

**BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN**

**XÂY DỰNG WEBSITE BÁN KHÓA HỌC ONLINE
SỬ DỤNG NodeJS**

CBHD: ThS. Nguyễn Thị Hương Lan
Sinh viên: Dương Quốc Khánh
Mã sinh viên: 2020605888

Hà Nội – Năm 2024

**BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN**

**XÂY DỰNG WEBSITE BÁN KHÓA HỌC ONLINE
SỬ DỤNG NodeJS**

CBHD: ThS. Nguyễn Thị Hương Lan

Sinh viên: Dương Quốc Khánh

Mã sinh viên: 2020605888

Hà Nội – Năm 2024

DƯƠNG QUỐC KHÁNH

CÔNG NGHỆ THÔNG TIN

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến Quý thầy cô Khoa Công nghệ thông tin - Trường Đại học Công nghiệp Hà Nội đã truyền đạt cho em những kiến thức vô cùng quý báu, bổ ích và tạo điều kiện giúp em để hoàn thành đề tài của mình một cách tốt nhất, đặc biệt em xin gửi lời cảm ơn chân thành đến ThS.Nguyễn Thị Hương Lan là người đã trực tiếp hướng dẫn và tận tình giúp đỡ em trong suốt quá trình thực hiện đề tài cho đến khi hoàn thành bài báo cáo này.

Trong quá trình nghiên cứu đề tài, em đã cố gắng hoàn thành tốt nhất báo cáo đồ án tốt nghiệp. Tuy nhiên kiến thức chuyên ngành của bản thân còn nhiều hạn chế. Vì vậy trong báo cáo không tránh khỏi những thiếu sót, em rất mong nhận được sự đóng góp của tất cả các thầy cô giáo để đồ án của em được đầy đủ và hoàn chỉnh hơn .

Em xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT	i
DANH MỤC HÌNH ẢNH	ii
DANH MỤC BẢNG BIỂU	iv
MỞ ĐẦU	1
1. Lý do chọn đề tài	1
2. Mục tiêu nghiên cứu	2
3. Đối tượng nghiên cứu	2
4. Phạm vi nghiên cứu	2
5. Phương pháp nghiên cứu	3
CHƯƠNG 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU.....	4
1.1. Giới thiệu chung	4
1.2. Tổng quan	4
1.2.1. Tổng quan về lập trình Website.....	4
1.2.2. Cơ sở dữ liệu	6
1.2.3. Frontend	11
1.2.4. Backend.....	13
CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	19
2.1. Khảo sát hệ thống	19
2.2. Giới thiệu về hệ thống	23
2.2.1. Về hệ thống	24
2.2.2. Về người sử dụng	24
2.2.3. Về chức năng.....	25
2.3. Thiết kế hệ thống.....	27
2.3.1. Mô hình hóa Use case	27

2.3.2.	Mô tả chi tiết Use case	29
2.4.	Thiết kế giao diện	46
2.4.1.	Biểu đồ dịch chuyển màn của hệ thống	46
2.4.2.	Giao diện chức năng đăng nhập	47
2.4.3.	Giao diện màn hình đăng ký	47
2.4.4.	Giao diện thông tin khóa học	48
2.4.3	Giao diện quản lý khóa học	48
2.4.5.	Giao diện quản lý thông tin cá nhân	49
2.4.6.	Giao diện tạo khóa học	49
2.4.7.	Giao diện của khóa học	50
2.4.8.	Giao diện khi tham gia bài giảng	50
2.4.9.	Giao diện tạo bài tập	51
2.5.	Cơ sở dữ liệu	52
2.5.1.	Sơ đồ thực thể liên kết	52
2.5.2.	Tổng quan cơ sở dữ liệu	53
CHƯƠNG 3:	KẾT QUẢ	57
3.1.	Về chức năng	57
3.2.	Về giao diện	57
3.2.1.	Giao diện chức năng đăng nhập	57
3.2.2.	Giao diện màn hình đăng ký	58
3.2.3.	Giao diện thông tin khóa học	58
3.2.4.	Giao diện quản lý khóa học	59
3.2.5.	Giao diện quản lý thông tin cá nhân	59
3.2.6.	Giao diện tạo khóa học	60

3.2.7. Giao diện của khóa học	60
3.2.8. Giao diện khi tham gia bài giảng.....	61
3.2.9. Giao diện tạo bài tập.....	61
CHƯƠNG 4: KIỂM THỬ.....	62
4.1. Kế hoạch kiểm thử.....	62
4.2. Thiết kế và thực thi kiểm thử	63
4.3. Báo cáo kiểm thử	72
KẾT LUẬN	73
TÀI LIỆU THAM KHẢO.....	75

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

STT	Viết tắt	Dịch nghĩa
1	SDK	Software Development Kit
2	FCM	Firebase Cloud Messaging
3	IDE	Integrated Development Environment
4	VPN	Virtual Private Network
5	EMM	Enterprise Mobility Manager
6	OTA	Over The Air
7	CNTT	Công nghệ thông tin
8	SI	Systems Integrators
9	ISV	Independent Software Vendors
10	API	Application Programming Interface
11	HTML	HyperText Markup Language
12	CSS	Cascading Style Sheets
13	JVM	Java Virtual Machine
14	UC	Use case

DANH MỤC HÌNH ẢNH

Hình 1.1: Lập trình Website.....	5
Hình 1.2: MongoDB.....	10
Hình 2.1: Mô hình tổng quan hệ thống	27
Hình 2.2: Biểu đồ Usecase tổng quan	27
Hình 2.3: Phân rã Usecase theo tác nhân học sinh	28
Hình 2.4: Phân rã Usecase theo tác nhân giáo viên	28
Hình 2.5: Biểu đồ trình tự UC tạo khóa học	29
Hình 2.6: Biểu đồ VOPC UC tạo khóa học	29
Hình 2.7: Biểu đồ trình tự UC quản lý khóa học	30
Hình 2.8: Biểu đồ VOPC UC quản lý khóa học	30
Hình 2.9: Biểu đồ trình tự UC tạo bài kiểm tra	31
Hình 2.10: Biểu đồ VOPC UC tạo bài kiểm tra.....	32
Hình 2.11: Biểu đồ trình tự UC làm bài kiểm tra	47
Hình 2.12: Biểu đồ VOPC UC làm bài kiểm tra	31
Hình 2.13: Biểu đồ trình tự UC tham gia khóa học	32
Hình 2.14: Biểu đồ VOPC UC tham gia khóa học	33
Hình 2.15: Biểu đồ trình tự UC tạo khóa học	35
Hình 2.16: Biểu đồ VOPC UC tạo khóa học	36
Hình 2.17: Biểu đồ trình tự UC đăng tải tài liệu.....	38
Hình 2.18: Biểu đồ VOPC UC đăng tải tài liệu	40
Hình 2.19: Biểu đồ dịch chuyển màn của hệ thống	46
Hình 2.20: Thiết kế giao diện đăng nhập	47
Hình 2.21: Thiết kế giao diện đăng ký.....	47
Hình 2.22: Thiết kế giao diện thông tin khóa học.....	48
Hình 2.23: Thiết kế giao diện quản lý khóa học	48
Hình 2.24: Thiết kế giao diện quản lý thông tin cá nhân.....	49
Hình 2.25: Thiết kế giao diện tạo khóa học	49
Hình 2.26: Thiết kế giao diện của khóa học	50

Hình 2.27: Thiết kế giao diện xem chi tiết một	50
Hình 2.28: Thiết kế giao diện tham gia bài giảng.....	51
Hình 2.29: Sơ đồ thực thể	52
Hình 2.30: Sơ đồ tổng quan cơ sở dữ liệu.....	53
Hình 3.1: Giao diện màn hình đăng nhập	58
Hình 3.2: Giao diện màn hình đăng ký	58
Hình 3.3: Giao diện thông tin khóa học	59
Hình 3.4: Giao diện quản lý thông tin cá nhân	59
Hình 3.5: Giao diện tạo khóa học	60
Hình 3.6: Giao diện của khóa học.....	60
Hình 3.7: Giao diện tạo bài tập	61
Hình 3.8: Giao diện tham gia bài giảng	61

DANH MỤC BẢNG BIỂU

Bảng 2.1: Mô tả UC tạo khóa học.....	29
Bảng 2.2: Mô tả UC quản lý khóa học.....	31
Bảng 2.3: Mô tả UC tạo bài kiểm tra	32
Bảng 2.4: Mô tả UC làm bài kiểm tra	33
Bảng 2.5: Mô tả UC tham gia khóa học.....	39
Bảng 2.6: Mô tả UC tạo khóa học.....	39
Bảng 2.7: Mô tả UC đăng tải tài liệu	38
Bảng 2.8: Giải thích ý nghĩa thực thể	52
Bảng 2.9: Mô tả chi tiết User (Người sử dụng)	55
Bảng 2.10: Mô tả chi tiết Courses (Khóa học).....	55
Bảng 2.11: Mô tả chi tiết Notification (Thông báo lớp học)	55
Bảng 2.12: Mô tả chi tiết Exams (Bài kiểm tra)	56
Bảng 2.13: Mô tả chi tiết Questions (Câu hỏi)	56
Bảng 2.14: Mô tả chi tiết Answer (Câu trả lời)	56
Bảng 4.1: Kiểm thử các chức năng của ứng dụng	65

MỞ ĐẦU

1. Lý do chọn đề tài

Hiện nay, việc học trực tuyến chắc hẳn không còn xa lạ đối với tất cả chúng ta. Kể từ cuối năm 2019, hình thức học trực tuyến hay còn gọi là Elearning xuất hiện nhiều hơn ở Việt Nam vì tình dịch bệnh Covid 19. Trên tình hình thực tế, việc học trực tuyến có lẽ không còn quá xa lạ đối với nhiều người nhưng chắc hẳn độ phổ biến của nó ở nước ta thì không hẳn quá rộng rãi. Trong thời kỳ khó khăn dịch bệnh như vậy, học trực tuyến như là một giải pháp để khắc phục việc học sinh, sinh viên không thể tới trường nhưng vẫn có thể đảm bảo tiến độ học tập một cách hiệu quả nhất.

Với hệ thống khóa học online là hệ thống giúp học viên có thể tham gia học trực tuyến và trao đổi kiến thức giữa học sinh, sinh viên với nhau viên hoặc những điều còn chưa rõ với giáo viên. So với học truyền thống, học trực tuyến cung cấp cho học sinh, sinh viên sự kết hợp hài hòa giữa nhìn, nghe và sự chủ động tích cực trong hoạt động. Chính nhờ vào lợi ích đó, học tập qua mạng đã mang lại rất nhiều hiệu quả cho việc học như: thu hút được nhiều đối tượng học sinh, sinh viên trên phạm vi toàn cầu, cắt giảm được nhiều chi phí xuất bản, in ấn tài liệu. Ngoài ra, học trực tuyến còn giúp cho người học có khả năng tự kiểm soát tốc độ học của mình sao có phù hợp với bản thân, vẫn đảm bảo được chất lượng học tập mà không cần phải có những phần hướng dẫn. Chính vì những đặc điểm trên, hệ thống học trực tuyến đang là một giải pháp tối ưu với sự thu hút đông đảo học viên về nhiều trình độ và cấp học khác nhau. Tuy nhiên vẫn còn một số mặt hạn chế khi học trực tuyến.

Ngày nay, có nhiều cách để có thể tiếp cận việc học trực tuyến. Có thể kể đến những hình thức như giáo viên có thể đăng tải những video quay sẵn lên từng bài học, có các bài tập, câu hỏi trong mỗi bài học. Học viên có thể chủ động xem và làm các bài tập, bài giảng đó một cách chủ động. Như đã kể trên, chúng ta thấy đa phần việc học như vậy không có tương tác trực tiếp với giáo

viên. Không phải tất cả học sinh đều có khả năng tự học tốt, đôi lúc việc học thật sự cần sự tương tác trực tiếp giữa người dạy và người học.

Chính vì thế, trong khuôn khổ lần này, em sẽ phát triển hệ thống khóa học trực tuyến để giải quyết vấn đề trên. Hệ thống ưu tiên vào tính dễ sử dụng, không đòi hỏi người dùng phải có chuyên môn nhiều về máy tính nói chung, có thể đáp ứng.

2. Mục tiêu nghiên cứu

- Phân tích, hiểu nghiệp vụ bài toán thực tế.
- Phân tích và thiết kế hệ thống.
- Phát triển, kiểm thử và vận hành hệ thống đảm bảo hoạt động ổn định.
- Tìm hiểu kiểm thử đơn vị.
- Xây dựng ứng dụng, báo cáo và tài liệu kèm theo.

3. Đối tượng nghiên cứu

Đối tượng của hệ thống là học sinh, sinh viên, giáo viên có nhu cầu sử dụng hệ thống để có thể dạy - học trực tuyến. Đối tượng này bao gồm cả những người rành hoặc không rành về công nghệ, cho nên trang web cần có giao diện trực quan, dễ dùng, hỗ trợ người dùng tương tác với nhau nhiều nhất có thể.

Từ đây, khi nói đến người dùng, ta sẽ hiểu là người sử dụng hệ thống để học trực tuyến. Khi nói đến giáo viên, ta nói tới người tương tác với học sinh, là giáo viên dạy học sinh.

4. Phạm vi nghiên cứu

Đối tượng của đề tài "Xây dựng website bán khoá học online sử dụng NodeJS" là các trung tâm, các giáo viên có nhu cầu chia sẻ khoá học. Đây là những cá nhân, tổ chức quan tâm đến việc quản lý và bán khoá học một cách hiệu quả, tối ưu hóa quy trình kinh doanh và tạo trải nghiệm tốt cho khách hàng. Phạm vi nghiên cứu của đề tài này sẽ tập trung vào việc xây dựng một hệ thống website quản lý bán khoá học. Nghiên cứu sẽ tập trung vào việc phân tích và thiết kế giao diện, tích hợp các chức năng quản lý và bán khoá học, xây dựng

một cơ sở dữ liệu để lưu trữ thông tin về khoá học, tài liệu học, giá cả và thông tin bài giảng.

5. Phương pháp nghiên cứu

Phương pháp nghiên cứu về mặt lý thuyết:

- + Tiến hành thu thập các tài liệu có liên quan đến đề tài.
- + Tổng hợp các tài liệu đã thu thập và tiến hành phân tích.
- + Chọn lọc các tài liệu nghiên cứu để thực hiện báo cáo đồ án tốt nghiệp.

Phương pháp nghiên cứu trong thực nghiệm:

- + Phân tích các sản phẩm có chức năng tương tự.
- + Tiến hành xây dựng sản phẩm theo phân tích và yêu cầu thực tế.
- + Áp dụng công nghệ vào xây dựng hệ thống.
- + Xây dựng kế hoạch kiểm thử, thiết kế và thực thi kiểm thử.
- + Tổng hợp tất các kiến thức đã học hoàn thành báo cáo đồ án.

CHƯƠNG 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU

1.1. Giới thiệu chung

Đề tài "Xây dựng website bán khoá học online sử dụng NodeJS" nhằm tạo ra một hệ thống website hoàn chỉnh để quản lý và bán các khoá học online. Website này sẽ cung cấp một giao diện trực quan và dễ sử dụng cho khách hàng để tìm hiểu và mua khoá học. Khách hàng có thể xem thông tin chi tiết về các khoá học được đăng tải.

Ngoài ra, website cũng sẽ cung cấp chức năng quản lý cho quản trị viên. Quản trị viên có quyền kiểm soát hệ thống khóa học và người dùng. Mục tiêu của đề tài này là xây dựng một website quản lý bán khoá học chuyên nghiệp, giúp cung cấp khóa học chất lượng và dịch vụ tốt nhất cho khách hàng. Đồng thời, nó cũng giúp tối ưu hóa quá trình quản lý và bán khoá học cho các giáo viên có nhu cầu.

Về website này sẽ có rất nhiều vấn đề khác nhau. Trong đề tài này, em tập trung nghiên cứu về xây dựng các chức năng liên quan tới các khóa học và phát triển một ứng dụng website bán khoá học cho khách hàng với các công cụ được sử dụng là NodeJS, ReactJS, MongoDB ...làm cơ sở lý thuyết.

1.2. Tổng quan

1.2.1. Tổng quan về lập trình Website

Website còn gọi là trang web (có thể nhầm lẫn với “web page”) hoặc trang mạng, là một tập hợp trang web, thường chỉ nằm trong một tên miền hoặc tên miền phụ trên World Wide Web của Internet. Một trang web là tập tin HTML hoặc XHTML có thể truy nhập dùng giao thức HTTP. Trang mạng có thể được xây dựng từ các tệp tin HTML (trang mạng tĩnh) hoặc vận hành bằng các CMS chạy trên máy chủ (trang mạng động). Một website có thể được xây dựng từ nhiều ngôn ngữ lập trình khác nhau như Java, PHP, ASP.NET,....



Hình 1.1. Lập trình Website

Website có 2 dạng cơ bản là website tĩnh và website động.

- Website tĩnh: Quản trị viên không thể tùy ý thay đổi nội dung và hình ảnh. Dạng web này cần lập trình viên có kiến thức về HTML cơ bản. Website tĩnh viết trên nền tảng HTML CSS và một số hiệu ứng từ Java.
- Website động: Là website có bộ công cụ quản trị để tùy biến nội dung dành cho người quản trị. Người quản trị có thể tùy ý thay đổi nội dung và hình ảnh. Ngôn ngữ xây dựng website động thường là PHP, ASP.NET, Java,...

Google Firebase Cloud Message

Firebase Cloud Messaging (FCM) là một dịch vụ gửi thông báo, tin nhắn đa nền tảng được cung cấp bởi Google, cho phép bạn gửi tin nhắn, thông báo một cách đáng tin cậy và hoàn toàn miễn phí tới các thiết bị đã được đăng ký với FCM.

Nguyên tắc hoạt động: Các thiết bị client sẽ đăng ký `device_token` lên cho FCM. Các thông báo, tin nhắn được soạn và gửi từ một ứng dụng, từ Notifications composer của firebase cung cấp, FCM sẽ nhận những thông

báo này và xử lý gửi về các thiết bị đã đăng ký với FCM từ trước. Khi các thiết bị có kết nối mạng thì thông báo sẽ được gửi về ứng dụng thành công.

Với FCM, có 2 loại thông báo, tin nhắn mà bạn có thể gửi tới ứng dụng, đó là:

- *Notification messages*: Đôi khi được gọi là "thông báo (tin nhắn) hiển thị", chúng được xử lý tự động bởi FCM SDK. Notification messages chứa các key dữ liệu đã được định nghĩa trước. Sử dụng Notification messages khi bạn chỉ muốn hiển thị các thông báo đến các ứng dụng clients.
- *Data messages*: Là thông báo (tin nhắn) sẽ được xử lý bởi các ứng dụng client. Data messages chứa các cặp key - value do người dùng định nghĩa. Sử dụng Data messages khi bạn muốn xử lý các thông báo trên chính ứng dụng của bạn.

Thông báo Firebase hoạt động khác nhau tùy thuộc vào trạng thái của ứng dụng. Trong android, để nhận Firebase message thì cần phải tạo một service và extend `FirebaseMessagingService`. Để xử lý các thông báo nhận được bạn cần override phương thức `onMessageReceived`. Phương thức này xử lý được hầu hết các loại tin nhắn, ngoại trừ các trường hợp sau:

- Nhận Notification messages khi ứng dụng đang ở trạng thái background. Trong trường hợp này, thông báo sẽ được gửi đến khay hệ thống của thiết bị, khi người dùng chạm vào thông báo sẽ mở trình khởi chạy ứng dụng mặc định.
- Nhận thông báo có chứa cả Notification messages và Data messages khi ứng dụng đang ở trạng thái background. Trong trường hợp này, phần Notification sẽ được gửi đến khay hệ thống, còn phần Data sẽ được sử dụng cho trình khởi chạy các Activity khi chạm vào thông báo.

1.2.2. Cơ sở dữ liệu

1.2.3.1. NoSQL Database

Thuật ngữ NoSQL được giới thiệu lần đầu vào năm 1998 sử dụng làm tên gọi chung cho các lightweight open source relational database (cơ sở dữ liệu quan hệ nguồn mở nhỏ) nhưng không sử dụng SQL cho truy vấn. Vào năm

2009, Eric Evans, nhân viên của Rackspace giới thiệu lại thuật ngữ NoSQL trong một hội thảo về cơ sở dữ liệu nguồn mở phân tán. Thuật ngữ NoSQL đánh dấu bước phát triển của thế hệ database mới: distributed (phân tán) và non-relational (không ràng buộc). Đây là cũng 2 đặc tính quan trọng nhất.

❖ Có một vài lí do chứng minh cho sự ra đời của NoSQL:

- Sở dĩ người ta phát triển NoSQL xuất phát từ yêu cầu cần những database có khả năng lưu trữ dữ liệu với lượng cực lớn, truy vấn dữ liệu với tốc độ cao mà không đòi hỏi quá nhiều về năng lực phần cứng cũng như tài nguyên hệ thống và tăng khả năng chịu lỗi.
- Giải quyết được một số vấn đề mà relational database không giải quyết được.

