và thực tiễn của đề tài: Trong giáo dục, dự án giúp nâng cao chất lượng qua tài liệu và kiểm tra trực tuyến. Sinh viên được hỗ trợ tối ưu hóa học với tài nguyên đa phương tiện và công cụ học trực tuyến. Đồng thời, giảng viên tiết kiệm thời gian với quản lý dữ liệu tự động. Dự án kết hợp giáo dục và công nghệ, tạo cầu nối giữa lý thuyết và thực tế trong học tập.

**1.4. Nội dung thực hiện**

Dự án “Xây dựng website hỗ trợ dạy và học môn Cơ sở dữ liệu” gồm các nội dung công việc và kết quả đạt được cụ thể bên dưới:

* Khảo sát và phân tích thiết kế hệ thống
* Thiết kế đặc tả hệ thống
* Xây dựng cơ sở dữ liệu
* Thiết kế và đặc tả giao diện
* Lập trình hệ thống
* Triển khai thực nghiệm hệ thống trên mạng Internet

**1.5. Phương pháp tiếp cận**

* Gặp gỡ trực tiếp, phỏng vấn và trao đổi giáo viên, sinh viên về phân tích thiết kế hệ thống xây dựng.
* Đọc tài liệu cung cấp thông tin về nghiệp vụ xây dựng website hỗ trợ dạy và học cho trường học và các nội dung liên quan.
* Tham hệ thống website hỗ trợ dạy và học các môn học của các doanh nghiệp khác.

1. **CƠ SỞ LÝ THUYẾT**

**2.1. Quy trình phát triển phần mềm**

Quy trình xây dựng phần mềm bao gồm tập hợp các thao tác và kết quả

tương quan sử dụng trong việc phát triển để sản xuất ra một xe máy phần mềm. Mỗi

một phần mềm lại có đặc điểm và yêu cầu khác nhau, tuy nhiên tất cả đều cần phải

trải qua các bước sau:

**Bước 1: Phân tích yêu cầu**

Phân tích yêu cầu là công việc bao gồm các tác vụ xác định yêu cầu cho một hệ thống mới hoặc được thay đổi dựa trên cơ sở là các nhu cầu trong quá trình sử dụng.Việc phân tích yêu cầu có ý nghĩa quan trọng đối với thành công của một dự án.Các yêu cầu phải có tính đo được, kiểm thử được, có liên quan đến các nhu cầu

hoặc cơ hội doanh nghiệp đã được xác định, và phải được định nghĩa ở mức độ chi

tiết đủ cho việc thiết kế hệ thống.

**Bước 2: Thiết kế phần mềm**

Là một quá trình giải quyết vấn đề và lập kế hoạch cho một giải pháp phần

mềm. Sau khi các mục đích và các đặc điểm kỹ thuật của phần mềm được giải quyết, lập trình viên sẽ thiết kế hoặc thuê người thiết kế để phát triển một kế hoạch cho giải pháp phần mềm. Nó bao gồm các thành phần cấp thấp, các vấn đề thuật toán cũng như một khung nhìn kiến trúc. Thiết kế chức năng, cơ sở dữ liệu và giao diện.

**Bước 3: Lập trình máy tính**

Lập trình máy tính (gọi tắt là lập trình) là kỹ thuật cài đặt một hoặc nhiều thuật toán trừu tượng có liên quan với nhau bằng một hoặc nhiều ngôn ngữ lập trình để tạo ra một chương trình máy tính có các thành tố nghệ thuật, khoa học, toán học, kỹ nghệ. Các ngôn ngữ lập trình khác nhau hỗ trợ các phong cách lập trình khác nhau.Một phần của công việc lập trình là việc lựa chọn một trong những ngôn ngữ phù hợp nhất với các bài toán cần giải quyết. Các ngôn ngữ lập trình khác nhau đòi hỏi lập trình viên phải xử lý các chi tiết ở mức độ khác nhau khi cài đặt các thuật toán.Sự thống nhất trong các cách xử lý sẽ tạo thuận lợi cho việc lập trình và hiệu quả của chương trình.

**Bước 4: Kiểm thử phần mềm**

Kiểm thử phần mềm là một cuộc kiểm tra được tiến hành để cung cấp cho các bên liên quan thông tin về chất lượng của xe máy hoặc dịch vụ được kiểm thử.

Kiểm thử có thể cung cấp cho doanh nghiệp một quan điểm, một cách nhìn độc đáo

về phần mềm để từ đó đánh giá và thấu hiểu được những rủi ro trong quá trình triển

khai phần mềm. Tùy thuộc vào từng phương pháp, việc kiểm thử có thể được thực

hiển bất cứ lúc nào trong quá trình phát triển phần mềm. Theo truyền thống thì các

nỗ lực kiểm thử được tiến hành sau khi các yêu cầu được xác định và việc lập trình

được hoàn tất trong phương pháp phát triển “Agile” thì việc kiểm thử được tiến hành liên tục trong suốt quá trình xây dựng phần mềm. Như vậy, mỗi một phương

pháp kiểm thử bị chi phối theo một quy trình phát triển phần mềm nhất định.

**Bước 5: Triển khai phần mềm**

Sau khi phần mềm được kiểm thử và khắc phục những sai sót sẽ được triển

khai đưa vào sử dụng trong thực tế. Đối với những phần mềm thiết kế theo thỏa

thuận với khách hàng, việc triển khai đơn giản chỉ là hướng dẫn cho khách hàng

cách sử dụng đạt hiệu quả cao. Với những phần mềm mang tính thông dụng, việc

triển khai còn qua các chương trình giới thiệu và đưa xe máy ra thị trường. Trong

quá trình triển khai cũng luôn đánh giá hiệu quả sử dụng của phần mềm, xem xét

những nhược điểm để lên kế hoạch thiết kế phần mềm hiệu quả hơn.

**Bước 6: Bảo trì phần mềm**

Bảo trì phần mềm bao gồm điều chỉnh các lỗi mà chưa được phát hiện trong

các giai đoạn trước chu kỳ sống của phần mềm, nâng cấp tính năng sử dụng và an

toàn vận hành của phần mềm. Bảo trì phần mềm có thể chiếm đến 65%-75% công

sức chu kỳ sống của một phần mềm. Quá trình phát triển phần mềm bao gồm rất

nhiều giai đoạn: thu thập yêu cầu, phân tích, xây dựng, kiểm tra, triển khai và bảo

trì phần mềm. Nhiệm vụ của giai đoạn bảo trì phần mềm là giữ cho phần mềm được

cập nhật khi môi trường thay đổi và yêu cầu người sử dụng thay đổi. Mỗi một giai

đoạn xây dựng phần mềm lại đòi hỏi các kỹ năng phân tích và ứng dụng kiến thức

công nghệ khác nhau. Để xây dựng phần mềm thiết thực, mang lại hiệu quả kinh tế

cao đòi hỏi con người lập trình viên phải tuân thủ các yêu cầu trong từng giai đoạn

thiết kế.

**2.2. Phương pháp phân tích thiết kế hướng đối tượng**

**Khái niệm OOAD (Object Oriented Analysis and Design)**

Phân tích và thiết kế hướng đối tượng là một kỹ thuật tiếp cận phổ biến dùng để phân tích, thiết kế một ứng dụng, hệ thống. Nó dựa trên bộ các nguyên tắc chung, đó là một tập các hướng dẫn để giúp chúng ta tránh khỏi một thiết kế xấu. 5 nguyên tắc SOLID trong thiết kế hướng đối tượng:

* Một lớp chỉ nên có một lý do để thay đổi, tức là một lớp chỉ nên xử lý một chức năng đơn lẻ, duy nhất thôi. Nếu đặt nhiều chức năng vào trong một lớp sẽ dẫn đến sự phụ thuộc giữa các chức năng với nhau và mặc dù sau đó ta chỉ thay đổi ở một chức năng thì cũng phá vỡ các chức năng còn lại.
* Các lớp, module, chức năng nên dễ dàng Mở (Open) cho việc mở rộng (thêm chức năng mới) và Đóng (Close) cho việc thay đổi.
* Lớp dẫn xuất phải có khả năng thay thế được lớp cha của nó.
* Chương trình không nên buộc phải cài đặt một interface mà nó không sử dụng đến.
* Các module cấp cao không nên phụ thuộc vào các module cấp thấp. Cả hai nên phụ thuộc thông qua lớp trừu tượng. Lớp trừu tượng không nên phụ thuộc vào chi tiết. Chi tiết nên phụ thuộc vào trừu tượng

## Khái niệm UML

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư v.v..

OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.

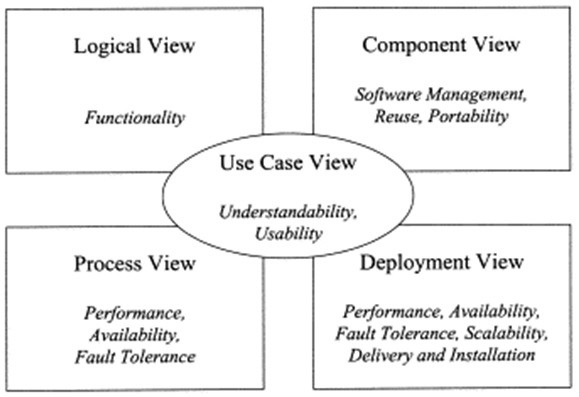
UML sử dụng để vẽ cho nhiều lĩnh vực khác nhau như phần mềm, cơ khí, xây dựng,...Sau đây chúng ta sẽ nghiên cứu cách sử dụng UML cho phân tích và thiết kế hướng đối tượng trong ngành phần mềm. OOAD sử dụng UML bao gồm các thành phần sau:

* View (góc nhìn)
* Diagram (bản vẽ)
* Notations (ký hiệu)
* Mechanisms (quy tắc, cơ chế)

Chúng ta sẽ tìm hiểu kỹ hơn các thành phần trên.

### View (góc nhìn)

Mỗi góc nhìn như thầy bói xem voi, nó không thể hiện hết hệ thống nhưng thể hiện rõ hệ thống ở một khía cạnh. Chính vì thế trong xây dựng có bản vẽ kiến trúc (nhìn về mặt kiến trúc), bản vẽ kết cấu (nhìn về mặt kết cấu), bản vẽ thi công (nhìn về mặt thi công). Trong phần mềm cũng như vậy, OOAD sử dụng UML có các góc nhìn sau:



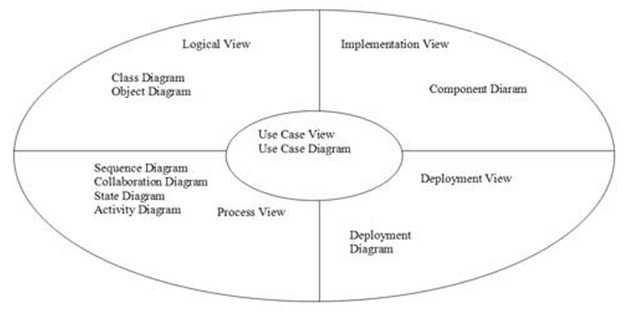
Trong đó:

* Use Case View: cung cấp góc nhìn về các ca sử dụng giúp chúng ta hiểu hệ thống có gì? ai dùng và dùng nó như thế nào.
* Logical View: cung cấp góc nhìn về cấu trúc hệ thống, xem nó được tổ chức như thế nào. Bên trong nó có gì.
* Process View: cung cấp góc nhìn động về hệ thống, xem các thành phần trong hệ thống tương tác với nhau như thế nào.
* Component View: Cũng là một góc nhìn về cấu trúc giúp chúng ta hiểu cách phân bổ và sử dụng lại các thành phần trong hệ thống ra sao.
* Deployment View: cung cấp góc nhìn về triển khai hệ thống, nó cũng ảnh hưởng lớn đến kiến trúc hệ thống.

Tập hợp các góc nhìn này sẽ giúp chúng ta hiểu rõ hệ thống cần phân tích, thiết kế. Trong hình trên chúng ta thấy góc nhìn Use Case View nằm ở giữa và chi phối tất cả các góc nhìn còn lại. Chính vì thế chúng ta thường thấy các tài liệu nói về 4 view + 1 chứ không phải 5 view nhằm nhấn mạnh vai trò của Use Case View.

### Diagram (Bản vẽ)

Diagram chúng ta có thể dịch là sơ đồ. Tuy nhiên ở đây chúng ta sử dụng từ bản vẽ cho dễ hình dung. Các bản vẽ được dùng để thể hiện các góc nhìn của hệ thống.

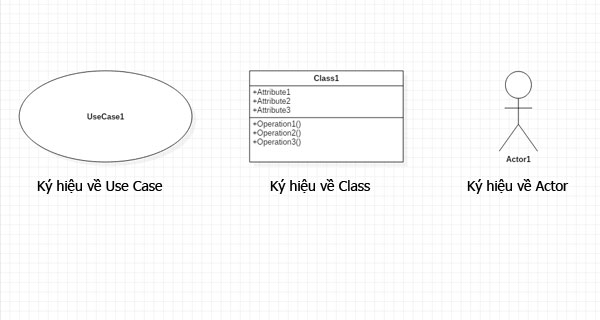


Trong đó:

* Use Case Diagram: bản vẽ mô tả về ca sử dụng của hệ thống. Bản vẽ này sẽ giúp chúng ta biết được ai sử dụng hệ thống, hệ thống có những chức năng gì. Lập được bản vẽ này chúng ta sẽ hiểu được yêu cầu của hệ thống cần xây dựng.
* Class Diagram: bản vẽ này mô tả cấu trúc của hệ thống, tức hệ thống được cấu tạo từ những thành phần nào. Nó mô tả khía cạnh tĩnh của hệ thống.
* Object Diagram: Tương tự như Class Diagram nhưng nó mô tả đến đối tượng thay vì lớp (Class).
* Sequence Diagram: là bản vẽ mô tả sự tương tác của các đối tượng trong hệ thống với nhau được mô tả tuần tự các bước tương tác theo thời gian.
* Collaboration Diagram: tương tự như sequence Diagram nhưng nhấn mạnh về sự tương tác thay vì tuần tự theo thời gian.
* State Diagram: bản vẽ mô tả sự thay đổi trạng thái của một đối tượng. Nó được dùng để theo dõi các đối tượng có trạng thái thay đổi nhiều trong hệ thống.
* Activity Diagram: bản vẽ mô tả các hoạt động của đối tượng, thường được sử dụng để hiểu về nghiệp vụ của hệ thống.
* Component Diagram: bản vẽ mô tả về việc bố trí các thành phần của hệ thống cũng như việc sử dụng các thành phần đó.
* Deployment Diagram: bản vẽ mô tả việc triển khai của hệ thống như việc kết nối, cài đặt, hiệu năng của hệ thống v.v…

### Notations (các ký hiệu)

Notations là các ký hiệu để vẽ, nó như từ vựng trong ngôn ngữ tự nhiên. Chúng ta phải biết từ vựng thì mới ghép thành câu, thành bài được. Chúng ta sẽ tìm hiểu kỹ các notations trong từng bản vẽ sau này. Dưới đây là vài ví dụ về notation.



Hình trên là ví dụ về ký hiệu của Use Case, Class, Actor, ngoài ra còn rất nhiều các ký hiệu khác

### Mechanisms (Rules)

Mechanisms là các quy tắc để lập nên bản vẽ, mỗi bản vẽ có quy tắc riêng và chúng ta phải nắm được để tạo nên các bản vẽ thiết kế đúng. Các quy tắc này chúng ta sẽ bàn kỹ trong các bài về các bản vẽ.

