

Trường Đại học Công nghệ - ĐHQGHN
Khoa Công nghệ thông tin

BÀI TẬP LỚN: PHÂN TÍCH & THIẾT KẾ HƯỚNG ĐỐI TƯỢNG

Giảng viên: PGS. TS Đặng Đức Hạnh

ThS Trần Mạnh Cường



ARCHITECTURAL ANALYSIS
ỨNG DỤNG HỌC TẬP TRỰC TUYẾN

Ngày: 08/04/2024

Chuẩn bị bởi: Nhóm 10: Nguyễn Quang Anh, Nguyễn Xuân Hòa, Hoàng Việt Hưng, Trần Đức Khải, Phạm Minh Vương.

Mục lục	
Lịch sử sửa đổi	4
1. Tổng quan	5
1.1. Giới thiệu	5
1.2. Đối tượng dự kiến và đề xuất cách đọc	5
1.3. Phạm vi dự án	7
1.4. Tài liệu tham khảo	7
2. Cơ chế phân tích	7
2.1. Cơ chế lưu trữ (Persistence)	7
2.2. Cơ chế giao tiếp (Communication)	8
2.3. Cơ chế bảo mật (Security)	8
2.4. Các cơ chế khác	8
3. Các trừu tượng chính	9
4. Mô tả kiến trúc	10
5. Mục tiêu và ràng buộc về kiến trúc	11
6. Khung nhìn ca sử dụng	11
6.1. Tổng quan	11
6.2. Hiện thực hóa các ca sử dụng	13
6.3. Mô tả các ca sử dụng chính	14
7. Khung nhìn logic	16
7.1. Tổng quan	16
7.2. Các thiết kế kiến trúc của các gói chính	16
7.3. Gói trình diễn	18
7.4. Gói ứng dụng	19
7.5. Gói miền	19
7.6. Gói lưu trữ	20
8. Khung nhìn tiến trình	21
9. Khung nhìn triển khai	22
10. Khung nhìn thực thi	23

10.1. Tổng quan	23
11. Kích thước và hiệu suất	23
11.1. Sức chứa	23
11.2. Độ trễ & thời gian phản hồi	24
12. Chất lượng	24
Phụ lục: Thuật ngữ	25

Lịch sử sửa đổi

Họ tên	Thời gian	Lý do sửa đổi	Phiên bản
Nguyễn Quang Anh	08/04/2024	Khởi tạo mẫu tài liệu	1.0
Cả nhóm	10/04/2024	Viết yêu cầu phần 1	1.1
Cả nhóm	15/04/2024	Hoàn thiện version 1	1.2

1. Tổng quan

1.1. Giới thiệu

Đây là một báo cáo về chủ đề Phân tích và thiết kế hướng đối tượng của Nhóm 10 (sau gọi tắt là tác giả) về lựa chọn chủ đề giải quyết vấn đề.

Các yêu cầu về chức năng và phi chức năng của hệ thống đã được phân tích và tất cả các vấn đề và sự mơ hồ đã được giải quyết. Tài liệu này được xem xét bởi quản lý và tiếp thị dự án.

1.2. Đối tượng dự kiến và đề xuất cách đọc

Các đối tượng đọc khác nhau dành cho tài liệu này là:

- **Quản trị dự án:** Người phụ trách quản lý và chịu trách nhiệm về chất lượng hệ thống. Quản trị dự án nên đọc toàn bộ tài liệu để phục vụ việc lên kế hoạch và phân công công việc.
- **Nhà phát triển:** Người thực hiện nhiệm vụ phát triển hệ thống từ đầu vào là bản thiết kế và tài liệu để tạo thành đầu ra là một phiên bản có thể chạy được.
- **Người kiểm thử:** Người có nhiệm vụ đảm bảo rằng các yêu cầu là hợp lệ và phải xác nhận các yêu cầu. Tester nên đọc chi tiết để viết ca kiểm thử phù hợp.
- **Người viết tài liệu:** Người sẽ viết tài liệu trong tương lai (các báo cáo, biên bản).

Tài liệu cung cấp mô tả cấp cao về các mục tiêu của thiết kế kiến trúc, cùng với đặc tả ca sử dụng và các kiểu kiến trúc và các thành phần để triển khai được một thiết kế tốt nhất, phù hợp với yêu cầu. Nội dung thiết kế trong tài liệu tương đối chi tiết dành cho việc triển khai một cách dễ dàng.

Nội dung báo cáo bao gồm cơ chế phân tích, các khái niệm then chốt và các phần chính của kiến trúc:

- **Phần 1 Trình bày kiến trúc:** Mô tả kiến trúc phần mềm trong hệ thống hiện tại là gì và nó được biểu diễn ra sao? Trình bày những yếu tố cần thiết trong các nội dung của đặc tả.

