| ĐẠI HỌC BÁCH KHOA HÀ NỘI  TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG  ──────── \* ──────── | |
| --- | --- |
|  | |
| BÁO CÁO CUỐI KỲ  **MÔN: CHUYÊN ĐỀ** | |
| Đề tài: Xây dựng trang web chia sẻ kiến thức  Knowledge Sharing | |
|  | **Nhóm sinh viên thực hiện:** Nhóm 7   | **STT** | **MSSV** | **Họ tên** | | --- | --- | --- | | 1 | 20194185 | Trịnh Đức Tiệp | | 2 | 20194139 | Trần Văn Phúc |   **Mã lớp:** 141352  **GVHD:** TS. Nguyễn Nhất Hải |
| Hà Nội, tháng 6 năm 2023 | |

# MỤC LỤC

[**MỤC LỤC 2**](#_vlujv9q3u70r)

[**1. Mở đầu 4**](#_chtz5ipsec14)

[**2. Phân tích yêu cầu 5**](#_2ajip2u7r0qb)

[2.1. Mô tả yêu cầu 5](#_kbosjbgq6hjl)

[2.1.1. Mô tả yêu cầu khách hàng 5](#_hombx5jrs1qa)

[2.1.2. Mô tả yêu cầu về ràng buộc dữ liệu 6](#_rt2rs6sggw10)

[2.1.4. Mô tả yêu cầu phi chức năng 9](#_h55nb4xj0oex)

[2.2. Phân tích chức năng 9](#_cd0eqrruug3a)

[2.3.1. Nhóm chức năng quản lý tài khoản 9](#_3r8gsmngmxkh)

[2.3.2. Nhóm chức năng người học 10](#_26yduqog0r83)

[2.3.3. Nhóm chức năng người tạo khóa học 10](#_e1rhs5nu4zep)

[2.3.4. Nhóm chức năng quản trị viên 11](#_775m9uoozocy)

[2.3.5. Sơ đồ phân cấp chức năng. 11](#_18jel1rjp2ck)

[2.4. Xác định phạm vi 12](#_2m5x37azxltx)

[2.5. Phác họa giải pháp 12](#_ebppnpx28c6e)

[**3. Phân tích hệ thống 14**](#_gqsey6z0z6mz)

[3.1. Biểu đồ usecase tổng quan 14](#_8fugju1462zf)

[3.2. Đặc tả usecase 14](#_2wz0hji3o9ne)

[3.2.1. Phân rã usecase Quản lý cá nhân 1](#_psenhybxgcs7)

[3.2.2. Phân rã usecase Quản lý học tập 1](#_52ndqgfpyn97)

[3.2.3. Phân rã usecase Học tập 1](#_jqpbfboxnjya)

[3.2.4. Phân rã usecase Quản lý khóa học và bài học 1](#_9qx7efj8mgs)

[3.2.5. Phân rã usecase Quản lý hệ thống 1](#_iace2hnd2ncj)

[3.3. Biểu đồ trình tự 1](#_nbv7dsliaxlk)

[3.3.1. Trình tự Đăng ký tài khoản 1](#_ql1zc624gy9)

[3.3.2. Trình tự Quản lý Profile 1](#_v352sw6154y)

[3.2.3. Trình tự Yêu cầu tham gia khóa học 1](#_mz5ob7jw0bdi)

[3.2.4. Trình tự Đánh giá bài học 1](#_ddqb7e7m0gjo)

[3.2.5. Trình tự Chỉnh sửa khóa học 1](#_u58c948h0i0o)

[3.2.6. Trình tự Mời người dùng tham gia khóa học 1](#_xhddkk46dbe9)

[3.2.7. Trình tự Thay đổi mức hạn chế tài khoản 1](#_8qi69bb3v2tg)

[**4. Thiết kế hệ thống 1**](#_lpq8073vezjw)

[4.1. Thiết kế lớp chi tiết 1](#_p2zu7aswca9)

[4.1.1. Xác định các lớp lĩnh vực 1](#_pbvegxbm2pm6)

[4.1.2. Xác định các lớp tham gia usecase 1](#_8iov26s2w4f9)

[4.2. Thiết kế giao diện hệ thống 1](#_bspk3me2iayk)

[4.2.1. Giao diện đăng nhập 1](#_9dzrf6eyqdyh)

[4.2.2. Giao diện profile, chỉnh sửa profile 1](#_713sba62wnig)

[4.2.3. Giao diện Thêm khóa học, sửa khóa học 1](#_5cngbk31pmg6)

[4.2.4. Giao diện Thêm bài học, sửa bài học 1](#_9n68xlbh7o7n)

[4.2.5. Giao diện chi tiết khóa học 1](#_8h9n9ql6drw)

[4.2.6. Giao diện chi tiết bài học 1](#_v5a8l9lsenfy)

[4.2.7. Giao diện đánh giá, bình luận 1](#_45vtl27xm8l)

[4.2.8. Giao diện quản lý bài học trong khóa học 1](#_lam5ppu5wpyy)

[4.3. Thiết kế cơ sở dữ liệu 1](#_r4bg18qzyu9)

[4.3.1. Sơ đồ thực thể liên kết: 1](#_lrno3ieufel2)

[**4.3.2. Sơ đồ tổng quan cơ sở dữ liệu: 1**](#_dw6qldmpydvl)

[**4.3.3. Cấu trúc của 1 số bảng trong cơ sở dữ liệu: 1**](#_6r5yr7qnzmy)

[**4.4. Thiết kế API giao tiếp giữa frontend và backend 1**](#_nj645wa47knt)

[4.4.1. Nhóm API chức năng User 1](#_jyqlcwey2b2o)

[4.4.2. Nhóm API chức năng Người học 1](#_kndnqz7qeq3j)

[4.4.3. Nhóm API chức năng Chủ khóa học 1](#_nppdedghovat)

[4.4.4. Nhóm API chức năng Tìm kiếm 1](#_eqax751f3aly)

[4.4.5. Nhóm API chức năng Admin 1](#_npdemuw3v4wu)

[**5. Cài đặt và triển khai hệ thống 1**](#_av4x5i1zxu8j)

[**KẾT LUẬN 1**](#_909g88ingp1y)

[**TÀI LIỆU THAM KHẢO 1**](#_b9du5a8ywrbk)

# 1. Mở đầu

Mục tiêu của đề tài này là tạo ra một hệ thống website chia sẻ kiến thức, cho phép người dùng tạo, tham gia các khóa học trực tuyến và chia sẻ kiến thức với nhau một cách thuận tiện, dễ dàng, nhanh chóng và tiết kiệm thời gian, tiền bạc trong việc tiếp cận kiến thức mới. Ngoài ra, các khoá học trong hệ thống này có thể được cung cấp miễn phí hoặc có tính phí. Nhờ đó, hệ thống sẽ tạo ra nguồn thu nhập cho người đăng khoá học và đồng thời cũng sẽ giúp đảm bảo chất lượng nội dung của các khóa học được cung cấp.

Mục đích của đề tài là mang đến một nền tảng chia sẻ kiến thức trực tuyến đáng tin cậy, dễ dàng sử dụng và hiệu quả cho tất cả mọi người. Hệ thống này sẽ tạo điều kiện thuận lợi để các cá nhân và tổ chức có thể chia sẻ, tìm kiếm và tiếp cận các kiến thức mới nhất, từ đó nâng cao trình độ và phát triển bản thân. Hơn nữa, sứ mệnh của đề tài còn là xây dựng cộng đồng học tập trực tuyến sôi động, giúp các thành viên có thể tương tác và hỗ trợ lẫn nhau trong quá trình học tập và chia sẻ kiến thức.

# 

# 2. Phân tích yêu cầu

## 2.1. Mô tả yêu cầu

### 2.1.1. Mô tả yêu cầu khách hàng

**Người dùng** cần đăng ký một tài khoản cá nhân để sử dụng, khai thác chức năng trang web. Nếu người dùng không có tài khoản thì hệ thống chỉ cho phép người dùng xem các bài học đơn lẻ và giới hạn các chức năng khác trong trang web. Người dùng có thể thực hiện các hoạt động quản lý tài khoản cá nhân như đổi mật khẩu, lấy lại mật khẩu đã quên. Bên cạnh đó, hệ thống cần lưu trữ thông tin cá nhân của người dùng, cho phép người dùng cập nhật thông tin cá nhân và thay đổi quyền riêng tư của các mục thông tin. Người dùng có thể liên hệ với nhau thông qua tính năng theo dõi tài khoản của người khác.

**Với vai trò là người học**, hệ thống cho phép người dùng tìm kiếm các khóa học và bài học mà mình quan tâm; đăng ký và tham gia học các khoá học cần thiết mà mình có nhu cầu nâng cao kiến thức. Sau khi học các khóa học, bài học, người dùng có thể đánh giá lại khóa học và bài học ấy bằng cách cho điểm chất lượng theo thang điểm 5 sao hoặc bình luận ý kiến ngay phía dưới của từng khóa học, bài học. Để lưu lại những bài học quan trọng mà mình quan tâm, hệ thống cho phép người dùng đánh dấu lại các bài học ấy để xem lại sau. Với những bài học có tính phí, người dùng có thể thực hiện thanh toán để học ngay, hoặc có thể gửi yêu cầu mở khóa học cho chủ khóa học để hai bên cùng nhau tự thương lượng. Bên cạnh đó, người dùng có thể chấp thuận hoặc từ chối những lời mời tham gia khóa học của những người dùng khác.

**Với vai trò là người cung cấp khóa học**, người dùng có thể biên soạn và đăng các khóa học và bài học lên hệ thống. Hệ thống cho phép người dùng có thể chỉnh sửa lại các khóa học để thêm hoặc bớt các bài học cho phù hợp hoặc chỉnh sửa lại nội dung và định dạng của từng bài học. Sau mỗi lần chỉnh sửa, hệ thống có trách nhiệm thông báo tới thành viên đã đăng ký khóa học về hành động chỉnh sửa này. Bên cạnh đó, người dùng có thể chỉnh sửa giá thành của các khóa học. Hệ thống cho phép người dùng có thể tự cấp quyền khóa học cho những người dùng khác tham gia vào khóa học của mình thông qua các hành động mời người khác tham gia khóa học hoặc chấp nhận yêu cầu tham gia khóa học của người khác. Đối với các nội dung bình luận bên dưới khóa học, bài học, người đăng bài có thể phản hồi lại các bình luận ấy để giải thích rõ hơn những vấn đề của người học. Ngoài ra, hệ thống cung cấp cho người dùng tính năng thống kê thông tin các khóa học và doanh thu tương ứng.

**Với vai trò là quản trị viên**, người dùng cần phải là người có quyền hạn, trách nhiệm lớn đối với hệ thống và phải là người hệ thống tin tưởng. Quản trị viên có thể quản lý các tài khoản của người dùng, kiểm tra các hoạt động bất thường của các tài khoản để đưa ra các phán quyết như chặn, hạn chế tính năng, hoặc thậm chí là xóa tài khoản. Một tài khoản có 4 mức hạn chế (xem yêu cầu về ràng buộc dữ liệu), tùy theo mức độ vi phạm của tài khoản mà quản trị viên có thể hạn chế tài khoản ở mức độ nào hoặc ân xá mở lại các tính năng cho các tài khoản đã bị hạn chế. Ngoài ra quản trị viên có có quyền quản lý toàn bộ các khóa học, bài học của các người dùng đăng lên. Những bài học hoặc khóa học có đánh giá thấp, điểm tin cậy kém (công thức tính điểm tin cậy xem ở yêu cầu ràng buộc dữ liệu) hoặc vi phạm tiêu chuẩn cộng đồng cũng sẽ đưa vào diện xem xét để quản trị viên đưa ra phán quyết là hủy bỏ hay giữ lại.