**2.3. Giới thiệu các xu hướng lập trình web mới**

Lập trình web đang ngày càng trở nên đa dạng và phức tạp, với sự xuất hiện của nhiều công nghệ và xu hướng mới. Dưới đây là một số xu hướng lập trình web mới đang thu hút sự chú ý của cộng đồng phát triển web.

**JAMstack (JavaScript, APIs, và Markup):**

JAMstack là một phương pháp phát triển web tập trung vào việc sử dụng JavaScript để tạo các trang tĩnh, kết hợp với việc sử dụng API để đưa dữ liệu vào trang. Các trang JAMstack thường nhanh, an toàn và dễ quản lý, đồng thời giảm tải cho máy chủ và tăng trải nghiệm người dùng.

**Serverless Computing:**

Serverless đang trở thành một xu hướng quan trọng trong lập trình web. Với kiến trúc này, các nhà phát triển không cần quan tâm đến quản lý máy chủ. Thay vào đó, họ tập trung vào việc viết mã và triển khai ứng dụng mà không cần lo lắng về hạ tầng.

**PWA (Progressive Web Apps):**

Progressive Web Apps đang ngày càng trở nên phổ biến vì khả năng cung cấp trải nghiệm người dùng tương tự như ứng dụng di động truyền thống. PWA có thể hoạt động offline, có khả năng đồng bộ dữ liệu tự động và tận dụng tính năng của trình duyệt để cung cấp trải nghiệm tốt nhất.

**GraphQL:**

GraphQL, một ngôn ngữ truy vấn dữ liệu, đang ngày càng được ưa chuộng bởi sự linh hoạt và khả năng tối ưu hóa truy vấn dữ liệu. Thay vì nhận toàn bộ dữ liệu từ một API như REST, GraphQL cho phép lấy chính xác những dữ liệu cần thiết, giảm lượng dữ liệu không cần thiết và tăng hiệu suất.

**Single Page Applications (SPA) Frameworks:**

Các framework SPA như React, Angular và VueJS vẫn tiếp tục là xu hướng quan trọng trong lập trình web. Chúng tập trung vào việc tạo ra ứng dụng web mà không cần tải lại trang, mang lại trải nghiệm người dùng mượt mà và nhanh chóng.

**2.4. Tổng quan về Angular**

Angular là một framework phát triển web mạnh mẽ và linh hoạt, được phát triển và duy trì bởi đội ngũ Google. Được ra mắt lần đầu tiên vào năm 2010 dưới tên gọi AngularJS, Angular đã trải qua nhiều cải tiến và phát triển thành một framework front-end đầy đủ tính năng, phù hợp cho việc xây dựng ứng dụng web động và phức tạp.

Một trong những đặc điểm nổi bật nhất của Angular là kiến trúc MVVM (Model-View-ViewModel), giúp tách biệt rõ ràng giữa dữ liệu, giao diện người dùng và logic điều khiển. Điều này giúp dễ dàng quản lý và duy trì mã nguồn, đặc biệt là đối với các dự án lớn.

Angular cung cấp một loạt các tính năng mạnh mẽ như two-way data binding, dependency injection, modularization, và một hệ thống điều hướng giúp tạo ra các ứng dụng linh hoạt và dễ mở rộng. Two-way data binding giúp tự động đồng bộ hóa dữ liệu giữa model và view, giảm thiểu lượng mã nguồn cần phải viết và duy trì.

Dependency injection trong Angular giúp quản lý và triển khai các thành phần của ứng dụng một cách hiệu quả, đồng thời tạo điều kiện thuận lợi cho việc kiểm thử (testing) và tái sử dụng mã nguồn. Hơn nữa, Angular có thể tích hợp dễ dàng với nhiều thư viện và công nghệ khác, đặc biệt là các thư viện của hệ sinh thái JavaScript.

Angular không chỉ hỗ trợ xây dựng ứng dụng web truyền thống mà còn có khả năng xây dựng ứng dụng Progressive Web Apps (PWA) và mobile apps sử dụng framework Ionic. Điều này làm cho Angular trở thành một lựa chọn đáng xem cho các dự án đa dạng và phức tạp.

Tổng cộng, Angular đưa ra một cách tiếp cận có tổ chức và mạnh mẽ để phát triển ứng dụng web, đồng thời giúp các nhà phát triển tận dụng tối đa sức mạnh của JavaScript và các công nghệ web hiện đại.

**2.5. Lập trình Web API NET CORE**

Lập trình Web API trên nền tảng .NET Core là một hành trình đầy hứa hẹn, mang đến cho các nhà phát triển môi trường phát triển linh hoạt và hiệu quả. .NET Core, là một framework mã nguồn mở, độc lập với hệ điều hành, đã trở thành một lựa chọn phổ biến cho việc xây dựng các ứng dụng web hiện đại và các dịch vụ API.

Một trong những điểm đặc biệt của .NET Core là khả năng chạy trên nhiều nền tảng, bao gồm Windows, Linux và macOS. Điều này mang lại sự linh hoạt và tính di động, cho phép phát triển ứng dụng mà không bị ràng buộc bởi hệ điều hành.

Khi lập trình Web API trên .NET Core, nhà phát triển có thể sử dụng ngôn ngữ lập trình C# mạnh mẽ và hiện đại. C# cung cấp tính an toàn, dễ đọc và dễ bảo trì, đồng thời tích hợp tốt với Visual Studio, một trong những IDE (Integrated Development Environment) hàng đầu, giúp tăng cường hiệu suất và trải nghiệm phát triển.

.NET Core cung cấp một số tính năng quan trọng như Dependency Injection, Middleware, và một hệ thống routing mạnh mẽ. Dependency Injection giúp quản lý các thành phần và tạo ra ứng dụng linh hoạt, trong khi Middleware cho phép xử lý các yêu cầu HTTP và phản hồi trước khi chúng đến đích. Hệ thống routing giúp dễ dàng xác định các điểm cuối API và xử lý yêu cầu từ client.

Đặc biệt, .NET Core hỗ trợ xác thực và ủy quyền thông qua Identity và JWT (JSON Web Tokens), cung cấp cơ chế bảo mật mạnh mẽ cho ứng dụng. Điều này làm cho việc xây dựng các dịch vụ API an toàn và bảo mật trở nên thuận tiện.

Tổng cộng, việc lập trình Web API trên .NET Core mang lại cho nhà phát triển một môi trường phát triển mạnh mẽ, đồng thời cung cấp các công cụ và tính năng để xây dựng các ứng dụng web API hiệu quả, linh hoạt và bảo mật.

1. **PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**
   1. **Đặc tả yêu cầu phần mềm**