- **Phần 2 Các mục tiêu và ràng buộc về kiến trúc:** Bao gồm các yêu cầu và mục tiêu có hướng tới kiến trúc như bảo mật, quyền riêng tư, tính linh hoạt, tái sử dụng,....
- **Phần 3 Khung nhìn ca sử dụng:** Liệt kê các ca sử dụng hoặc các tình huống từ mô hình ca sử dụng kể đến một số chức năng quan trọng của của hệ thống cuối hoặc nếu chúng có phạm vi kiến trúc lớn (có nhiều yếu tố kiến trúc, hoặc một đặc điểm kiến trúc đặc biệt).
- **Phần 4 Khung nhìn logic:** Mô tả các phần có ý nghĩa về mặt kiến trúc của thiết kế mô hình, chẳng hạn như phân tách các thành các hệ thống con và gói. Các lớp có ý nghĩa về mặt kiến trúc và mô tả những nhiệm vụ của chúng, cũng như các mối quan hệ, hoạt động với các lớp khác và các thuộc tính quan trọng.
- **Phần 5 Khung nhìn tiến trình:** Mô tả các phân tách các luồng thực hiện chương trình (tiến trình – process, luồng – thread, nhiệm vụ – task,...), đồng bộ giữa các luồng, phân bổ các đối tượng và lớp cho các luồng thực hiện khác nhau. Khung nhìn tiến trình tập trung vào các nhiệm vụ tương tranh tương tác với nhau như thế nào trong hệ thống đa nhiệm.
- **Phần 6 Khung nhìn triển khai:** Mô tả một hoặc nhiều mạng vật lý (phần cứng), cấu hình mà phần mềm được triển khai và chạy.
- **Phần 7 Khung nhìn thực thi:** Mô tả cấu trúc tổng thể của mô hình triển khai, phân tách phần mềm thành các lớp và các hệ thống con trong mô hình thực hiện, và bất kỳ thành phần có ý nghĩa kiến trúc.
- **Phần 8 Khung nhìn dữ liệu:** Mô tả về phối cảnh lưu trữ dữ liệu liên tục của hệ thống.
- **Phần 9 Kích thước và hiệu suất:** Mô tả về các đặc điểm kích thước chính của phần mềm tác động đến kiến trúc, cũng như các hạn chế hiệu suất mục tiêu.
- **Phần 10 Chất lượng:** Cách kiến trúc phần mềm đóng góp cho tất cả các khả năng (ngoài chức năng) của hệ thống: khả năng mở rộng, độ tin cậy, tính di động, v.v.

1.3. Phạm vi dự án

Ứng dụng học tập trực tuyến được xây dựng như một phương tiện hỗ trợ tìm kiếm, tạo các khóa học trực tuyến cho học sinh. Ứng dụng sẽ được phát triển dưới dạng một phần mềm Web và ứng dụng di động trên hệ điều hành Android và IOS. Người dùng cuối là những học sinh đang có nhu cầu tìm kiếm các khóa học online nhằm học tập thuận tiện mọi lúc mọi nơi. Người hướng dẫn có thể tạo khóa học, quản lý học sinh, giao bài tập, cung cấp học liệu. Học sinh có thể tìm kiếm và tham gia các khóa học phù hợp với định hướng của bản thân.

1.4. Tài liệu tham khảo

[1] IEEE Software Engineering Standards Committee, “IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications”, October 20, 1998.

[2] Slide môn học Phân tích và thiết kế hướng đối tượng do giảng viên cung cấp.

2. Cơ chế phân tích

2.1. Cơ chế lưu trữ (Persistence)

Đối với yêu cầu trên, chúng ta cần xác định những câu hỏi:

- **Độ chi tiết:** Phạm vi kích thước của các đối tượng là gì?
- **Số lượng:** Có bao nhiêu đối tượng cần phải phân tích?
- **Thời lượng:** Đối tượng thường cần được giữ trong bao lâu?
- **Cơ chế truy xuất:** Làm thế nào một đối tượng nhất định được xác định và truy xuất?
- **Tần suất cập nhật:** Các đối tượng có cố định hay không, có thường xuyên được cập nhật không?
- **Độ tin cậy:** Các đối tượng thành phần có gây ra lỗi cho hệ thống không?

2.2. Cơ chế giao tiếp (Communication)

Đối với tất cả các thành phần mô hình cần giao tiếp với các thành phần hoặc dịch vụ khác, chúng ta cần xác định:

- **Độ trễ:** thời gian chờ phản hồi trong việc tương tác các thành phần?
- **Tính đồng bộ:** sử dụng cơ chế giao tiếp không đồng bộ
- **Kích thước của thông điệp:** được gói trong JSON có đầy đủ thông tin về các yêu cầu/phản hồi.
- **Giao thức:** Điều khiển luồng, bộ đệm, v.v.

2.3. Cơ chế bảo mật (Security)

Với tất cả lớp, gói, các hệ thống con, cần đảm bảo các yếu tố về bảo mật sau đây:

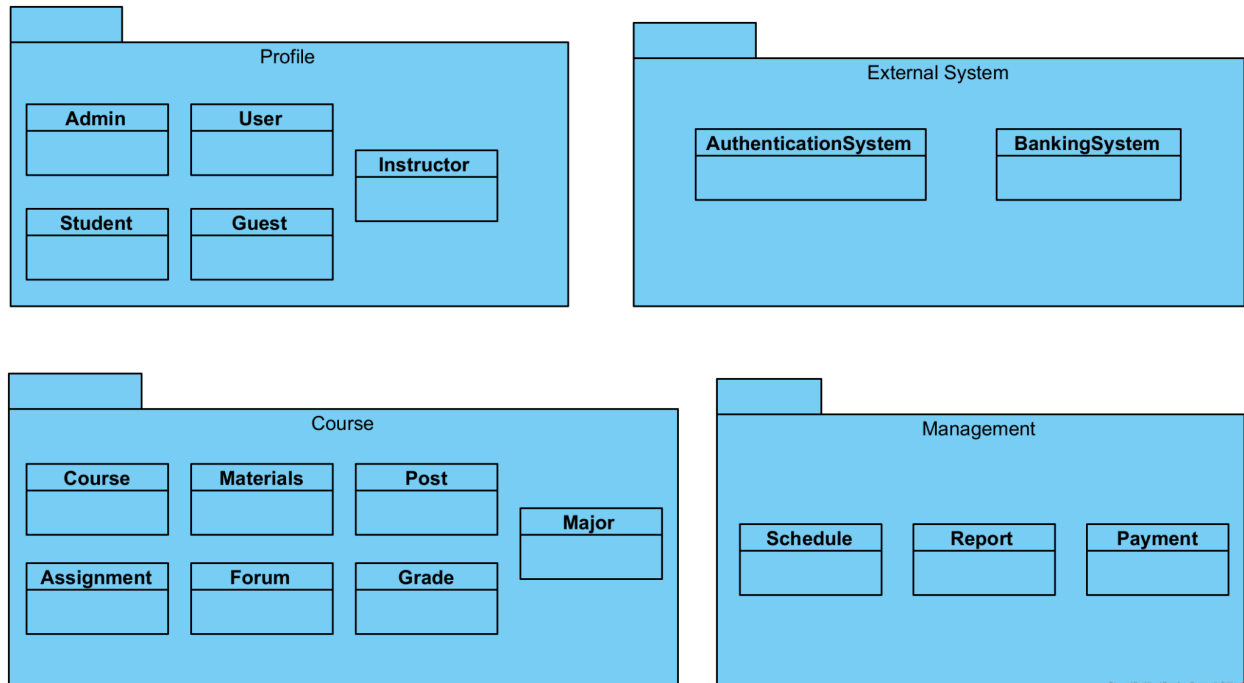
- **Độ chi tiết dữ liệu:** mức độ cụ thể được thể hiện bằng dữ liệu thực tế hoặc dữ liệu đa chiều ở trong kho dữ liệu.
- **Độ chi tiết về người dùng:** Có bao nhiêu vai trò của người dùng mà hệ thống có?
- **Luật:** Luật thiết lập các chuẩn chung để bảo vệ dữ liệu của người dùng.
- **Các quyền:** Các chức năng mà với vai trò của các người dùng có thể làm trong hệ thống?