### 

### 2.1.2. Mô tả yêu cầu về ràng buộc dữ liệu

a) Ràng buộc về tài khoản

Người dùng khi đăng ký cần cung cấp tên tài khoản là **email** chưa đăng ký trong hệ thống. Mật khẩu không được để rỗng.

b) Thông tin cá nhân người dùng (Profile)

Bao gồm các trường thông tin như tên, ảnh đại diện, ngày sinh, địa chỉ, nghề nghiệp – chức vụ, số điện thoại, email, liên kết xã hội, châm ngôn sống.

c) Quyền riêng tư

Quyền riêng từ bao gồm: **public** (công khai cho mọi người biết), **default** (chỉ cho phép người mình theo dõi biết), **private** (chỉ cho phép bản thân mình biết). Sắp xếp quyền riêng tư theo mức độ chặt dần: **public > default > private**. Quyền riêng tư mặc định nếu người dùng không chỉ định là **default**.

d) Ràng buộc về theo dõi người dùng

Mỗi người dùng chỉ được theo dõi tối đa 1.000 người dùng khác và trong ngày, mỗi người dùng chỉ được phép gửi tối đa 50 follows tới người dùng khác. Mỗi người dùng có số lượt người theo dõi mình là không giới hạn.

e) Ràng buộc về số lượng khóa học đăng ký tham gia

Mỗi người dùng chỉ được phép tham gia và gửi đăng ký tối đa tới 1000 khóa học. Nếu muốn đăng ký hoặc tham gia khóa học khác, cần rời khóa học cũ hoặc hủy đăng ký khóa học khác đang ở trong trạng thái request.

f) Ràng buộc về số lượng người học trong một khóa học

Mỗi khóa học không giới hạn số lượng người học khóa học ấy.

g) Ràng buộc về số lượng khóa học, bài học có thể tạo ra

Mỗi người dùng chỉ có thể tạo ra tối đa 1000 khóa học khác nhau. Trong một ngày, một người dùng có thể tạo ra tối đa 10 khóa học. Một người dùng có thể tạo ra không giới hạn các bài học, nhưng trong một ngày chỉ được tạo tối đa 100 bài học.

h) Ràng buộc về các loại khóa học

Có hai loại khóa học trong hệ thống là khóa học miễn phí và khóa học tính phí. Khóa học miễn phí – cung cấp cho toàn bộ người dùng; khóa học có tính phí – cung cấp cho những người thanh toán, những người được mời và những người được chấp nhận yêu cầu tham gia khóa học. Một khóa học là tập hợp danh sách các bài học được sắp xếp theo trật tự kiến thức của chủ khóa học. Các quyền thao tác của chủ khóa học được ràng buộc theo bảng phân quyền sau:

| Quyền | Khóa học miễn phí | Khóa học có tính phí |
| --- | --- | --- |
| Sửa nội dung | Được phép | Được phép |
| Thêm thành viên | Được phép | Được phép |
| Xóa khóa học | Được phép | Chỉ được phép khi khóa học không còn thành viên nào |
| Xóa thành viên | Được phép | Không được phép, chủ khóa học cần thương lượng với người học để tự hủy đăng ký khóa học |
| Chỉnh sửa giá thành | Không được phép, luôn luôn là khóa học miễn phí | Được phép, có thể sửa thành 0$ nhưng bản chất vẫn là khóa học có phí |

i) Ràng buộc về các loại bài học

Bài học public – bài học được public lên trang cá nhân của người dùng; bài học private – bài học trong trang cá nhân nhưng không được public, chỉ bản thân mới có thể xem; bài học tham chiếu – là một bài học thuộc một khóa học và được tham chiếu từ danh sách các bài viết trên trang cá nhân. Bài học public luôn luôn có trạng thái public, bài học private chỉ được phép xem bởi chủ bài học, bài học tham chiếu chỉ được phép xem trong khóa học chứa nó.

j) Điểm số của khóa học, bài học

Mỗi người dùng có thể đánh giá cho khóa học, bài học một lần và có thể chỉnh sửa sau này. Điểm số của khóa học, bài học được quy định theo thang điểm 5 sao, mỗi người dùng có thể đánh giá từ 0 sao đến 5 sao với ý nghĩa như sau:

| Điểm đánh giá | Ý nghĩa |
| --- | --- |
| **0 sao** | Bài học bị giảng sai trầm trọng về kiến thức hoặc vi phạm tiêu chuẩn cộng đồng. |
| **1 sao** | Bài học tệ, nhảm nhí và không có giá trị, không có tính ứng dụng trong cuộc sống hoặc bài học có một phần kiến thức lớn bị giảng sai. |
| **2 sao** | Bài học có kiến thức mang tính trung bình, có một số lỗi nhỏ, cách giảng dạy không hấp dẫn, không cuốn hút. |
| **3 sao** | Bài học quá dễ hoặc quá khó, chỉ phù hợp với một nhóm người nào đó chứ không phù hợp với bản thân. |
| **4 sao** | Bài giảng hay, hấp dẫn, kiến thức bổ ích giải đáp phần nào thắc mắc của bản thân. |
| **5 sao** | Bài học hoàn hảo, thuyết phục và giải đáp toàn bộ những vướng mắc của bản thân. |

Điểm số của khóa học, bài học được tính bằng trung bình cộng các đánh giá của người dùng.

k) Ràng buộc về 4 mức hạn chế tài khoản

Mức 0 – an toàn, đây là mức mà một người dùng thực hiện đầy đủ các nhóm quyền người dùng. Mức 1 – cảnh báo, đây là mức hạn chế người dùng không được phép tạo khóa học, bài học cho người khác. Mức 2 – kiểm điểm (chặn, block), đây là mức hạn chế người dùng bị chặn cả quyền tạo khóa học bài học, cả quyền đăng ký học khóa học bài học khác. Mức 3 – khai trừ (xóa tài khoản), đây là mức hạn chế cao nhất khi người dùng bị khai trừ khỏi hệ thống. Tổng hợp các mức hạn chế theo bảng dưới đây.

| Mức hạn chế tài khoản | Quyền tạo khóa học, bài học | Quyền đăng ký học, xem bài học | Quyền quản lý tài khoản cá nhân | Phán quyết |
| --- | --- | --- | --- | --- |
| Mức 0 | Có | Có | Có | An toàn |
| Mức 1 | Không | Có | Có | Cảnh cáo |
| Mức 2 | Không | Không | Có | Chặn |
| Mức 3 | Không | Không | Không | Xóa tài khoản |

l) Các dữ liệu lưu trữ trong hệ thống.

Dữ liệu lưu trữ trong hệ thống bao gồm:

1. Thông tin tài khoản; thông tin profile người dùng; thông tin về quan hệ theo dõi giữa các người dùng.

2. Dữ liệu các bài học; các bài học, khóa học được lưu; các đánh giá điểm số của bài học, khóa học; các bình luận của bài học, khóa học.

3. Dữ liệu các khóa học; bài học trong khóa học; khóa học đã đăng ký (thành viên của khóa học); yêu cầu tham gia khóa học course request; lời mời tham gia khóa học course invite; lịch sử thanh toán khóa học course payment.

2.1.3. Mô tả yêu cầu về các quy trình nghiệp vụ phức tạp

a) Về quy trình đăng ký tài khoản

Người dùng cung cấp tên, email (được dùng làm tên tài khoản sau này), mật khẩu cho hệ thống. Hệ thống kiểm tra hợp lệ rồi gửi mã thông báo tới email người dùng, thời hạn 2 phút. Người dùng nhận mã rồi nhập vào mã xác nhận hoàn tất quá trình đăng ký. Hệ thống chuyển người dùng tới trang cập nhật thông tin cá nhân.

b) Về quy trình lấy lại tài khoản bị quên mật khẩu.

Người dùng cấp email cho hệ thống để lấy lại mật khẩu, hệ thống kiểm tra email rồi chuyển người dùng tới trang nhập mật khẩu mới. Người dùng nhập mật khẩu mới rồi bấm nút lấy mã xác nhận, hệ thống gửi mã xác nhận vào email. Người dùng kiểm tra email, nhập mã xác nhận rồi bấm nút gửi yêu cầu đổi mật khẩu. Hệ thống kiểm tra mã xác nhận và đổi lại mật khẩu cho người dùng.

c) Về quy trình đăng ký khóa học.

Với khóa học miễn phí, người dùng chỉ cần bấm đăng ký, hệ thống tự kiểm tra thông tin rồi thêm người dùng vào khóa học. Với khóa học có tính phí, người dùng có hai lựa chọn là mua khóa học ngay hoặc gửi yêu cầu tham gia khóa học. Khi bấm mua khóa học ngay, người dùng được chuyển sang trang thanh toán. Tại đây, người dùng nhập số tiền rồi bấm thanh toán, hệ thống kiểm tra giao dịch rồi cấp khóa học ngay cho người dùng. Trong trường hợp người dùng không muốn thanh toán mà bấm gửi yêu cầu tham gia khóa học, hệ thống sẽ tạo một request gửi tới chủ khóa học. Lúc này, người dùng cần đợi hoặc chủ động liên hệ với chủ khóa học để hai bên tự thương lượng với nhau. Chủ khóa học chỉ cần xét duyệt yêu cầu là hệ thống sẽ tự xử lý thêm người dùng vào khóa học.

d) Về quy trình thêm bài học vào khóa học.

Khi chủ khóa học muốn thêm một bài học mới vào khóa học hiện có, chủ khóa học có 2 lựa chọn đó là chọn một bài học từ danh sách các bài học đã có sẵn của chủ khóa học hoặc tạo ra một bài học mới hoàn toàn. Nếu chủ khóa học chọn thêm bài học từ danh sách bài học có sẵn, hệ thống hiển thị danh sách các bài học và chủ khóa học chỉ cần chọn bài học phù hợp. Nếu chủ khóa học muốn thêm một bài học mới, hệ thống chuyển sang giao diện tạo bài học mới. Chủ khóa học soạn thảo bài học và chọn thêm bài học. Bài học sẽ tự động thêm vào danh sách bài học của chủ khóa học và khóa học tương ứng cũng sẽ thêm một bài học tham chiếu vào bài học mà chủ khóa học vừa tạo.

### 

### 2.1.4. Mô tả yêu cầu phi chức năng

Hệ thống cần đảm bảo hoạt động một cách hiệu quả, đáp ứng nhu cầu của người dùng với tốc độ nhanh và độ tin cậy cao. Đồng thời, hệ thống cần đảm bảo khả năng mở rộng để đáp ứng nhu cầu sử dụng của người dùng trong tương lai. Hơn nữa, hệ thống cũng cần đảm bảo tính bảo mật và an toàn cho thông tin cá nhân của người dùng.

Giao diện người dùng cần phải dễ sử dụng, có tính trực quan cao, hình ảnh, màu sắc và bố cục hài hoà, mượt mà giúp cho việc tiếp thu, chia sẻ kiến thức trở lên dễ dàng hơn.

## 

## 2.2. Phân tích chức năng

Bảng tổng hợp các chức năng cần có của hệ thống.

### 

### 2.3.1. Nhóm chức năng quản lý tài khoản

| STT | Tên chức năng | Mô tả chức năng |
| --- | --- | --- |
| 1 | Đăng nhập | Tạo một phiên làm việc mới cho người dùng, yêu cầu người dùng nhập email và mật khẩu |
| 2 | Đăng ký | Tạo một tài khoản mới cho người dùng |
| 3 | Đổi mật khẩu | Giúp người dùng đổi mật khẩu mới khi người dùng muốn thay đổi hoặc bị lộ |
| 4 | Lấy lại tài khoản  (Quên mật khẩu) | Giúp người dùng lấy lại mật khẩu khi bị quên, yêu cầu cần xác thực email |
| 5 | Hủy tài khoản | Hủy tài khoản khi người dùng không muốn sử dụng hệ thống nữa |
| 6 | Quản lý Profile | Thay đổi thông tin người dùng và quyền truy cập của từng mục thông tin |
| 7 | Theo dõi người dùng | Cho phép người dùng theo dõi hoạt động của người dùng khác khi quan tâm tới các bài viết và khóa học của họ |

### 

### 2.3.2. Nhóm chức năng người học

| STT | Tên chức năng | Mô tả chức năng |
| --- | --- | --- |
| 1 | Đăng ký khóa học | Đăng ký tham gia khóa học để có thể truy cập vào các bài học có trong khóa học ấy |
| 2 | Thanh toán khóa học | Thực hiện các bước trả tiền cho khóa học tính phí để tham gia ngay khóa học ấy |
| 3 | Gửi yêu cầu tham gia khóa học | Gửi lời đề nghị tới chủ khóa học để xin phép được tham gia khóa học ấy, cần phải có sự chấp nhận của chủ khóa học mới có thể tham gia |
| 4 | Hủy đăng ký khóa học | Rời bỏ khóa học đang học |
| 5 | Tìm kiếm khóa học, bài học | Thực hiện tìm kiếm các khóa học và bài học mà mình quan tâm dựa trên các từ khóa tìm kiếm |
| 6 | Xác nhận lời mời tham gia khóa học | Đồng ý hoặc từ chối tham gia khóa học mà chủ khóa học đã gửi lời mời tới mình |
| 7 | Xem bài học | Xem chi tiết nội dung trình bày của bài học |
| 8 | Đánh giá khóa học, bài học | Thực hiện đánh giá khóa học, bài học bằng cách cho điểm hoặc bình luận phía dưới để chia sẻ cảm xúc hoặc hỏi những vấn đề chưa hiểu trong bài học |
| 9 | Lưu khóa học, bài học | Đánh dấu lại những khóa học, bài học mà mình quan tâm để lưu lại và xem lại sau |

### 

### 2.3.3. Nhóm chức năng người tạo khóa học

| STT | Tên chức năng | Mô tả chức năng |
| --- | --- | --- |
| 1 | Tạo khóa học | Tạo một khóa học mới cho bản thân mình |
| 2 | Xóa khóa học | Xóa khóa học trong danh sách khóa học của bản thân |
| 3 | Sửa khóa học | Thực hiện các thao tác cập nhật thông tin của khóa học như chỉnh sửa giá thành khóa học, thêm bài học vào khóa học hoặc xóa bài học khỏi khóa học |
| 4 | Thêm bài học | Thêm một bài học vào danh sách bài học trên trang cá nhân |
| 5 | Sửa bài học | Sửa chi tiết nội dung của bài học như tên bài học, nội dung bài học, định dạng trình bày, … |
| 6 | Xóa bài học | Xóa bỏ hoàn toàn bài học khỏi trang cá nhân và các bài học tham chiếu liên quan |
| 7 | Xem danh sách khóa học | Hiển thị danh sách các khóa học của một người chủ khóa học |
| 8 | Xem danh sách thành viên khóa học | Chủ khóa học xem danh sách mọi người đang tham gia học khóa học ấy |
| 9 | Mời người tham gia khóa học | Chủ khóa học muốn mời danh sách những người cụ thể tham gia vào học khóa học của mình |
| 10 | Duyệt yêu cầu tham gia khóa học | Chủ khóa học đồng ý hoặc từ chối các lời đề nghị xin tham gia khóa học của bản thân |

### 

### 2.3.4. Nhóm chức năng quản trị viên

| STT | Tên chức năng | Mô tả chức năng |
| --- | --- | --- |
| 1 | Hạn chế tài khoản | Quản trị viên quyết định mức hạn chế cho tài khoản nếu vi phạm tiêu chuẩn cộng đồng |
| 2 | Ân xá tài khoản | Quản trị viên phục hồi các quyền thực hiện trên hệ thống cho các tài khoản đã thực hiện xong án phạt |
| 3 | Xóa tài khoản | Quản trị viên xóa bỏ các tài khoản vi phạm nghiêm trọng các tiêu chuẩn cộng đồng của hệ thống |
| 4 | Tìm kiếm tài khoản | Tìm kiếm danh sách tài khoản |
| 5 | Tìm kiếm khóa học | Tìm kiếm danh sách khóa học |
| 6 | Xóa khóa học | Xóa khóa học vi phạm khỏi hệ thống |
| 7 | Tìm kiếm bài học | Tìm kiếm danh sách bài học |
| 8 | Xóa bài học | Xóa bài học vi phạm khỏi hệ thống |

### 2.3.5. Sơ đồ phân cấp chức năng.

Mô hình hóa toàn bộ chức năng của hệ thống thông qua sơ đồ phân cấp chức năng như ở dưới.