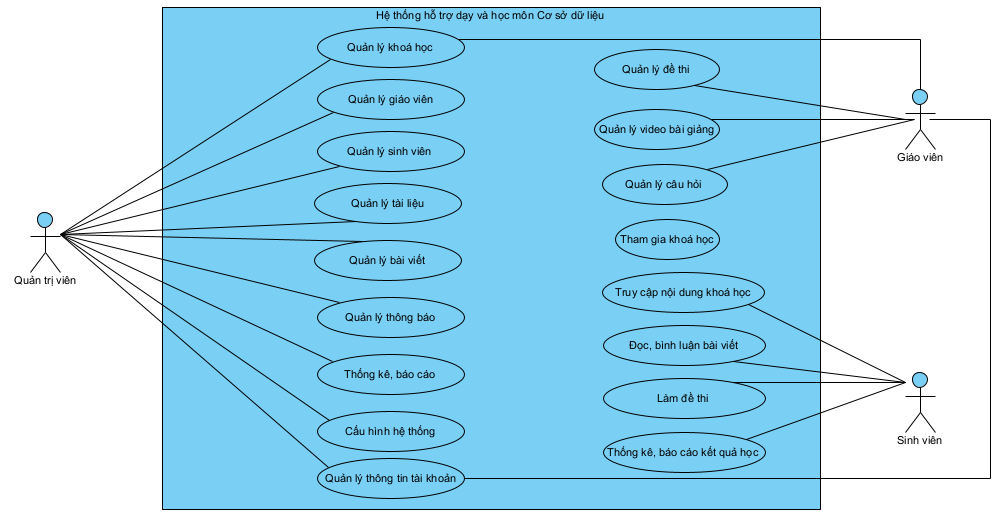
Dự án hỗ trợ dạy và học môn Cơ sở dữ liệu nhằm xây dựng một hệ thống website cung cấp nền tảng học tập trực tuyến cho giảng viên và sinh viên trong lĩnh vực Cơ Sở Dữ Liệu. Website này cho phép giảng viên tạo, quản lý các môn học, chủ đề, và tài liệu giảng dạy. Sinh viên có thể xem và tương tác với nội dung học tập, thảo luận trong diễn đàn, và thực hiện các bài kiểm tra trực tuyến. Các yêu cầu của hệ thống được mô tả như sau:

**Phân hệ quản trị hệ thống**

QTV có thể thao tác quản lý các khóa đang có trên hệ thống. Khi cần thêm một khoá học mới, QTV sử dụng chức năng "Thêm khoá học" và nhập các thông tin liên quan như Tên khóa học, mã khoá học, và mô tả ngắn về nội dung của khoá học,...Quản trị viên cũng có khả năng gán các Giảng viên phụ trách và xác định các học phần liên quan. QTV có thể thực hiện "Xem thông tin khoá học",

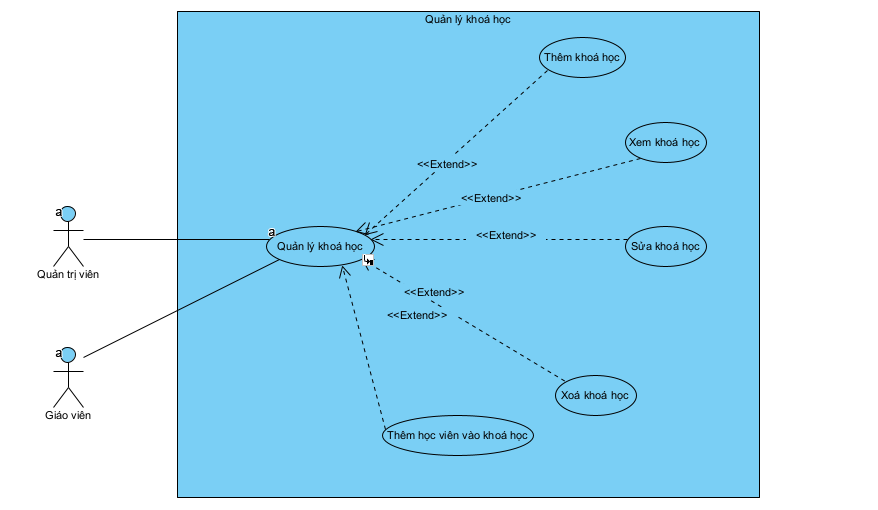
Quản trị viên thực có thao tác liên quan đến Quản lý người dùng, bao gồm Sinh viên và Giảng viên. Khi cần thêm một người dùng mới, Quản trị viên sử dụng chức năng "Thêm Người Dùng". Tại đây, họ nhập thông tin cần thiết như tên, email, và vai trò của người dùng (Sinh viên hoặc Giảng viên). Đồng thời, Quản trị viên có thể đặt mật khẩu mặc định hoặc yêu cầu người dùng tạo mật khẩu của mình.Với các người dùng hiện có, quản trị viên có thể "Xem và quản lý thông tin người dùng" cho phép Quản trị viên xem danh sách người dùng, kiểm tra thông tin cá nhân, và thực hiện các chỉnh sửa nếu cần thiết. Nếu một người dùng quên mật khẩu, Quản trị viên có thể sử dụng chức năng "Đặt lại mật khẩu" để tạo một mật khẩu mới hoặc gửi một liên kết đặt lại mật khẩu cho người dùng. Quản trị viên có thể sử dụng chức năng "Khóa người dùng" để tạm thời khóa người dùng trong trường hợp vi phạm hoặc nếu có nghi ngờ về bảo mật.

* + 1. ***Các yêu cầu chức năng***

******

***Hình 3.1: Biểu đồ usecase tổng quát hệ thống***

* **Usecase quản lý khoá học**
* **Biểu đồ usecase:**

****

**Hình 3.2: Biểu đồ usecase quản lý khoá học**

* **Dòng sự kiện:**

| 1. Yêu cầu chức năng quản lý khoá học |
| --- |
| 2. SYSTEM Hệ thống chuyển người dùng đến trang quản lý khoá học và lấy danh sách khoá học hiển thị lên màn hình |
| 3. Lựa chọn ca sử dụng mong muốn |
| 3.1. if Lựa chọn chức năng thêm khoá học |
| 3.1.1. SYSTEM Thực hiện ca sử dụng thêm khoá học |
| 3.1. end if |
| 3.2. if Lựa chọn chức năng sửa khoá học |
| 3.2.1. SYSTEM Thực hiện ca sử dụng sửa khoá học |
| 3.2. end if |
| 3.3. if Lựa chọn chức năng xóa khoá học |
| 3.3.1. Thực hiện ca sử dụng xóa khoá học |
| 3.3. end if |

* **Dòng sự kiện phụ:**
* **Thêm khoá học**

| 1. Người dùng yêu cầu chức năng nhập mới thông tin khoá học |
| --- |
| 2. SYSTEM Hiển thị giao diện nhập thông tin khoá học |
| 3. Thực hiện nhập thông tin khoá học |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận thêm thông tin khoá học |
| 6. SYSTEM Lưu thông tin khoá học mới vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý khoá học và lấy danh sách các khoá học mới sau khi lưu vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. SYSTEM Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác nhập |
| 1. SYSTEM Hệ thống đóng giao diện chức năng nhập thông tin khoá học và trở lại giao diện chức năng quản lý thông tin khoá học |

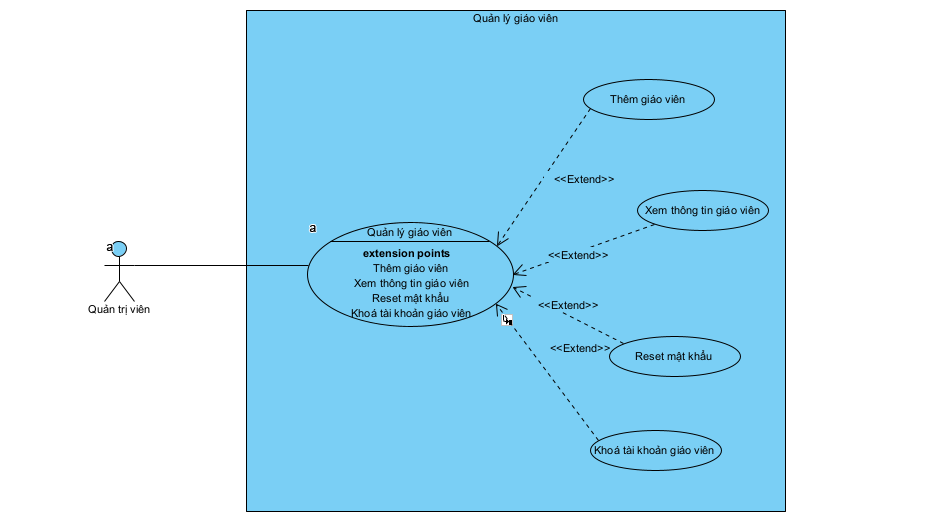
* **Sửa khoá học**

| 1. Yêu cầu chức năng sửa thông tin khoá học |
| --- |
| 2. SYSTEM Hiển thị giao diện sửa thông tin khoá học và hiển thị thông tin khoá học cần sửa |
| 3. Người dùng thực hiện sửa thông tin khoá học |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận sửa thông tin khoá học |
| 6. SYSTEM Cập thông tin khoá học mới sau khi sửa vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý khoá học và lấy danh sách các khoá học mới, sau khi vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác sửa thông tin khoá học |
| 1. Hệ thống đóng giao diện chức năng sửa thông tin khoá học và trở lại giao diện chức năng quản lý thông tin khoá học |