❖ Một số đặc điểm chung về NoSQL:

- High Scalability: Gần như không có một giới hạn cho dữ liệu và người dùng trên hệ thống.
- High Availability: Do chấp nhận sự trùng lặp trong lưu trữ nên nếu một node (commodity machine) nào đó bị chết cũng không ảnh hưởng tới toàn bộ hệ thống.
- Atomicity: Độc lập data state trong các operation.
- Consistency: chấp nhận tính nhất quán yếu, có thể không thấy ngay được sự thay đổi mặc dù đã cập nhật dữ liệu.
- Durability: dữ liệu có thể tồn tại trong bộ nhớ máy tính nhưng đồng thời cũng được lưu trữ lại đĩa cứng.
- Deployment Flexibility: việc bổ sung thêm/loại bỏ các node, hệ thống sẽ tự động nhận biết để lưu trữ mà không cần phải can thiệp bằng tay. Hệ thống cũng không đòi hỏi cấu hình phần cứng mạnh, đồng nhất.
- Modeling flexibility: Key-Value pairs, Hierarchical data (dữ liệu cấu trúc), Graphs.
- Query Flexibility: Multi-Gets, Range queries (load một tập giá trị dựa vào một dãy các khóa).

a. Key value stores

Lưu trữ kiểu key-value là kiểu lưu trữ dữ liệu NoSQL đơn giản nhất sử dụng từ một API. Chúng ta có thể nhận được giá trị cho khóa, đặt một giá trị cho một khóa, hoặc xóa một khóa từ dữ liệu. Ví dụ, giá trị là ‘blob’ được lưu trữ thì chúng ta không cần quan tâm hoặc biết những gì ở bên trong. Từ các cặp giá trị được lưu trữ luôn luôn sử dụng truy cập thông qua khóa chính và thường có hiệu năng truy cập tốt và có thể dễ dàng thu nhỏ lại.

b. Column oriented database

Cơ sở dữ liệu column-family lưu trữ dữ liệu trong nhiều cột trong mỗi dòng với key cho từng dòng. Column families là một nhóm các dữ liệu liên quan được truy cập cùng với nhau. Ví dụ, với khách hàng, chúng ta thường xuyên sử dụng thông tin cá nhân trong cùng một lúc chứ không phải hóa đơn của họ. Cassandra là một trong số cơ sở dữ liệu column-family phổ biến. Ngoài ra còn có một số cơ sở dữ liệu khác như HBase, Hypertable và Amazon DynamoDB. Cassandra có thể được miêu tả nhanh và khả năng mở rộng dễ dàng với các thao tác viết thông qua các cụm. Các cụm không có node master, vì thế bất kỳ việc đọc và ghi nào đều có thể được xử lý bởi bất kỳ node nào trong cụm.

c. Graph database

Kiểu đồ thị này cho phép bạn lưu trữ các thực thể và quan hệ giữa các thực thể. Các đối tượng này còn được gọi là các nút, trong đó có các thuộc tính. Mỗi nút là một thể hiện của một đối tượng trong ứng dụng. Quan hệ được gọi là các cạnh, có thể có các thuộc tính. Cạnh có ý nghĩa định hướng; các nút được tổ chức bởi các mối quan hệ. Các tổ chức của đồ thị cho phép các dữ liệu được lưu trữ một lần và được giải thích theo nhiều cách khác nhau dựa trên các mối quan hệ. Thông thường, khi chúng ta lưu trữ một cấu trúc đồ thị giống như trong RDBMS, nó là một loại duy nhất của mối quan hệ. Việc tăng thêm một mối quan hệ có nghĩa là rất nhiều thay đổi sơ đồ và di chuyển dữ liệu, mà không phải là trường hợp khó khi chúng ta đang sử dụng cơ sở dữ liệu đồ thị. Trong

cơ sở dữ liệu đồ thị, băng qua các thành phần tham gia hoặc các mối quan hệ là rất nhanh. Các mối quan hệ giữa các node không được tính vào thời gian truy vấn nhưng thực sự tồn tại như là một mối quan hệ. Đi qua các mối quan hệ là nhanh hơn so với tính toán cho mỗi truy vấn.

d. Document Oriented databases

Tài liệu là nguyên lý chính của cơ sở dữ liệu kiểu dữ liệu. Dữ liệu lưu trữ và lấy ra là các tài liệu với định dạng XML, JSON, BSON,... Tài liệu miêu tả chính nó, kế thừa từ cấu trúc dữ liệu cây. Có thể nói cơ sở dữ liệu tài liệu là 1 phần của key-value. Cơ sở dữ liệu kiểu tài liệu như MongoDB cung cấp ngôn ngữ truy vấn đa dạng và cú pháp như là cơ sở dữ liệu như đánh index,... Một số cơ sở dữ liệu tài liệu phổ biến mà chúng ta hay gặp là MongoDB, CouchDB, Terastore, OrientDB, RavenDB.

❖ Ưu điểm của NoSQL:

- Không sử dụng SQL
- Không khai báo ngôn ngữ truy vấn dữ liệu
- Không định nghĩa schema
- Có 1 số nhóm dạng: Key-Value pair storage, Column Store, Document Store, Graph databases
- Dữ liệu phi cấu trúc và không thể đoán trước
- Ưu tiên cho hiệu năng cao, tính sẵn sàng cao và khả năng mở rộng.

❖ Nhược điểm của NoSQL:

- Không thích hợp cho mọi loại dữ liệu.
- Thiếu tính nhất quán (consistency).
- Hạn chế trong việc truy vấn phức tạp.
- Công cụ quản lý không phát triển đồng đều.
- Khả năng mở rộng không đồng đều.

Tuy nhiên, nhược điểm này không đồng nghĩa với việc NoSQL không có ưu điểm. Mỗi loại cơ sở dữ liệu đều có các trường hợp sử dụng riêng và cần phải được đánh giá kỹ lưỡng để đảm bảo phù hợp với nhu cầu cụ thể của dự án.

1.2.3.2. MongoDB

MongoDB là một hệ thống cơ sở dữ liệu NoSQL mã nguồn mở có thiết kế dựa trên việc lưu trữ dữ liệu dưới dạng tài liệu JSON linh hoạt, gọi là Bản ghi. Điều này mang lại sự linh hoạt và dễ dàng trong việc lưu trữ và truy xuất dữ liệu, đồng thời giúp giải quyết các vấn đề về khả năng mở rộng và hiệu suất trong các ứng dụng web, di động và phân tích dữ liệu lớn.



Hình 1.2. MongoDB

Dưới đây là một số đặc điểm chính của MongoDB:

- Mô Hình Lưu Trữ Tài Liệu: MongoDB lưu trữ dữ liệu dưới dạng các tài liệu JSON, giúp biểu diễn dữ liệu theo cách tự nhiên và linh hoạt. Điều này giúp dễ dàng thêm, sửa đổi hoặc loại bỏ các trường dữ liệu mà không cần thay đổi cấu trúc cơ sở dữ liệu.
- Khả năng mở rộng ngang: MongoDB cho phép mở rộng cơ sở dữ liệu ngang, tức là chia dữ liệu thành các phân đoạn và phân phối chúng trên nhiều máy chủ. Điều này giúp tăng khả năng chịu tải và hiệu suất của hệ thống một cách dễ dàng và tự động.
- Ngôn ngữ truy vấn linh hoạt: MongoDB cung cấp một ngôn ngữ truy vấn mạnh mẽ và linh hoạt, bao gồm các tính năng như truy vấn nhúng, truy vấn

định dạng JSON và các toán tử phức tạp cho phép truy vấn dữ liệu theo nhiều tiêu chí khác nhau.

- Tích hợp dễ dàng với các ngôn ngữ lập trình: MongoDB có các thư viện và API chính thức cho nhiều ngôn ngữ lập trình phổ biến như JavaScript, Python, Java và C#. Điều này giúp các nhà phát triển tích hợp và tương tác với cơ sở dữ liệu một cách dễ dàng và hiệu quả.
- Độ bền cao và sao lưu tự động: MongoDB có các tính năng bảo đảm sẵn sàng cao và tự động sao lưu, giúp đảm bảo tính toàn vẹn và khả dụng của dữ liệu ngay cả trong điều kiện mất kết nối hoặc lỗi hệ thống.

Với những ưu điểm trên, MongoDB đã trở thành một lựa chọn phổ biến cho các ứng dụng hiện đại đòi hỏi tính linh hoạt, mở rộng và hiệu suất cao.

1.2.3. Frontend

1.2.3.1. ReactJS

React là thư viện viết bằng Javascript được tạo ra với mục đích dùng để xây dựng giao diện người dùng (User interface - UI).

ReactJS được xây dựng xung quanh các component, chứ không dùng template như các framework khác. Tạo một component bằng phương thức `createClass` của đối tượng React.

Đặc tính của React:

- Props: Khi sử dụng components, chúng ta có thể thêm các attributes gọi là props. Những attributes này được gọi ra trong component bằng `this.props`.
- Trạng thái State: Mọi component đều có một object state và một object props. State được thiết lập sử dụng method `setState`.
- Dữ liệu một chiều: Trong react dòng dữ liệu của ứng dụng là một chiều thông qua state và props, có nghĩa là trong cấu trúc nhiều tầng components, một component cha có thể quản lý state và truyền xuống component bên dưới thông qua props. State có thể được update sử dụng method `setState` để đảm bảo UI được cập nhật liên tục, kết quả sẽ được truyền xuống component con sử dụng attributes thông qua `this.props`.

1.2.3.2. Redux

Redux là một thư viện Javascript giúp tạo ra một lớp quản lý trạng thái của ứng dụng, được dựa trên nền tảng tư tưởng của kiến trúc Flux do Facebook giới thiệu, redux thường kết hợp hoàn hảo với React.

Nguyên lý:

- Nguồn dữ liệu tin cậy duy nhất: State của toàn bộ ứng dụng được chứa trong một object tree nằm trong Store duy nhất
- Trạng thái chỉ được phép đọc: Cách duy nhất để thay đổi State của ứng dụng là phát một Action
- Thay đổi chỉ bằng hàm thuần túy: Để chỉ ra cách mà State được biến đổi bởi Action chúng ta dùng các pure function gọi là Reducer.

Cấu trúc của mô hình Redux gồm 4 phần:

- Action: là nơi mang các thông tin dùng để gửi từ ứng dụng đến Store.
- Reducer: là nơi xác định State thay đổi như thế nào.
- Store: là nơi quản lý State, cho phép truy cập state qua `getState()`, update State qua `dispatch(action)`, đăng kí listener qua `subscribe(listener)`.
- View: hiển thị dữ liệu được cung cấp bởi store. Hooks chính thức được giới thiệu trong phiên bản React 16.8. Nó cho phép chúng ta sử dụng state và các tính năng khác của React mà không phải dùng đến Class.

Từ phiên bản 16.8 trở đi, người dùng đã có thể sử dụng state trong stateless (functional) component, việc mà từ trước tới nay bắt buộc phải khai báo Class. Có thể thấy, các nhà phát triển React họ đang muốn hướng đến 1 tương lai Functional Programming thay vì sử dụng những Class. Cộng với việc không sử dụng Class kế thừa từ React Component nữa nên giờ đây kích thước bundle sẽ được giảm đáng kể bởi code sử dụng Hooks.

Để có thể thay thế được Class thì React Hooks cung cấp cho người dùng một bộ các built-in Hooks, giúp chúng ta sử dụng được các thành phần tạo nên React, có 2 loại built-in đó là: Basic Hooks và Additional Hooks.

Một số basic hooks hay dùng: useState, useEffect.

Một số basic hooks hay dùng: useReducer, useMemo, useCallback.

Java là một ngôn ngữ lập trình hướng đối tượng, dựa trên lớp được thiết kế để có càng ít phụ thuộc thực thi càng tốt. Nó là ngôn ngữ lập trình có mục đích chung cho phép các nhà phát triển ứng dụng viết một lần, chạy ở mọi nơi (WORA), nghĩa là mã Java đã biên dịch có thể chạy trên tất cả các nền tảng hỗ trợ Java mà không cần biên dịch lại. Các ứng dụng Java thường được biên dịch thành bytecode có thể chạy trên bất kỳ máy ảo Java (JVM) nào bất kể kiến trúc máy tính bên dưới. Cú pháp của Java tương tự như C và C++, nhưng có ít cơ sở cấp thấp hơn các ngôn ngữ trên. Java runtime cung cấp các khả năng động (chẳng hạn như phản ánh và sửa đổi mã thời gian chạy) thường không có sẵn trong các ngôn ngữ biên dịch truyền thống.

Java ban đầu được James Gosling tại Sun Microsystems (sau đó đã được Oracle mua lại) phát triển và được phát hành vào năm 1995 như một thành phần cốt lõi của nền tảng Java của Sun Microsystems. Các trình biên dịch Java, máy ảo và thư viện lớp thực thi gốc và tham chiếu ban đầu được Sun phát hành theo giấy phép độc quyền. Kể từ tháng 5 năm 2007, tuân theo các thông số kỹ thuật của Quy trình Cộng đồng Java, Sun đã cấp phép hầu hết các công nghệ Java của mình theo Giấy phép Công cộng GNU. Oracle cung cấp Máy ảo Java HotSpot của riêng mình, tuy nhiên việc triển khai tham chiếu chính thức là OpenJDK JVM, là phần mềm mã nguồn mở miễn phí và được hầu hết các nhà phát triển sử dụng và là JVM mặc định cho hầu hết các bản phân phối Linux.

1.2.4. Backend

1.2.4.1. NodeJS

NodeJS là một nền tảng được xây dựng trên V8 JavaScript Engine – trình thông dịch thực thi mã Javascript, giúp xây dựng các ứng dụng web một cách

đơn giản và dễ dàng mở rộng. NodeJS được phát triển bởi Ryan Dahl và năm 2009 và có thể chạy trên nhiều hệ điều hành khác nhau: OS X, Microsoft Windows, Linux. Ưu điểm của NodeJS là tốc độ xử lý nhanh. Nhờ cơ chế xử lý bất đồng bộ, NodeJS có thể xử lý hàng ngàn kết nối cùng một lúc mà không gặp khó khăn nào. NodeJS chạy đa nền tảng phía server sử dụng kiến trúc hướng sự kiện Event-driven, với cộng đồng sử dụng lớn và hoàn toàn miễn phí.

ExpressJS là một framework nhỏ, nhưng linh hoạt được xây dựng trên nền tảng của Nodejs. Nó cung cấp tính năng mạnh mẽ để phát triển web và mobile. Framework này giúp cho việc phát triển ứng dụng được rút ngắn đi rất nhiều. Nó hỗ trợ Rest API, cho phép định nghĩa Middleware hỗ trợ cho việc tổ chức và tái sử dụng code, nó giúp định nghĩa routes và các request method đến server một cách dễ dàng.

1.2.4.2. JWT

JSON Web Mã (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.

Nói một cách khác, JWT là sự kết hợp (bởi dấu .) một Object Header dưới định dạng JSON được encode base64, một payload object dưới định dạng JSON được encode base64 và một Signature cho URI cũng được mã hóa base64 nốt. Ứng dụng JWT thường gặp, đó là:

Authentication: Tình huống thường gặp nhất, khi user logged in, mỗi request tiếp đó đều kèm theo chuỗi token JWT, cho phép người dùng có thể truy cập đường dẫn, dịch vụ và tài nguyên được phép ứng với token đó.

Information Exchange: JSON Web Token cũng là một cách hữu hiệu và bảo mật để trao đổi thông tin giữa nhiều ứng dụng. Redis (Remote Dictionary Server) là một mã nguồn mở dùng để lưu trữ dữ liệu có cấu trúc có thể sử dụng như một cơ sở dữ liệu NoSql. Ưu điểm lớn nhất của redis là lưu trữ dữ liệu trên

Ram, nên việc đọc ghi dữ liệu rất nhanh chóng, thích hợp để làm bộ nhớ đệm. MongoDB là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table- based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ có các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh. MongoDB lần đầu ra đời bởi MongoDB Inc., tại thời điểm đó là thế hệ 10, vào tháng Mười năm 2007, nó là một phần của sản phẩm PaaS (Platform as a Service) tương tự như Windows Azure và Google App Engine. Sau đó nó đã được chuyển thành nguồn mở từ năm 2009. MongoDB đã trở thành một trong những NoSQL database nổi trội nhất bây giờ, được dùng làm backend cho rất nhiều website như eBay, SourceForge và The New York Times.

Ưu điểm của MongoDB:

- Dữ liệu lưu trữ phi cấu trúc, không có tính ràng buộc, toàn vẹn nên tính sẵn sàng cao, hiệu suất lớn và dễ dàng mở rộng lưu trữ.
- Dữ liệu được caching (ghi đệm) lên RAM, hạn chế truy cập vào ổ cứng nên tốc độ đọc và ghi cao.

1.2.4.3. Restful API

API là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng dưới dạng phổ biến như JSON hay XML. REST (REpresentational State Transfer) - là một dạng chuyển đổi cấu trúc dữ liệu, là một phong cách kiến trúc cho việc thiết kế các ứng dụng có kết nối. Nó sử dụng HTTP đơn giản để tạo cho giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE, v.v đến một URL để xử lý dữ liệu. RESTful API là một tiêu chuẩn

dùng trong việc thiết kế API cho các ứng dụng web (thiết kế web services) để tiện cho việc quản lý các resource. Nó được sử dụng phổ biến để giao tiếp giữa các ứng dụng (web, mobile, . . .) với nhau. Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE. . .) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API. Chương 3 em đã trình bày về các công nghệ sử dụng trong hệ thống

Tổng quan về Kiểm thử đơn vị

Kiểm thử đơn vị là một quá trình phát triển phần mềm trong đó các phần nhỏ nhất có thể kiểm thử của một ứng dụng, được gọi là các đơn vị, được xem xét kỹ lưỡng và độc lập để hoạt động đúng.

Mục tiêu chính của kiểm thử đơn vị là lấy phần mềm có thể kiểm thử nhỏ nhất trong ứng dụng, cô lập nó khỏi phần còn lại của mã và xác định xem nó có hoạt động chính xác như mong đợi hay không.

Mỗi đơn vị được kiểm tra riêng biệt trước khi tích hợp vào các mô-đun để kiểm tra các giao diện giữa các mô-đun. Sử dụng kiểm thử đơn vị hiệu quả, một tỷ lệ lớn các lỗi được xác định. Kiểm thử đơn vị được thực hiện bởi các nhà phát triển.

Mỗi mô-đun được phát triển bởi các nhà thiết kế cần phải được kiểm tra riêng lẻ để xác minh hoạt động đúng để bất kỳ mô-đun bị lỗi nào có thể được sửa ngay lập tức thay vì để nó tồn tại và sau đó gây ra một số vấn đề lớn trong giai đoạn tích hợp.

Các loại kiểm thử đơn vị:

- Kiểm thử đơn vị tự động: là quá trình kiểm tra các tập lệnh và xác định lỗi tự động. Nó được thực hiện với sự trợ giúp của một công cụ kiểm tra tự động chuyên dụng. Các nhà phát triển ngày nay sử dụng phương pháp này để kiểm

tra các tập lệnh vì nó tiết kiệm thời gian, giảm lỗi của con người và cũng cải thiện chất lượng phần mềm.

– Ưu điểm:

- + Giảm chi phí trong thời gian dài vì nó không yêu cầu một nhóm.
- + Nhanh hơn so với kiểm thử đơn vị thủ công.
- + Các trường hợp kiểm thử có thể lặp lại để xác định lỗi không bị phát hiện trước đó.

- + Các trường hợp kiểm thử có thể tái sử dụng để tiết kiệm chi phí bảo trì.
- + Nó cung cấp kết quả chính xác do thiếu sự can thiệp của con người.
- + Tăng năng suất của nhà phát triển bằng cách tự động hóa các tác vụ.
- + Cải thiện khả năng mở rộng.
- + Cho phép phân lô nhiều kịch bản thử nghiệm.

– Nhược điểm:

- + Cài đặt đắt hơn so với kiểm thử thủ công.
- + Yêu cầu một người kiểm tra chuyên nghiệp.
- + Nó không hỗ trợ thử nghiệm ngẫu nhiên,
- + Có giới hạn về môi trường.
- + Vì kiểm thử tự động không liên quan đến con người nên nó không thể đảm bảo tính thân thiện với con người.
- + Nó không phù hợp để thay đổi động tác thiết kế đồ họa giao diện người dùng.

➤ Kiểm thử đơn vị thủ công: là quá trình kiểm tra tập lệnh và xác định lỗi theo cách thủ công. Các thử nghiệm như vậy thường được thực hiện bởi một nhà phân tích QA mà không sử dụng các công cụ tự động hóa. Kiểm thử đơn vị thủ công là lý tưởng khi có một số lượng nhỏ các kiểm thử đơn vị được thực hiện, khi cần sửa chữa nhanh hoặc khi tự động hóa trở nên tốn kém.

– Ưu điểm:

- + Nó không yêu cầu kiến thức mã hóa.

- + Vì nó liên quan đến sự quan sát của con người, người thử nghiệm có thể tìm thấy các vấn đề về khả năng sử dụng và giao diện người dùng.
- + Không có giới hạn môi trường.
- + Nó phù hợp để thay đổi động tác thiết kế đồ họa giao diện người dùng.
- Nhược điểm:
 - + Đòi hỏi nguồn nhân lực.
 - + Quá trình này chậm.
 - + Các trường hợp kiểm thử không được tái sử dụng. Vì vậy bất cứ khi nào có sự thay đổi nhỏ, các trường hợp kiểm thử phải được viết lại từ đầu.
 - + Không có tích hợp để kiểm tra tải và hiệu suất.
 - + Không thể được sử dụng để so sánh hai cơ sở dữ liệu.
 - + Việc lặp lại các bài kiểm tra có thể tốn kém và tốn thời gian.

CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Khảo sát hệ thống

Trong thời đại của Công nghệ 4.0, sự phát triển của internet và thiết bị kỹ thuật số đã mở ra một cánh cửa rộng lớn cho việc học tập và đào tạo trực tuyến. Ngày nay, có nhiều trang web bán khóa học online xuất sắc nhưng đôi khi, việc tìm ra những trang web uy tín và chất lượng có thể là một thách thức. Một số cái tên có thể kể đến như: Unica, Edumall hay Karo ...

- **Unica - Chia sẻ kiến thức thực tiễn:**

Unica là một trong những nền tảng đào tạo trực tuyến hàng đầu tại Việt Nam. Unica không chỉ mang đến hàng ngàn khóa học đa dạng mà còn là cổng kết nối chuyên gia với học viên. Với hơn 450 giảng viên có uy tín, Unica đã thành công đào tạo hơn 200,000 học viên và cung cấp hơn 800 khóa học. Trang web này nổi bật với sự đa dạng của danh mục khóa học, bao gồm ngoại ngữ, marketing, tin học, thiết kế, quay video, kinh doanh, làm đẹp, và nhiều lĩnh vực khác. Phương pháp giảng dạy của Unica là khoa học, với hình ảnh trực quan, giúp học viên dễ tiếp cận kiến thức một cách hiệu quả. Họ cung cấp tính linh hoạt cho học viên, cho phép học bất kỳ lúc nào, ở bất kỳ đâu. Unica cam kết hoàn tiền nếu học viên không hài lòng và thường xuyên có các chương trình khuyến mãi hấp dẫn để giúp học viên tiết kiệm chi phí.