## 2.4. Xác định phạm vi

Do hệ thống chia sẻ kiến thức sau khi phân tích còn khá rộng, nên đề tài này sẽ giới hạn lại phạm vi thực hiện, cụ thể như sau:

Về đối tượng phục vụ và quản lý. Hệ thống tập trung phục vụ các đối tượng chính là học sinh, sinh viên, giáo viên, cộng đồng lập trình viên. Hệ thống cũng tập trung vào quản lý các kiến thức liên quan đến giáo dục trong nhà trường hoặc các kiến thức trong ngành công nghệ thông tin.

Về địa lý. Hệ thống tập trung phục vụ cho các đối tượng ở Việt Nam.

Về chức năng. Các chức năng cơ bản mà hệ thống có gần như tuân theo các phân tích ở trên. Riêng chức năng thanh toán khóa học có tính phí, hệ thống chỉ mô phỏng lại chức năng thanh toán, chứ không thực hiện như đúng theo trình tự thanh toán trong thực tế là người dùng cần liên kết tài khoản ngân hàng và hệ thống cần hợp tác với ngân hàng để làm nhiệm vụ chuyển tiền.

## 

## 2.5. Phác họa giải pháp

Qua những phân tích về yêu cầu hệ thống ở trên, giải pháp của bài toán được dự kiến phác họa như sau:

Về kiến trúc hệ thống. Hệ thống dự kiến sẽ được xây dựng tuân theo hai mô hình kiến trúc phổ biến, đó là mô hình MVC (Model-View-Controller) và mô hình Client-Server. Trong đó, tầng View sẽ đóng vai trò là Frontend, hai tầng Model và Controller đóng vai trò là Backend. Frontend và Backend sẽ giao tiếp với nhau thông qua các API mà Backend cung cấp.

Về nền tảng phát triển. Hệ thống được xây dựng trên nền tảng Website. Web là nền tảng rất phổ biến, dễ sử dụng và dễ tiếp cận với mọi người. Với mọi thiết bị chỉ cần có kết nối Internet và trình duyệt, người dùng có thể truy cập trang web tại mọi lúc, mọi nơi.

Về công cụ phát triển. Hệ thống dự kiến được xây dựng dựa vào bộ công cụ phát triển cho lập trình viên như sau:

+ Ngôn ngữ lập trình: Các ngôn ngữ dùng để lập trình Web cơ bản như HTML, CSS, JavaScript, …

+ Thư viện, framework, hệ quản trị cơ sở dữ liệu: Frontend sẽ sử dụng framework ReactJs và Bootstrap. Backend sử dụng NodeJs, ExpressJs. Database sử dụng hệ quản trị cơ sở dữ liệu MySQL.

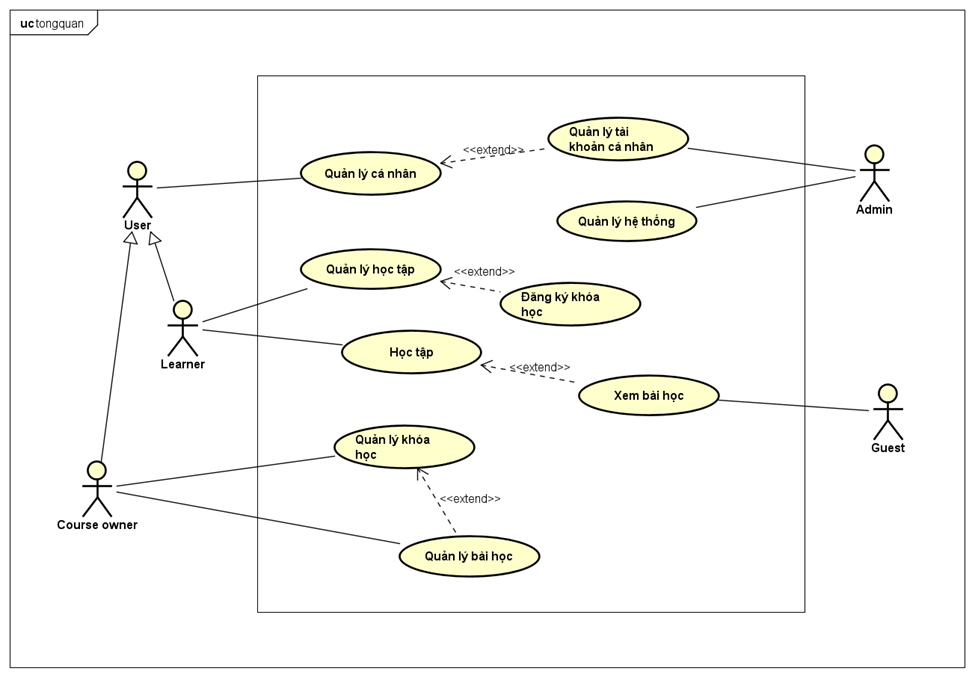
+ Công cụ khác: IDE sử dụng Visual Studio Code, Sublime Text. Công cụ test API Postman, công cụ quản lý mã nguồn git/github, công cụ kiểm thử chrome devtool, công cụ quản lý sản phẩm docker, công cụ quản lý triển khai ci/cd Jenkins, …

# 

# 3. Phân tích hệ thống

## 3.1. Biểu đồ usecase tổng quan

Theo các phân tích chi tiết về yêu cầu ở trên, biểu đồ usecase tổng quan chỉ ra rõ rằng hệ thống sẽ gồm 4 tác nhân chính (User, Learner, Course Owner, Admin) và một tác nhân phụ là Client. Chi tiết hơn, tác nhân user thực hiện nhóm usecase quản lý cá nhân, Learner thực hiện quản lý học tập, đăng ký khóa học và học tập. Course owner thì thực hiện quản lý khóa học và bài học. Admin thực hiện quản lý tài khoản cá nhân và quản lý hệ thống.



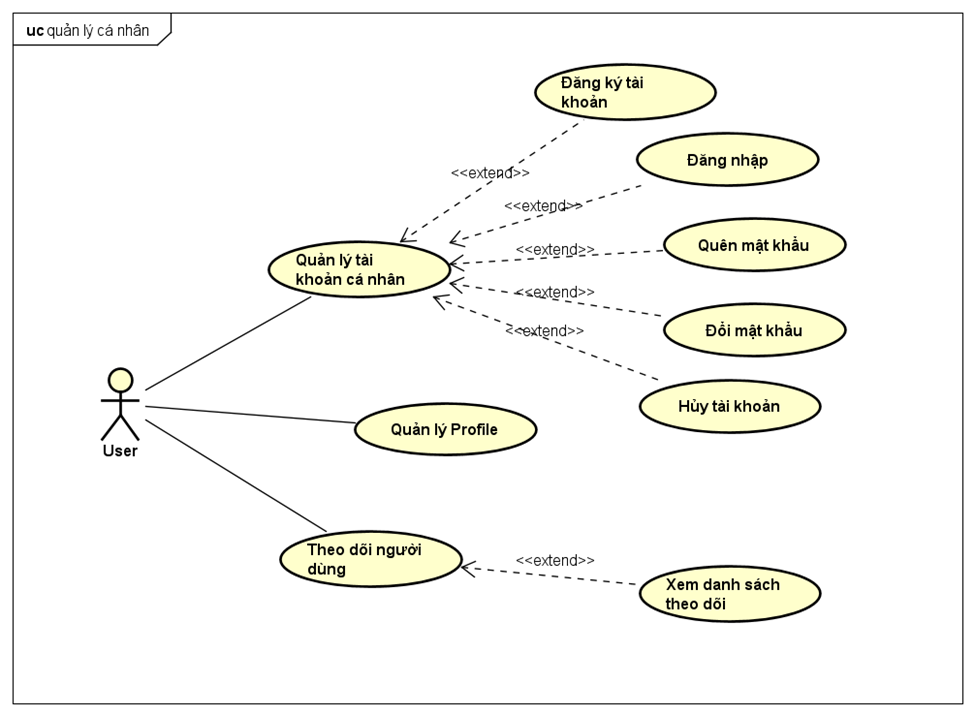
## 3.2. Đặc tả usecase

Sau đây nhóm sẽ phân rã một các usecase và đặc tả một số usecase chính trong mỗi nhóm usecase được phân rã mức 2.

### 

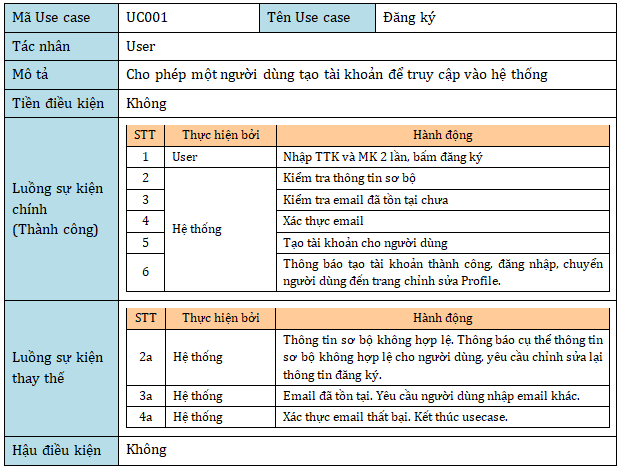
### 3.2.1. Phân rã usecase Quản lý cá nhân

Usecase Quản lý cá nhân được phân rã mức hai theo sơ đồ usecase dưới đây.



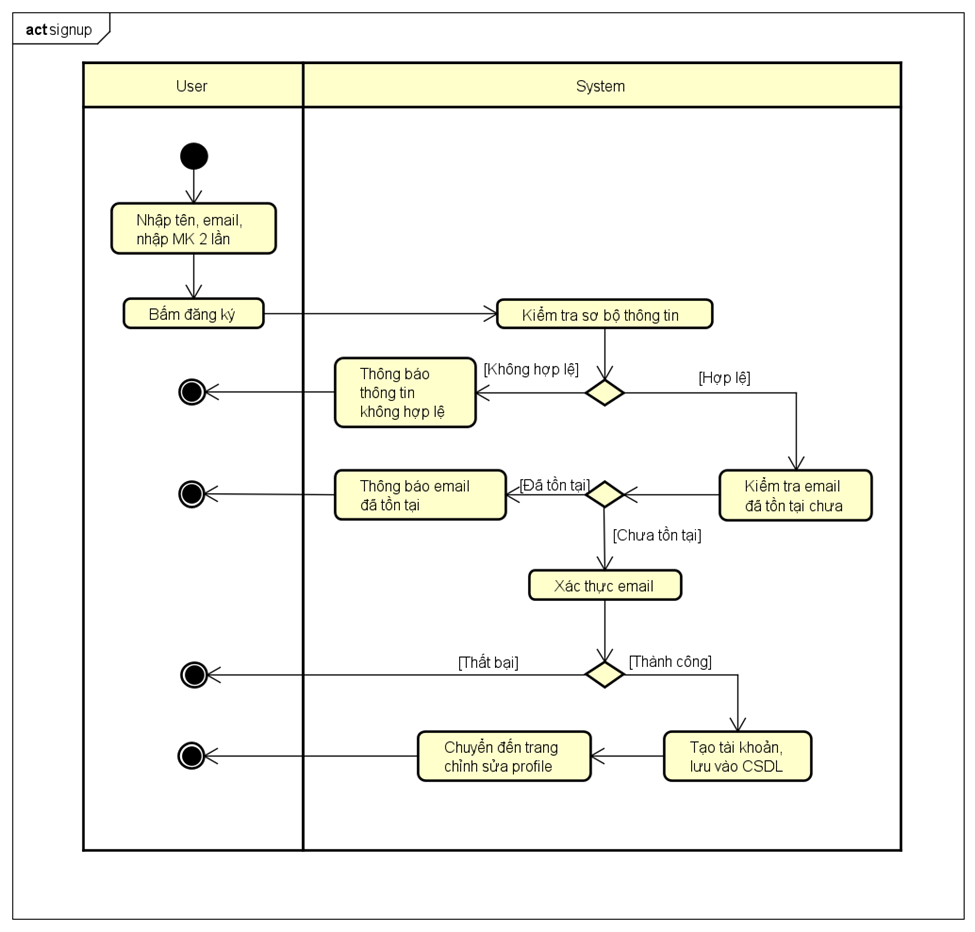
Theo sơ đồ usecase phân rã mức hai này, User và Admin sẽ có chung nhóm usecase là Quản lý tài khoản cá nhân. Ngoài ra, User còn có thêm usecase Quản lý Profile và quản lý Theo dõi người dùng.

Đặc tả usecase Đăng ký tài khoản:



Biểu đồ hoạt động Đăng ký:

Hoạt động đăng ký yêu cầu hệ thống kiểm tra tính hợp lệ của thông tin đăng ký tài khoản một cách chặt chẽ (kiểm tra sơ bộ và kiểm tra email có hợp lệ). Sau đó bước quan trọng nhất là xác thực email thông qua mã code gửi tới email người dùng. Khi đảm bảo thành công các bước kiểm tra ấy hệ thống mới tiến hành tạo tài khoản cho người dùng.