* **Xoá khoá học**

| 1. Người dùng chọn khoá học cần xóa thông tin |
| --- |
| 2. SYSTEM Hiển thị hộp thoại thông báo xác nhận xóa thông tin khoá học |
| 3. Người dùng lựa chọn xác nhận xóa thông tin khoá học |
| 3.1. if Xác nhận xóa thông tin khoá học |
| 3.1.1. SYSTEM Xóa thông tin khoá học đã chọn trong hệ thống và thông báo thao tác thành công |
| 3.1. end if |
| 3.2. if Hủy thao tác xóa thông tin khoá học |
| 3.2.1. SYSTEM Đóng hộp thoại yêu cầu xác nhận xóa thông tin khoá học |
| 3.2. end if |

* **Usecase quản lý giáo viên**
* **Biểu đồ usecase:**

****

* **Dòng sự kiện:**

| 1. Yêu cầu chức năng quản lý giáo viên |
| --- |
| 2. SYSTEM Hệ thống chuyển người dùng đến trang quản lý giáo viên và lấy danh sách khoá học hiển thị lên màn hình |
| 3. Lựa chọn ca sử dụng mong muốn |
| 3.1. if Lựa chọn chức năng thêm giáo viên |
| 3.1.1. SYSTEM Thực hiện ca sử dụng thêm giáo viên |
| 3.1. end if |
| 3.2. if Lựa chọn chức năng xem thông tin giáo viên |
| 3.2.1. SYSTEM Thực hiện ca sử dụng xem thông tin giáo viên |
| 3.2. end if |
| 3.3. if Lựa chọn chức năng reset mật khẩu |
| 3.3.1. Thực hiện ca sử dụng reset mật khẩu |
| 3.3. end if  3.4. if Lựa chọn chức năng khoá tài khoản giáo viên  3.2.1. SYSTEM Thực hiện ca sử dụng khoá tài khoản giáo viên  end if |

* **Dòng sự kiện phụ**
* **Thêm giáo viên:**

| 1. Yêu cầu chức năng xem thông tin giáo viên |
| --- |
| 2. SYSTEM Chuyển đến trang xem thông giáo viên và lấy thông tin giáo viên hiển thị lên màn hình |
| 3. Người dùng thực hiện sửa thông tin giáo viên |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận sửa thông tin giáo viên |
| 6. SYSTEM Cập thông tin giáo viên mới sau khi sửa vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý giáo viên và lấy danh sách các khoá giáo viên, sau khi vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác sửa thông tin giáo viên |
| 1. Hệ thống đóng giao diện chức năng sửa thông tin giáo viên và trở lại giao diện chức năng quản lý thông tin giáo viên |

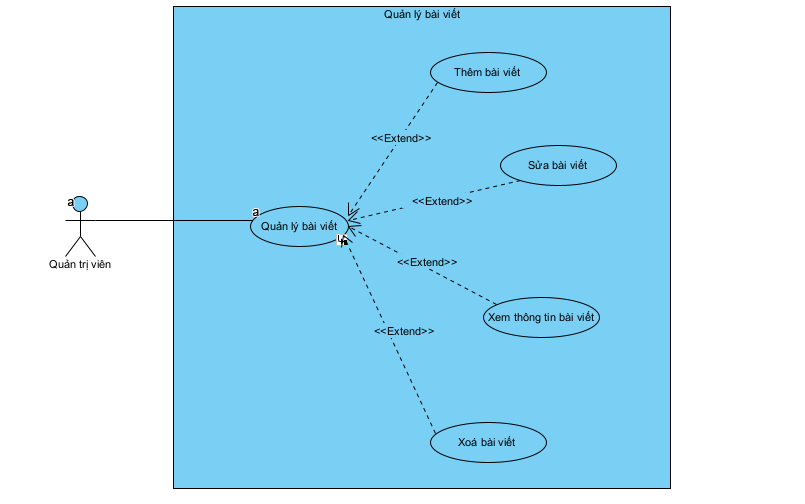
* **Xem thông tin giáo viên**

| 1. Yêu cầu chức năng xem thông tin giáo viên |
| --- |
| 2. SYSTEM Chuyển đến trang xem thông giáo viên và lấy thông tin giáo viên hiển thị lên màn hình |
| 3. Người dùng thực hiện sửa thông tin giáo viên |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận sửa thông tin giáo viên |
| 6. SYSTEM Cập thông tin giáo viên mới sau khi sửa vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý giáo viên và lấy danh sách các khoá giáo viên, sau khi vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác sửa thông tin giáo viên |
| 1. Hệ thống đóng giao diện chức năng sửa thông tin giáo viên và trở lại giao diện chức năng quản lý thông tin giáo viên |

* **Reset mật khẩu**

| 1. Người dùng yêu cầu chức năng reset mật khẩu |
| --- |
| 2. SYSTEM Hệ thống hiển thị màn hình xác nhận người dùng khoá tài khoản giáo viên |
| 3. Người dùng lựa chọn hành động mong muốn  3.1. if Người dùng xác nhận khoá tài khoản giáo viên  3.2.1. SYSTEM Hệ thống thực hiện khoá tài khoản giáo viên  end if  3.2. if Người dùng bỏ qua yêu cầu xác nhận khoá tài khoản giáo viên  3.2.1. SYSTEM Hệ thống đóng màn hình yêu cầu khoá tài khoản giáo viên  end if |
| 4. SYSTEM Hệ thống quay trở lại màn hình hiển thị danh sách giáo viên |
|  |

* **Usecase quản lý bài viết**
* **Biểu đồ usecase:**

****

* **Dòng sự kiện:**

| 1. Yêu cầu chức năng quản lý bài viết |
| --- |
| 2. SYSTEM Hệ thống chuyển người dùng đến trang quản lý bài viết và lấy danh sách bài viết hiển thị lên màn hình |
| 3. Lựa chọn ca sử dụng mong muốn |
| 3.1. if Lựa chọn chức năng thêm bài viết |
| 3.1.1. SYSTEM Thực hiện ca sử dụng thêm bài viết |
| 3.1. end if |
| 3.2. if Lựa chọn chức năng xem thông tin bài viết |
| 3.2.1. SYSTEM Thực hiện ca sử dụng xem thông tin bài viết |
| 3.2. end if |
| 3.3. if Lựa chọn chức năng sửa bài viết |
| 3.3.1. Thực hiện ca sử dụng sửa bài viết |
| 3.3. end if  3.4. if Lựa chọn chức năng xoá bài viết  3.2.1. SYSTEM Thực hiện ca sử dụng sửa bài viết  end if |

* **Dòng sự kiện phụ**
* **Thêm bài viết:**

| 1. Người dùng yêu cầu chức năng nhập mới thông tin bài viết |
| --- |
| 2. SYSTEM Hiển thị giao diện nhập thông tin bài viết |
| 3. Thực hiện nhập thông tin bài viết |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận thêm thông tin bài viết |
| 6. SYSTEM Lưu thông tin bài viết mới vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý bài viết và lấy danh sách các khoá học mới sau khi lưu vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. SYSTEM Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác nhập |
| 1. SYSTEM Hệ thống đóng giao diện chức năng nhập thông tin bài viết và trở lại giao diện chức năng quản lý thông tin khoá học |