2.4. Các cơ chế khác

Tất cả những cơ chế phân tích khác:

- **Sự thừa thãi:** là sự nhân đôi của các thành phần của một hệ thống với mục đích tăng độ tin cậy của hệ thống, thường là dưới dạng dự phòng hoặc đảm bảo tính ổn định trong trường hợp có lỗi.
- **Phát hiện/xử lý/báo cáo lỗi:** Các lỗi nên được phát hiện/xử lý/báo cáo như thế nào?
- **Quản lý tiến trình:** Tiến trình nên được hoàn thiện như thế nào?
- **Phân phối:** Dữ liệu nên được lưu ở máy chủ nào và được lưu trữ ra sao?

3. Các trừu tượng chính



- **User:** Người dùng có tài khoản, truy cập vào ứng dụng nói chung.
Cơ chế thiết kế: Privilege types, Persistence, Security.
- **Admin:** Quản trị viên của ứng dụng.
Cơ chế thiết kế: Privilege types, Persistence, Security.
- **Student:** Học sinh, tham gia học các khóa học trên ứng dụng.
Cơ chế thiết kế: Privilege types, Persistence, Security.
- **Instructor:** Người hướng dẫn khóa học trên hệ thống. Người hướng dẫn cũng có thể đóng vai trò là học sinh trong khóa học khác.
Cơ chế thiết kế: Privilege types, Persistence, Security.
- **Guest:** Khách truy cập ứng dụng, chưa có tài khoản.
Cơ chế thiết kế: Privilege types.
- **Course:** Khóa học với các tài liệu, bài tập, diễn đàn trao đổi.
Cơ chế thiết kế: Persistence, Security.
- **Materials:** Các tài liệu trong khóa học.

Cơ chế thiết kế: Persistence, Security.

- **Assignment:** Bài tập trong khóa học.

Cơ chế thiết kế: Persistence, Security.

- **Forum:** Diễn đàn trao đổi trong khóa học, nơi học sinh có thể trao đổi thông tin với học viên khác và với người hướng dẫn.

Cơ chế thiết kế: Persistence, Security.

- **Grade:** Điểm của bài tập.

Cơ chế thiết kế: Persistence, Security.

- **Major:** Chuyên ngành của khóa học.

Cơ chế thiết kế: Persistence, Security.

- **Schedule:** Lịch trình do học sinh sắp xếp.

Cơ chế thiết kế: Persistence, Security.

- **Payment:** Các giao dịch trong ứng dụng.

Cơ chế thiết kế: Persistence, Security.

- **Report:** Báo cáo trong ứng dụng.

Cơ chế thiết kế: Persistence, Security.

- **AutheticationSystem:** Hệ thống xác thực người dùng đăng nhập vào ứng dụng.

Cơ chế thiết kế: Legacy Interface.

- **BankingSystem:** Hệ thống hỗ trợ giao dịch, thanh toán.

Cơ chế thiết kế: Legacy Interface.

4. Mô tả kiến trúc

Tài liệu này trình bày kiến trúc dưới dạng khung nhìn ca sử dụng, khung nhìn thực thi, khung nhìn tiến trình, khung nhìn triển khai, khung nhìn logic. Các khung nhìn này được trình bày dưới dạng mô hình Rose Rational và sử dụng ngôn ngữ mô hình UML.

5. Mục tiêu và ràng buộc về kiến trúc

Có một số yêu cầu chính và các ràng buộc hệ thống trong thiết kế kiến trúc của hệ thống, bao gồm:

- Tất cả các chức năng phải phù hợp với các trình duyệt người dùng thường sử dụng.
- Các thông tin về tài khoản tín dụng của khách hàng phải được bảo mật trong hợp gửi thông tin về máy chủ hoặc cơ sở dữ liệu.
- Hiệu suất và khả năng chịu tải được mô tả trong tài liệu đặc tả bổ sung phải được đánh giá với thiết kế kiến trúc.

6. Khung nhìn ca sử dụng

6.1. Tổng quan

Khung nhìn ca sử dụng được xem là một tài liệu quan trọng để mô tả các chức năng, hành vi và nhiệm vụ của hệ thống. Khung nhìn này thể hiện mọi yêu cầu của hệ thống, do vậy khung nhìn này luôn đứng trước mọi khung nhìn khác, được sử dụng để điều khiển, thúc đẩy và thẩm định các công việc của tất cả các giai đoạn trong quá trình phát triển hệ thống.