❖ Ưu điểm:

- Các khóa học rất gần gũi và cần thiết trong cuộc sống hằng ngày, bên cạnh đó có những khóa học còn nâng cao kỹ năng, kiến thức cũng như trình độ của học viên.
- Đào tạo trực tuyến thông qua Video với số lượng lớn khóa học (hơn 800 khóa học) thuộc nhiều lĩnh vực khác nhau như lập trình, ngoại ngữ, kinh doanh, marketing, yoga... bao gồm các khóa học từ cơ bản tới nâng cao, nội dung dạy bằng tiếng Việt tạo điều kiện thuận lợi nhất cho học viên.
- Chỉ cần thanh toán một lần duy nhất, bạn có thể sở hữu khóa học mãi mãi để học tập lâu dài.

- Bạn có thể bắt đầu công việc học tập của mình mọi lúc mọi nơi chỉ với một chiếc máy tính có internet.

❖ **Nhược điểm:**

- Một số khóa học gốc của Unica có giá thành khá cao vì không có khuyến mãi.
- Những khóa học có chất lượng không đồng đều vì khóa học đến từ những giảng viên, chuyên gia khác nhau. Bạn cần tham khảo các đánh giá của học viên trước đó hoặc nhờ sự tư vấn của Unica để quyết định đăng ký khóa học cho phù hợp.
- Trong trường hợp có sai sót, lỗi xảy ra, Unica sẽ hoàn tiền khóa học cho bạn.

- **Edumall - Học trực tuyến lớn nhất Đông Nam Á:**

Edumall tự hào là "siêu thị" các khóa học trực tuyến ngắn hạn lớn nhất Đông Nam Á, với hàng nghìn khóa học thuộc mọi lĩnh vực. Edumall nổi tiếng với sứ mệnh "Triệu người dùng nâng trí tuệ" và với đội ngũ giảng viên chuyên nghiệp, giàu kinh nghiệm và mạng lưới học viên rộng khắp cả nước. Edumall đặc biệt dễ tiếp cận cho học viên bằng cách áp dụng phương pháp sư phạm LIPE (Xem bài giảng video - Hỏi, đáp với thầy và bạn học - Thực hành - Đánh giá kết quả). Phương pháp này đã được áp dụng thành công tại các trường đại học hàng đầu thế giới như Harvard và Stanford. Học viên có thể học tại nhà, tại văn phòng, quán cà phê hoặc bất kỳ nơi đâu, bất cứ khi nào, miễn là có thiết bị kết nối internet. Edumall cũng nổi bật với việc chỉ mất chi phí một lần để mua một khóa học, và sau đó bạn có thể học khóa học đó mãi mãi, ngay cả khi nội dung được cập nhật. Điều này giúp tiết kiệm chi phí học tập và đảm bảo việc tiếp nhận kiến thức trở nên dễ dàng và thuận lợi. Các khóa học của Edumall thường được biểu diễn dưới dạng video, do giảng viên biên soạn từ trước và đưa lên hệ thống website edumall.vn. Tổng thời gian trung bình mỗi khóa học kéo dài 3-4 tiếng và được chia thành nhiều bài giảng nhỏ có thời lượng từ 3-5 phút để học viên dễ tiếp nhận và vận dụng thực tiễn.

❖ Ưu điểm:

- Không giới hạn những chủ đề khóa học chia sẻ với những quy mô hàng nghìn khóa học trên nhiều nền tảng khác nhau với những lĩnh vực giảng dạy vô cùng phong phú.
- Danh tiếng và tập khách hàng đã có sẵn là do các khóa học được quảng cáo với tên chung là tên nền tảng giáo viên đang giảng dạy. Họ quảng cáo các khóa học như là một phần họ là sở hữu và đào tạo khóa học đó là đội ngũ của Edumall.
- Các giảng viên chỉ cần tập chung tạo nên một chất lượng cao về khóa học và chỉ cần tập trung đào tạo thật tốt. Tất cả giao dịch bán khóa học bên Edumall hỗ trợ bạn.
- Edumall có một nền tảng đã có danh tiếng sẵn do vậy lộ trình phát triển có kế hoạch truyền thông được xây dựng bài bản. Họ sẽ thay bạn quảng cáo các khóa học và thu hút học viên vì vậy bạn không cần quá lo ngại về vấn đề học viên.
- Những vấn đề liên quan đến quản lý như hỗ trợ người học và thu các học phí giúp họ cải thiện được những chất lượng của website, các ứng dụng đều do edumall phụ trách do vậy bạn có thể dành nhiều thời gian hơn cho các lĩnh vực chuyên môn đó chính là việc tạo lập và thiết kế những khóa học để có thể tập trung vào việc dạy học trực tuyến.

❖ Nhược điểm:

- Về chi phí: Tuy không là người trực tiếp thu phí của các giảng viên tuy nhiên nếu dạy trên đây thì một phần doanh thu bán được từ khóa học sẽ bị trích lại. Khoản chiết khấu này có thể lên đến tầm 80 – 85% do các chi phí họ phải bỏ ra để cho các hoạt động quảng cáo.
- Về chuyên môn: Điều này là tốt cho các học viên và nền tảng khi xây dựng được danh tiếng với đội ngũ thầy cô chuyên nghiệp. Tuy nhiên có nhiều giáo viên không trực tiếp giảng dạy tại các trường đại học nào hoặc không có những chứng chỉ về sư phạm dù đã được kiểm chứng về những

kiến thức và kỹ năng. Yêu cầu về các chứng chỉ chuyên môn là bất lợi đối với những thầy cô bắt đầu vào nghề giảng dạy hay dạy thêm ngoài các công việc chính.

- Thiếu quyền kiểm soát các hoạt động trao đổi mua bán đều phụ thuộc vào các đơn vị cung cấp các nền tảng và giáo viên chỉ có thể nhận lại phần trăm sau cùng do vậy nó cần có yêu cầu về sự minh bạch của tài chính
- Vấn đề cạnh tranh: Trên nền tảng bán khóa học online cùng chủ đề trong phần giới thiệu sẽ không được phản ánh để nêu bật được những đặc điểm của sản phẩm của bạn trong khi nhiệm vụ của chợ khóa học không chỉ là bán khóa học trên nền tảng mà việc bán khóa học nào đó doanh thu về thầy cô nào là dường như không quan trọng vì họ vẫn nhận được số tiền hoa hồng từ việc bán khóa học.
- Không tạo nên được danh tiếng riêng do các khóa học khi quảng cáo đều để tên dưới nền tảng khóa học do vậy các giáo viên đã trở thành một đội ngũ của Edumall như vậy việc xây dựng và tạo dựng thương hiệu riêng khó có thể thực hiện

- Karo - Nền tảng cung cấp các khóa học online chuyên biệt:

Mục tiêu của Karo Education không chỉ dừng lại ở việc cung cấp kiến thức và đào tạo mà còn chú trọng đến việc xây dựng mối quan hệ đối tác lâu dài với khách hàng, hỗ trợ Khách hàng trong việc thực hiện các dự án số hóa và cùng nhau đạt được những thành công đáng kể.

Mặc dù tuổi đời còn non trẻ nhưng Karo biết cách tập trung vào các lĩnh vực quan trọng, cần thiết trong thời đại 4.0 Thông qua việc hướng dẫn và tư vấn sẽ giúp các khách hàng hiểu rõ hơn về tiềm năng của các công nghệ mới như Trí tuệ nhân tạo (AI), Internet of Things (IoT), blockchain và máy học (Machine Learning). Đồng thời, công ty cũng đảm bảo rằng khách hàng biết cách tích

hợp những công nghệ này vào quy trình kinh doanh của họ để tối đa hóa lợi ích và tăng cường độ cạnh tranh trong thị trường ngày càng cạnh tranh khốc liệt.

❖ Ưu điểm

- các khóa học miễn phí.
- Có thể chi trả.
- Chính sách hoàn tiền trong 30 ngày cho các khóa học đã mua.
- Truy cập trọn đời.
- Chứng chỉ hoàn thành Udemy.
- Các khóa học đa dạng.
- Xử lý thanh toán tích hợp.

❖ Nhược điểm

- Giấy chứng nhận không được công nhận;
- Vấn đề kiểm soát chất lượng;
- Người hướng dẫn không phản hồi;
- Cơ cấu giá không đồng nhất.

2.2. Giới thiệu về hệ thống

Hệ thống website mà em xây dựng là một trang web bán khóa học, hướng đến cả người học và người giảng dạy. Đối tượng nghiên cứu chính là các cá nhân muốn truy cập, tham gia và tạo ra các khóa học trực tuyến. Trang web sẽ phục vụ cả hai đối tượng này bằng cách cung cấp một nền tảng trực tuyến linh hoạt và dễ sử dụng.

Phạm vi nghiên cứu sẽ bao gồm việc phát triển các tính năng tìm kiếm giúp khách hàng là các học viên tìm kiếm và mua khóa học một cách dễ dàng, phù hợp với nhu cầu của họ. Đồng thời, hệ thống tập trung vào hiệu năng sử dụng cũng như các yếu tố liên quan tới an toàn và bảo mật cho người dùng. Nghiên cứu cũng xem xét các yếu tố quản lý như quản lý thông tin khách hàng, thông tin khóa học, tài liệu liên quan và hệ thống bài tập, bài kiểm tra... Tăng cường việc quản lý và hiệu quả kinh doanh đem lại trải nghiệm tốt cho người dùng.

2.2.1. Về hệ thống

- Hệ thống sử dụng giao diện web đơn giản, đẹp, thân thiện với người dùng, có sự thống nhất về bố cục và màu sắc của các phần tử trong một trang và giữa các trang với nhau.
- Đảm bảo tất cả thông tin cá nhân cũng như các giao dịch mua bán đã được thực hiện được bảo mật, tuân thủ các quy định về quyền riêng tư như GDPR hay CCPA.
- Tích hợp với cổng thanh toán an toàn và phổ biến cụ thể là Momo, sau khi thực hiện thanh toán hệ thống sẽ cấp quyền truy cập nội dung khóa học cho các học viên.
- Cung cấp công cụ tìm kiếm mạnh mẽ giúp học viên có thể tìm được các khóa học phù hợp với nhu cầu của bản thân.
- Hỗ trợ giải đáp thắc mắc cho các học viên thông qua việc bình luận đưa ra các câu hỏi về các nội dung còn chưa rõ.
- Cung cấp các chức năng phục vụ cho việc tổ chức và quản lý các tài liệu và tài nguyên học liệu cho các khóa học, giảng viên có khả năng đăng tải và quản lý nội dung khóa học bao gồm video bài giảng, tài liệu và hệ thống bài tập, bài kiểm tra.
- Hỗ trợ theo dõi mức độ hoàn thành khóa học và các hành vi của người dùng để có thể hiểu hơn nhu cầu và trải nghiệm của người dùng.

2.2.2. Về người sử dụng

Giao diện dễ sử dụng và trực quan để người dùng có thể dễ dàng tìm kiếm và truy cập các khóa học một cách thuận tiện. Thiết kế một cách hấp dẫn và chuyên nghiệp để tạo ra ấn tượng tích cực đầu tiên và khuyến khích người dùng khám phá thêm.

Nâng cao trải nghiệm mua hàng với thủ tục thanh toán đơn giản, quy trình mua hàng cần được tối ưu hóa để người dùng có thể hoàn thành thanh toán một cách nhanh chóng và dễ dàng.

Cung cấp thông tin mô tả chi tiết về mỗi khóa học để người dùng có thể đưa ra quyết định mua hàng thông minh hơn.

Lưu trữ và sửa đổi thông tin, cho phép người dùng lưu trữ thông tin cá nhân để thuận tiện cho việc truy cập.

Cho phép người dùng thảo luận, chia sẻ kinh nghiệm và tương tác với nhau, phản hồi và đưa ra yêu cầu hỗ trợ thông qua bình luận dưới mỗi bài giảng, người học tương tác với giảng viên, đặt câu hỏi và nhận phản hồi để tối ưu hóa quá trình học tập.

Đảm bảo rằng các khóa học cung cấp nội dung chất lượng và hữu ích cho người học.

Cung cấp phản hồi và thông báo để thông báo cho người dùng về các khóa học mới, cập nhật và sự kiện quan trọng.

2.2.3. Về chức năng

❖ Các chức năng chính của hệ thống

- Quản lý người dùng bao gồm:
 - + Đăng ký và đăng nhập tài khoản.
 - + Quản lý thông tin cá nhân.

Quản lý khóa học:

- + Tạo, chỉnh sửa và xóa khóa học.
- + Cung cấp thông tin chi tiết về mỗi khóa học.
- + Tạo Modules cho khóa học.
- + Tạo hệ thống Assignment và Exam.

Thanh toán:

- + Tích hợp thanh toán an toàn thông qua cổng điện tử Momo.
- + Cung cấp quyền truy cập sau khi thanh toán.

Tìm kiếm:

- + Tìm kiếm khóa học theo từ khóa

Quản lý nội dung:

- + Tải lên và quản lý nội dung khóa học như video bài giảng, tài liệu, hệ thống bài tập và kiểm tra.

Tương tác và hỗ trợ:

- + Phản hồi và đưa ra thắc mắc sau bài học.
- + Thông báo cần thiết về khóa học để kiểm soát tốt hơn.

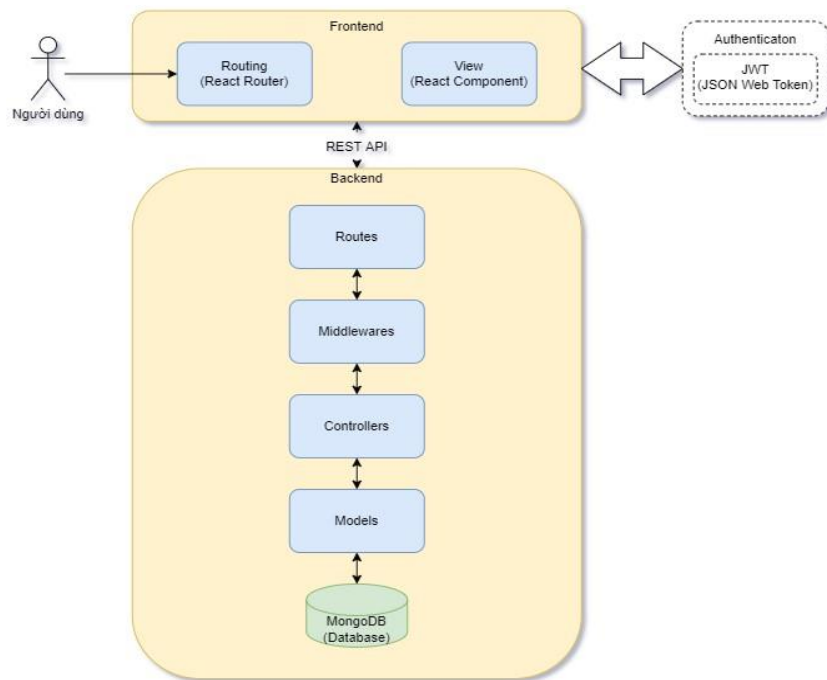
Bảo mật và quản lý dữ liệu:

- + Bảo mật dữ liệu cá nhân của người dùng.
- + Tuân thủ các quy định về quyền riêng tư.

❖ Yêu cầu phi chức năng

- Thời gian phản hồi nhanh chóng.
- Giao diện đẹp và dễ sử dụng.
- Hệ thống hoạt động ổn định và không gặp sự cố thường xuyên.
- Đảm bảo hiệu suất cao và đáp ứng tốt trong điều kiện tải cao.
- Dễ vận hành và bảo trì.

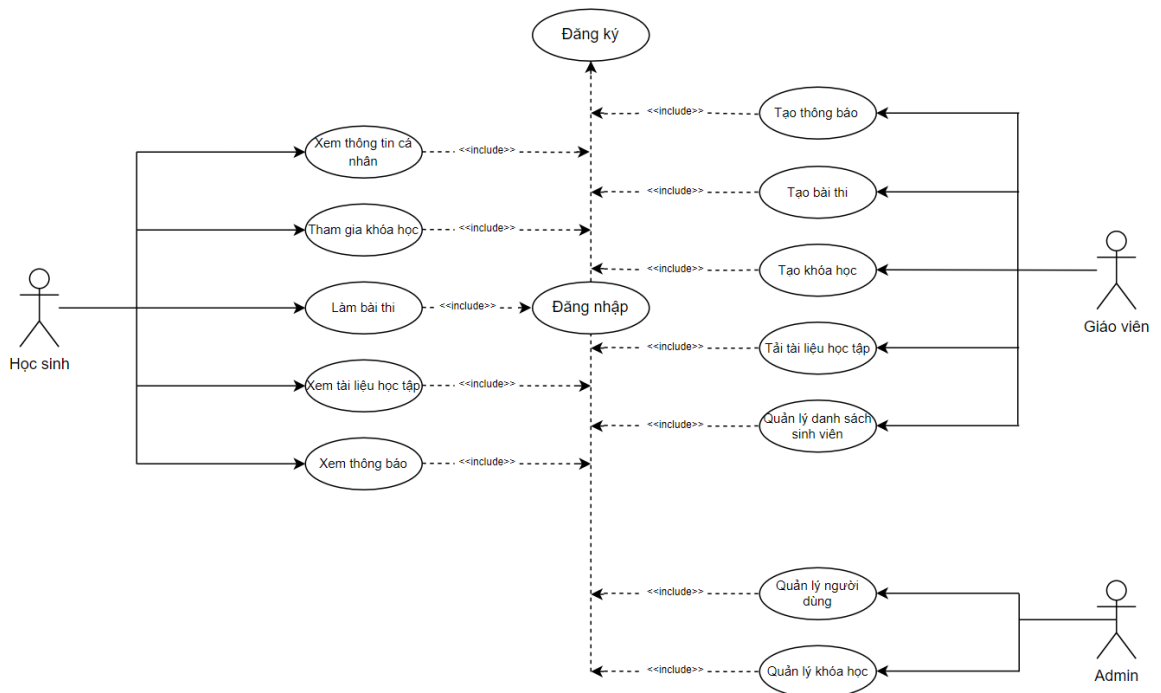
2.3. Thiết kế hệ thống



Hình 2.1. Mô hình tổng quan hệ thống

2.3.1. Mô hình hóa Use case

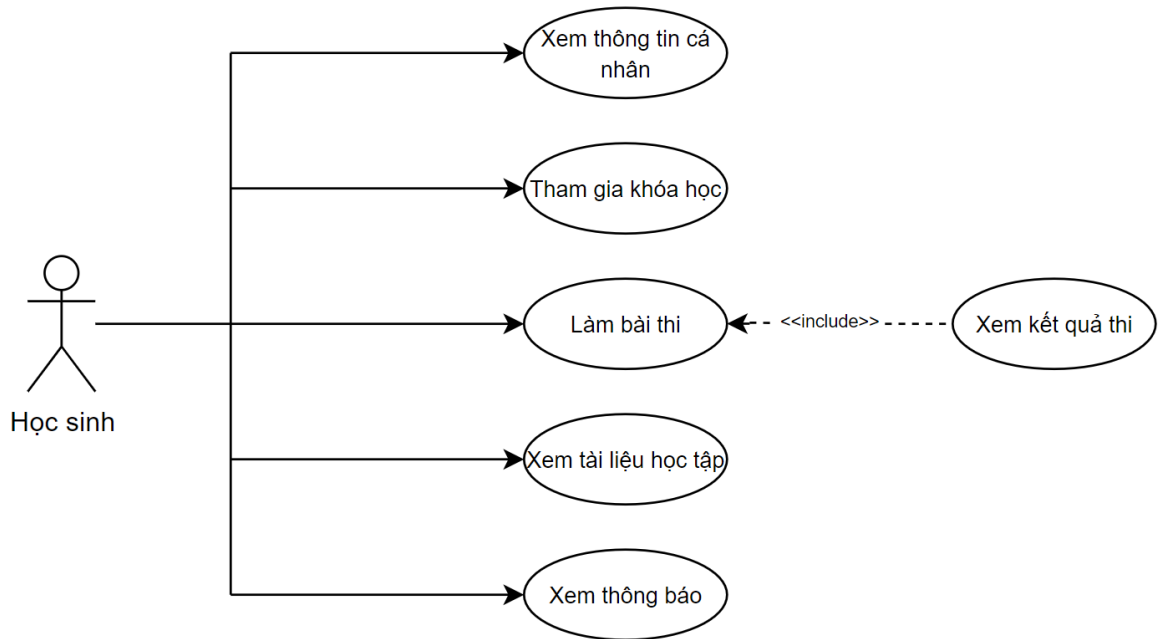
2.3.1.1. Biểu đồ Usecase tổng quan



Hình 2.2. Biểu đồ Usecase tổng quan

2.3.1.2. Biểu đồ Usecase thứ cấp

a. Phân rã UC theo tác nhân Học sinh



Hình 2.3. Phân rã Usecase theo tác nhân học sinh

b. Phân rã UC theo tác nhân Giáo viên



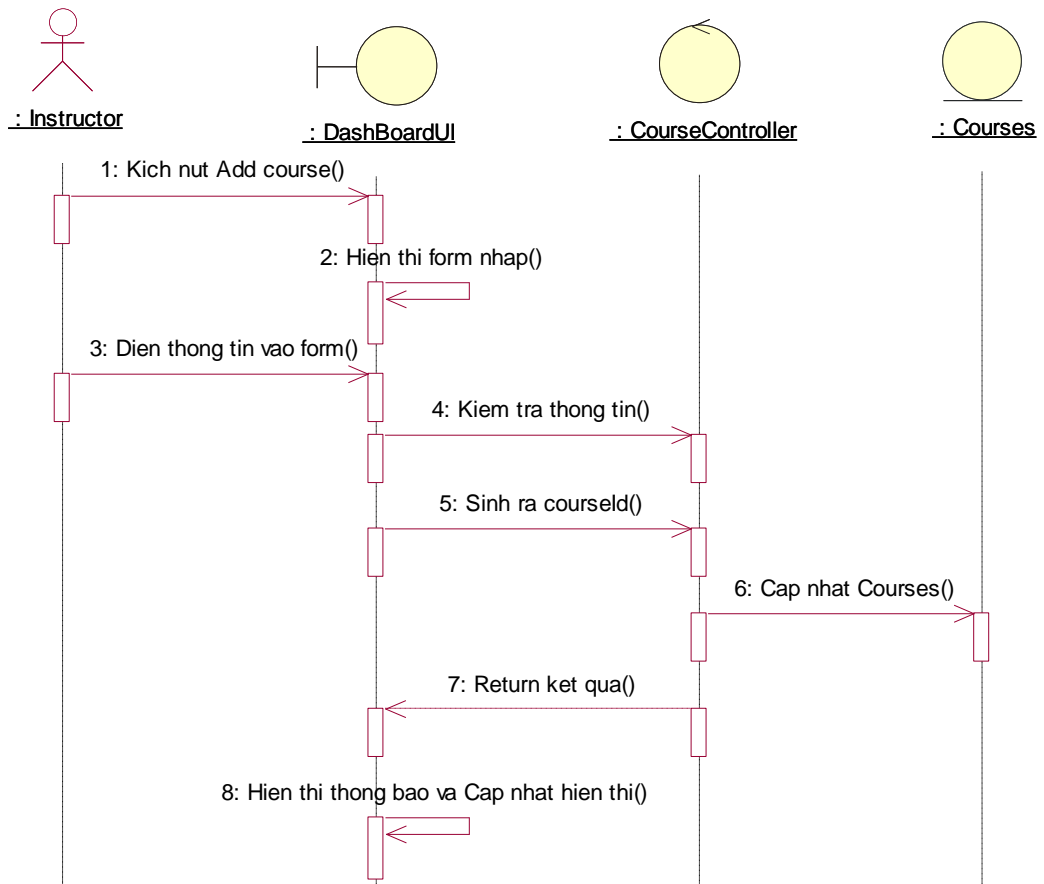
Hình 2.4. Phân rã Usecase theo tác nhân giáo viên