* **Sửa bài viết:**

| 1. Yêu cầu chức năng sửa thông tin bài viết |
| --- |
| 2. SYSTEM Hiển thị giao diện sửa thông tin bài viết và hiển thị thông tin bài viết cần sửa |
| 3. Người dùng thực hiện sửa thông tin bài viết |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận sửa thông tin bài viết |
| 6. SYSTEM Cập thông tin bài viết mới sau khi sửa vào trong hệ thống |
| 7. SYSTEM Chuyển người dùng trở trang quản lý bài viết và lấy danh sách các bài viết mới, sau khi vào trong hệ thống hiển thị lên giao diện |
| Extension: |
| 4.a. Thông tin người dùng nhập không hợp lệ |
| 1. Hệ thống hiển thị thông báo lỗi và yêu cầu người dùng nhập lại |
| 5.a. Hủy thao tác sửa thông tin bài viết |
| 1. Hệ thống đóng giao diện chức năng sửa thông tin bài viết và trở lại giao diện chức năng quản lý thông tin bài viết |

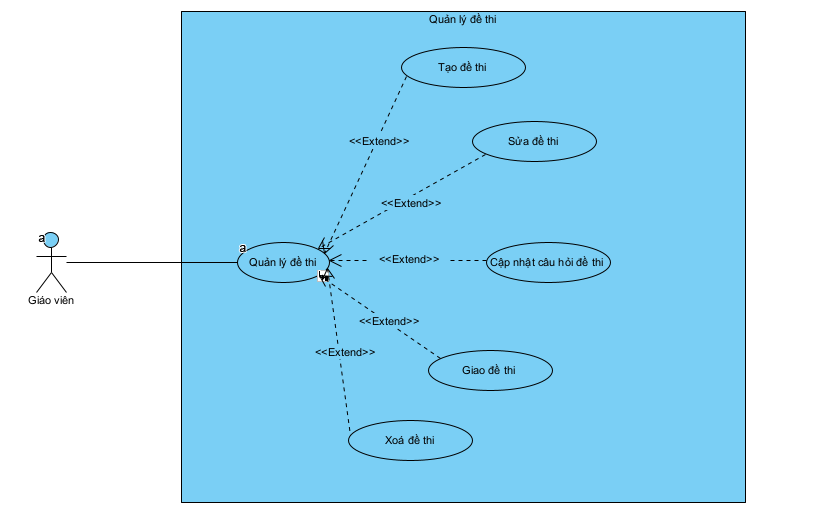
* **Xem bài viết:**

| 1. Yêu cầu chức năng xem thông tin bài viết |
| --- |
| 2. SYSTEM Chuyển đến trang xem thông giáo viên và lấy thông tin giáo viên hiển thị lên màn hình |
| 3. Người dùng yêu cầu đóng trang xem thông tin bài viết |
| 4. SYSTEM Hệ thống quay trở lại trang quản lý bài viết |
|  |

* **Xoá bài viết:**

| 1. Người dùng chọn bài viết cần xóa thông tin |
| --- |
| 2. SYSTEM Hiển thị hộp thoại thông báo xác nhận xóa thông tin bài viết |
| 3. Người dùng lựa chọn xác nhận xóa thông tin bài viết |
| 3.1. if Xác nhận xóa thông tin bài viết |
| 3.1.1. SYSTEM Xóa thông tin bài viết đã chọn trong hệ thống và thông báo thao tác thành công |
| 3.1. end if |
| 3.2. if Hủy thao tác xóa thông tin bài viết |
| 3.2.1. SYSTEM Đóng hộp thoại yêu cầu xác nhận xóa thông tin bài viết |
| 3.2. end if |

* **Quản lý đề thi**
* **Biểu đồ usecase:**



* **Dòng sự kiện:**

| 1. Giáo viên yêu cầu chức năng tạo đề thi |
| --- |
| 2. SYSTEM Hệ thống chuyển người dùng đến trang quản lý đề thi và lấy danh sách đề thi hiển thị lên màn hình |
| 3. Lựa chọn ca sử dụng mong muốn |
| 3.1. if Lựa chọn chức năng tạo đề thi |
| 3.1.1. SYSTEM Thực hiện ca sử dụng thêm đề thi |
| 3.1. end if |
| 3.2. if Lựa chọn chức năng sửa thông tin đề thi |
| 3.2.1. SYSTEM Thực hiện ca sử dụng sửa thông tin đề thi |
| 3.2. end if |
| 3.3. if Lựa chọn chức năng cập nhật câu hỏi đề thi |
| 3.3.1. Thực hiện ca sử dụng cập nhật câu hỏi đề thi |
| 3.3. end if  3.4. if Lựa chọn chức năng giao đề thi  3.2.1. SYSTEM Thực hiện ca sử dụng giao đề thi  end if  3.5. if Lựa chọn chức năng xoá đề thi  3.2.1. SYSTEM Thực hiện ca sử dụng xoá đề thi  end if |

* **Dòng sự kiện phụ:**
* **Tạo đề thi:**

| 1. Người dùng yêu cầu chức năng tạo đề thi |
| --- |
| 2. SYSTEM Hiển thị giao diện tạo đề thi cho phép người dùng nhập |
| 3. Người dùng thực hiện nhập thông tin đề thi |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Người dùng chọn tiếp tục |
| 6. SYSTEM Hệ thống hiển thị danh sách câu hỏi lựa chọn thêm vào đề thi  7. Người dùng thực hiện lựa chọn câu hỏi thêm vào đề thi  8. Người dùng yêu cầu lưu đề thi |
| 9.SYSTEM Hệ thống kiểm tra tính hợp lệ dữ liệu  10. SYSTEM Hệ thống lưu thông tin đề thi, chuyển người dùng trở trang quản lý bài viết và lấy danh sách các khoá học mới sau khi lưu vào trong hệ thống hiển thị lên giao diện |
|  |

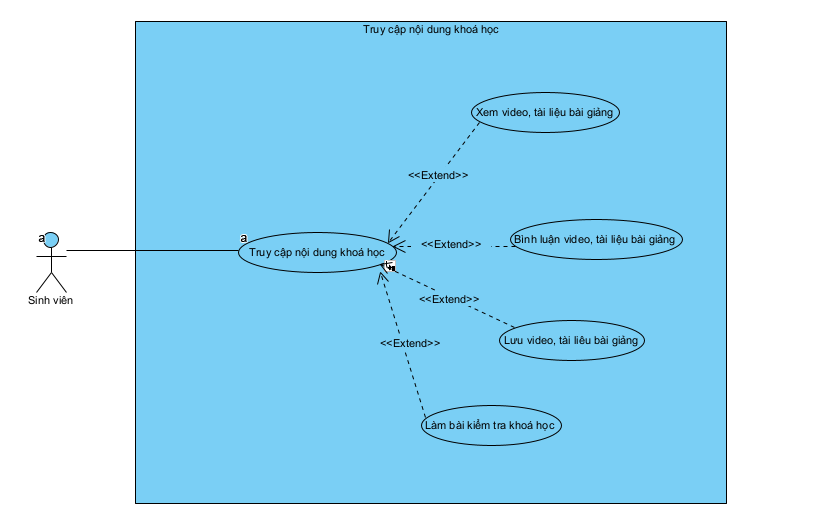
* **Sửa đề thi:**

| 1. Người dùng yêu cầu chức năng sửa đề thi |
| --- |
| 2. SYSTEM Hiển thị giao diện thị thông tin đề thi người dùng lựa chọn sửa |
| 3. Người dùng thực hiện sửa thông tin đề thi |
| 4. SYSTEM Hệ thống kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Người dùng chọn tiếp tục |
| 6. SYSTEM Hệ thống hiển thị danh sách câu hỏi đã được thêm vào đề thi  7. Người dùng thực hiện lựa chọn thêm, sửa, xoá câu hỏi tại đề thi  8. Người dùng yêu cầu lưu đề thi |
| 9. SYSTEM Hệ thống lưu thông tin đề thi, chuyển người dùng trở trang quản lý bài viết và lấy danh sách các khoá học mới sau khi lưu vào trong hệ thống hiển thị lên giao diện |
|  |