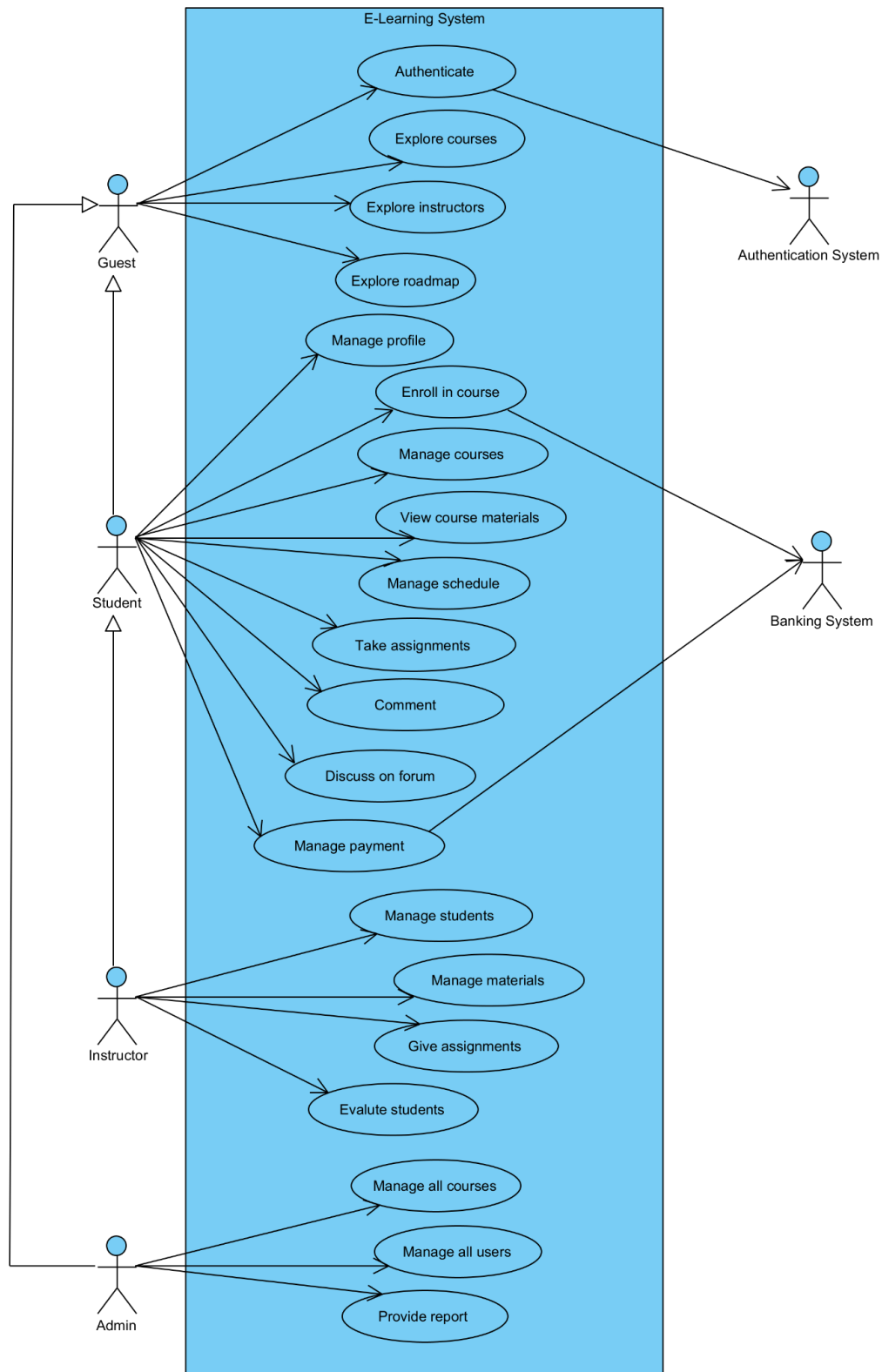
Dưới đây là các ca sử dụng có trong hệ thống, các ca sử dụng có ý nghĩa quan trọng sẽ được in đậm:

- **Đăng nhập (Authenticate)**
- Tìm hiểu thông tin khóa học (Explore courses)
- Tìm hiểu thông tin người hướng dẫn (Explore instructors)
- Tìm hiểu thông tin lộ trình học tập (Explore roadmaps)
- **Quản lý khóa học của bản thân (Manage courses)**
- **Đăng ký khóa học mới (Enroll in course)**
- Quản lý thanh toán (Manage payment)
- Xem tài liệu trong khóa học (View course materials)
- Lên lịch học (Manage schedule)
- Quản lý thông tin cá nhân (Manage profile)

- **Thực hiện bài tập (Take assignments)**
- Đánh giá người hướng dẫn (Comment)
- Quản lý học viên của khóa học (Manage students)
- Quản lý học liệu của khóa học (Manage materials)
- Giao bài tập cho học viên (Give assignments)
- **Đánh giá học viên (Evaluate students)**
- Quản lý tất cả khóa học (Manage all courses)
- Quản lý tất cả tài khoản (Manage all users)
- Cung cấp báo cáo (Provide report)
- **Trao đổi thông tin trên diễn đàn (Discuss on forum)**

6.2. Hiện thực hóa các ca sử dụng

Biểu đồ dưới đây biểu diễn các ca sử dụng trong hệ thống



6.3. Mô tả các ca sử dụng chính

Đăng nhập (Authenticate)

Ca sử dụng này mô tả cách người dùng đăng nhập vào *Ứng dụng học tập trực tuyến*. Người dùng có thể đăng nhập vào ứng dụng thông qua địa chỉ email hoặc tài khoản Google, Facebook, Github.

Tìm hiểu thông tin khóa học (Explore courses)

Ca sử dụng này mô tả người dùng xem thông tin chung của khóa học.

Tìm hiểu thông tin người hướng dẫn (Explore instructors)

Ca sử dụng này mô tả người dùng xem thông tin của người hướng dẫn.

Tìm hiểu thông tin lộ trình học (Explore roadmaps)

Ca sử dụng này mô tả người dùng xem thông tin chung của lộ trình học tập.

Quản lý thông tin cá nhân (Manage profile)

Ca sử dụng này mô tả cách người hướng dẫn và học sinh quản lý (xem, cập nhật) những thông tin cá nhân của mình ở trên hệ thống *Học tập trực tuyến* (tên, tuổi, thông tin ngân hàng ...).