2.3.2. Mô tả chi tiết Use case

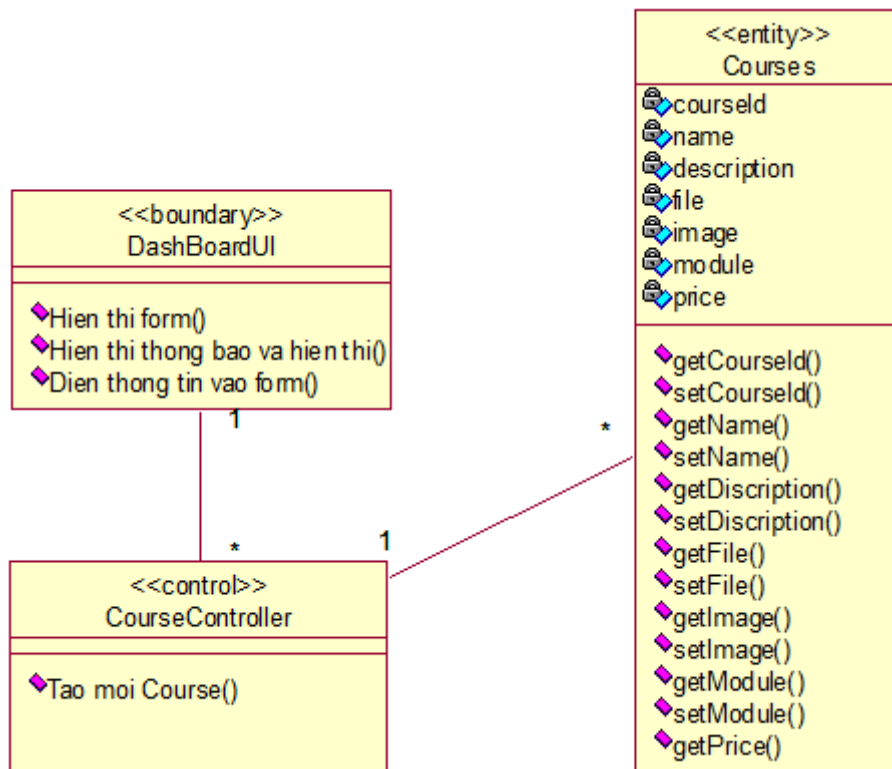
2.3.2.1. Mô tả UC tạo khóa học

Bảng 2.1. Mô tả UC tạo khóa học

Mã Usecase	Tên Usecase		Tạo khoá học
Tác nhân			Giáo viên
Tiền điều kiện			Người dùng đăng nhập vào hệ thống với vai trò là giáo viên
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Người dùng	Chọn chức năng tạo khóa học
	2	Hệ thống	Hiển thị form để nhập dữ liệu
	3	Người dùng	Nhập thông tin khóa học
	4	Người dùng	Nhấn gửi dữ liệu
	5	Hệ thống	Kiểm tra dữ liệu
	6	Hệ thống	Lưu thông tin vào cơ sở dữ liệu
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	7	Người dùng	Hủy chọn chức năng thêm khóa học
	8	Hệ thống	Ẩn form thêm khóa học
	9	Người dùng	Nhập thông tin không phù hợp
	10	Hệ thống	Thông báo thông tin nhập không hợp lệ
Hậu điều kiện			Không



Hình 2.5. Biểu đồ trình tự UC tạo khóa học

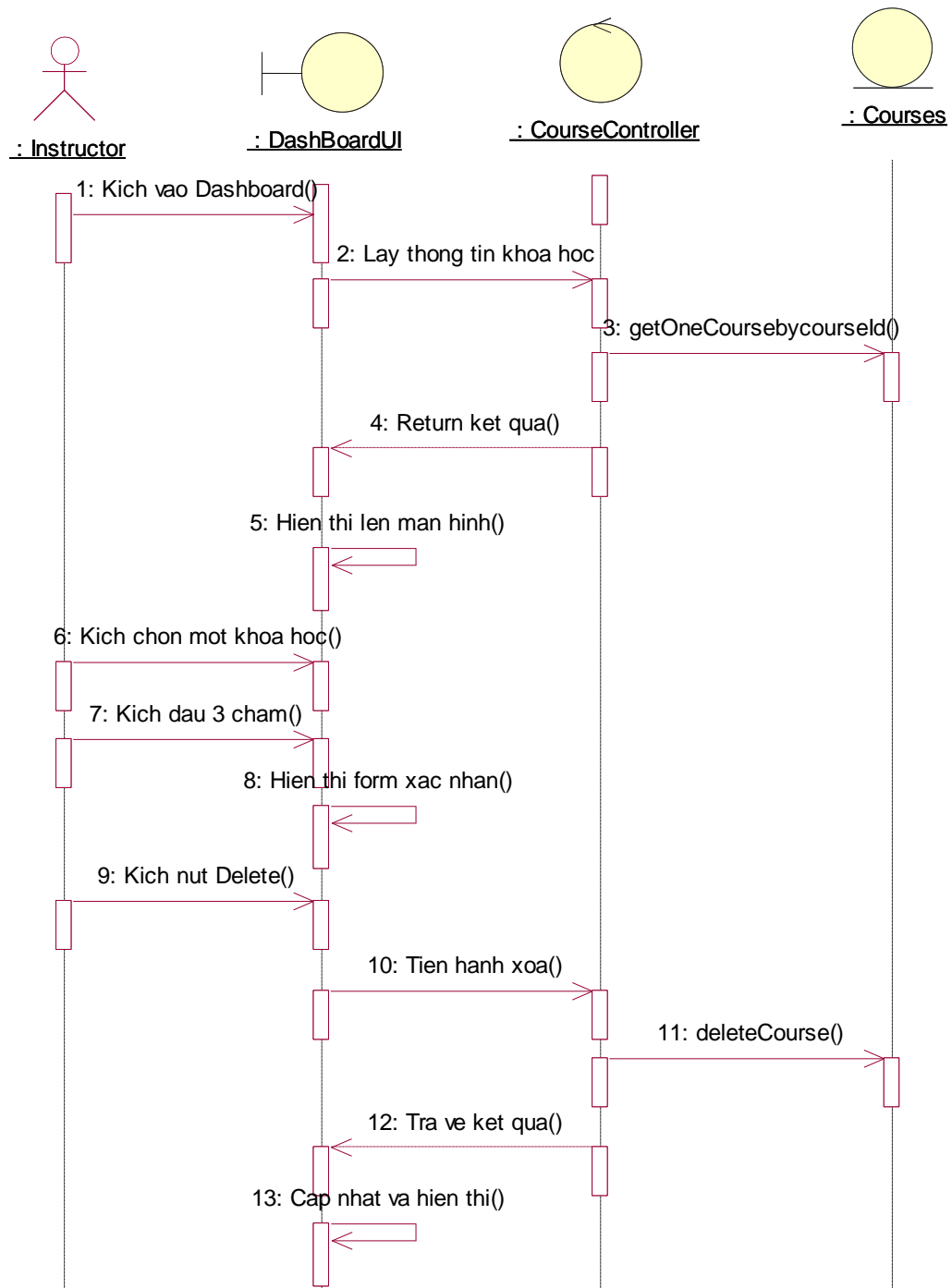


Hình 2.6. Biểu đồ VOPC UC tạo khóa học

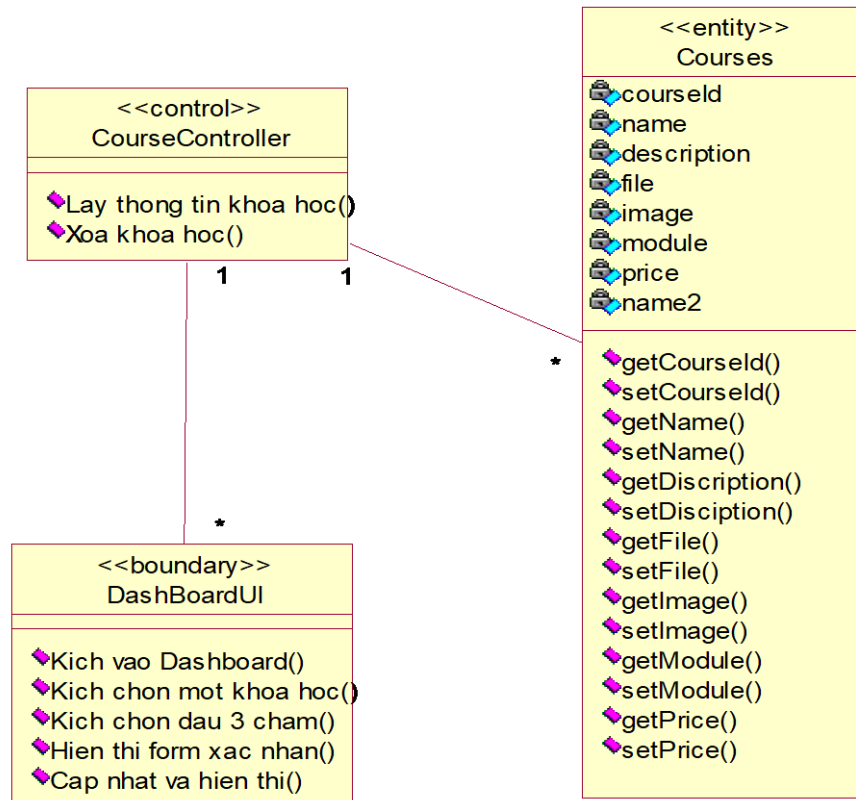
2.3.2.2. Mô tả UC quản lý khóa học

Bảng 2.2. Mô tả UC Quản lý khoá học

Mã usecase	Tên usecase		Quản lý khoá học
Tác nhân	Giáo viên		
Tiền điều kiện	Đăng nhập thành công và thuộc bộ phận giáo viên		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Giáo viên	Chọn chức năng xem thông tin khoá học
	2	Hệ thống	Tìm kiếm dữ liệu trong cơ sở dữ liệu
	3	Hệ thống	Hiển thị thông tin khoá học
	4	Giáo viên	Chọn chức năng xoá khoá học
	5	Hệ thống	Xoá khoá học
	6	Hệ thống	Trả về danh sách khoá học
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	7	Giáo viên	Hủy chọn chức năng xoá khoá học
	8	Hệ thống	Ẩn form xoá khoá học
Hậu điều kiện	Không		



Hình 2.7. Biểu đồ trình tự UC quản lý khóa học

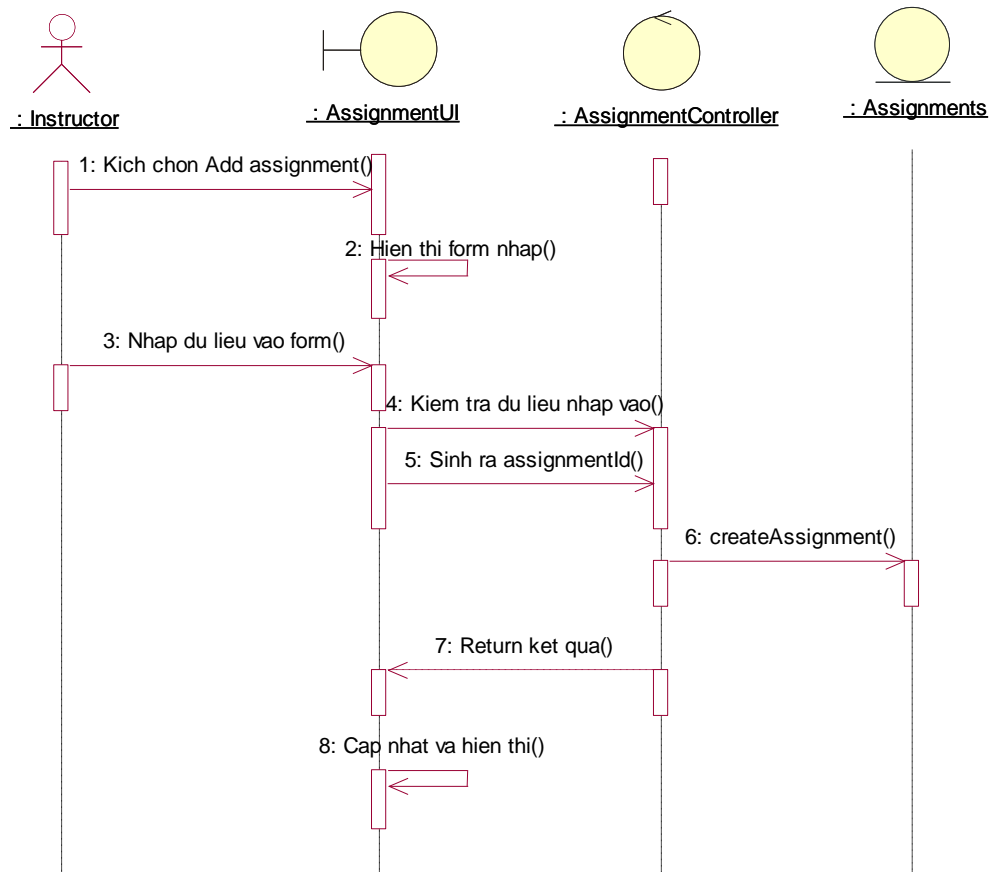


Hình 2.8. Biểu đồ VOPC UC quản lý khóa học

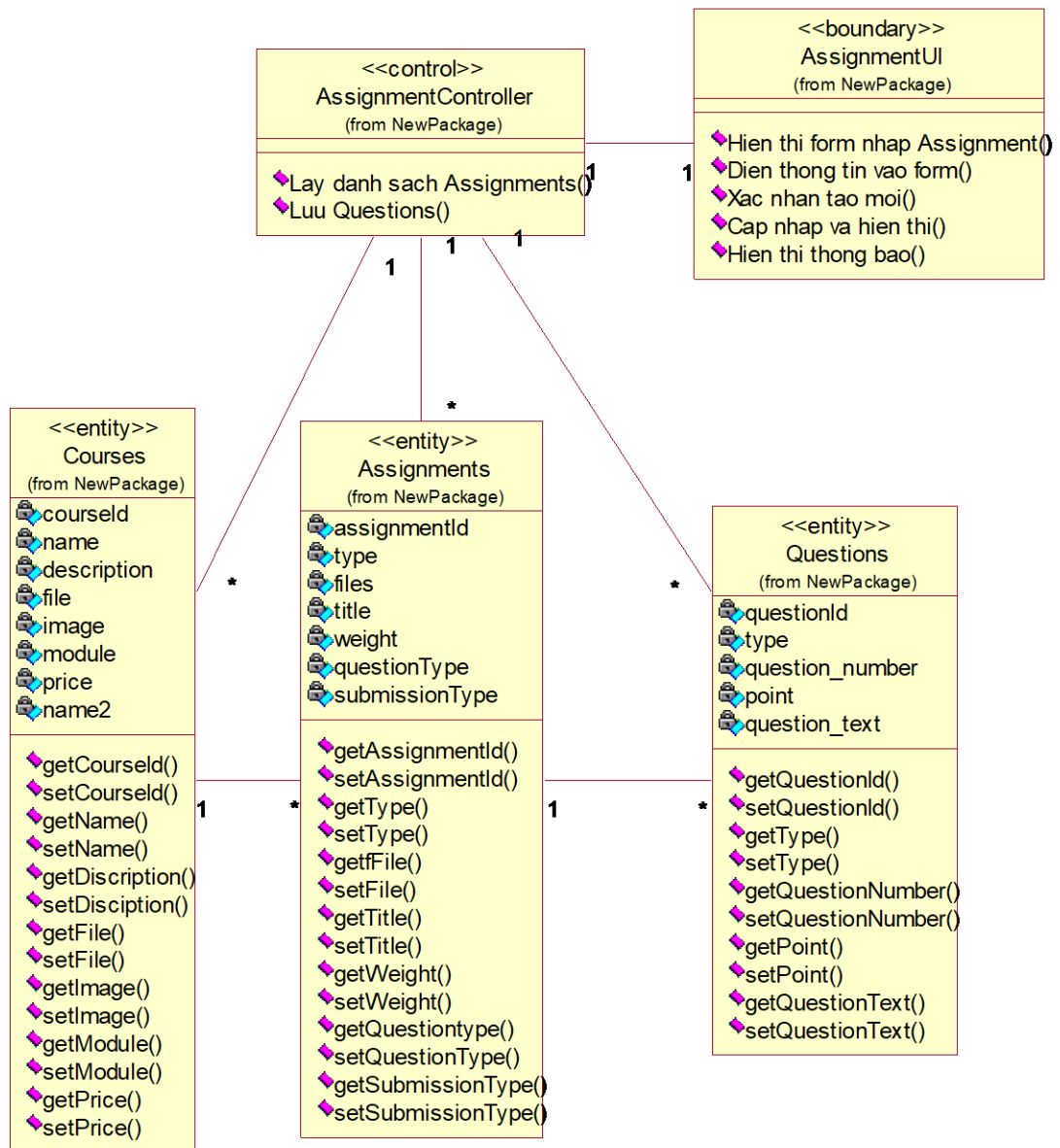
2.3.2.3. Mô tả UC tạo bài kiểm tra

Bảng 2.3. Mô tả UC tạo bài kiểm tra

Mã usecase	Tên usecase		Quản lý người dùng
Tác nhân			Giáo viên
Tiền điều kiện	Đăng nhập thành công và thuộc bộ phận giáo viên		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Giáo viên	Chọn chức năng tạo bài kiểm tra
	2	Hệ thống	Hiển thị giao diện tạo bài kiểm tra
	3	Giáo viên	Nhập thông tin bài kiểm tra và thông tin của từng câu hỏi
	4	Hệ thống	Lưu bài kiểm tra vào cơ sở dữ liệu
	5	Hệ thống	Thông báo tạo bài kiểm tra thành công
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	6	Quản trị viên	Hủy chọn chức năng sửa thông tin người dùng
Hậu điều kiện	Không		