## 

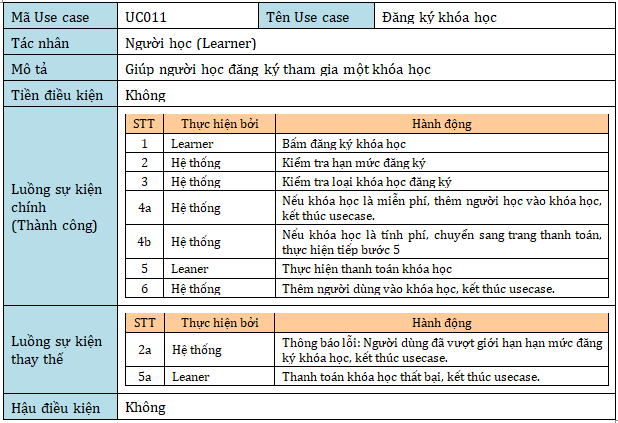
### 3.2.2. Phân rã usecase Quản lý học tập

Usecase Quản lý học tập được phân rã mức hai theo sơ đồ usecase dưới đây.

## 

Nhóm usecase Quản lý học tập có tác nhân chính là người học (Learner). Nhóm usecase này sẽ chứa những usecase liên quan đến các hoạt động hỗ trợ quản lý danh sách khóa học cho người học như Đăng ký, hủy đăng ký khóa học, Tìm kiếm khóa học, bài học và Quản lý danh sách các lời mời tham gia khóa học.

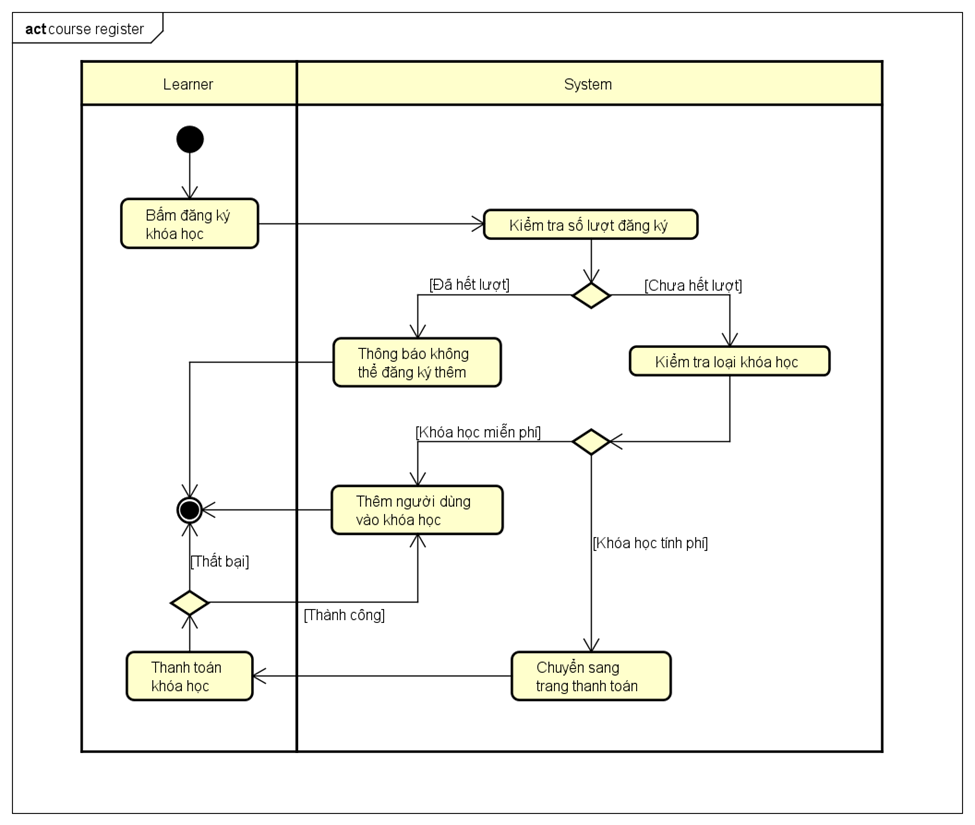
Đặc tả usecase Đăng ký khóa học:



## 

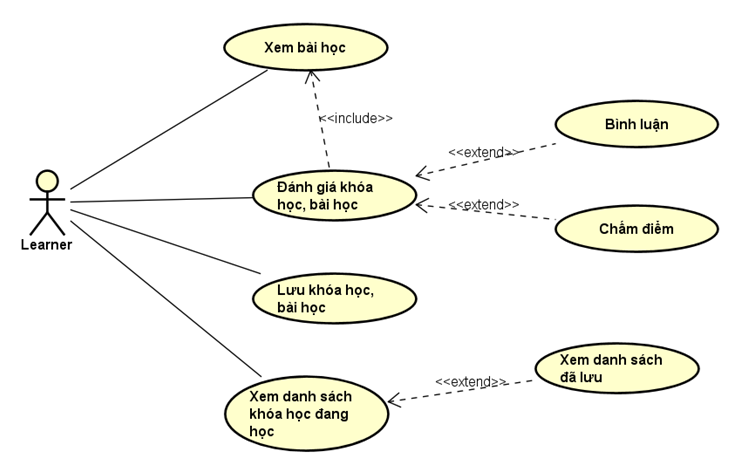
Biểu đồ hoạt động Đăng ký khóa học:

Hoạt động đăng ký khóa học gộp chung cả hai loại khóa học là Khóa học miễn phí và Khóa học tính phí. Trước khi thực hiện đăng ký khóa học, cần kiểm tra số lượt đăng ký và số lượt khóa học của người học còn trong hạn mức cho phép hay không. Sau đó chia làm 2 trường hợp xử lý. Với khóa học miễn phí thì chỉ cần thêm người học vào khóa học là xong. Còn với khóa học có tính phí thì cần đưa người dùng chuyển sang usecase thanh toán khóa học để xử lý tiếp.



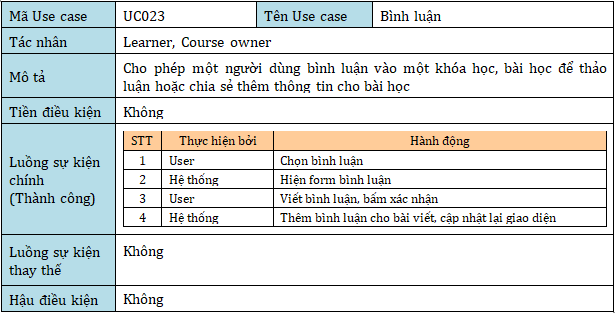
### 3.2.3. Phân rã usecase Học tập

Nhóm usecase này phục vụ tác nhân chính là Learner và có phân rã như sau:

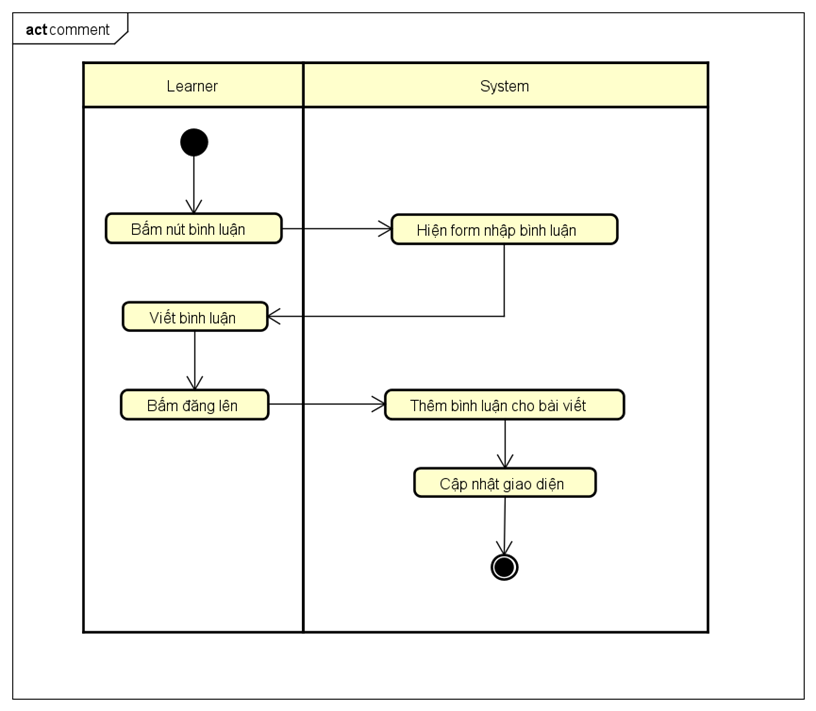


Các usecase này phục vụ cho quá trình học tập và bàn luận về những khóa học, bài học.

Đặc tả usecase Bình luận bài học:

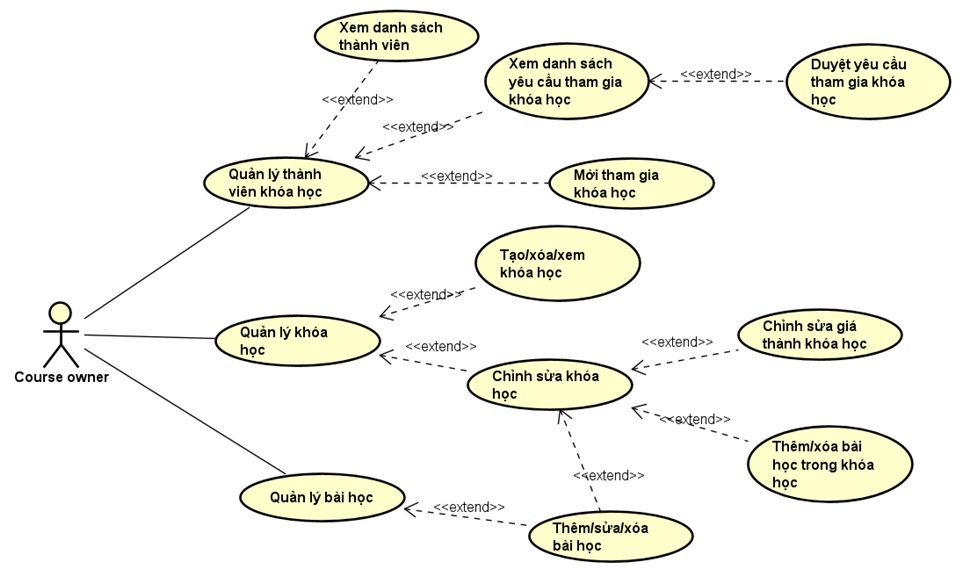


Biểu đồ hoạt động:

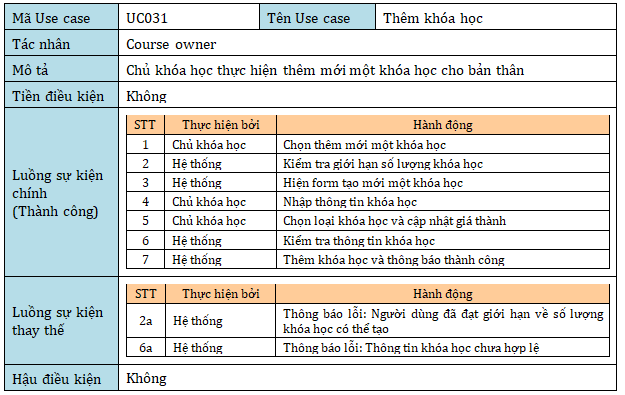


### 3.2.4. Phân rã usecase Quản lý khóa học và bài học

Usecase Quản lý khóa học và bài học là nhóm usecase phục vụ cho tác nhân chính là những người đăng bài học, khóa học muốn chia sẻ kiến thức cho mọi người. Ta gọi chung họ là những chủ khóa học hay course owner. Nhóm usecase này cho phép chủ khóa học có thể dễ dàng quản lý danh sách các khóa học và bài học của mình qua các hành động thêm sửa xóa cơ bản và quản lý danh sách thành viên của khóa học của mình.