* **Giao đề thi:**

| 1. Người dùng yêu cầu chức năng giao đề thi |
| --- |
| 2. SYSTEM Hiển thị bảng hiển thị danh sách người dùng giao đề, thời gian |
| 3. Người dùng lựa chọn lớp giao đề, thời gian thực hiện |
| 4. SYSTEM Kiểm tra tính hợp lệ của thông tin nhập vào |
| 5. Xác nhận giao đề thi |
| 6. SYSTEM Lưu thông tin bài viết mới vào trong hệ thống |
| 7. SYSTEM Chuyển thông tin đề thi tới những người được giao trong danh sách người dùng được chọn |
|  |

* **Truy cập nội dung khoá học**
* **Biểu đồ usecase:**

****

* **Luồng sự kiện:**

| 1. Người dùng yêu cầu truy cập nội dung khoá học |
| --- |
| 2. SYSTEM Hệ thống lấy thông tin chi tiết nội dung khoá học và hiển thị lên giao diện người dùng |
| 3. Người dùng lựa chọn ca sử dụng mong muốn |
| 3.1. if Người dùng lựa chọn xem video, tài liệu bài giảng |
| 3.1.1. SYSTEM Hệ thống thực hiện ca Xem video, tài liệu bài giảng |
| 3.1. end if |
| 3.2. if Người dùng lựa chọn chức năng bình luận video bài giảng |
| 3.2.1. SYSTEM Thực hiện ca sử dụng bình luận video, bài giảng |
| 3.2. end if |
| 3.3. if Lựa chọn chức năng lưu video, bài giảng |
| 3.3.1. Thực hiện ca sử dụng Lưu video, bài giảng |
| 3.3. end if  3.4. if Lựa chọn chức năng khoá làm bài kiểm tra khoá học  3.2.1. SYSTEM Thực hiện ca sử dụng làm bài kiểm tra khoá học  end if |

* **Luồng sự kiện phụ:**
* **Xem video, tài liệu bài giảng**

| 1. Người dùng yêu cầu truy xem video, tài liệu bài giảng |
| --- |
| 2. SYSTEM Hệ thống lấy thông tin nội dung video, bài giảng hiển thị cho người dùng |
|  |

* **Bình luận video bài giảng**

| 1. Người dùng yêu cầu chức năng bình luận video, bài giảng |
| --- |
| 2. SYSTEM Hệ thống hiển thị danh sách bình luận video, tài liệu bài giảng  3. Người dùng nhập nội dung bình luận và xác nhận  4. SYSTEM Hệ thống lưu thông tin bình luận người dùng và hiển thị lại màn hình |
|  |

* **Lưu video, bài giảng**

| 1. Người dùng yêu cầu chức năng lưu video, bài giảng |
| --- |
| 2. SYSTEM Hệ thống lưu thông tin lưu bài giảng của người dùng và thông báo thành công tới người dùng |
|  |

* **Làm bài kiểm tra khoá học**

| 1. Người dùng yêu cầu chức năng làm bài kiểm tra khoá học |
| --- |
| 2. SYSTEM Hệ thống hiển thị thông tin bài kiểm tra khoá học lên giao diện  3. Người dùng thực hiện bài kiểm tra khoá học  4. Người dùng xác nhận nộp bài  5. SYSTEM Hệ thống lưu thông tin bài làm người dùng, tính số điểm đạt được và hiển thị thông báo tới người dùng. |
|  |

* + 1. ***Biểu đồ lớp thực thể***

Biểu đồ lớp thực thể hệ thống gồm:

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã khoá học | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên khoá học | Tên của khoá học |
| 3 | Thời lượng khoá học | Thời lượng khoá học |
| 4 | Mô tả | Mô tả |

**Bảng 3.1: Bảng KhoaHoc**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã chủ đề | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên chủ đề | Tên của khoá học |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.2: Bảng ChuDe**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã bài học | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên bài học | Tên của khoá học |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.3: Bảng BaiHoc**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.4: Bảng Video**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.5: Bảng TaiLieu**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.6: Bảng CauHoi**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.7: Bảng DapAn**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.8: Bảng CauHoi**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.9: Bảng BaiViet**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

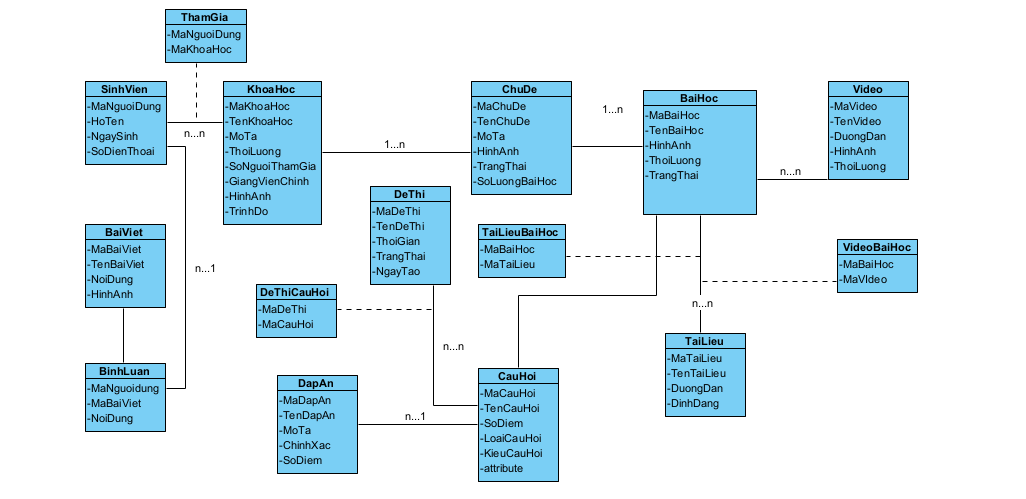
**Bảng 3.10: Bảng BinhLuan**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.11: Bảng SinhVien**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.12: Bảng GiaoVien**

****

**Hình 3.10: Biểu đồ lớp thực thể**

* + 1. ***Các yêu cầu phi chức năng***

Dưới đây là một số yêu cầu phi chức năng có thể áp dụng cho hệ thống hỗ trợ dạy và học môn Cơ sở dữ liệu:

**Hiệu suất:**

* Hệ thống cần hỗ trợ một số lượng lớn người dùng đồng thời mà không ảnh hưởng đến hiệu suất của nó.
* Thời gian phản hồi từ hệ thống đối với người dùng nên làm việc tối đa trong khoảng thời gian chấp nhận được.

**Bảo mật:**

* Dữ liệu người dùng cần được bảo vệ an toàn, bao gồm cả thông tin cá nhân và thông tin đăng nhập.
* Cần có cơ chế xác thực mạnh mẽ để đảm bảo rằng chỉ những người dùng có quyền mới có thể truy cập các tính năng và dữ liệu nhất định.

**Duy trì và Quản lý:**

* Hệ thống cần có khả năng sao lưu và khôi phục dữ liệu để đảm bảo rằng không có dữ liệu nào bị mất trong trường hợp sự cố.
* Cần có giao diện quản trị dễ sử dụng để quản lý người dùng, khóa học, chủ đề, và các tùy chọn cấu hình khác của hệ thống.

**Tương thích:**

* Hệ thống cần tương thích với nhiều trình duyệt web phổ biến để đảm bảo sự tiện lợi cho người dùng.
* Nếu có ứng dụng di động, hệ thống cần tương thích với cả các nền tảng di động phổ biến.