Hình 2.9. Biểu đồ trình tự UC tạo bài kiểm tra

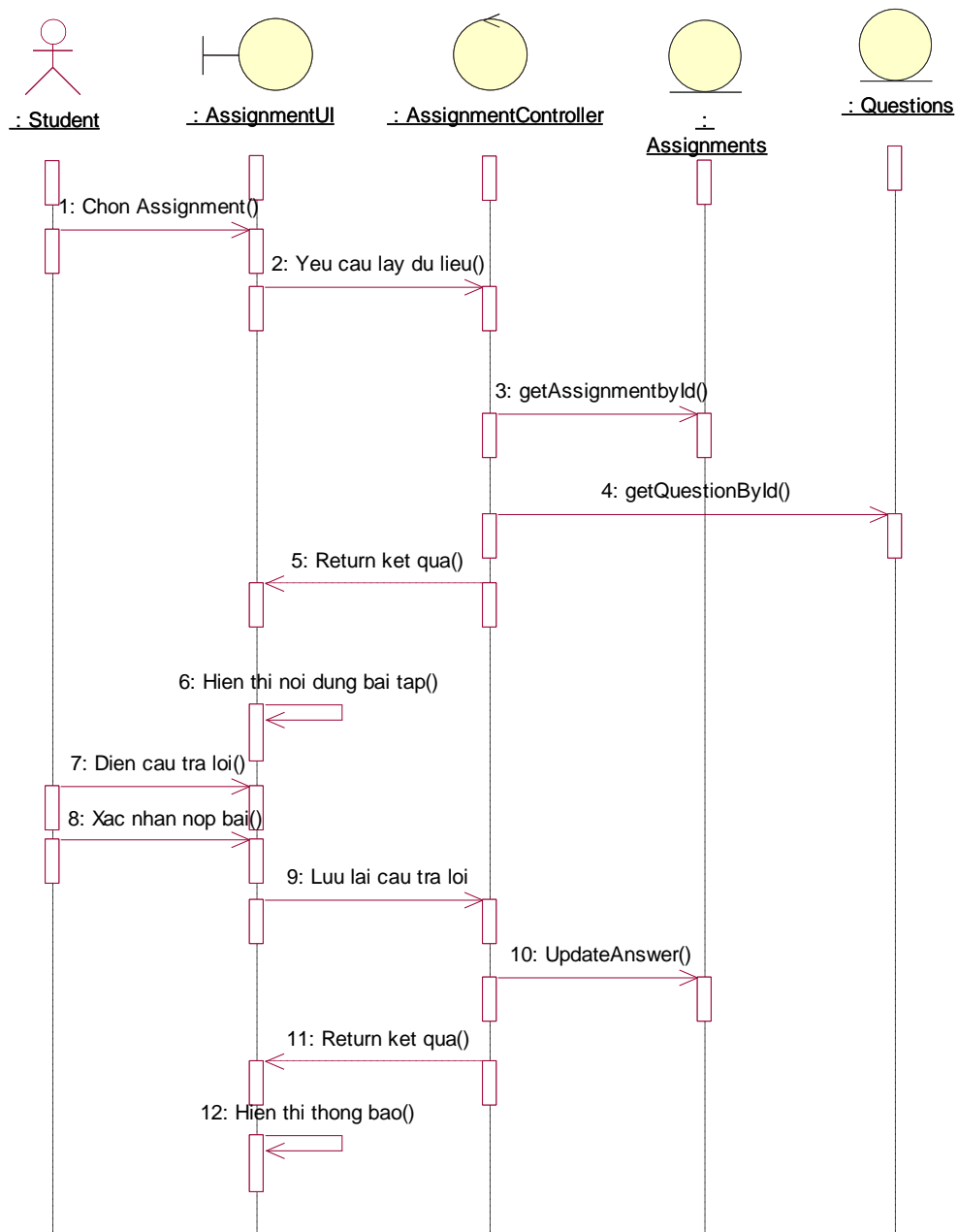


Hình 2.10. Biểu đồ VOPC UC tạo bài kiểm tra

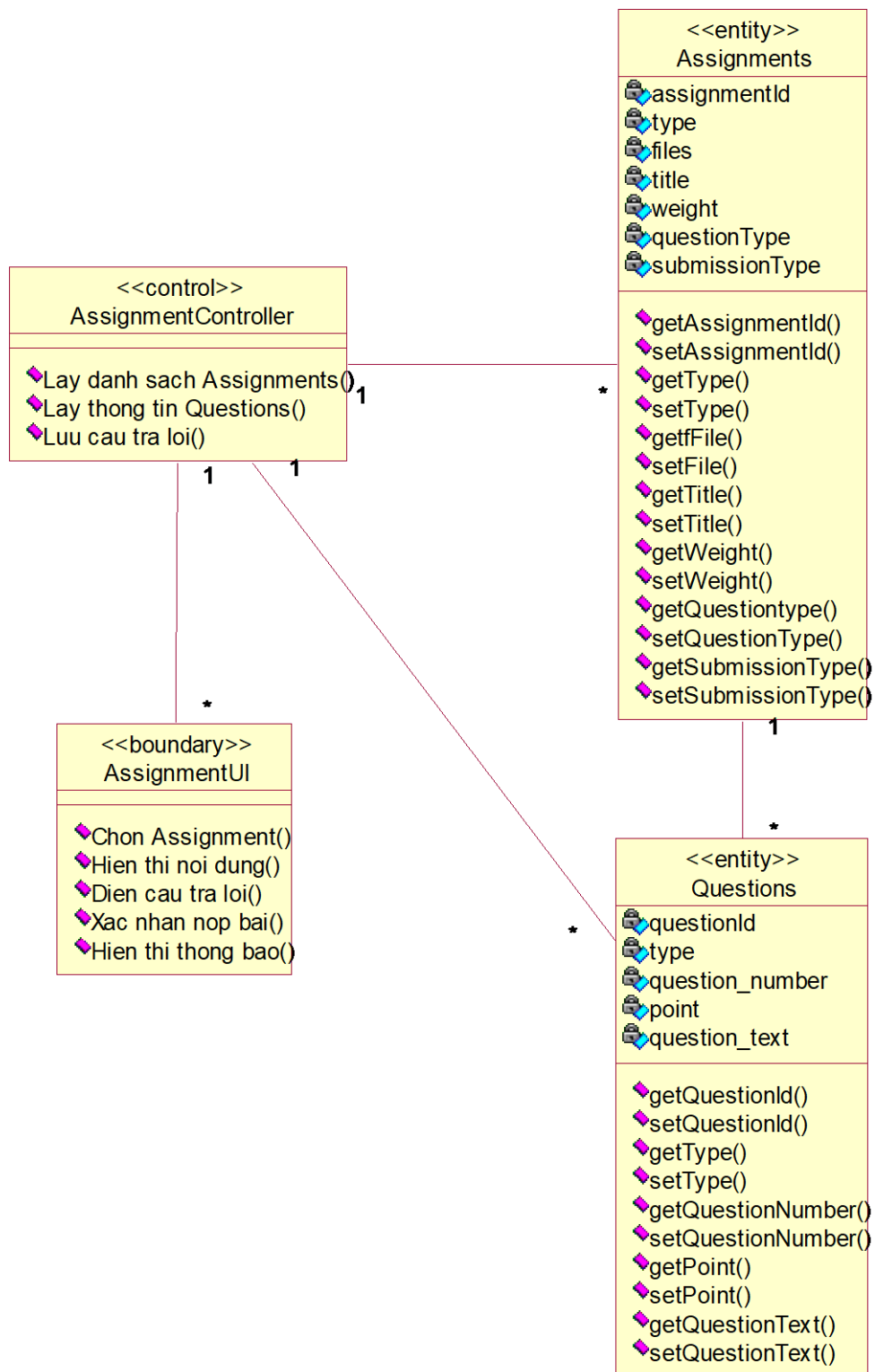
2.3.2.4. Mô tả UC làm bài kiểm tra

Bảng 2.4 Mô tả UC làm bài kiểm tra

Mã usecase	Tên usecase	Làm bài kiểm tra	
Tác nhân	Học sinh		
Tiền điều kiện	Đăng nhập thành công và thuộc bộ phận học sinh		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Học sinh	Chọn chức năng làm bài kiểm tra
	2	Hệ thống	Hiển thị giao diện để làm bài kiểm tra
	3	Học sinh	Làm bài kiểm tra bằng cách điền câu trả lời hoặc upload file
	4	Hệ thống	Lưu lại câu trả lời
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	7	Hệ thống	Nếu thông tin không hợp lệ báo lỗi cho người dùng
Hậu điều kiện	Không		



Hình 2.11. Biểu đồ trình tự UC làm bài kiểm tra

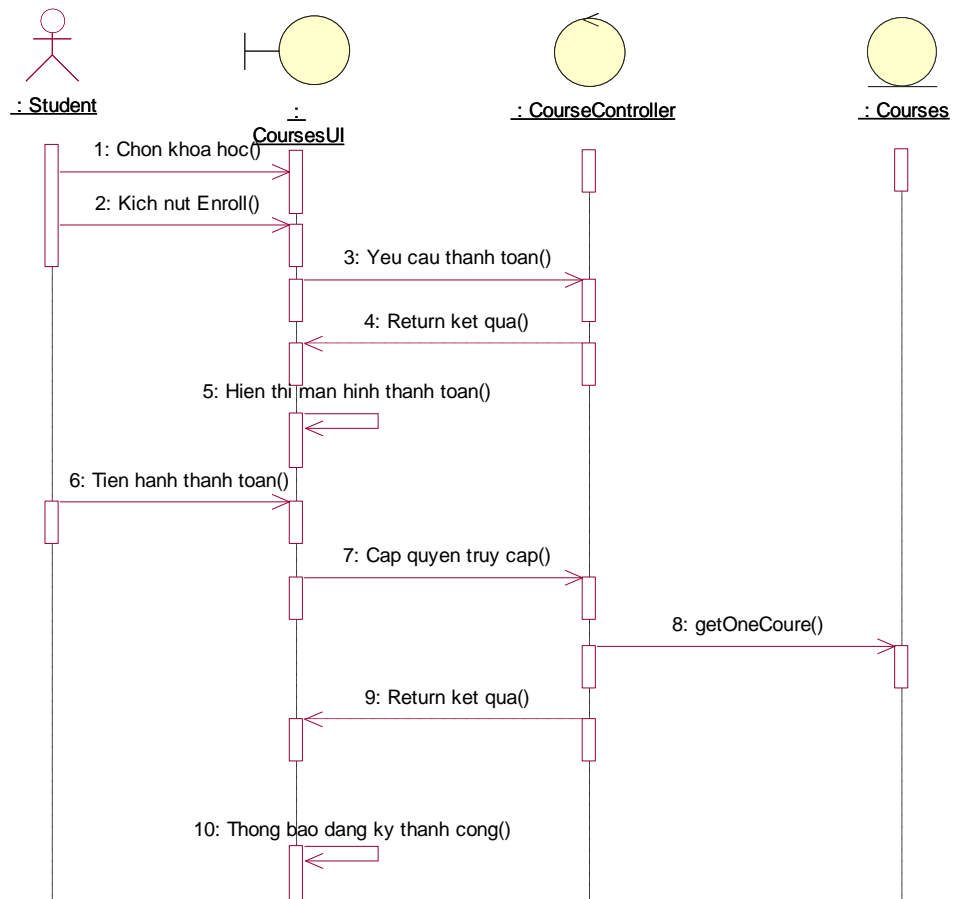


Hình 2.12. Biểu đồ VOPC UC làm bài kiểm tra

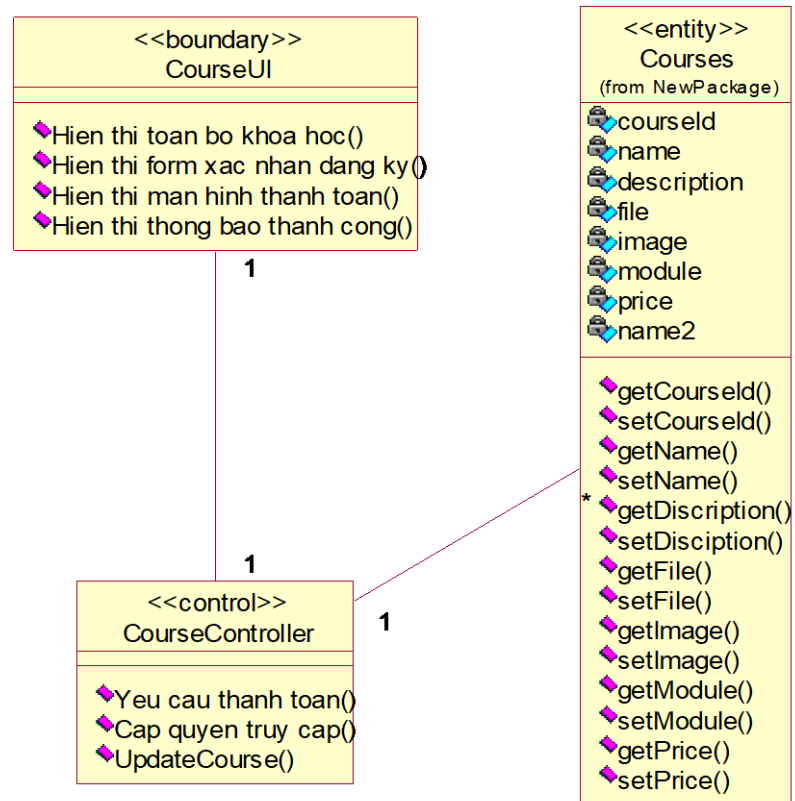
2.3.2.5. Mô tả UC tham gia khóa học

Bảng 2.5. Mô tả UC tham gia khóa học

Mã usecase	Tên usecase		Tham gia khóa học
Tác nhân	Học sinh		
Tiền điều kiện	Đăng nhập thành công, đã tham gia khoá học		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Học sinh	Chọn chức năng tham gia khóa học
	2	Hệ thống	Chuyển sang giao diện chi tiết của khóa học
	3	Học sinh	Học các bài giảng có ở trong khóa học
	4	Hệ thống	Với mỗi một chức năng trong khóa học, người dùng có thể ngoài học qua video có thể học qua các tài liệu như file bài tập
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	5	Hệ thống	Nếu thông tin không hợp lệ báo lỗi cho người dùng
Hậu điều kiện	Không		



Hình 2.13. Biểu đồ trình tự UC tham gia khóa học

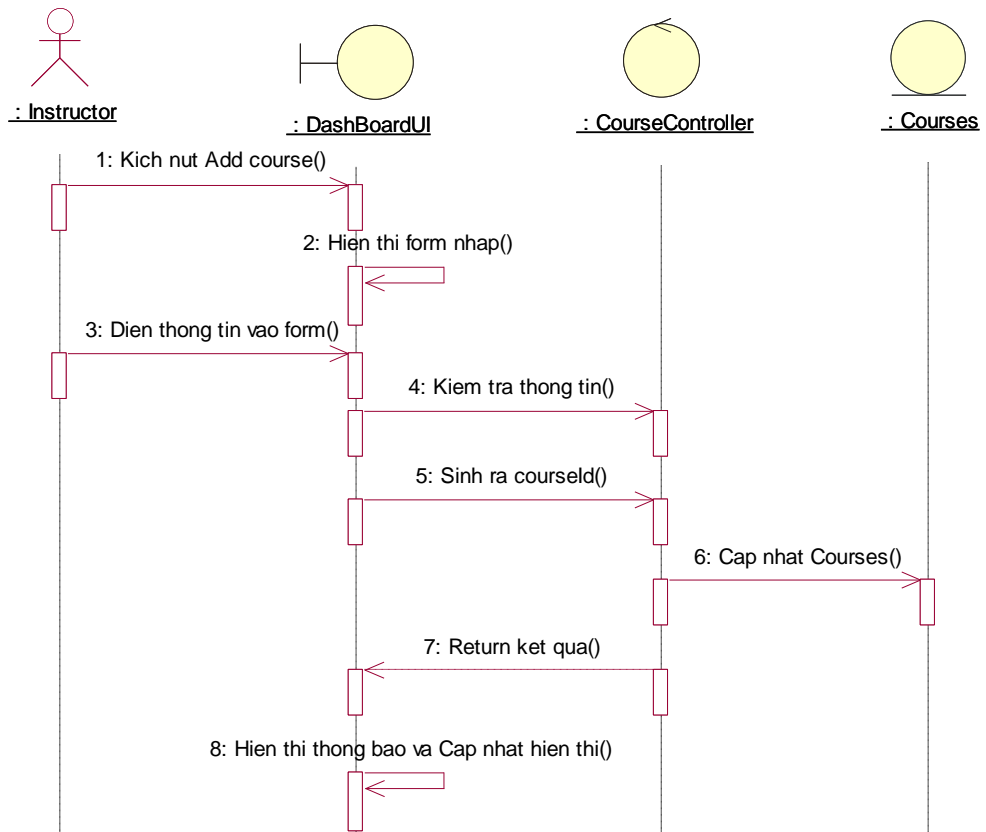


Hình 2.14. Biểu đồ VOPC UC tham gia khóa học

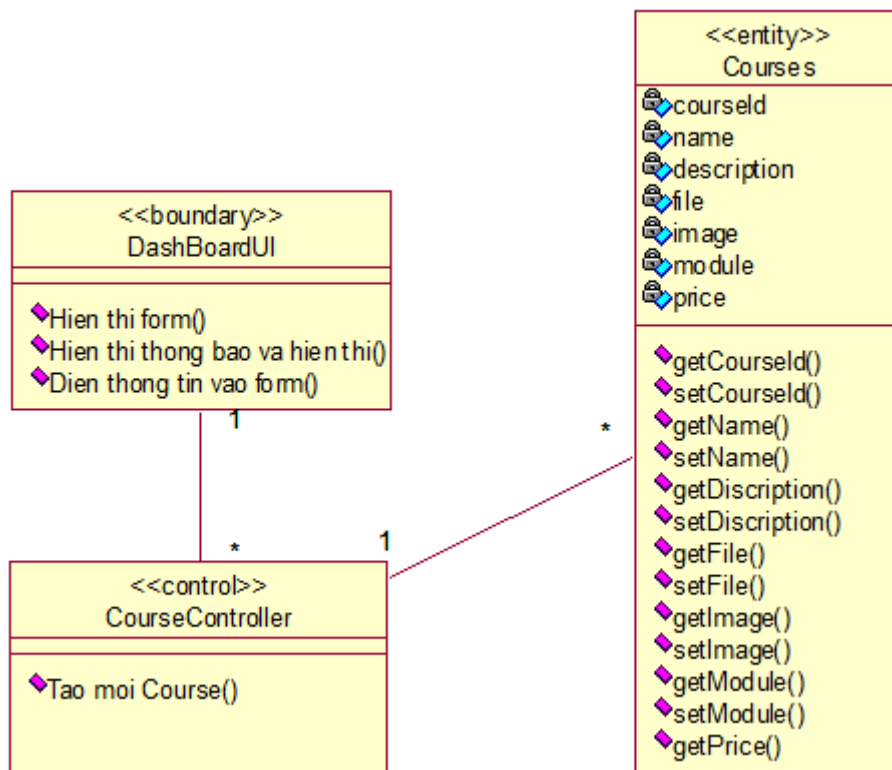
2.3.2.6. Mô tả UC tạo khóa học

Bảng 2.6. Mô tả UC tạo khóa học

Mã Usecase	Tên Usecase		Tạo khoá học
Tác nhân	Giáo viên		
Tiền điều kiện	Người dùng đăng nhập vào hệ thống với vai trò là giáo viên		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Người dùng	Chọn chức năng tạo khoá học
	2	Hệ thống	Hiện thị form để nhập dữ liệu
	3	Người dùng	Nhập thông tin khoá học
	4	Người dùng	Nhấn gửi dữ liệu
	5	Hệ thống	Kiểm tra dữ liệu
	6	Hệ thống	Lưu thông tin vào cơ sở dữ liệu
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	7	Người dùng	Hủy chọn chức năng thêm khoá học
	8	Hệ thống	Ẩn form thêm khóa học
	9	Người dùng	Nhập thông tin không phù hợp
	10	Hệ thống	Thông báo thông tin nhập không hợp lệ
Hậu điều kiện	Không		