Quản lý khóa học của bản thân (Manage courses)

Ca sử dụng này mô tả cách người dùng quản lý các khóa học mà mình đã và đang theo học (xem tiến độ các khóa học, xem thành tích học tập) trên hệ thống học tập trực tuyến.

Đăng ký khóa học mới (Enroll in course)

Ca sử dụng này mô tả cách người dùng đăng ký một khóa học mới trên hệ thống học tập trực tuyến.

Quản lý thanh toán (Manage payment)

Ca sử dụng này mô tả việc người dùng quản lý các giao dịch trên hệ thống.

Xem tài liệu trong khóa học (View course materials)

Ca sử dụng này mô tả việc người dùng xem các tài liệu, video bài giảng có trong khóa học.

Lên lịch học (Manage schedule)

Ca sử dụng này mô tả việc người dùng muốn lên lịch học của mình trên hệ thống.

Thực hiện bài tập (Take assignments)

Ca sử dụng này mô tả cách sinh viên thực hiện bài tập cho một khóa học.

Đánh giá người hướng dẫn (Comment)

Ca sử dụng này mô tả cách học sinh đánh giá hiệu suất của giáo viên.

Quản lý học viên của khóa học (Manage students)

Ca sử dụng này mô tả cách người hướng dẫn (Instructor) có thể quản lý danh sách học viên của khóa học trong hệ thống *Học tập trực tuyến (E-learning)*, bao gồm chức năng xem danh sách, thêm học viên mới, xóa học viên khỏi khóa học và cập nhật thông tin của học viên.

Quản lý học liệu của khóa học (Manage materials)

Ca sử dụng này mô tả cách người hướng dẫn (Instructor) có thể quản lý các học liệu của khóa học trong hệ thống *Học tập trực tuyến (E-learning)*, bao gồm chức năng bao gồm chức năng tải lên tài liệu mới, sửa đổi tài liệu đã có, xóa tài liệu không cần thiết và sắp xếp thứ tự hiển thị của các tài liệu.

Giao bài tập cho học viên (Give assignments)

Ca sử dụng này mô tả cách người hướng dẫn (Instructor) có thể giao bài tập cho học viên trong khóa học trên hệ thống *Học tập trực tuyến* bao gồm chức năng tạo bài tập mới, thiết lập thời hạn nộp bài, đính kèm tài liệu liên quan, và gửi bài tập cho học viên.

Đánh giá học viên (Evaluate students)

Ca sử dụng này mô tả cách người hướng dẫn (Instructor) có thể đánh giá, chấm điểm, phản hồi về bài tập hoặc bài kiểm tra của học viên trong khóa học trên hệ thống *Học tập trực tuyến (E-learning)*.

Quản lý tất cả khóa học (Manage all courses)

Ca sử dụng này mô tả cách Admin có thể quản lý tất cả các khóa học trên *hệ thống học tập trực tuyến (E-learning)*, bao gồm chức năng xem danh sách khóa học, thêm khóa học mới, chỉnh sửa thông tin khóa học, xóa khóa học và gán người hướng dẫn (Instructor) cho các khóa học.

Quản lý tất cả tài khoản (Manage all users)

Ca sử dụng này mô tả cách Admin có thể quản lý tất cả tài khoản người dùng trên hệ thống *Học tập trực tuyến (E-learning)*, bao gồm chức năng xem danh sách tài khoản, thêm tài khoản mới, chỉnh sửa thông tin tài khoản, xóa tài khoản và gán vai trò (roles) cho các tài khoản.

Cung cấp báo cáo (Provide report)

Ca sử dụng này mô tả cách Admin có thể tạo và xem các báo cáo liên quan đến hoạt động của hệ thống *Học tập trực tuyến (E-learning)*, bao gồm số liệu thống kê về người dùng, khóa học, hoạt động của học viên, và các số liệu khác để giám sát và đánh giá hiệu quả của hệ thống.

Trao đổi thông tin trên diễn đàn (Discuss on forum)

Ca sử dụng này mô tả cách sinh viên có thể tham gia vào diễn đàn thảo luận về một khóa học cụ thể.