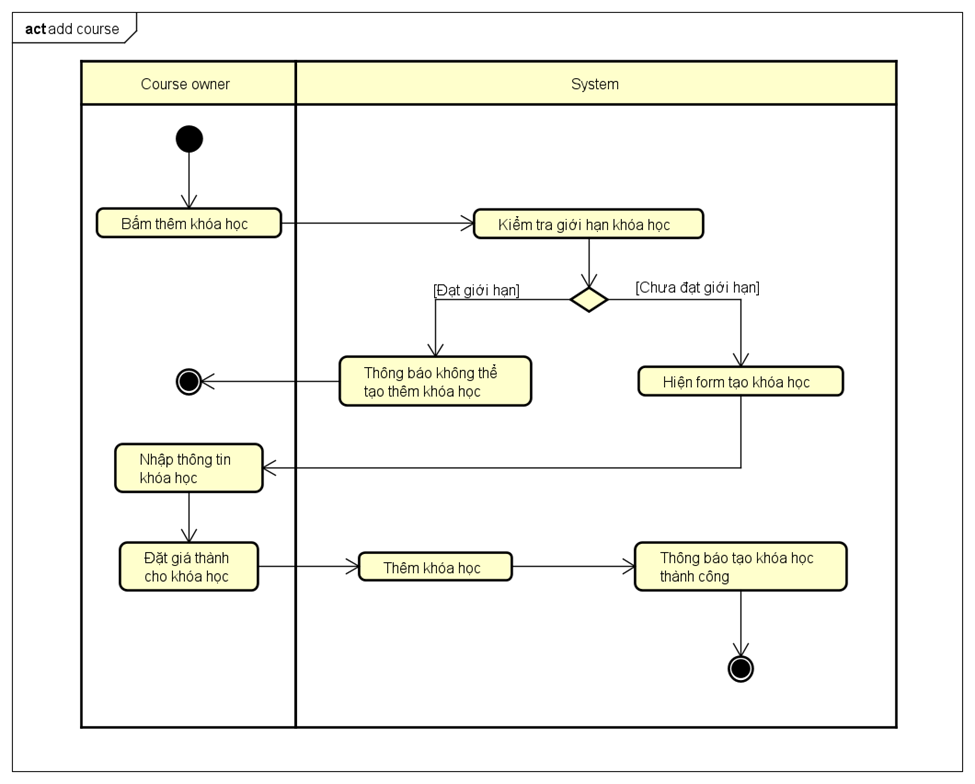


Đặc tả usecase Thêm khóa học:



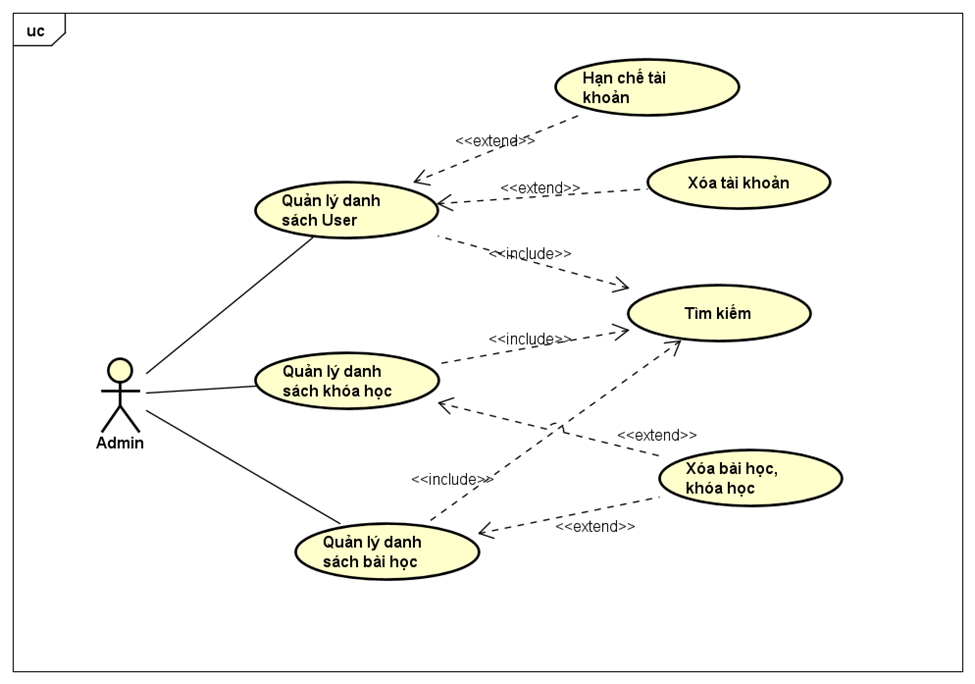
Biểu đồ hoạt động Thêm khóa học:

Hoạt động thêm khóa học chú trọng vào bước kiểm tra giới hạn khóa học của một người dùng, cụ thể như sau:

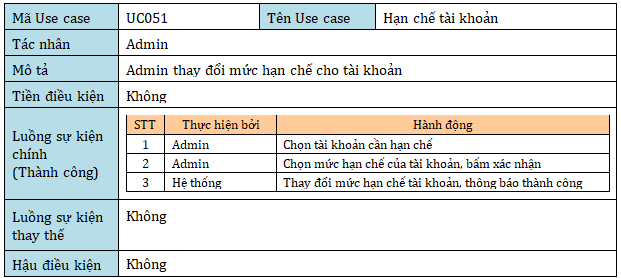


### 3.2.5. Phân rã usecase Quản lý hệ thống

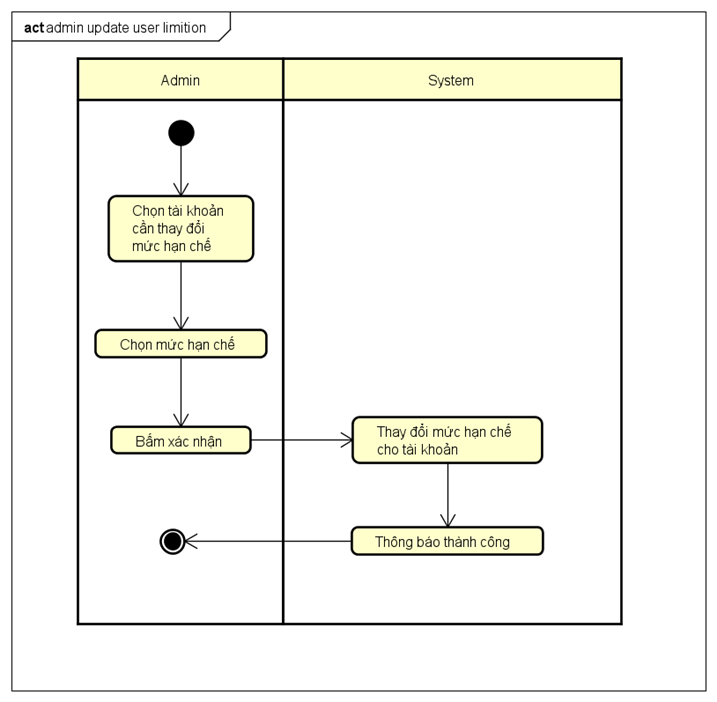
Nhóm usecase quản lý hệ thống hỗ trợ cho Admin các chức năng cơ bản để quản trị hệ thống như quản trị tài khoản (hạn chế tài khoản và xóa tài khoản) hay quản trị khóa học, bài học (xóa khóa học, xóa bài học).



Đặc tả usecase Hạn chế tài khoản:



Biểu đồ hoạt động Hạn chế tài khoản:



## 3.3. Biểu đồ trình tự

Biểu đồ trình tự giúp chúng ta thấy được rõ hơn về thứ tự trước sau mà các lớp, các đối tượng giao tiếp với nhau trong luồng hoạt động của từng usecase. Bên cạnh đó, ta có thể nhìn nhận rõ ràng về trách nhiệm của từng lớp, từng đối tượng và toàn bộ vòng đời của nó khi thực hiện usecase.

Một biểu đồ trình tự cần biểu diễn các thông tin bao gồm: Các tác nhân và đối tượng tham gia trình tự được dàn hàng ngang; Vòng đời và thời gian sống của từng đối tượng được vẽ theo đường thẳng đứng; Sự giao tiếp lẫn nhau giữa các đối tượng thông qua mũi tên thông điệp và một số thông tin bổ sung khác.

### 3.3.1. Trình tự Đăng ký tài khoản

Trình tự đăng ký tài khoản do tác nhân user thực hiện và có sự tham gia của các object gồm 1 màn đăng ký, 1 controller xác thực, 1 controller email và 2 enity là account và profile. Trình tự gồm 2 luồng hoạt động chính là kiểm tra thông tin đăng ký và xác thực email bằng code.

# 

### 3.3.2. Trình tự Quản lý Profile

Trình tự quản lý Profile do tác nhân user thực hiện, giúp tác nhân cập nhật được thông tin Profile cũng như visible của các mục thông tin ấy.

# 

# 

### 3.2.3. Trình tự Yêu cầu tham gia khóa học

Trình tự gồm hai tác nhân tham gia là người học và chủ khóa học. Tương ứng với 2 luồng hoạt động là gửi yêu cầu và xác nhận yêu cầu.

# 

# 

### 3.2.4. Trình tự Đánh giá bài học

Người học thực hiện việc chấm điểm cho khóa học và thực hiện bình luận cho khóa học ấy.

# 

# 

### 3.2.5. Trình tự Chỉnh sửa khóa học

Chủ khóa học chỉnh sửa thông tin của một khóa học như thế nào và cập nhật danh sách các bài học trong khóa học ấy ra sao sẽ được thể hiện qua biểu đồ trình tự này.

# 

### 3.2.6. Trình tự Mời người dùng tham gia khóa học

Trình tự thêm người dùng vào khóa học thể hiện việc chủ khóa học có thể thực hiện chủ động mời người dùng khác tham gia học tập khóa học của mình. Ngoài ra trình tự cũng thể hiện việc người học có thể đồng ý lời mời để tham gia vào khóa học mà chủ khóa học đã gửi lời mời tới mình.

# 

# 

### 3.2.7. Trình tự Thay đổi mức hạn chế tài khoản

Đây là một trình tự đơn giản, thể hiện cho việc admin của hệ thống quản lý danh sách tài khoản đang hoạt động trong hệ thống thông qua việc thay đổi mức hạn chế của tài khoản.

# 

# 

# 4. Thiết kế hệ thống

## 4.1. Thiết kế lớp chi tiết

### 4.1.1. Xác định các lớp lĩnh vực

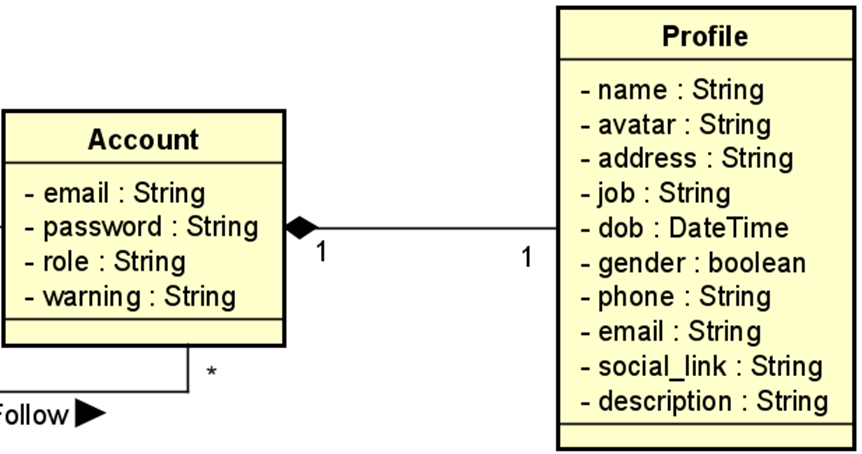
a). Lớp Account (Tài khoản)

Do yêu cầu của hệ thống là cần phải đăng nhập vào mới sử dụng được các tài nguyên nên cần có lớp Account để thế hiện cho tài khoản đăng nhập của người dùng. Bên cạnh đó, lớp Account cũng chịu trách nhiệm thể hiện các thông tin của một tài khoản người dùng như vai trò của người dùng là gì và mức cảnh báo hiện tại của tài khoản hiện là như thế nào. Ngoài ra, lớp Account còn có một liên kết đệ quy n-n tới chính mình thông quan quan hệ theo dõi để thực hiện chức năng theo dõi giữa các tài khoản khác nhau trong hệ thống.

## 

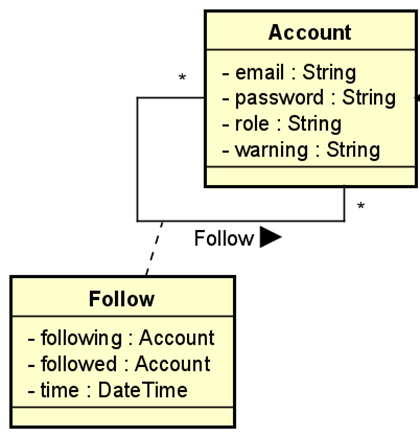
b) Lớp Profile

Để lưu trữ thông tin chi tiết của người dùng như tên, ngày sinh, giới tính, địa chỉ, … ta cần sử dụng một lớp riêng biệt đảm nhiệm trách nhiệm này, gọi lớp này là Profile. Lớp Profile có liên kết 1-1 với lớp Account thể hiện quan hệ “Một Account có duy nhất một Profile”.



c) Lớp Follow (Theo dõi)

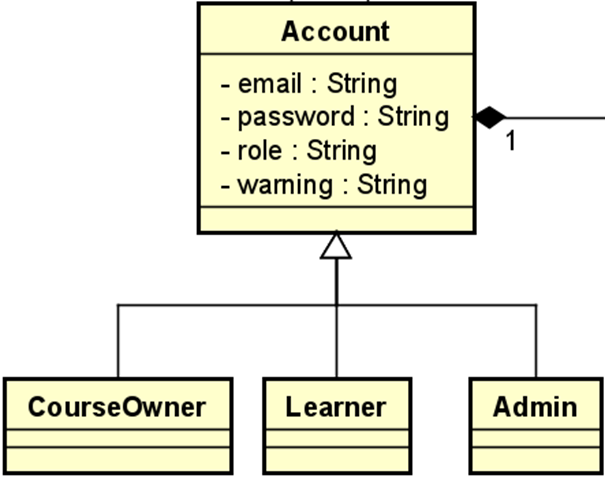
Lớp Follow thể hiện quan hệ theo dõi giữa hai tài khoản người dùng khác nhau. Đây là quan hệ n-n thể hiện một tài khoản có thể theo dõi nhiều tài khoản khác và một tài khoản cũng có thể được nhiều tài khoản khác theo dõi lại. Một account sẽ lưu trữ một mảng các đối tượng Follow mà Account đó được theo dõi hoặc chính Account đó tự thực hiện theo dõi người khác.



d) Lớp CourseOwner (Chủ khóa học), Learner (Người học) và Admin (Quản trị viên)

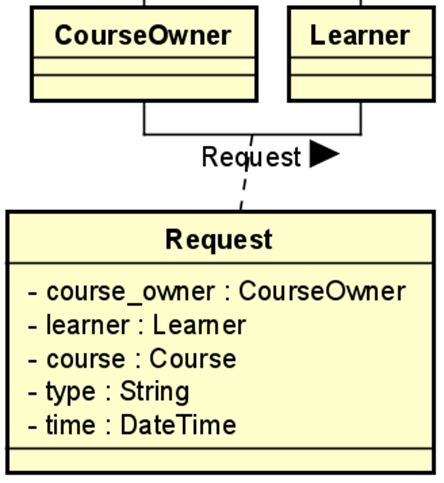
Đây là ba lớp con cùng kế thừa lớp cha là Account. Ba lớp này có cùng thông tin nhưng vai trò của chúng là khác nhau: Lớp CourseOwner đại diện cho chủ khóa học tham gia vào các nghiệp vụ chia sẻ kiến thức còn lớp Learner đại diện cho người học thì lại tham gia các nghiệp vụ học tập các kiến thức có trong hệ thống. Lớp Admin lại đại diện cho nhóm những người quản trị viên tham gia vào quản trị hệ thống.