Hình 2.15. Biểu đồ trình tự UC tạo khóa học

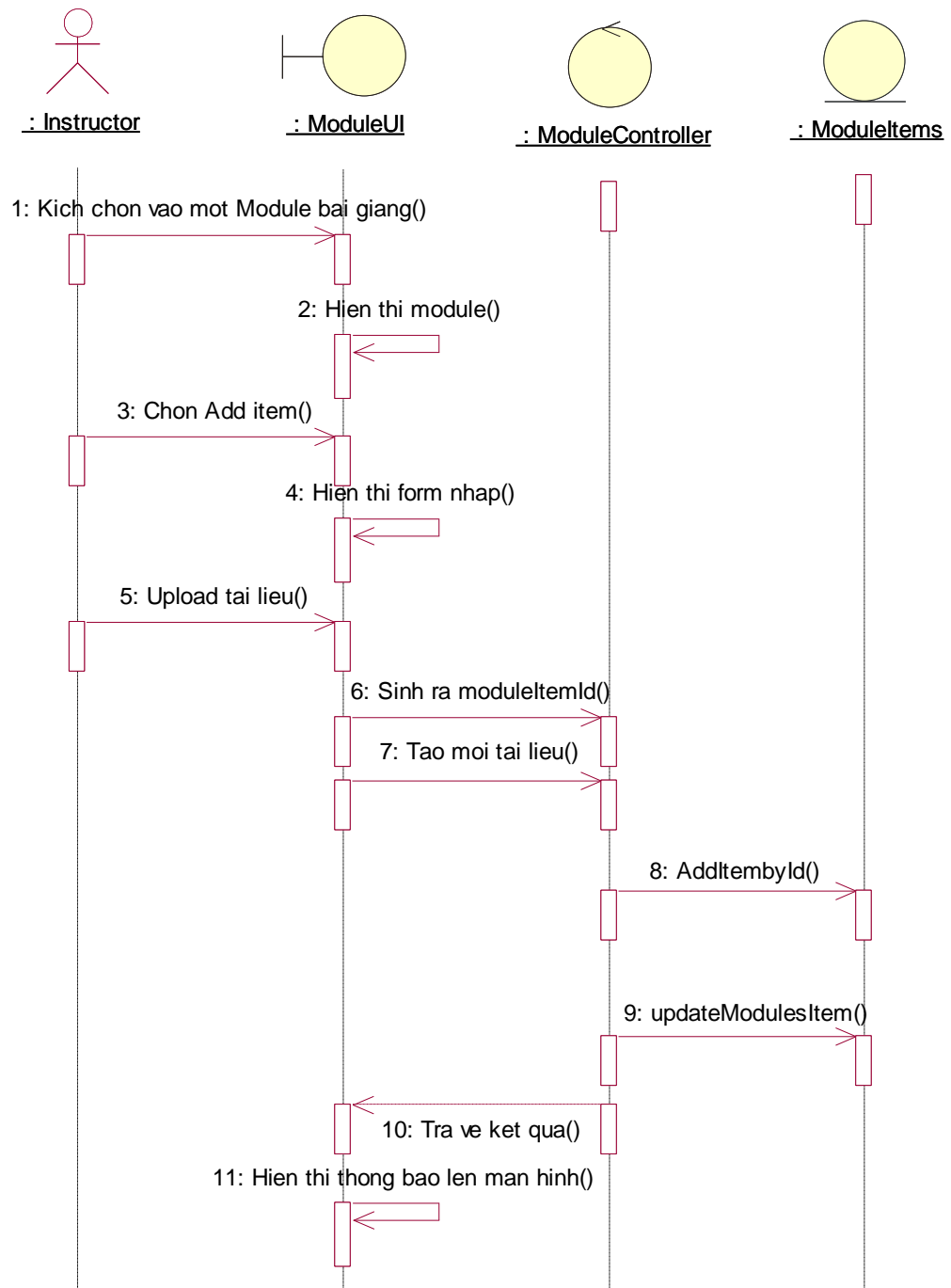


Hình 2.16. Biểu đồ VOPC UC tạo khóa học

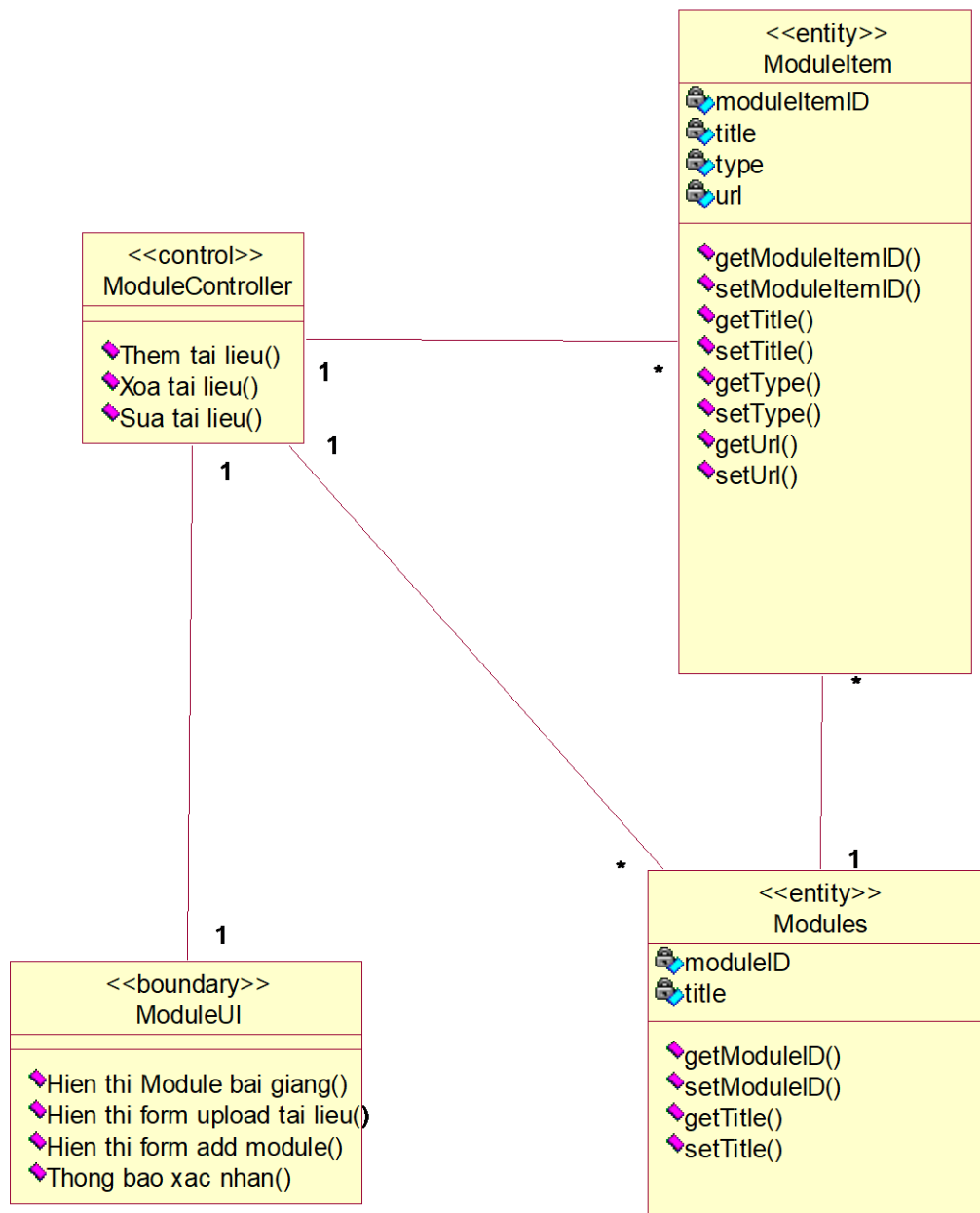
2.3.2.7. Mô tả UC đăng tải tài liệu

Bảng 2.7. Mô tả UC đăng tải tài liệu

Mã usecase	Tên usecase		Đăng tải tài liệu lớp học
Tác nhân	Giáo viên		
Tiền điều kiện	Đăng nhập thành công và thuộc bộ phận giáo viên		
Luồng sự kiện chính	STT	Thực hiện	Hành động
	1	Giáo viên	Chọn chức năng tải lên file trong khoá học
	2	Hệ thống	Hiển thị form để tải lên file
	3	Giáo viên	Chọn file pdf cần tải lên
	4	Giáo viên	Gửi dữ liệu
	5	Hệ thống	Kiểm tra file hợp lệ
	6	Hệ thống	Trả về danh sách file tải lên mới
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	7	Giáo viên	Hủy chọn chức năng tải lên file
	8	Hệ thống	Ẩn form tải lên
	9	Giáo viên	Tải lên file không đúng định dạng hoặc đã tồn tại
	10	Hệ thống	Thông báo dữ liệu không hợp lệ
Hậu điều kiện	Không		



Hình 2.17. Biểu đồ trình tự UC đăng tải tài liệu

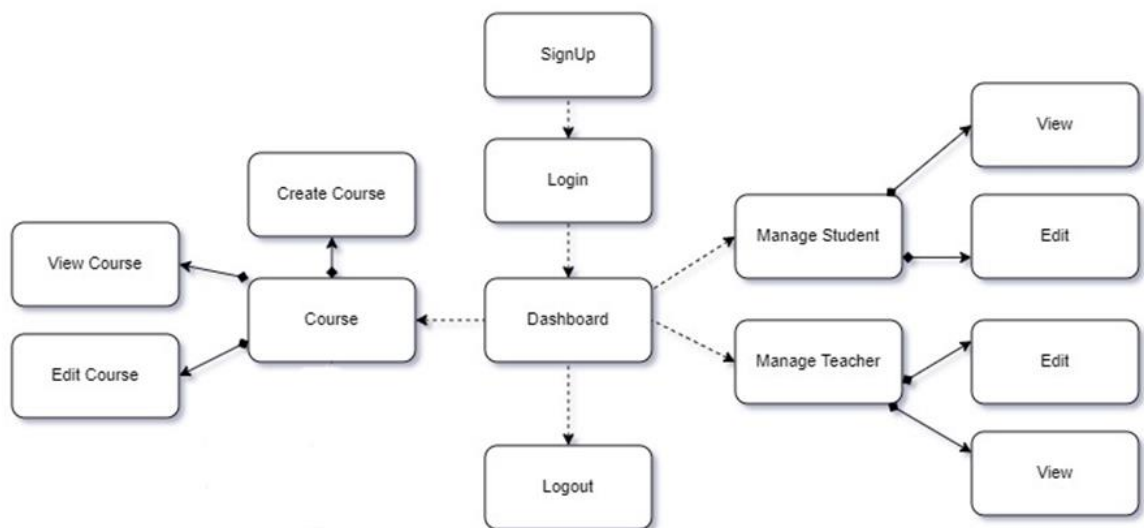


Hình 2.18. Biểu đồ VOPC UC đăng tải tài liệu

2.4. Thiết kế giao diện

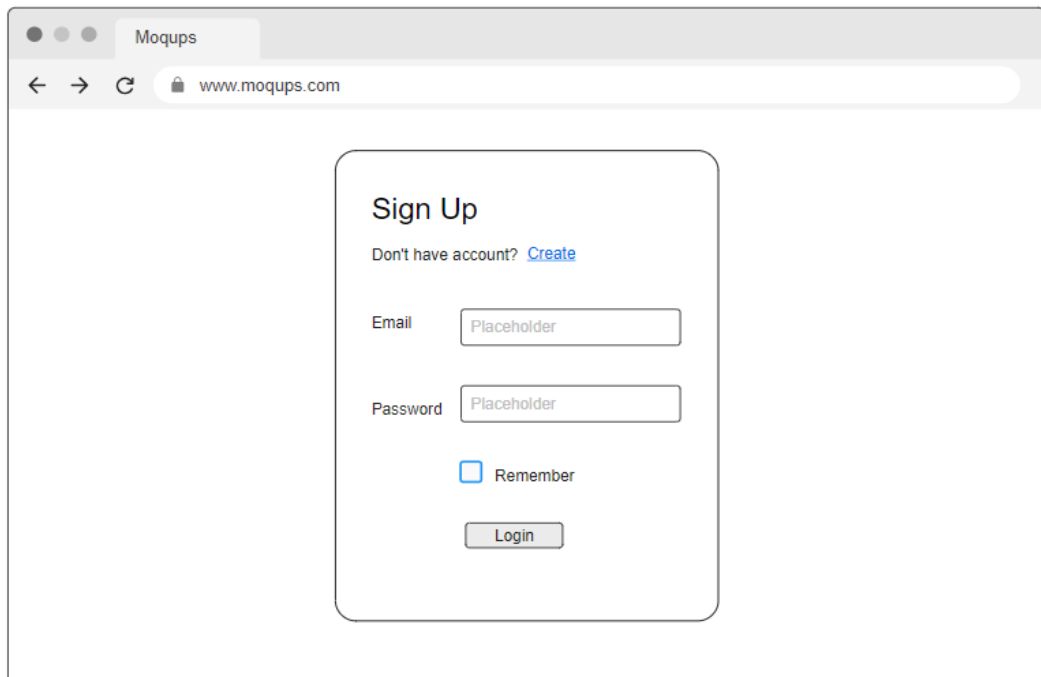
2.4.1. Biểu đồ dịch chuyển màn của hệ thống

Sau khi đăng nhập vào hệ thống, người dùng sẽ được dịch chuyển đến màn hình Dashboard, sau đó người dùng có thể chọn các chức năng bên thanh sidebar và chuyển đến màn hình mong muốn ví dụ course. Hình dưới là biểu đồ dịch chuyển màn hình trong hệ thống lớp học trực tuyến, mỗi một hình chữ nhật tương ứng với một màn hình, các hướng mũi tên sẽ giải thích sự dịch chuyển giữa các màn hình, mũi tên nét liền có đuôi là một hình thoi biểu thị màn hình dịch chuyển tới là một màn hình con của màn hình hiện tại.



Hình 2.19. Biểu đồ dịch chuyển màn của hệ thống

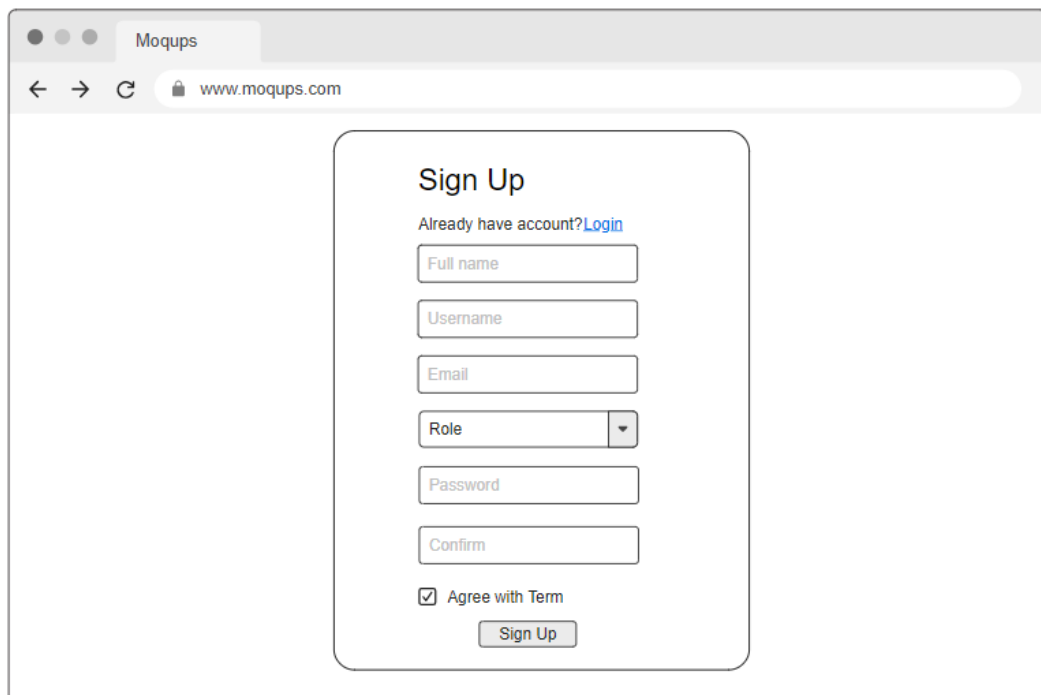
2.4.2. Giao diện chức năng đăng nhập



The image shows a web browser window with the address bar displaying 'www.moqups.com'. The main content area features a 'Sign Up' form. At the top of the form is the title 'Sign Up'. Below it is a link 'Don't have account? Create'. The form contains two input fields: 'Email' and 'Password', both with 'Placeholder' text. Below these fields is a checkbox labeled 'Remember' and a 'Login' button.

Hình 2.20. Thiết kế giao diện đăng nhập

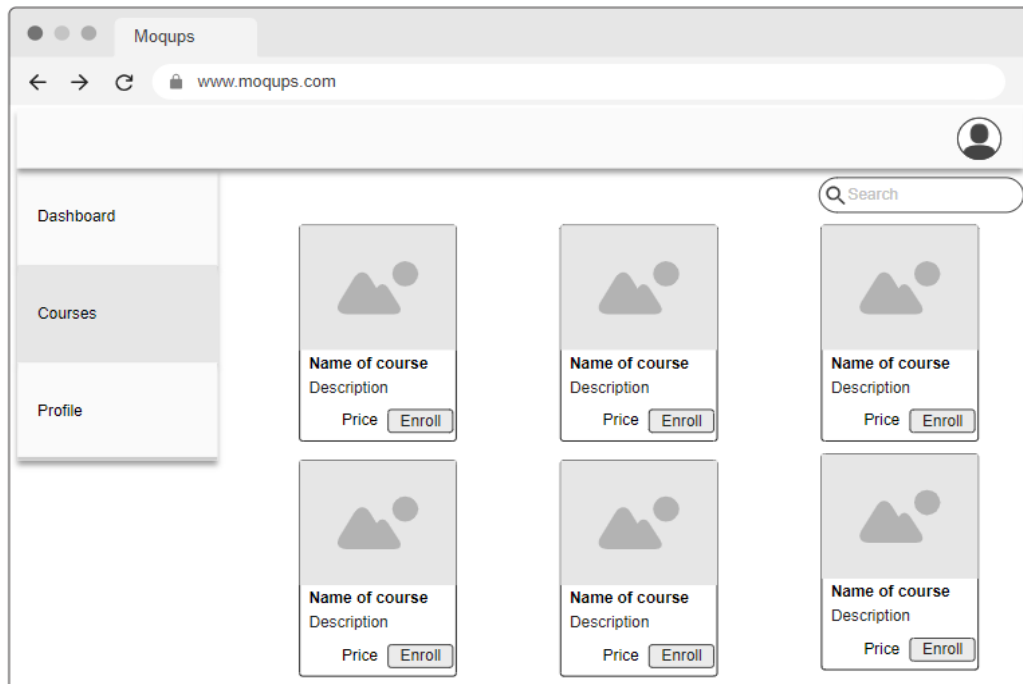
2.4.3. Giao diện màn hình đăng ký



The image shows a web browser window with the address bar displaying 'www.moqups.com'. The main content area features a 'Sign Up' form. At the top of the form is the title 'Sign Up'. Below it is a link 'Already have account? Login'. The form contains several input fields: 'Full name', 'Username', 'Email', 'Role' (a dropdown menu), 'Password', and 'Confirm'. Below these fields is a checkbox labeled 'Agree with Term' and a 'Sign Up' button.

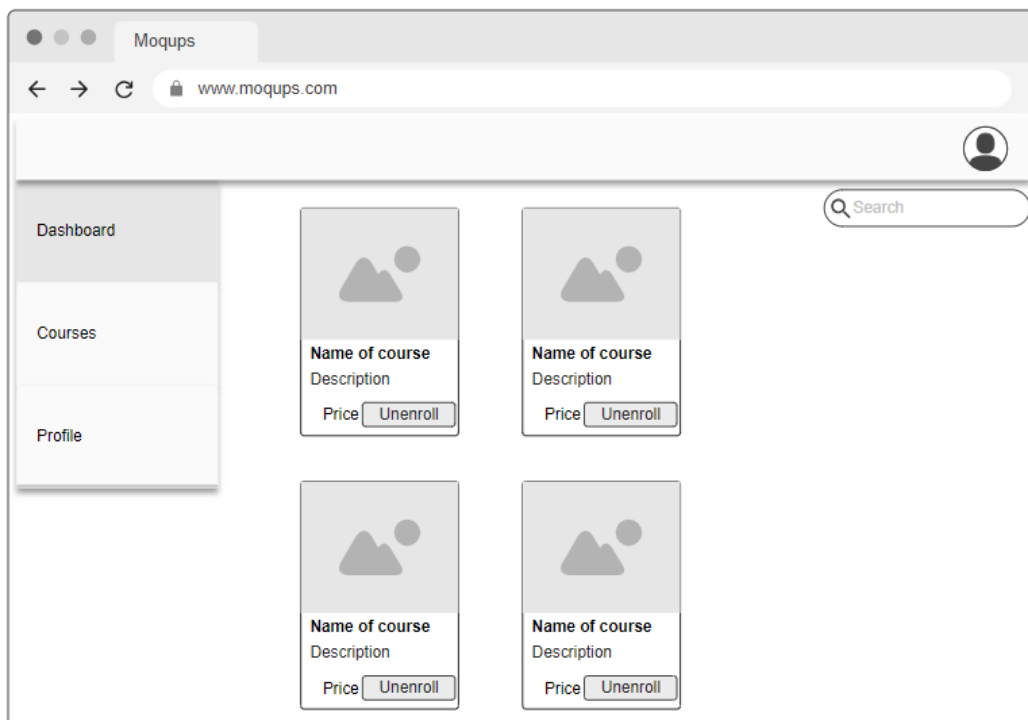
Hình 2.21. Thiết kế giao diện đăng ký

2.4.4. Giao diện thông tin khóa học



Hình 2.22. Thiết kế giao diện thông tin khóa học

2.4.3 Giao diện quản lý khóa học



Hình 2.23: Thiết kế giao diện quản lý khóa học

2.4.5. Giao diện quản lý thông tin cá nhân

The screenshot shows a web browser window with the URL www.moqups.com. The page has a sidebar on the left with links for "Dashboard", "Courses", and "Profile". The "Profile" link is selected. The main content area displays a user profile form. On the left, there is a circular profile picture placeholder labeled "Full name" and a "Role" label. On the right, there are input fields for "Name", "Username", "Email", and "Mobile". Below these is a "Photo" section with a small image icon and a "Save changes" button. A search bar is located in the top right corner.

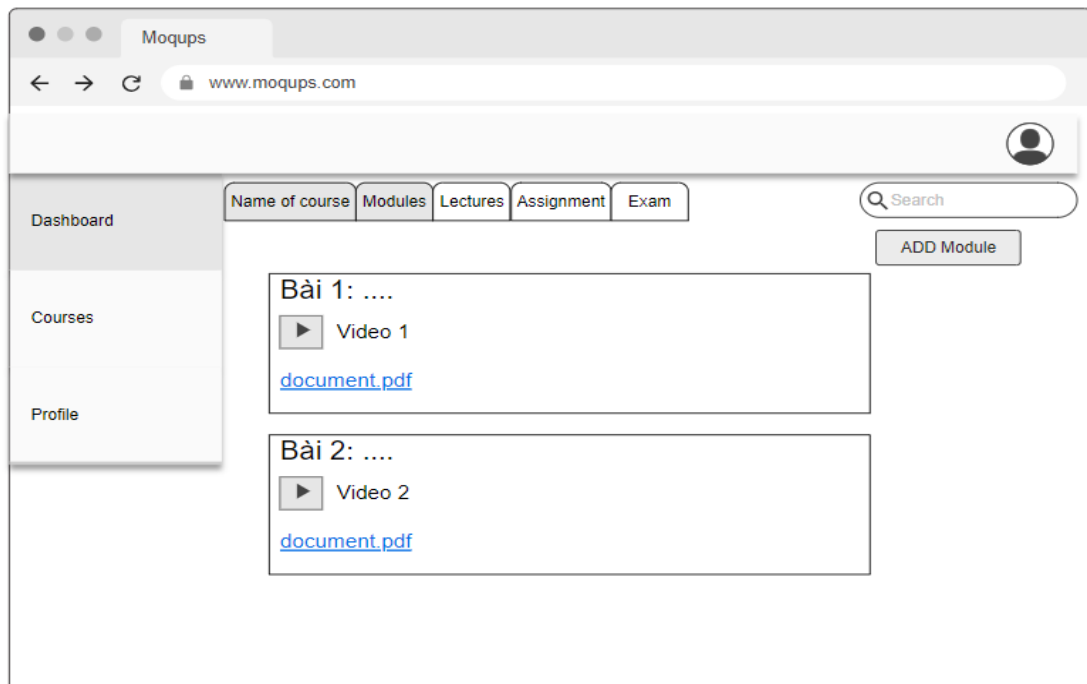
Hình 2.24. Thiết kế giao diện quản lý thông tin cá nhân

2.4.6. Giao diện tạo khóa học

The screenshot shows the Moqups website with the "Courses" link selected in the sidebar. A modal form is open in the center of the screen for creating a new course. The modal contains input fields for "Name", "Description", "Image" (with a label "Url of image"), and "Price". At the bottom of the modal are "Cancel" and "Submit" buttons. In the background, a list of existing courses is visible, each with a "Name course", "Description", "Price", and an "Unenroll" button. An "ADD Course" button is also visible in the top right corner of the main content area.