**Mở rộng và Mô đun:**

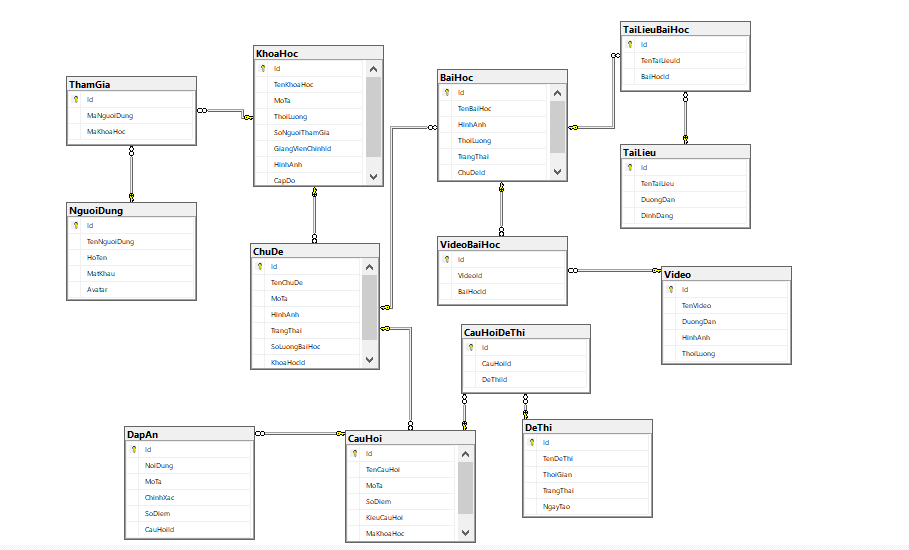
* Kiến trúc của hệ thống cần thiết kế linh hoạt để dễ dàng mở rộng và tích hợp các tính năng mới trong tương lai.

**Hiệu suất Tải:**

* Hệ thống cần được kiểm thử đối với các tình huống tải cao để đảm bảo rằng nó có thể xử lý số lượng người dùng đồng thời lớn.

**Giao diện Người dùng:**

* Giao diện người dùng cần được thiết kế để tối ưu hóa trải nghiệm người dùng và dễ sử dụng.
  1. **Thiết kế hệ thống**
     1. ***Thiết kế cơ sở dữ liệu***
* Mô hình cơ sở dữ liệu quan hệ



* Mô tả cấu trúc bảng

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã khoá học | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên khoá học | Tên của khoá học |
| 3 | Thời lượng khoá học | Thời lượng khoá học |
| 4 | Mô tả | Mô tả |

**Bảng 3.1: Bảng KhoaHoc**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã chủ đề | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên chủ đề | Tên của khoá học |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.2: Bảng ChuDe**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã bài học | Mỗi khoá học được đánh dấu một mã khác nhau |
| 2 | Tên bài học | Tên của khoá học |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.3: Bảng BaiHoc**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.4: Bảng Video**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.5: Bảng TaiLieu**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.6: Bảng CauHoi**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.7: Bảng DapAn**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.8: Bảng CauHoi**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.9: Bảng BaiViet**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.10: Bảng BinhLuan**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.11: Bảng SinhVien**

| STT | Tên thuộc tính | Mô tả |
| --- | --- | --- |
| 1 | Mã video | Mỗi video được đánh dấu một mã khác nhau |
| 2 | Tên video | Tên của video |
| 3 | Mô tả | Mô tả |
| 4 | Trạng thái | Trạng thái chủ đề |

**Bảng 3.12: Bảng GiaoVien**

* Trình bày các thủ tục hoặc trigger được xây dựng nếu có
  + 1. ***Thiết kế lớp đối tượng***
       1. Biểu đồ lớp VOPC của các ca sử dụng
* Biểu đồ VOPC usecase quản lý khoá học:

<*phần này trình bày biểu đồ lớp VOPC mức phân tích cho từng ca sử dụng*>

* + - 1. Biểu đồ tuần tự

<*phần này trình bày biểu đồ tuần tự cho từng ca sử dụng*>

* + - 1. Biểu đồ lớp chi tiết

<*phần này trình bày biểu đồ lớp VOPC mức thiết kế, sau khi xác định được đầy đủ được các phương thức cho các lớp, tách lớp, gộp lớp*>

* + 1. ***Thiết kế giao diện***

1. **TRIỂN KHAI WEBSITE**
   1. **Triển khai các chức năng cho phân hệ người dùng**

<*Phần này trình bày các kết quả đã được triển khai cài đặt cho phân hệ người dùng – 3 chức năng tiêu biểu*>

Để xây dựng được các chức năng của trang người dùng theo thiết kế đã được trình bày ở chương 3, đồ án đã sử dụng HTML, CSS, và JavaScript, Jquery, AngularJS để thiết kế giao diện và thao tác dữ liệu các trang theo yêu cầu. Tiếp theo, đồ án sẽ trình bày các kỹ thuật được sử dụng để xây dựng các trang.

* + 1. ***Trang chủ (Ví dụ)***

a) Phía font end

* Xây dựng bố cục trang Home bằng các thẻ HTML

<Phần này trình bày cách thức sử dụng các thẻ HTML để phân chia được bố cục trang Chủ>.

* Kỹ thuật định dạng bằng CSS

<Phần này trình bày kỹ thuật sử dụng CSS để định các phần của trang Home hiển thị theo bố cục đã thiết kế>.

* Sử dụng Javascript, Jquery, AngularJS để lập trình các chức năng

<Phần này trình bày thuật toán hoặc các bước xử lý để có được các chức năng theo yêu cầu, ví dụ chức năng chọn mua hàng, tìm kiếm, ….>

b) Phía backend

* Triển khai các lớp tầng DataAccess

<Phần này trình bày cách thức xử lý để thực hiện được các thao tác trên cơ sở dữ liệu để phục vụ các xử lý ở trang chủ>

* Triển khai lớp tầng Bussiness

<Phần này trình bày về cách thức cài đặt lớp ở tầng Bussiness để đáp ứng được các xử lý được gọi ở tầng Controller>

* Triển khai lớp tầng Controller

<Phần này trình bày về cách thức cài đặt các Action để đáp ứng được các xử lý được gọi ở tầng View cho trang chủ>

* + 1. ***Trang Xem Sản phẩm (Ví dụ)***
    2. ***Trang Quản lý giỏ (Ví dụ)***

***……***

* 1. **Triển khai các chức năng cho phân hệ quản trị nội dung (nếu có)**

<*Phần này trình bày các kết quả đã được triển khai cho phân hệ trang quản trị*>

* 1. **Kiểm thử và triển khai ứng dụng**
     1. ***Kiểm thử***

<Thực thi và kiểm tra, sửa lỗi tất cả các chức năng đáp ứng yêu cầu): Xây dựng và mô tả dữ liệu/hành động cho input và kết quả cho Output.>

* + 1. ***Đóng gói ứng dụng***

<Trình bày và minh chứng quy trình đóng gói sản phẩm của đồ án>

* + 1. ***Triển khai ứng dụng***

<Trình bày điều kiện, môi trường triển khai ứng dụng: phần cứng, phần mềm>

**KẾT LUẬN**

* Trình bày kết quả đạt được: Kiến thức, sản phẩm
* Những hạn chế của đề tài: Sản phẩm (ví dụ: chức năng còn thiếu, chưa hoàn thiện….); Kỹ năng (phân tích thiết kế hệ thống, lập trình…)

Hướng phát triển của đề tài: (Giải quyết những hạn chế của đề tài)

**TÀI LIỆU THAM KHẢO**

| [1] | Tên tác giả (năm XB), Tên sách, NXB, Nơi XB |
| --- | --- |
| [2] | ……….. |