Sơ đồ cụ thể của ba lớp trên như sau:



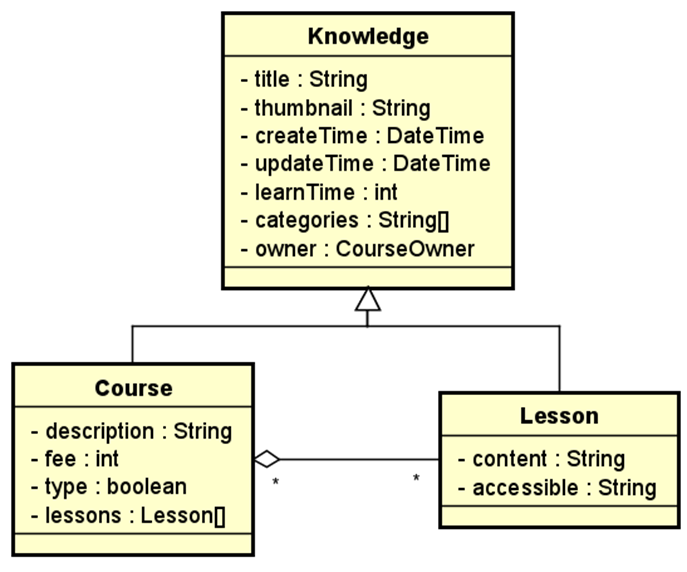
e) Lớp Request (Yêu cầu, Lời mời)

Giữa CourseOwner và Leaner tồn tại một mối quan hệ n-n, đó là quan hệ yêu cầu tham gia khóa học hoặc lời mời tham gia khóa học. Cũng tương tự như quan hệ Follow, đây là quan hệ n-n có nghĩa rằng một người có thể invite/request tới nhiều người khác hoặc cũng có thể được invite/request bởi nhiều người khác nhau. Mối quan hệ giữa hai lớp này được thực thể hóa thành lớp Request để lưu trữ thông tin request/invite khác nhau. Quan hệ cụ thể như hình sau:

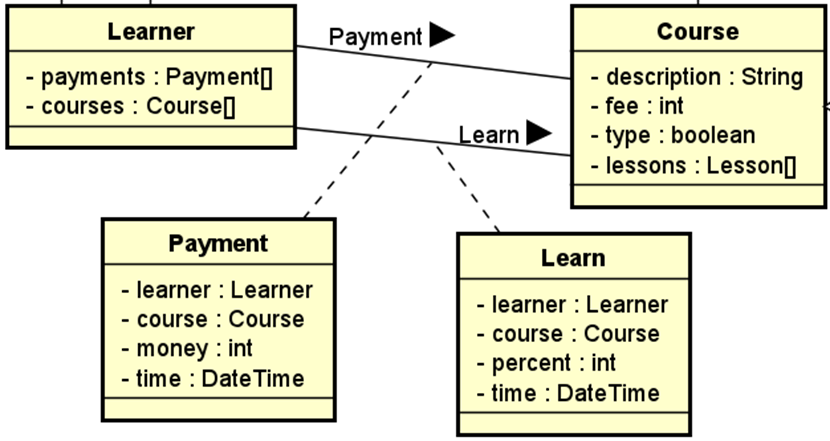


f) Lớp Knowledge (Kiến thức), Course (Khóa học) và Lesson (Bài học)

Lớp Knowledge là lớp cha đại diện cho 2 lớp kế thừa là lớp Course và Lesson chứa các thông tin chung mà cả Course và Lesson đều cần có. Lớp Course đại diện cho khóa học còn lớp Lesson lại đại diện cho bài học. Hai lớp Lesson và Course có quan hệ với nhau, cụ thể là một khóa học có thể chứa nhiều bài học và một bài học cũng có thể nằm trong nhiều khóa học khác nhau.

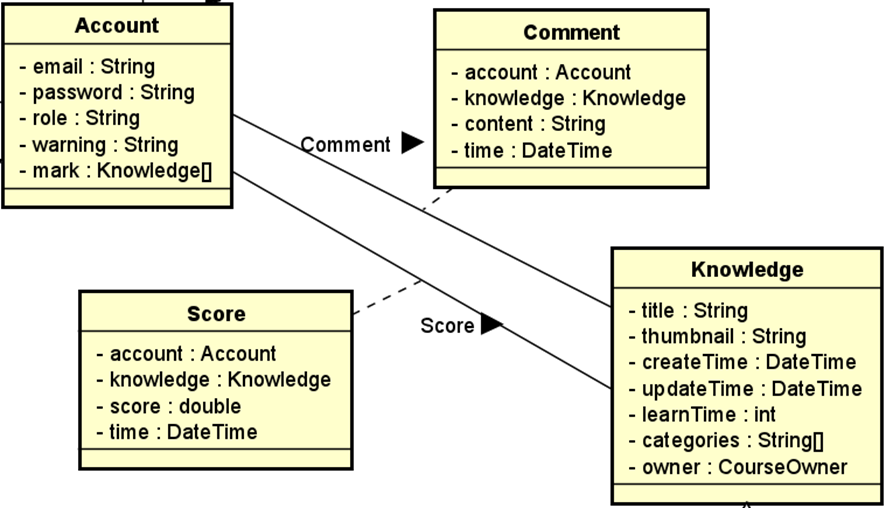


g) Lớp Payment (Thanh toán) và Learn (Học tập)



Hai lớp Payment và Learner này biểu diễn mối quan hệ giữa Người học và Khóa học. Trong khi lớp Payment lưu trữ thông tin thanh toán khóa học khi người học đăng ký mua khóa học, thì lớp Learn sẽ lưu trữ thông tin liên quan đến tiến độ học tập của một người học trong một khóa học cụ thể. Cả hai quan hệ này đều là quan hệ n-n.

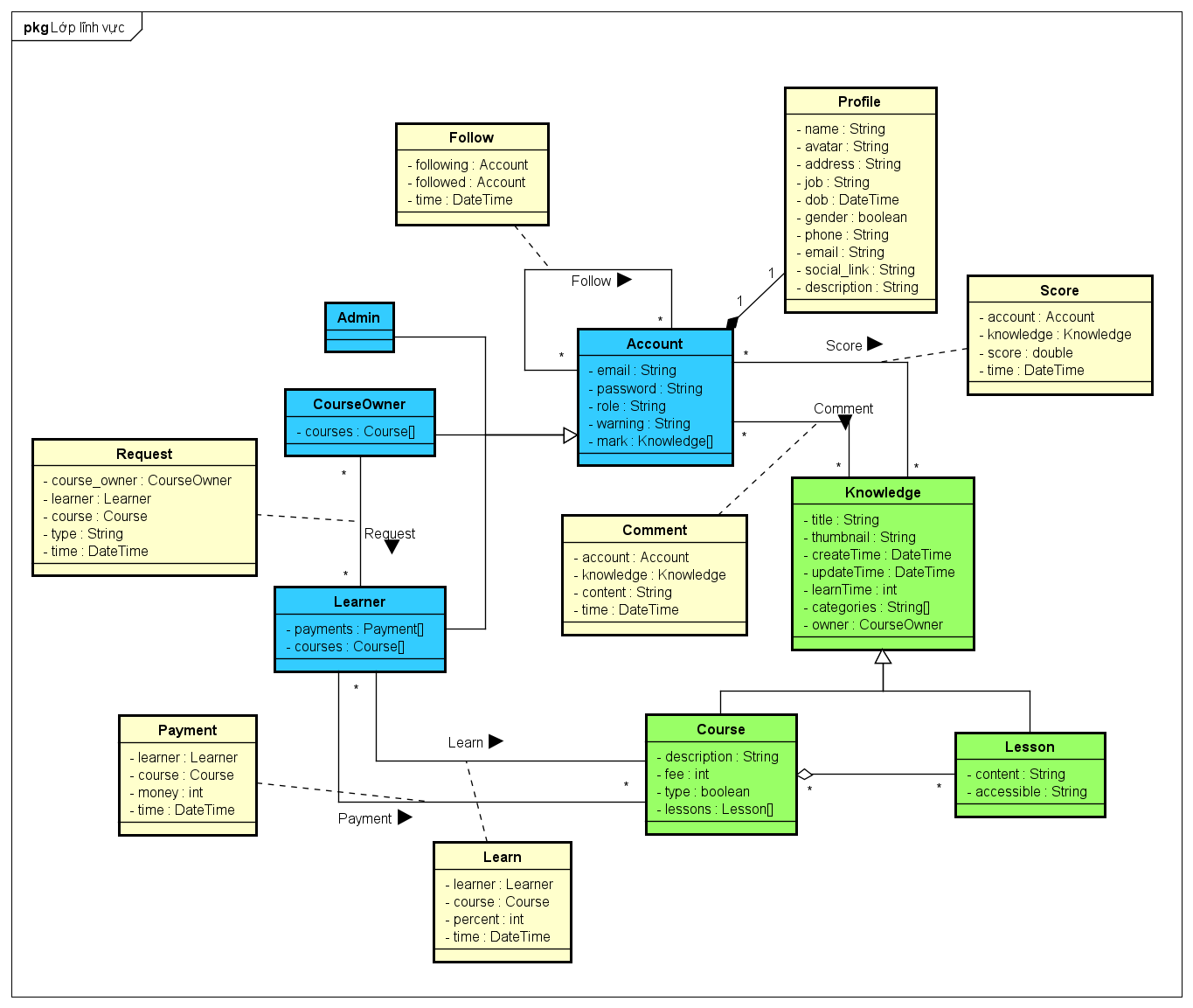
h) Các lớp liên quan đến quan hệ giữa Account và Knowledge (lớp Comment, Score)



Hai lớp quan trọng trong quan hệ giữa Account và Knowledge là lớp Comment và lớp Score. Lớp Comment lưu trữ những thông tin liên quan đến bình luận của người dùng tới một mục kiến thức (Course hoặc Lesson). Lớp Score lưu danh sách đánh giá điểm số cho của một người dùng cho một mục kiến thức. Hai lớp này hỗ trợ Account có thể tương tác với khóa học và bài học bằng cách cho điểm và bình luận. Ngoài ra, để Account có thể lưu trữ thông tin danh sách các mục kiến thức mà mình quan tâm đánh dấu, lớp Account có thêm một trường thuộc tính là mark có kiểu là một mảng của Knowledge.

i) Biểu đồ lớp lĩnh vực

Qua các bước phân tích các lớp lĩnh vực tham gia vào hệ thống, dưới đây sẽ là biểu đồ tổng hợp lại toàn bộ các lớp ấy và quan hệ giữa chúng để có thể có được cái nhìn tổng quan nhất về toàn bộ lớp thực thể hay lớp lĩnh vực xuất hiện trong hệ thống.