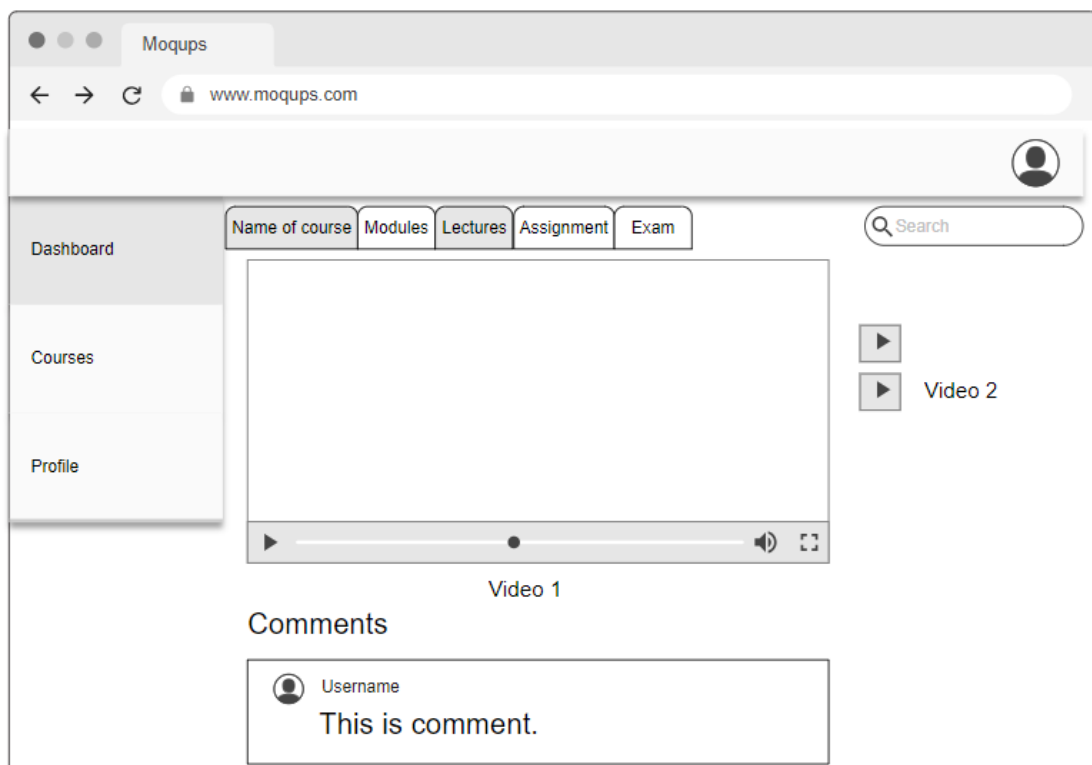
Hình 2.25. Thiết kế giao diện tạo khóa học

2.4.7. Giao diện của khóa học



Hình 2.26. Thiết kế giao diện khóa học

2.4.8. Giao diện khi tham gia bài giảng



Hình 2.27. Thiết kế giao diện tham gia bài giảng

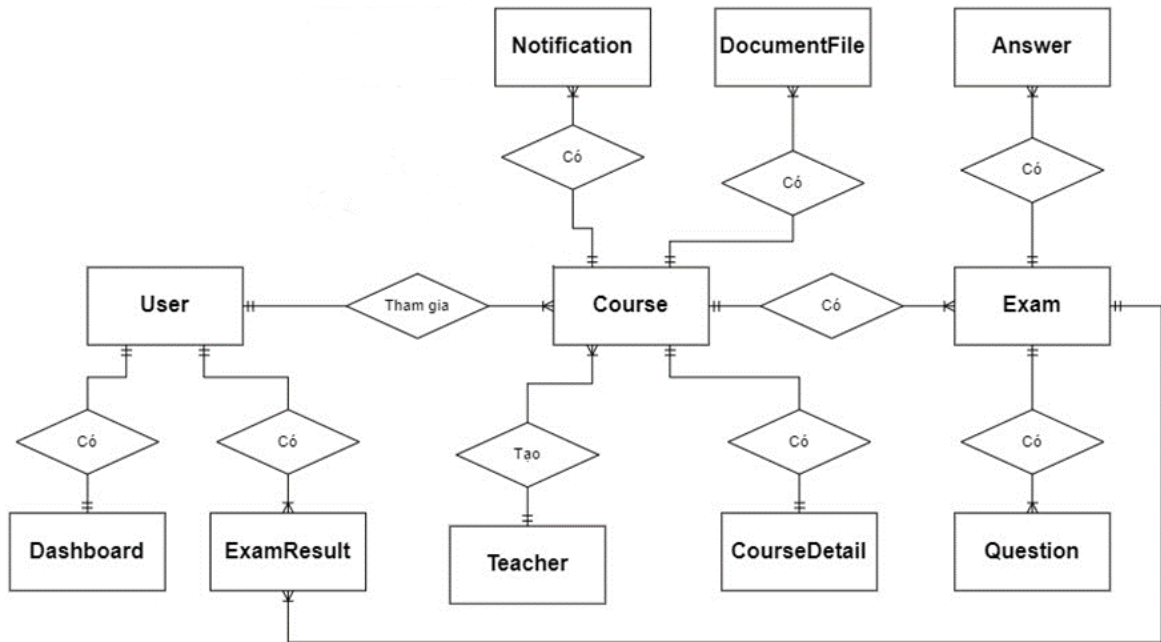
2.4.9. Giao diện tạo bài tập

The screenshot shows a web browser window with the address bar displaying "www.moqups.com". The page title is "Moqups". The interface features a sidebar on the left with navigation links: "Dashboard", "Courses", and "Profile". The main content area has a top navigation bar with tabs: "Name of course", "Modules", "Lectures", "Assignment", and "Exam". A search bar is located in the top right corner. The "Assignment" tab is selected, and the page title is "Assignment 1". The main content area contains a form for creating a question. The form has a title "Question 1" and a "Score" input field. Below the title are two large text input fields labeled "Question" and "Answer". At the bottom of the form are "Add" and "Delete" buttons. To the right of the form is a sidebar with additional settings: "Type" (dropdown), "Answer type" (dropdown), "05/12/2024" (date picker), and "Weight" (input field).

Hình 2.28. Thiết kế giao diện tạo bài tập

2.5. Cơ sở dữ liệu

2.5.1. Sơ đồ thực thể liên kết



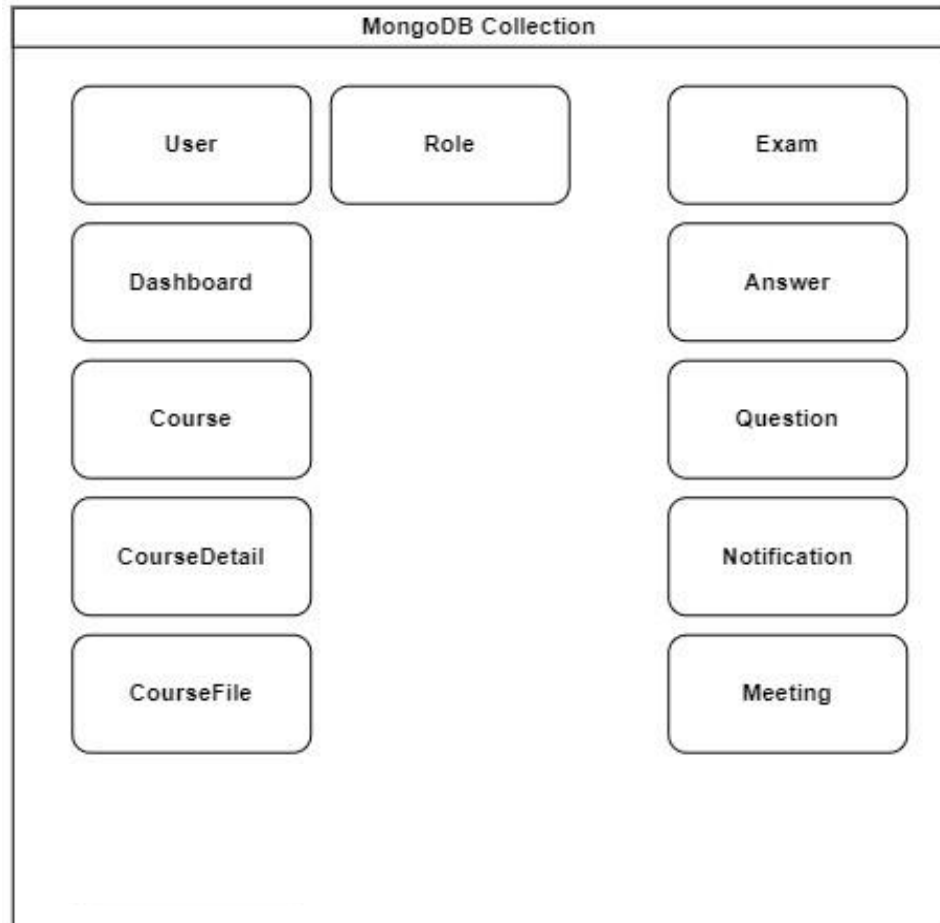
Hình 2.29. Sơ đồ thực thể

Bảng 2.8. Giải thích ý nghĩa thực thể

Tên trường	Ý nghĩa
Dashboard	Bảng tin
User	Người dùng
Teacher	Giáo viên
Course	Khoá học
DocumentFile	Tài liệu học tập
Question	Câu hỏi
Answer	Câu trả lời
Exam	Thi
ExamResult	Kết quả thi
Notification	Thông báo
Meeting	Cuộc họp

2.5.2. Tổng quan cơ sở dữ liệu

Hệ thống học trực tuyến sử dụng cơ sở dữ liệu phi quan hệ (NoSQL) với những ưu điều đã nêu ở phần trước, là CSDL lưu trữ dữ liệu đám mây và phục vụ cho các chức năng khác nhau.



Hình 2.30. Sơ đồ tổng quan cơ sở dữ liệu

Mô tả về các collections có trong CSDL:

- Users (Người dùng)
Lưu trữ thông tin về người dùng
- Course (Khoá học)
Lưu trữ thông tin về lớp học, thông tin lớp học được giáo viên nhập vào trong quá trình tạo lớp học
- Exams (Bài thi)
Lưu trữ các thông tin của bài thi: tên bài thi, câu hỏi, tổng điểm.
- Questions (Câu hỏi)

Lưu trữ các câu hỏi thuộc bài kiểm tra bao gồm, nội dung câu hỏi, ID các đáp án.

- Answers (Câu hỏi)

Lưu trữ các câu trả lời thuộc câu hỏi

- Notification (Thông báo lớp học)

Lưu trữ thông tin báo lớp học do giáo viên tạo

- ExamResult (Kết quả thi)

Lưu trữ thông tin về kết quả bài kiểm tra của người dùng

- CourseFile (Tài liệu khoá học)

Lưu trữ thông tin về tài liệu trong khoá học.

Bảng 2.9. Mô tả chi tiết Users (Người sử dụng)

Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectID	ID định danh của User
name	String	Tên hiển thị của User
username	String	Username dùng để đăng nhập
email	String	Email đăng kí của người dùng
password	String	Mật khẩu dùng để đăng nhập được mã hóa
roles	String	Các quyền của tài khoản

Bảng 2.10. Mô tả chi tiết Courses (Khóa học)

Tên trường	Kiểu dữ liệu	Mô tả
id	ObjectID	ID của khoá học
name	String	Tên khóa học
description	String	Mô tả về khóa học
files	Array	Các file bài tập
status	String	Trạng thái khóa học
enrollments	Array	Id Sinh Viên tham gia
Price	Number	Giá khóa học

Bảng 2.11. Mô tả chi tiết Notification (Thông báo lớp học)

Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectID	ID định danh
content	String	Nội dung thông báo lưu dạng text
course_id	ObjectID	ID của khoá học
author_id	ObjectID	ID người tạo
author_name	String	Tên của người tạo thông báo

Bảng 2.12. Mô tả chi tiết Exams (Bài kiểm tra)

Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectID	ID định danh
title	String	Tên bài kiểm tra
description	String	Mô tả bài kiểm tra
course_id	String	ID khoá học
author_id	ObjectID	ID của người tạo bài kiểm tra
author_name	String	Tên người tạo bài kiểm tra
questions	Array ObjectID	Chứa ID của các câu hỏi thuộc bài kiểm tra

Bảng 2.13. Mô tả chi tiết Questions (Câu hỏi)

Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectID	ID định danh
question	String	Nội dung câu hỏi
exam_id	ObjectID	ID bài kiểm tra
exam_course_id	ObjectID	ID khóa học
answers	Array ObjectID	Chứa ID của các câu trả lời thuộc bài kiểm tra
author_name	String	Tên người tạo
point	Number	Số điểm của câu hỏi

Bảng 2.14. Mô tả chi tiết Answer (Câu trả lời)

Tên trường	Kiểu dữ liệu	Mô tả
_id	ObjectID	ID định danh
answer	String	Nội dung câu trả lời
question_id	ObjectID	Chứa ID của câu hỏi

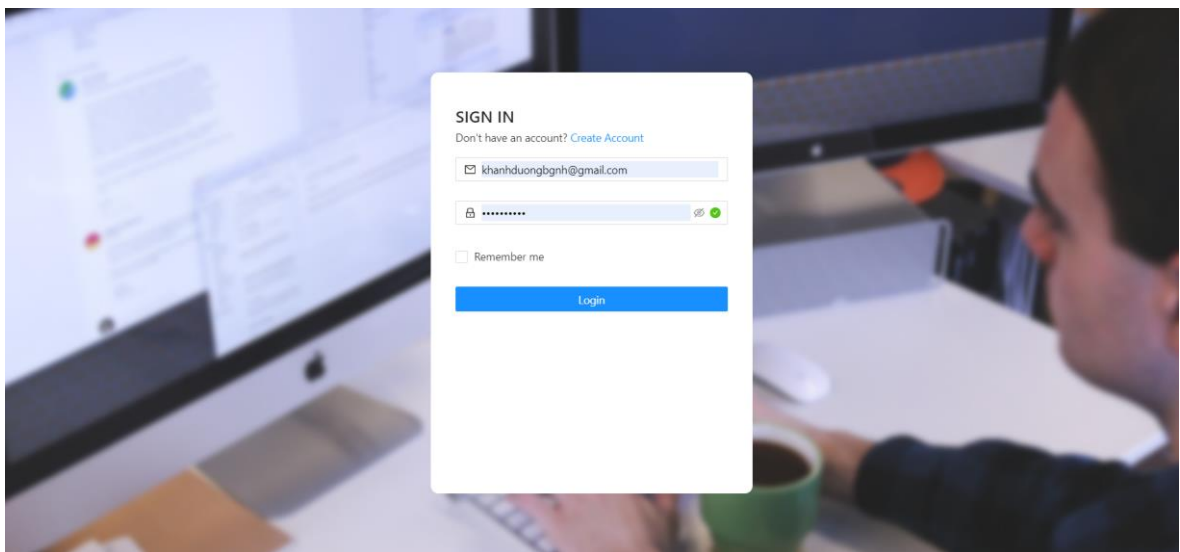
CHƯƠNG 3: KẾT QUẢ

3.1. Về chức năng

Đã đạt được có nhiều chức năng quan trọng nhằm cung cấp trải nghiệm tốt nhất cho người dùng. Trang chủ được thiết kế hấp dẫn với thông tin tổng quan về các khóa học nổi bật và các chương trình mới. Trang danh mục khóa học giúp người dùng dễ dàng tìm kiếm và lựa chọn khóa học phù hợp theo chủ đề, mức độ khó khăn hoặc giáo viên. Mỗi khóa học được trình bày chi tiết trên trang riêng, bao gồm mô tả, giáo viên, giá cả và đánh giá từ người dùng. Hệ thống tìm kiếm và bộ lọc linh hoạt giúp người dùng thuận tiện trong việc tìm kiếm khóa học theo nhu cầu cụ thể. Để đăng ký và thanh toán, người dùng có thể sử dụng giao diện đơn giản và an toàn. Hệ thống quản lý người dùng cho phép họ quản lý thông tin cá nhân, theo dõi tiến độ học tập và lịch sử thanh toán. Nền tảng học trực tuyến cung cấp môi trường học chất lượng với video giảng dạy, tài liệu học và bài kiểm tra. Cuối cùng, tính năng tương tác và hỗ trợ như trò chuyện trực tuyến và diễn đàn hỗ trợ giúp người dùng giải đáp thắc mắc và nhận sự hỗ trợ kịp thời.