7. Khung nhìn logic

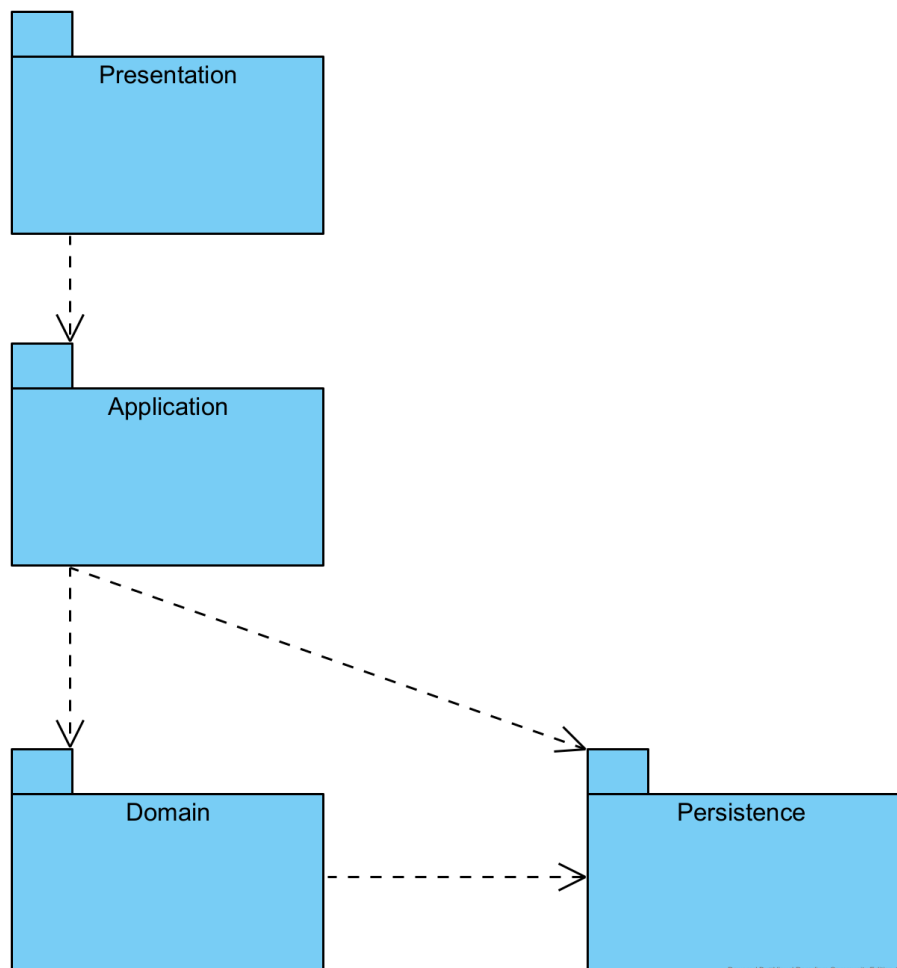
7.1. Tổng quan

Khung nhìn logic biểu diễn tổ chức của các lớp có ý nghĩa nhất và các quan hệ của chúng với nhau. Khung nhìn logic tập trung vào hệ thống cài đặt hành vi trong UC như thế nào. Nó bao gồm các lớp, biểu đồ lớp, biểu đồ đối tượng (khía cạnh tĩnh của khung nhìn), biểu đồ tương tác, biểu đồ biến đổi trạng thái (khía cạnh động của khung nhìn) và các gói. Hầu hết mọi người trong dự án đều quan tâm đến khung nhìn logic. Thông thường đội ngũ phát triển phần mềm tiếp cận khung nhìn logic theo hai bước. Bước thứ nhất là nhận ra các lớp phân tích (analysis class). Các lớp này độc lập với ngôn ngữ. Trong UML các lớp này được biểu diễn bằng các biểu tượng sau: Lớp phân tích có thể xuất hiện cả ở trong biểu đồ tương tác của khung nhìn UC. Một khi đã nhận ra các lớp phân tích thì đội ngũ phát triển phần mềm chuyển chúng sang lớp thiết kế (design class). Đó là những lớp phụ thuộc ngôn ngữ. Khung nhìn logic tập trung vào cấu trúc logic của hệ thống. Trong khung nhìn này ta sẽ nhận ra các bộ phận hệ thống, khảo sát thông tin và hành vi, khảo sát quan hệ giữa các bộ phận. Cần cẩn thận khi gán thông tin và hành vi cho lớp, nhóm các lớp, khảo sát quan hệ giữa các lớp và gói để đảm bảo khả năng sử dụng lại.

7.2. Các thiết kế kiến trúc của các gói chính

Khung nhìn logic của hệ thống bao gồm 4 gói:

- **Trình diễn (Presentation):** chứa các lớp cho mỗi biểu mẫu mà các tác nhân sử dụng để giao tiếp với hệ thống.
- **Ứng dụng (Application):** chứa các lớp xử lý chính cho hệ thống.
- **Miền (Domain):** chứa các gói hỗ trợ các thực thể chính của hệ thống.
- **Lưu trữ (Persistence):** chứa các lớp để lưu trữ các đối tượng cụ thể trong hệ thống. Tại điểm này trong thiết kế, chỉ thông tin cá nhân được lưu trữ, mặc dù các đối tượng nội dung có thể được lưu trữ ở một số điểm trong tương lai (việc chọn một hệ thống quản lý nội dung được đóng gói có thể loại bỏ nhu cầu cho việc này).



7.3. Gói trình diễn

Gói này chứa các lớp cho từng chức năng của người dùng thể hiện ở tầng trình diễn. Các lớp này bao gồm:

- LoginView
- LoginWith3rdPartyView
- SearchView
- CourseView
- CourseListView
- InstructorView
- InstructorListView
- MajorListView
- ProfileView
- PersonalCourseListView
- PaymentView
- BankingServicesView
- ScheduleView
- AssignmentView
- EvaluationView
- ManageCourseView
- ManageMaterialsView
- EvaluateStudentsView
- CourseManagementView
- UserManagementView
- ReportView
- ForumView

7.4. Gói ứng dụng

Gói này chứa các lớp cho chức năng xử lý chính trong hệ thống. Các lớp điều khiển tồn tại để hỗ trợ quản lý thông tin, tài liệu, xử lý các tính năng có trên hệ thống. Các lớp này bao gồm:

- ManageLogin
- ManageCourse
- ManageInstructor
- ManageMajor
- ManagePersonalCourse
- ManagePayment
- ManageSchedule
- ManageProfile
- AssignWork
- EvaluateInstructor
- ManageMaterials
- EvaluateStudents
- ManageAllUsers
- DirectCourse
- DirectForum
- DirectTopic

7.5. Gói miền

Profile Package: Bao gồm tất cả các lớp liên quan đến người dùng phổ thông, gồm có các lớp:

- Admin
- Student
- Instructor
- Guest

- User

Course Package: Bao gồm các lớp liên quan đến thông tin của các khóa học, gồm có các lớp:

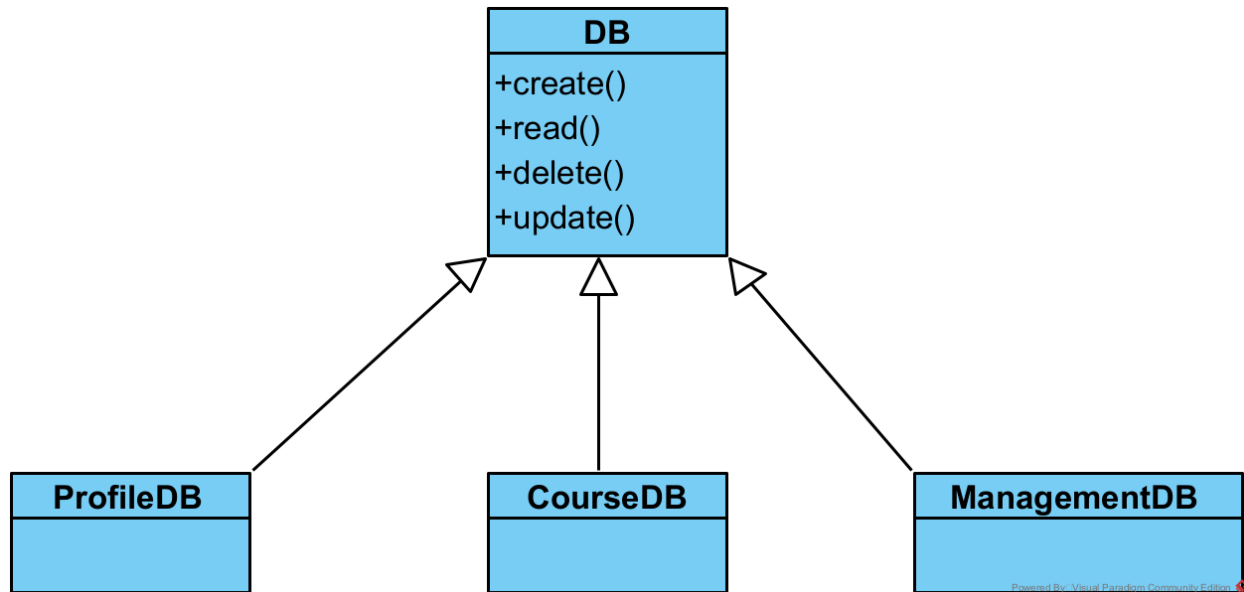
- Course
- Major
- Assignment
- Grade
- Materials
- Forum
- Post

Management Package: Bao gồm các lớp liên quan đến công tác quản lý trong ứng dụng, bao gồm các lớp:

- Payment
- Schedule
- Report

7.6. Gói lưu trữ

Persistence package diagram cho thấy sự tổng quát hóa giữa lớp trừu tượng DB và các lớp con ProfileDB, CourseDB và ManagementDB. Bốn hoạt động create, read, update và delete đề cập đến bốn chức năng chính được thực hiện trong cơ sở dữ liệu của hệ thống.



8. Khung nhìn tiến trình

Tiến trình Quản lý người dùng: Đây là tiến trình quản lý thông tin người dùng và các hoạt động của họ trên ứng dụng.

Tiến trình này bao gồm các bước sau:

- Nhận yêu cầu đăng ký, đăng nhập hoặc cập nhật thông tin người dùng từ giao diện người dùng.
- Xử lý yêu cầu và truy vấn cơ sở dữ liệu để lưu trữ hoặc cập nhật thông tin người dùng.
- Kiểm tra tính hợp lệ của thông tin người dùng và xử lý các lỗi khi cần thiết.

Tiến trình đăng ký khóa học

Tiến trình này bao gồm các bước sau:

- Người dùng chọn khóa học muốn đăng ký.
- Người sử dụng thanh toán thông qua các phương thức thanh toán qua ví điện tử.
- Hệ thống xử lý thanh toán, kiểm tra tính hợp lệ của thông tin thanh toán và cập nhật trạng thái thanh toán.
- Hệ thống gửi thông tin thanh toán thành công hoặc không thành công cho người dùng.

- Hệ thống gửi thông tin người dùng đăng ký khóa học thành công hay chưa.

Tiến trình đánh giá người hướng dẫn: Cho phép người dùng đánh giá người hướng dẫn mà họ đã học, đồng thời cung cấp thông tin đánh giá cho người dùng khác tham khảo.

Tiến trình này bao gồm các bước sau:

- Khởi động tiến trình: Người dùng truy cập vào ứng dụng học tập trực tuyến, tìm chọn người hướng dẫn đã hoặc đang học và truy cập chức năng đánh giá.
- Hiện thị giao diện đánh giá: Hệ thống hiện thị giao diện đánh giá cho người dùng nhập thông tin về đánh giá, bao gồm số sao (rate), nội dung đánh giá.
- Nhập thông tin đánh giá: Người dùng nhập thông tin đánh giá vào các trường tương ứng trên giao diện đánh giá.
- Lưu thông tin đánh giá: Hệ thống lưu thông tin đánh giá vào cơ sở dữ liệu của ứng dụng.
- Xử lý thông tin đánh giá: Hệ thống sử dụng các thuật toán đánh giá để tính toán điểm đánh giá trung bình cho người hướng dẫn dựa trên đánh giá của người dùng.
- Cập nhật thông tin đánh giá: Hệ thống cập nhật điểm đánh giá trung bình và số lượng đánh giá cho người hướng dẫn vào cơ sở dữ liệu của ứng dụng.
- Hiện thị thông tin đánh giá: Hệ thống hiện thị thông tin đánh giá của người hướng dẫn cho người dùng khác tham khảo trên trang giới thiệu về người hướng dẫn.
- Kết thúc quá trình: Tiến trình đánh giá người hướng dẫn kết thúc và quay trở lại giao diện người hướng dẫn vừa đánh giá.

9. Khung nhìn triển khai

Cấu trúc phần cứng của hệ thống sẽ bao gồm một máy chủ (máy chủ) để lưu trữ và xử lý dữ liệu, và các thiết bị người dùng như điện thoại thông minh (điện thoại thông minh), máy tính bảng (máy tính bảng) hoặc máy tính để bàn (máy tính để bàn) để cung cấp giao diện người dùng. Máy chủ sẽ được đặt tại một trung tâm dữ liệu (data center) để đảm bảo tính an toàn, tin cậy và mở rộng của hệ thống.

Phần mềm của hệ thống sẽ được phát triển khai thác trên nền tảng điện toán đám mây (điện toán đám mây) để đảm bảo tính linh hoạt và khả năng mở rộng. Các phần mềm là ứng dụng Web.

Phần mềm của hệ thống sẽ được phát triển bằng ngôn ngữ lập trình TypeScript và các Framework phổ biến như Next.js, ReactJs. Hệ thống sẽ sử dụng các công nghệ và kỹ thuật phần mềm như máy học (machine learning) để cung cấp chức năng gợi ý các món ăn và tìm kiếm theo từ khóa.