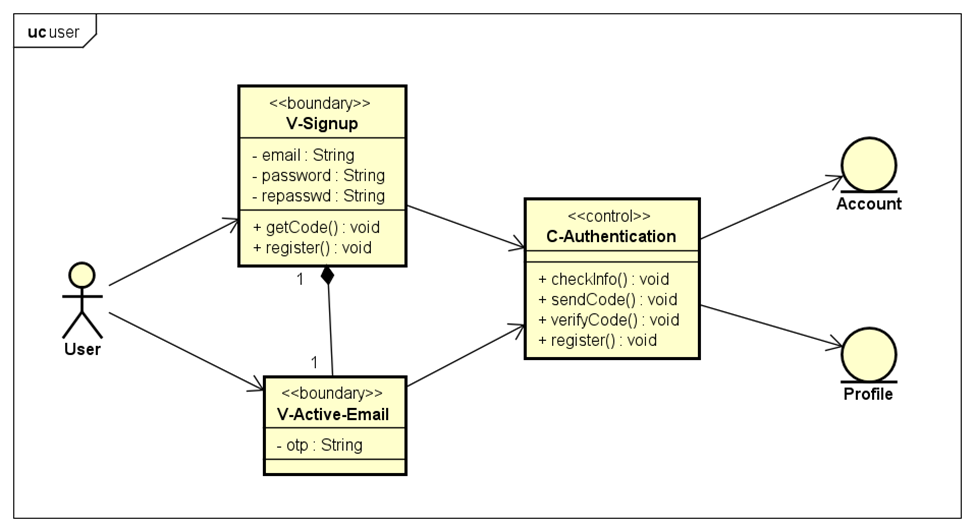
### 

### 4.1.2. Xác định các lớp tham gia usecase

Khi xác định biểu đồ lớp tham gia usecase, ta sẽ xác định thêm các lớp biên và các lớp điều khiển cùng tham gia để hoàn thành các usecase ấy. Sau đây chúng ta sẽ phân tích một số usecase điển hình để xác định các lớp biên, lớp điều khiển ấy.

a) Biểu đồ lớp tham gia usecase Đăng ký tài khoản

Dựa vào đặc tả usecase Đăng ký tài khoản, ta xác định có 2 lớp boundary và một lớp control tham gia usecase này, cụ thể:

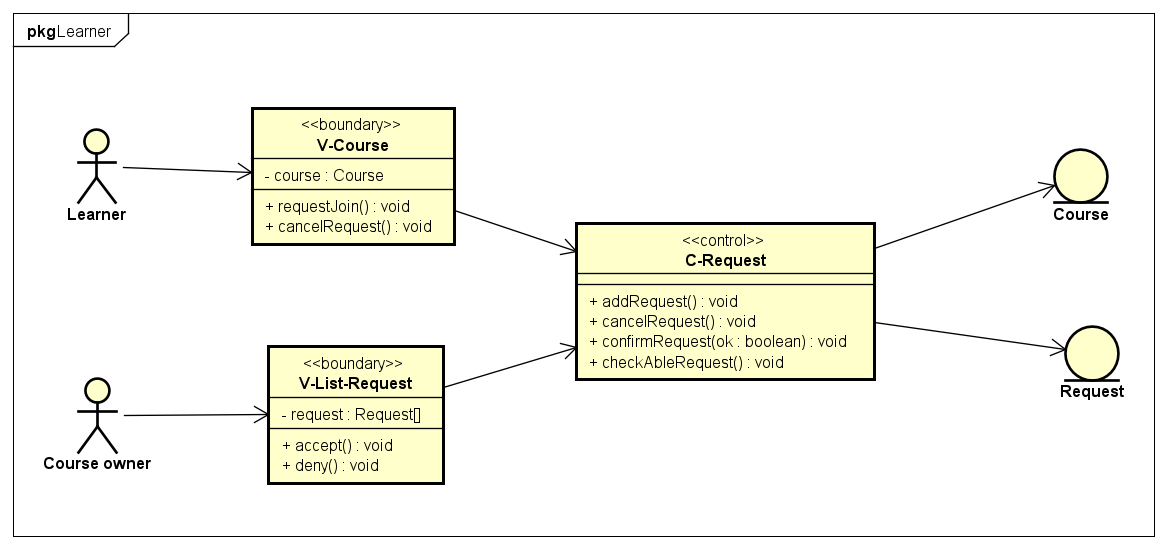


Lớp V-Signup là lớp biên chứa input người dùng nhập vào bao gồm email, password, repasswd và có các phương thức như getCode() để kiểm tra thông tin nhập vào lấy mã otp xác thực và register() để đăng ký tài khoản.

Lớp Boundary V-Active-Email chứa một trường input là otp yêu cầu người dùng nhập vào. Lớp này là một phần của màn hình đăng ký.

Lớp control C-Authentication chứa các phương thức như checkInfo() để kiểm tra thông tin tài khoản đăng ký của người dùng, sendCode() để gửi code otp về email, verifyCode() giúp xác thực code và register() để tiến hành đăng ký tài khoản mới cho user.

b) Biểu đồ lớp tham gia usecase Yêu cầu tham gia khóa học

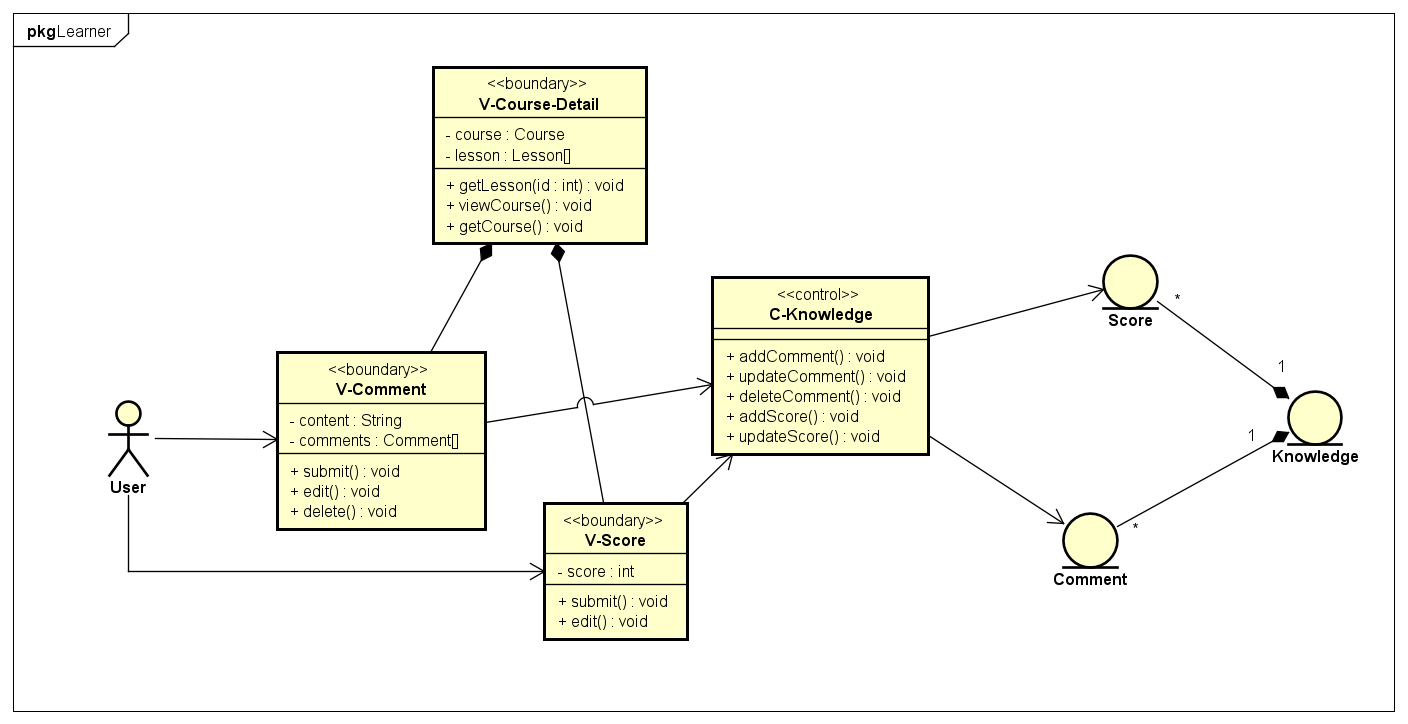


Learner gửi yêu cầu tham gia khóa học thông qua V-Course của khóa học tương ứng và cũng có thể hủy yêu cầu bằng hai phương thức requestJoin() và cancelRequest() của lớp boundary này.

Course owner có thể xác nhận request hoặc từ chối request thông qua hai phương thức accept() và deny() của lớp boundary V-List-Request.

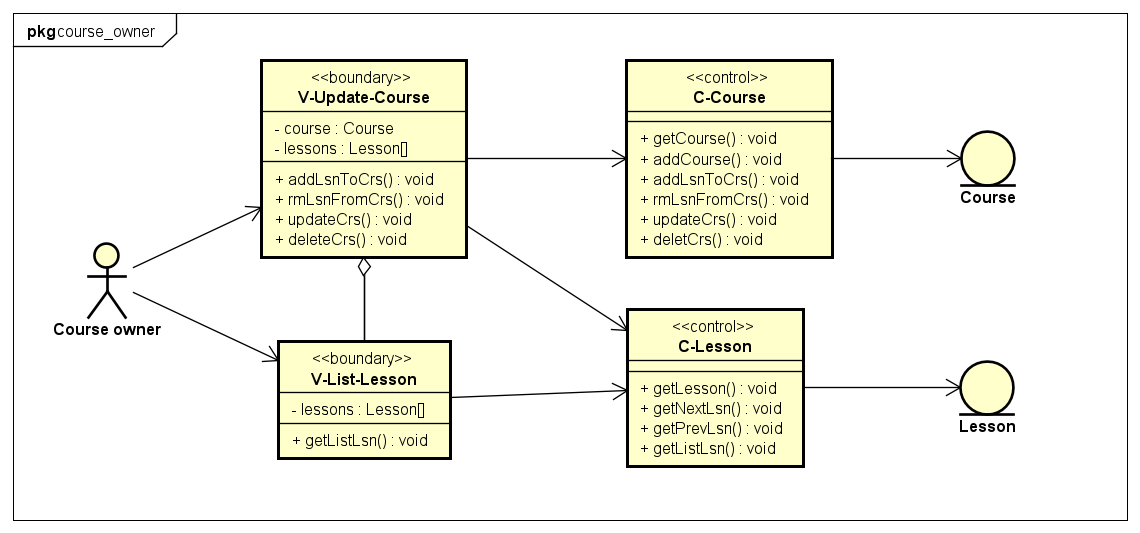
Lớp control C-Request giúp xử lý các yêu cầu gửi từ lớp boundary gồm có các phương thức addRequest(), cancelRequest(), confirmRequest() và checkableRe -quest() tương ứng với từng nghiệp vụ.

c) Biểu đồ lớp tham gia usecase Đánh giá bài học



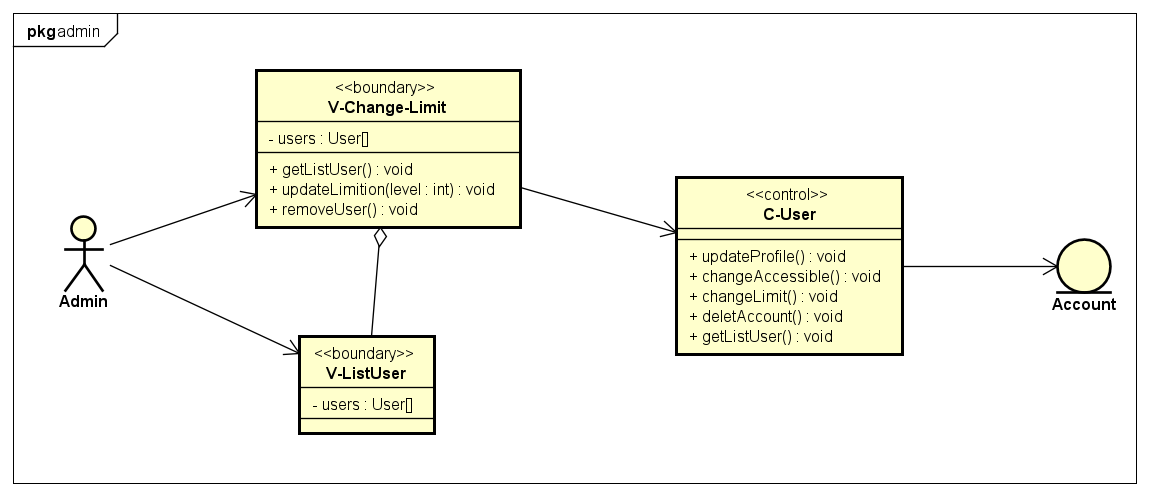
Tham gia vào nghiệp vụ Đánh giá bài học gồm 2 lớp boundary đều là bộ phận của boundary V-Course-Lesson là lớp V-Comment và lớp V-Score. V-Comment hỗ trợ người dùng có thể xem các bình luận của một mục kiến thức, bên cạnh đó có thể bình luận, sửa và xóa theo các phương thức mà nó hỗ trợ. V-Score cũng cho phép user chấm điểm và sửa điểm cho mục kiến thức. Lớp control C-Knowledge hỗ trợ các request gửi từ boundary bằng 5 phương thức tương ứng.

d) Biểu đồ lớp tham gia usecase Chỉnh sửa khóa học



Nghiệp vụ cập nhật khóa học chỉ thay mỗi boundary V-Create-Course thành V-Update-Course với các phương thức được hỗ trợ là addLsnToCrs(): thêm bài học vào khóa học, rmLsnFromCrs(): xóa bài học khỏi khóa học, updateCrs(): cập nhật khóa học và deleteCrs(): xóa khóa học. Lớp control C-Course cũng tương ứng thêm các phương thức xử lý nghiệp vụ tương ứng.