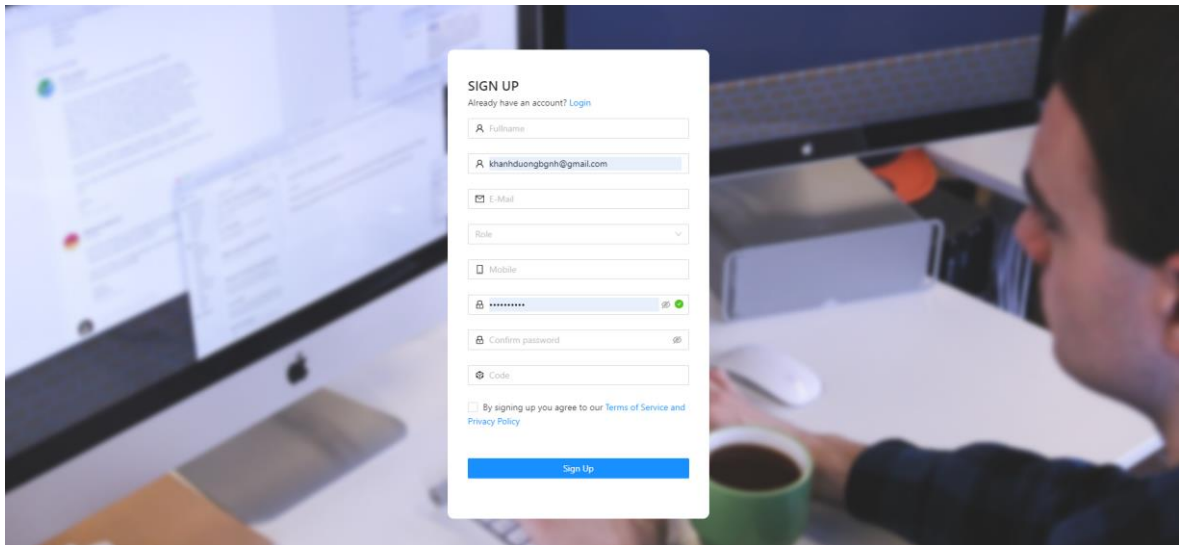
3.2. Về giao diện

3.2.1. Giao diện chức năng đăng nhập



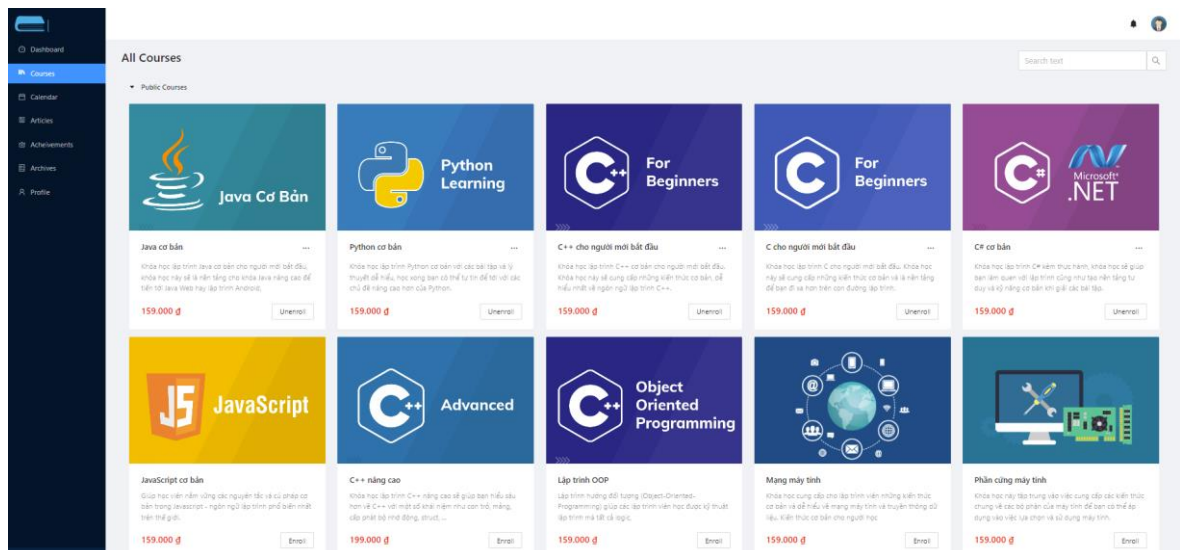
Hình 3.1. Giao diện đăng nhập

3.2.2. Giao diện màn hình đăng ký



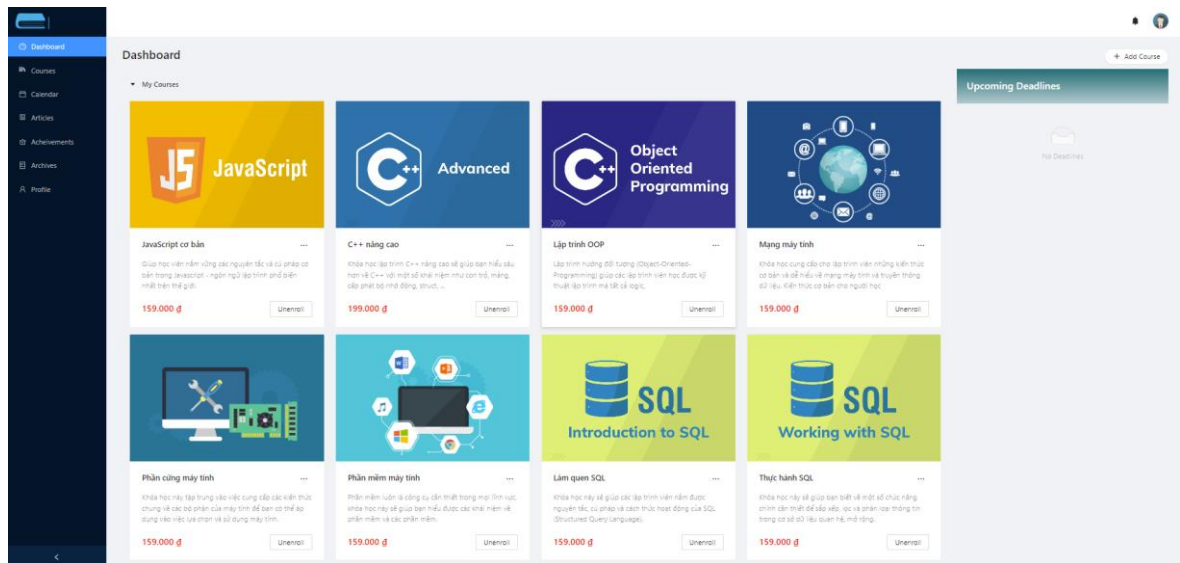
Hình 3.2. Giao diện màn hình đăng ký

3.2.3. Giao diện thông tin khóa học



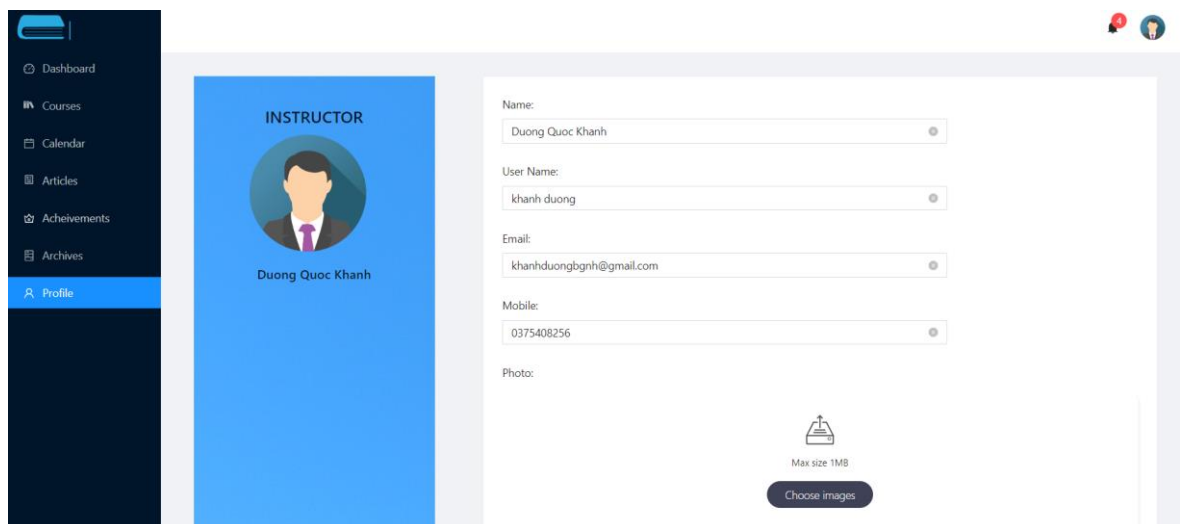
Hình 3.3. Giao diện thông tin khóa học

3.2.4. Giao diện quản lý khóa học



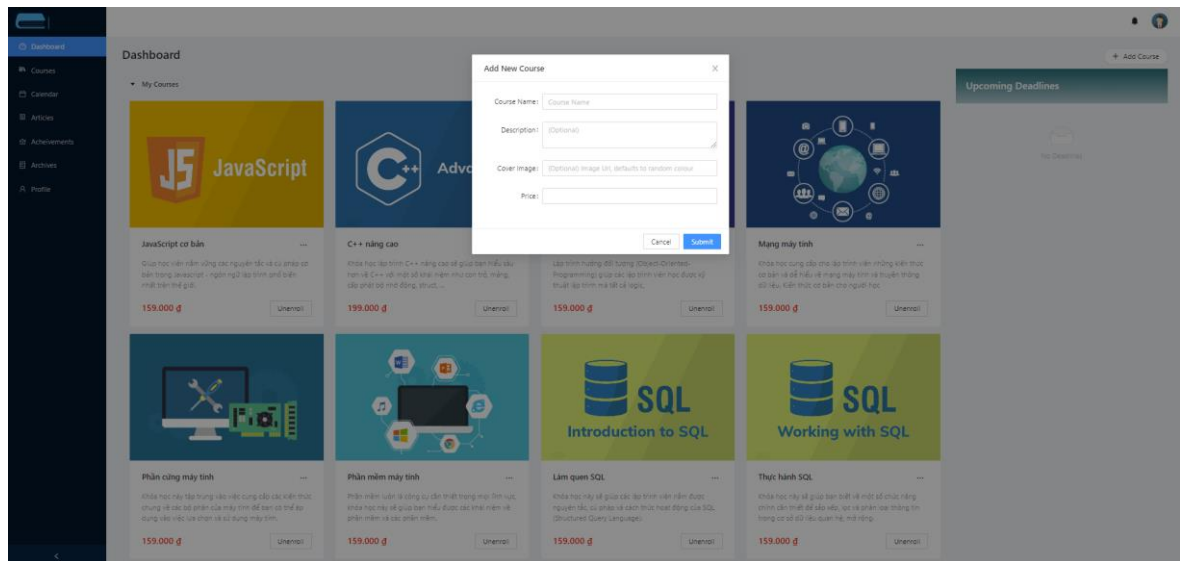
Hình 3.4: Giao diện quản lý khóa học

3.2.5. Giao diện quản lý thông tin cá nhân



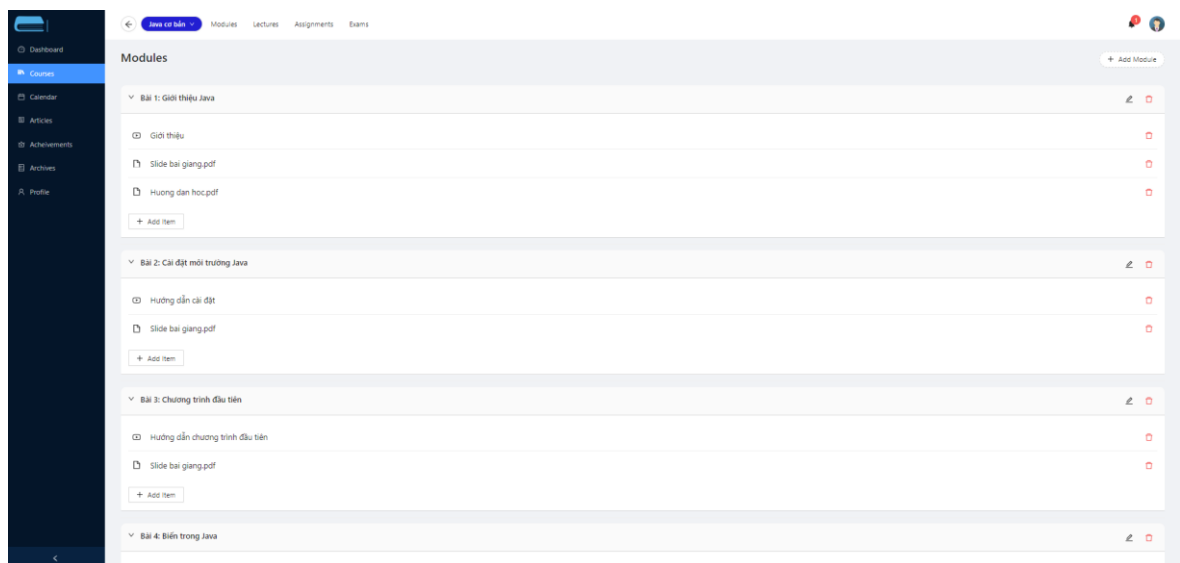
Hình 3.5: Giao diện quản lý thông tin cá nhân

3.2.6. Giao diện tạo khóa học



Hình 3.6. Giao diện tạo khóa học

3.2.7. Giao diện của khóa học



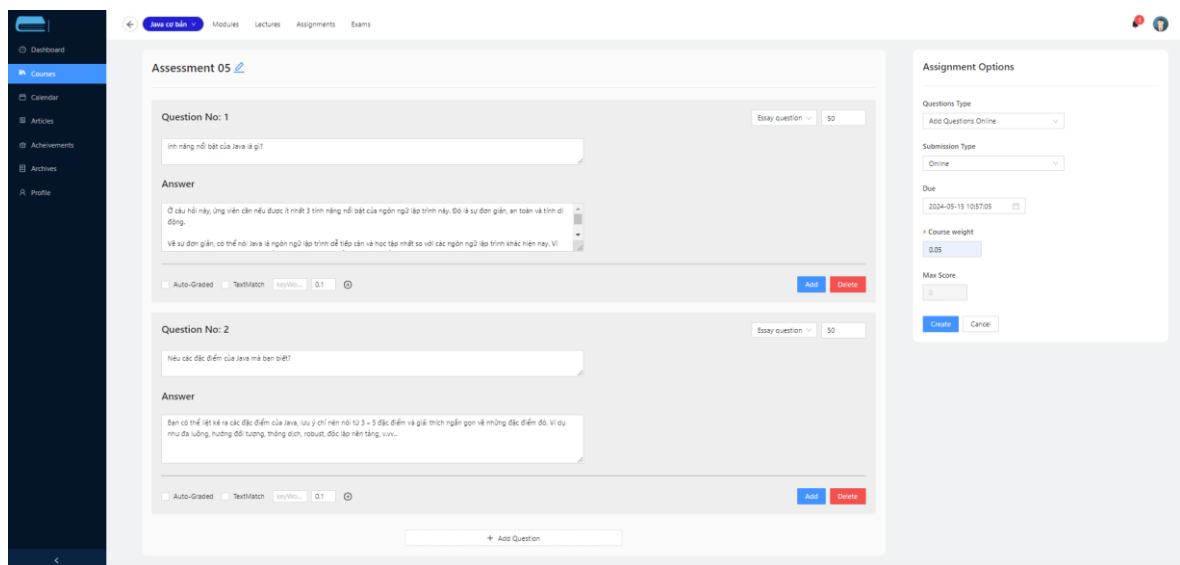
Hình 3.7. Giao diện của khóa học

3.2.8. Giao diện khi tham gia bài giảng



Hình 3.8. Giao diện tham gia bài giảng

3.2.9. Giao diện tạo bài tập



Hình 3.9. Giao diện tạo bài tập

CHƯƠNG 4: KIỂM THỬ

4.1. Kế hoạch kiểm thử

Kế hoạch kiểm thử chi tiết hơn với các bước cụ thể và các nhiệm vụ:

❖ Xác định mục tiêu kiểm thử:

- Chức năng: Đảm bảo tất cả các chức năng của website hoạt động chính xác và đáp ứng nhu cầu của người dùng.
- Giao diện người dùng: Đảm bảo giao diện người dùng thân thiện, dễ sử dụng và phản hồi đúng trên các thiết bị và trình duyệt khác nhau.
- Bảo mật: Bảo vệ dữ liệu người dùng và giao dịch, ngăn chặn các lỗ hổng bảo mật có thể bị tấn công.
- Hiệu suất: Đảm bảo website hoạt động mượt mà và đáp ứng dự kiến dưới tải lớn.

❖ Xác định phạm vi kiểm thử:

Kiểm thử chức năng:

- Đăng nhập và đăng ký: Kiểm tra tính năng đăng nhập và đăng ký tài khoản.
- Tìm kiếm khóa học: Kiểm tra tính năng tìm kiếm khóa học theo từ khóa.
- Quản lý thanh toán: Kiểm tra việc thanh toán khóa học.
- Quản lý tài khoản: Kiểm tra tính năng quản lý thông tin cá nhân.

Kiểm thử giao diện người dùng:

- Kiểm tra trên các trình duyệt phổ biến như Chrome, Edger....

Kiểm thử bảo mật:

- Đảm bảo kết nối được bảo mật và dữ liệu được mã hóa.

Kiểm thử hiệu suất:

- Kiểm tra thời gian tải trang: Đo thời gian tải trang chính và các trang con dưới các điều kiện mạng khác nhau.
- Kiểm tra khả năng chịu tải: Sử dụng công cụ để thử nghiệm khả năng chịu tải và đáp ứng của website dưới tải cao.

Lập kế hoạch kiểm thử:

- Xây dựng kịch bản kiểm thử: Xác định các bước cụ thể để thực hiện kiểm thử từng chức năng.
 - ❖ Thực hiện kiểm thử:
 - Thực hiện kịch bản kiểm thử: Thực hiện các kịch bản kiểm thử theo kế hoạch đã lập.
 - Ghi lại kết quả: Ghi lại kết quả kiểm thử, bao gồm cả lỗi và các vấn đề phát hiện được.
 - ❖ Phân tích và báo cáo kết quả:
 - Phân tích kết quả: Phân tích kết quả kiểm thử để xác định và ưu tiên các vấn đề cần sửa đổi.
 - Tạo báo cáo: Tạo báo cáo về kết quả kiểm thử và đề xuất các biện pháp cải thiện.
 - ❖ Kiểm tra và triển khai:
 - Kiểm tra lại và sửa lỗi: Kiểm tra lại sau khi sửa các lỗi phát hiện được.
 - Triển khai: Triển khai website sau khi kiểm thử và sửa lỗi hoàn tất.
- Kế hoạch được tạo dựng sẽ giúp đảm bảo rằng mọi khía cạnh của website bán khóa học được kiểm tra một cách cẩn thận và tổ chức trước khi được đi vào vận hành và sử dụng.

4.2. Thiết kế và thực thi kiểm thử

Bảng 4.1: Kiểm thử các chức năng của ứng dụng

STT	Nội dung	Mục đích kiểm thử	Đầu vào	Đầu ra mong muốn	Kết quả
1	Tài khoản	Đăng ký tài khoản	Full name: “Duong Quoc Khanh” Username: Khanh Email: khanhduongbgnh@gmail.com Role: “Student” Mobile: 0375408256 Password: Khanh2207# Confirm password: Khanh2207# Code: 111111 Đồng ý với các điều khoản: Yes	Tạo tài khoản thành công và quay lại màn hình Login	Pass
			Để trống một hoặc nhiều trường dữ liệu	Đưa ra thông báo yêu cầu nhập đầy đủ thông tin	Pass
			Nhập sai định dạng Email: Email: abc@cd@gmail.comm	Đưa ra thông báo yêu cầu nhập đúng định dạng Email	Pass

			Password và Confirm password không nhất quán: Password: Khanh2207# Confirm password: Khanh2207@	Đưa ra cảnh báo yêu cầu hai trường mật khẩu cần giống nhau	Pass
		Đăng nhập tài khoản	Email: khanhduongbgnh@gmail.com Password: Khanh2207#	Đăng nhập thành công	Pass
			Email: khanhduongbgnh@gmail.com Password: 123456	Đăng nhập thất bại và đưa ra cảnh báo	Pass

			Thông tin rỗng	Thông báo yêu cầu điền đầy đủ thông tin	Pass
		Sửa đổi thông tin cá nhân	Tại Profile điền mới một hoặc nhiều trường dữ liệu: Name: Duong Van Khanh Username: Van Khanh Email: khanh123@gmail.com Mobile: 0123456789 Chọn Save changes	Thông báo cập nhật thông tin thành công	Pass
2	Khóa học	Hiển thị Danh sách khóa học	Kích chọn mục Course	Hiển thị toàn bộ khóa học của hệ thống với thông tin bao gồm: Tên khóa học, mô tả khái quát nội dung, giá cả.	Pass

		Đăng ký khóa học	Tại mục Courses kích nút Enroll trên mỗi khóa học và xác nhận đăng ký rồi tiến hành thanh toán.	Màn hình chuyển tới trang thanh toán thông qua cổng điện tử Momo. Thanh toán thành công sẽ cấp quyền truy cập cho học viên.	Pass
		Hủy khóa học	Kích nút Unenroll trên khóa học đã đăng ký và xác nhận.	Hệ thống sẽ xóa bỏ quyền truy cập của học viên	Pass
		Xem bài giảng	Tại Dashboard, chọn khóa học đã đăng ký, với khóa học truy cập mục Modules và kích chọn mục cần xem.	Học viên có thể xem tài liệu, video có trong Module.	Pass

			Tại Dashboard, chọn khóa học đã đăng ký, với khóa học truy cập mục Lectures và kích chọn video cần mở.	Hiển thị Danh sách video bài giảng, video sẽ được phát khi được kích chọn.	Pass
		Gửi phản hồi và yêu cầu hỗ trợ thông qua bình luận	Tại mỗi video bài giảng, người dùng nhập thông tin vào form bình luận như: “Em cần giải thích về mục A trong bài học.”	Thông điệp sẽ được hiển thị dưới video bài giảng.	Pass
				Thông báo cho người dùng khác khi có bình luận mới	Fail
		Làm bài tập, làm bài kiểm tra	Tại mục Assignments hoặc Exam, học viên tham gia làm bài. Với dạng trắc nghiệm: học viên thực hiện chọn đáp án trực tiếp. Với dạng tự luận: học viên có thể thực hiện gửi câu trả lời trực tiếp hoặc kéo thả file.	Câu trả lời được ghi lại.	Pass

				Thực hiện chấm điểm tự động với Assignment dạng trắc nghiệm	Fail
				Cảnh báo và không cho phép nộp bài khi hết giờ.	Fail
		Tạo khóa học	<p>Tại Dashboard với Role giáo viên kích chọn Add Course và nhập nội dung cho từng trường dữ liệu:</p> <p>Course Name: “Lập trình C/C++”</p> <p>Description: “Kiến thức cơ bản”</p> <p>Cover Image: (đường dẫn của một hình ảnh)</p> <p>Price: 599000</p> <p>Chọn Submit để thực hiện thêm khóa học</p>	Khóa học được tạo và hiển thị trên hệ thống.	Pass

			<p>Chọn Add Module và nhập dữ liệu: Module Name: “Bài 1: Giới thiệu tổng quan”</p>	<p>Module mới được tạo và hiển thị tại mục Modules.</p>	Pass
		<p>Thao tác với Modules của khóa học</p>	<p>Với Module chọn biểu tượng bút chì và nhập lại tên Module: Module Name: “Bài 1: Giới thiệu về C/C++”</p>	<p>Hệ thống cập nhật lại và hiển thị Module theo nội dung đã nhập</p>	Pass
			<p>Với Module chọn biểu tượng thùng rác và xác nhận.</p>	<p>Hệ thống sẽ xóa bỏ Module và cập nhật lại nội dung.</p>	Pass

		Tạo bài tập, bài kiểm tra	<p>Tại Assignment hoặc Exam chọn Add và nhập nội dung: Assesment title: “Ass01” Type: Essay question/ Choice question Question body: “Nội dung câu hỏi?” Add Choice: “A. Đáp án 1 B. Đáp án 2 ...” Tích chọn đúng cho đáp án của câu hỏi. Point: “1” Due: 10/05/2024 24:00:00” Course weight: “0.1” Chọn Create.</p>	Hệ thống lưu dữ liệu và cập nhật lại nội dung hiển thị.	Pass
--	--	---------------------------	--	---	------

4.3. Báo cáo kiểm thử

Tổng số ca kiểm thử: 23

Số ca đạt được đầu ra mong muốn: 20

Số ca chưa đạt được đầu ra mong muốn: 3

Hiện website vẫn còn gặp một số lỗi như:

- Hệ thống chưa thể đưa ra thông báo cho các thành viên trong lớp khi có bình luận được tạo mới trong phần trao đổi thông tin (Discussion)
- Với chức năng làm bài tập, bài kiểm tra dưới hình thức câu hỏi trắc nghiệm chưa thể thực hiện.
- Chưa kiểm soát được thời gian làm bài tập, bài kiểm tra thông qua deadline

KẾT LUẬN

❖ Đánh giá kết quả

- Phần hoàn thành

- + Về mặt công nghệ, em đã tìm hiểu và ứng dụng được NodeJS, ReactJS, MongoDB...
- + Phát triển, cài đặt và vận hành ổn định hệ thống.
- + Hoàn thiện bài báo cáo và phân tích thiết kế hệ thống.
- + Lập trình thành thạo với ngôn ngữ Javascript.

- Phần chưa hoàn thành

- + Cần hoàn thiện thêm giao diện và tối ưu hóa hệ thống để hạn chế ảnh hưởng đến quá trình trải nghiệm của người dùng.

❖ Hướng phát triển

Hệ thống hiện tại mới chỉ đáp ứng đủ các yêu cầu của một lớp học đơn giản. Còn nhiều chức năng còn thiếu ví dụ như: chat trong khoá học giữa các thành viên, có thêm bài học riêng biệt trong khoá học, cảnh báo học sinh chưa làm xong bài thi. Ngoài ra, chức năng meeting chưa thể cho phép quá nhiều người tham gia vào cùng 1 phòng họp. Vì vậy, cần phải hoàn thiện những chức năng thiếu sót và sử dụng dịch vụ media stream như Zoom để tăng trải nghiệm người dùng.

Kiến trúc hệ thống sẽ hướng tới Microservice, vì hệ thống hiện tại đang chỉ tách thành các service và kết nối với MongoDB. Vì thế trong tương lai cần phải tái cấu trúc code hệ thống theo hướng microservice.

Ngày nay, các ứng dụng về học trực tuyến không chỉ dừng lại ở nền tảng Web, mà còn có xu hướng phát triển phổ biến hơn ở nền tảng mobile, smartphone. Không thể phủ nhận rằng, đa số các gia đình ở thời điểm này đều trang bị trong nhà ít nhất là những chiếc smartphone, hay hơn nữa máy tính bảng. Vì vậy mà việc phát triển hệ thống theo cả hướng mobile là rất cần thiết, giúp cho người dùng thuận tiện hơn trong việc sử dụng cũng như hướng tiếp cận.

Em rất mong nhận được sự góp ý và giúp đỡ của thầy cô và nhà trường để em có thể phát triển, hoàn thiện ứng dụng tốt hơn trong thời gian tới.

TÀI LIỆU THAM KHẢO

- [1] NodeJS, Node.js. [Online]. Available: <https://nodejs.org> (visited on 07/13/2022).
- [2] reacjs, Reactjs. [Online]. Available: <https://reactjs.org> (visited on 07/13/2022).
- [3] MongoDB, Mongoddb. [Online]. Available: <https://mongodb.com> (vis- ited on 07/13/2022).
- [4] webRTC, Webrtc. [Online]. Available: <https://webrtc.org> (visited on 07/13/2022).
- [5] Redis, Redis. [Online]. Available: <https://redis.io> (visited on 07/13/2022).
- [6] socket, Socket. [Online]. Available: <https://socket.io> (visited on 07/13/2022).