Để đảm bảo tính toàn vẹn và bảo mật của dữ liệu, hệ thống sẽ sử dụng các phương thức bảo mật như mã hóa SSL (Secure Sockets Layer) để mã hóa dữ liệu giao tiếp và JWT (JSON Web Token) để xác thực người dùng.

10. Khung nhìn thực thi

10.1. Tổng quan

Tất cả phần mềm máy chủ nằm trong một lớp duy nhất. Trình duyệt của khách hàng cung cấp lớp thứ cấp thông qua kết nối Internet.

- **Lớp máy chủ:** Trong các máy chủ nói chung được quản lý bởi máy chủ mạng, nhưng các chương trình ứng dụng có thể một phần đảm nhận trách nhiệm này.
- **Lớp máy khách:** Lớp máy khách là nơi người dùng truy cập ứng dụng. Lớp máy chủ chấp nhận các yêu cầu thông qua kết nối internet từ lớp máy khách và chuyển các yêu cầu này đến tác nhân thích hợp. Sau đó, máy chủ chuyển tiếp phản hồi từ tác nhân trở lại lớp máy khách. Trong trường hợp này, máy khách chỉ đơn giản là một trình duyệt.

11. Kích thước và hiệu suất

Phần mềm được thiết kế sẽ hỗ trợ 10.000 người dùng đồng thời. Có thể vượt qua con số này bằng cách sử dụng kỹ thuật cân bằng tải.

11.1. Sức chứa

Tiểu mục này chỉ định các yêu cầu sau liên quan đến số lượng đối tượng tối thiểu mà hệ thống có thể hỗ trợ:

Hệ thống sẽ hỗ trợ tối thiểu 2.000 yêu cầu mở đồng thời.

Hệ thống sẽ hỗ trợ tối thiểu 500.000 khách hàng.

Hệ thống sẽ hỗ trợ tối thiểu 20.000 tương tác đồng thời.

11.2. Độ trễ & thời gian phản hồi

Tiêu mục này chỉ định các yêu cầu sau liên quan đến thời gian tối đa được phép cho hệ thống để thực thi các tác vụ cụ thể hoặc ca sử dụng kết thúc:

Hệ thống phải có khả năng hoàn thành 90% giao dịch trong chưa đầy 30 giây.

Thời gian tải thông tin khóa học và giảng viên không quá 1 giây.

Hệ thống phải cung cấp khả năng truy cập tới cơ sở dữ liệu và phản hồi chậm không quá 5 giây.

12. Chất lượng

Kiến trúc phần mềm cần đóng góp cho các khả năng của hệ thống:

- **Độ chính xác:** Ứng dụng cần đưa ra thông tin chính xác về các khóa học, các tài liệu phải đảm bảo chất lượng. Hệ thống kiến trúc hệ thống cần được thiết kế để thu thập và phân tích thông tin về các khóa học phù hợp với người dùng một cách chính xác và hiệu quả.
- **Tính tương thích:** Ứng dụng cần tương thích với nhiều thiết bị và hệ điều hành khác nhau để đáp ứng nhu cầu của nhiều người dùng. Hệ thống kiến trúc hệ thống cần được thiết kế để tương thích với nhiều phiên bản hệ thống điều hành và thiết bị khác nhau.
- **Hiệu quả:** Ứng dụng cần đáp ứng nhu cầu của người dùng một cách nhanh chóng và hiệu quả. Hệ thống kiến trúc cần được thiết kế để tối ưu hóa tốc độ và độ ổn định của ứng dụng.
- **Bảo mật:** Ứng dụng cần bảo vệ dữ liệu của người dùng khỏi các mối đe dọa bảo mật bên ngoài. Hệ thống kiến trúc cần được thiết kế để đảm bảo tính bảo mật của hệ thống và dữ liệu.
- **Cách sử dụng dễ dàng:** Hệ thống cần sử dụng dễ dàng và thân thiện với người dùng. Kiến trúc cần được thiết kế để cho phép người dùng tìm kiếm và lựa chọn các khóa học dễ dàng và nhanh chóng.
- **Tính mở rộng:** Hệ thống cần có khả năng mở rộng để đáp ứng yêu cầu của người dùng trong tương lai. Kiến trúc cần được thiết kế để cho phép các tính năng mới được thêm vào hệ thống một cách dễ dàng và hiệu quả.

- Khả năng hợp nhất: Hệ thống cần có khả năng hợp nhất với các dịch vụ và ứng dụng khác để đáp ứng yêu cầu của người dùng. Kiến trúc cần được thiết kế để cho phép.

Phụ lục: Thuật ngữ

Khung nhìn logic: Liên quan đến chức năng mà hệ thống cung cấp cho người dùng cuối.

Gói ứng dụng: Tập hợp các chương trình hoặc mô-đun được hướng vào một số ứng dụng chung và có thể được tùy chỉnh theo nhu cầu của một phiên bản cụ thể của ứng dụng đó.

Khung nhìn tiến trình: Liên quan đến các khía cạnh động của hệ thống, giải thích các quy trình của hệ thống và cách chúng giao tiếp và tập trung vào hành vi thời gian chạy của hệ thống.

Khung nhìn triển khai: Mô tả hệ thống theo quan điểm của kỹ sư hệ thống. Nó liên quan đến cấu trúc liên kết của các thành phần phần mềm trên lớp vật lý, cũng như các kết nối vật lý giữa các thành phần này.

Khung nhìn thực thi: Tập trung vào việc lấy khung nhìn logic và phân chia các thực thể logic thành các thành phần phần mềm thực tế. Nó mô tả tổ chức các mô-đun phần mềm tĩnh trong môi trường phát triển.

Lớp: Một nhóm các lớp có cùng một tập các phụ thuộc mô đun thời gian liên kết với các mô đun khác.

Privilege types: Phân quyền, đưa ra các vai trò khác nhau cho người dùng trong ứng dụng.

Legacy Interface: Kế thừa giao diện.