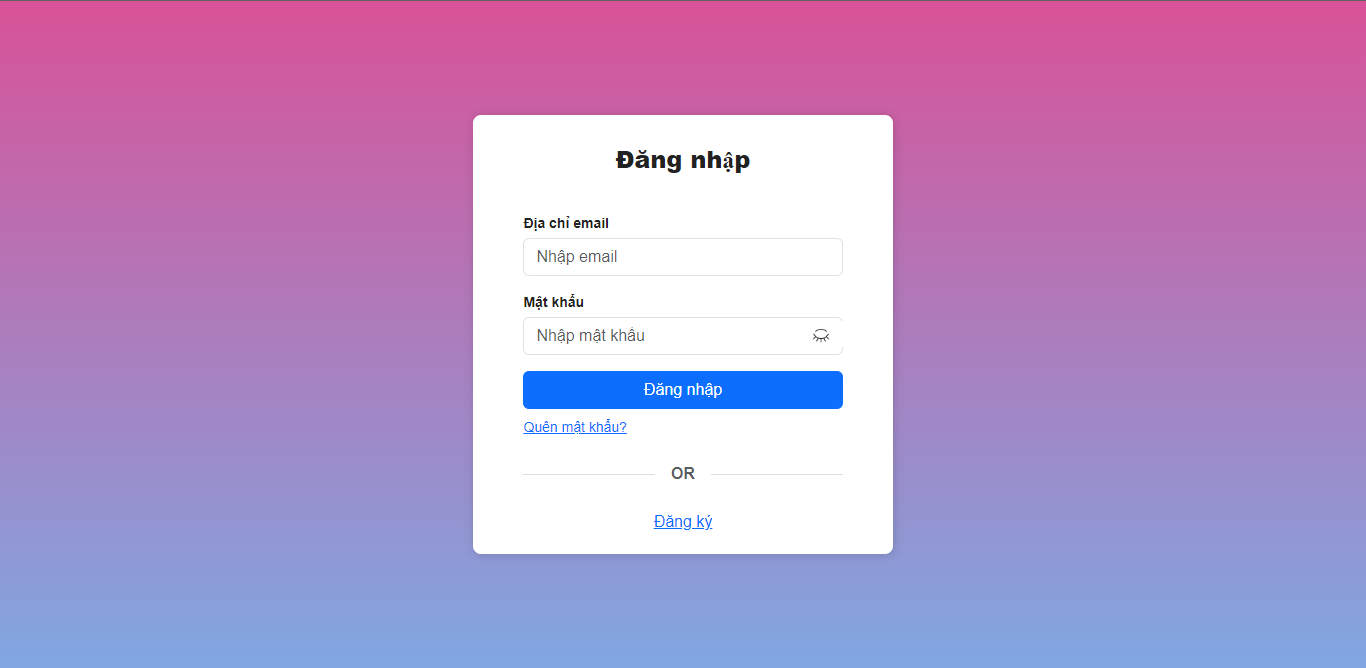
e) Biểu đồ lớp tham gia usecase Thay đổi mức hạn chế tài khoản



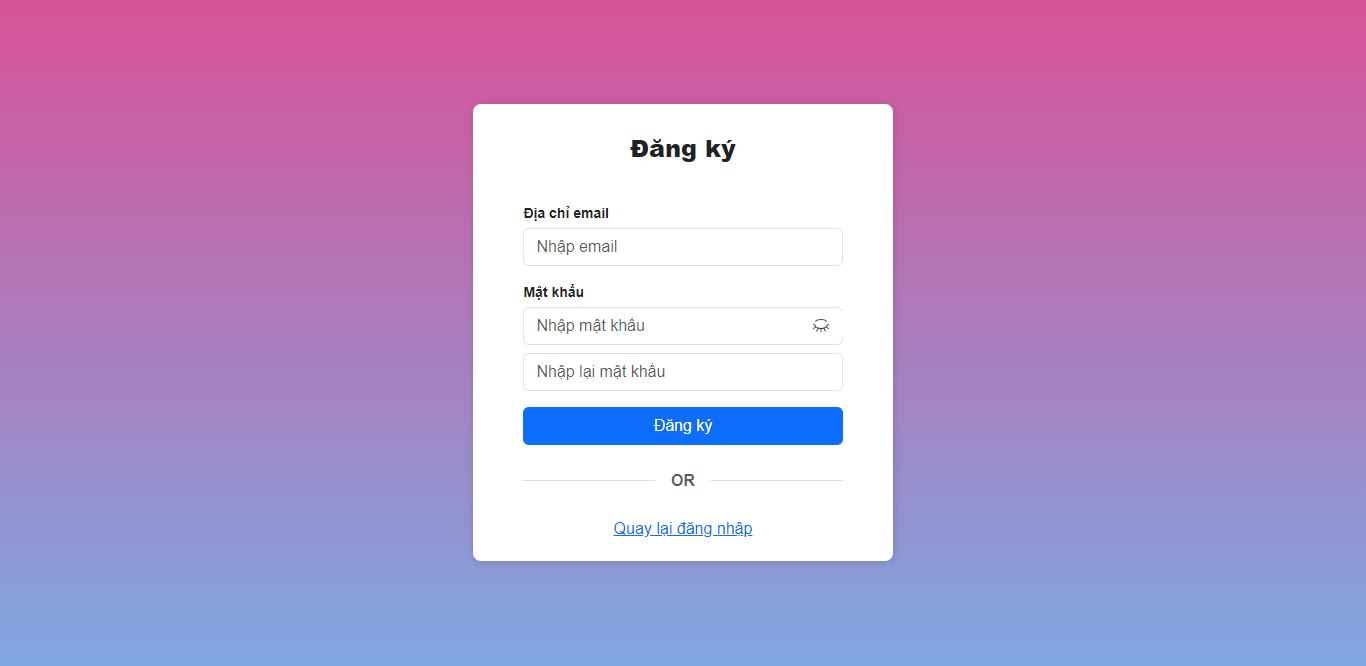
## 

## 4.2. Thiết kế giao diện hệ thống

### 4.2.1. Giao diện đăng nhập



4.3.2. Giao diện đăng ký



### 

### 4.2.2. Giao diện profile, chỉnh sửa profile

### 

### 

### 

### 

### 4.2.3. Giao diện Thêm khóa học, sửa khóa học

### 

### 

### 4.2.4. Giao diện Thêm bài học, sửa bài học

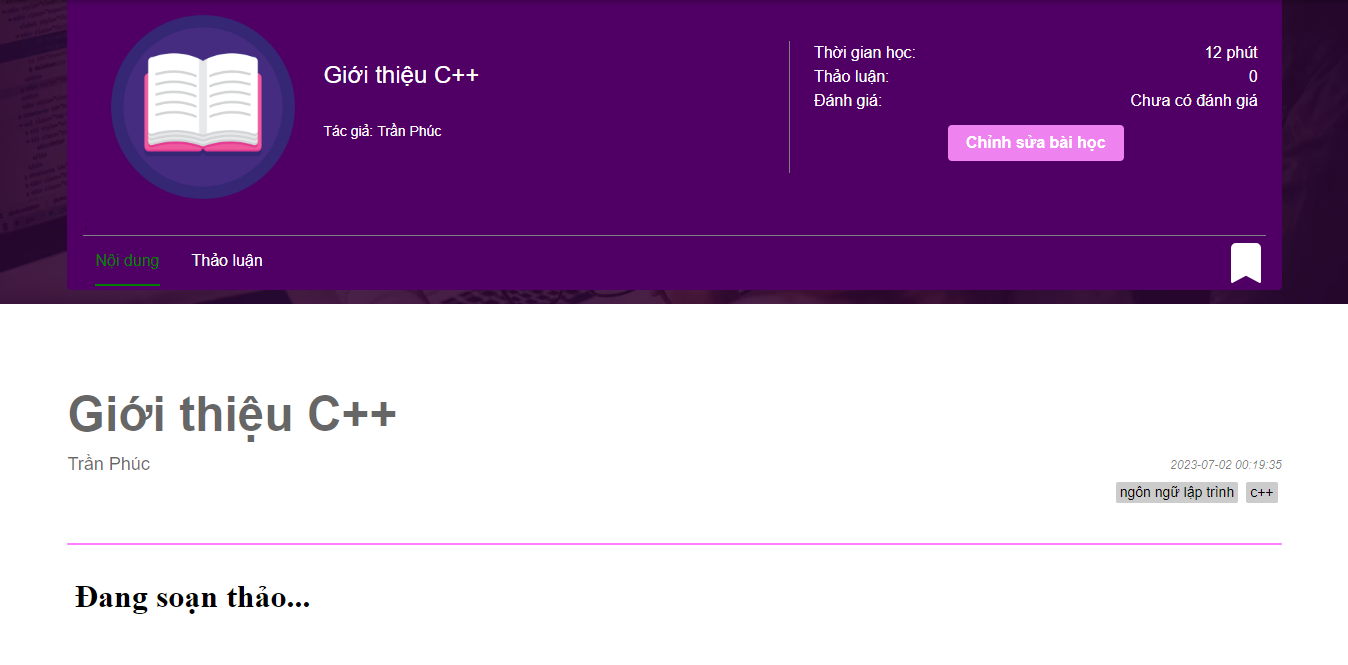
### 

### 4.2.5. Giao diện chi tiết khóa học

### 

### 

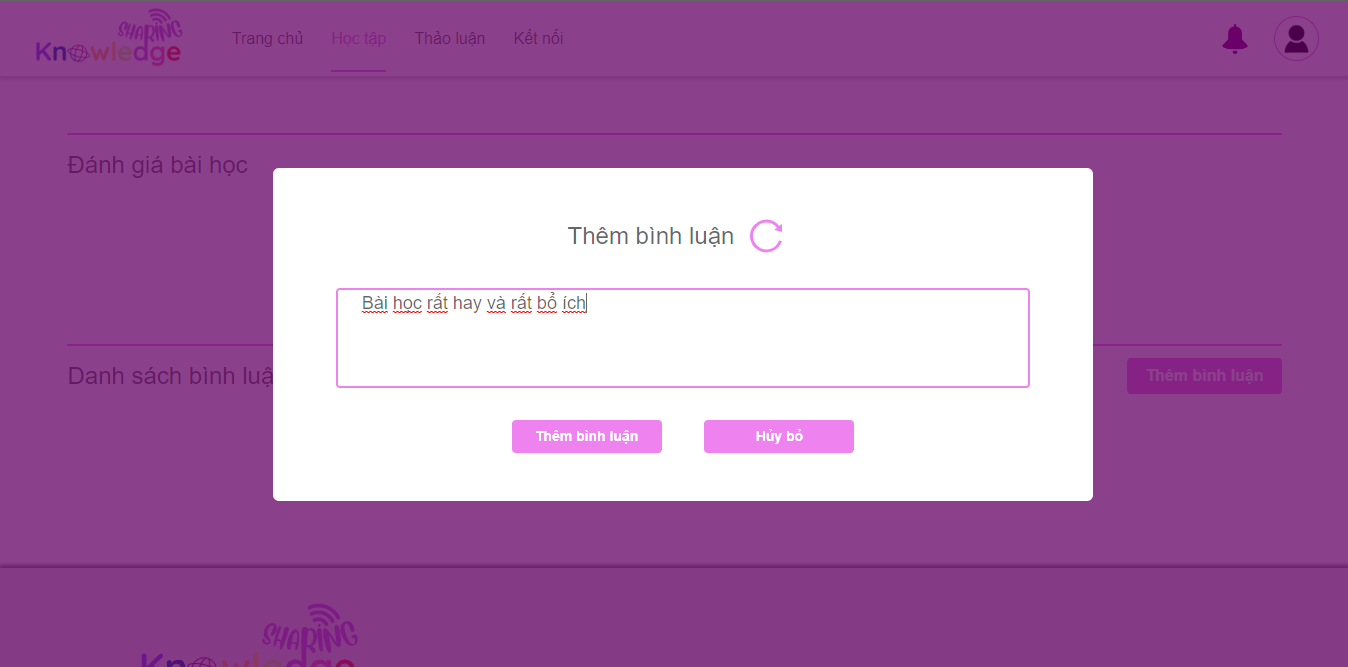
### 4.2.6. Giao diện chi tiết bài học

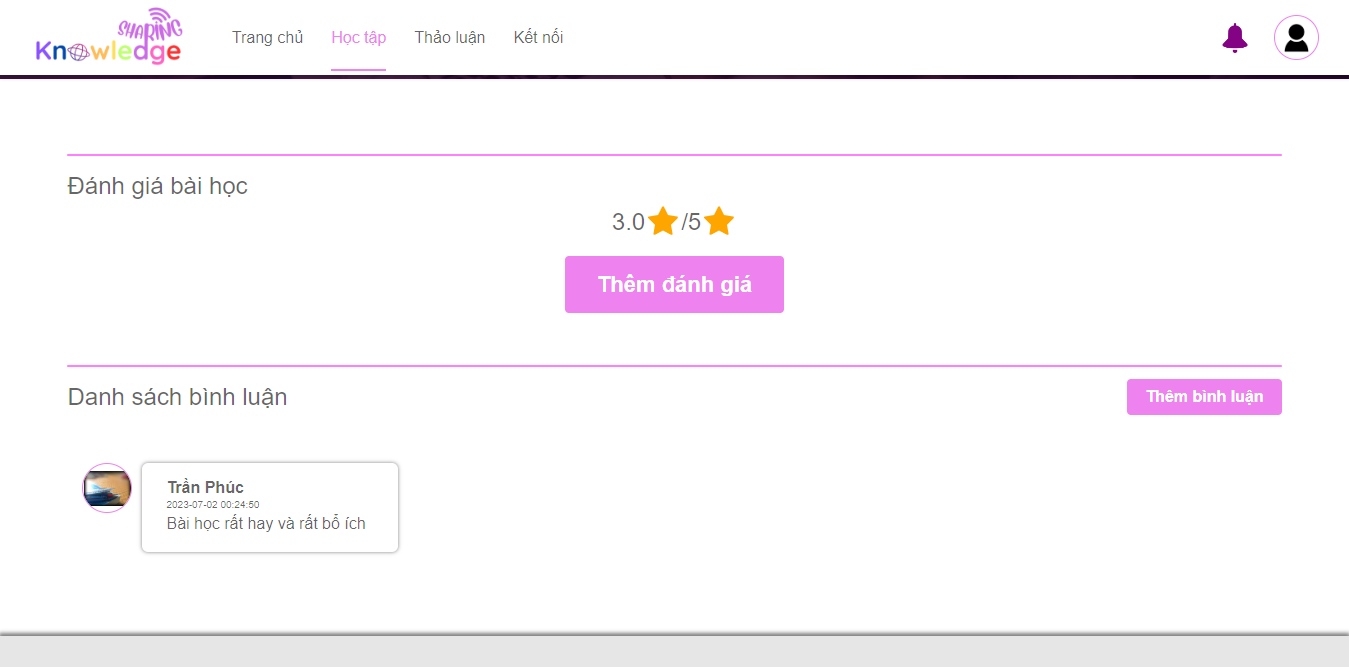


### 

### 4.2.7. Giao diện đánh giá, bình luận

### 





### 4.2.8. Giao diện quản lý bài học trong khóa học

## 

## 

## 

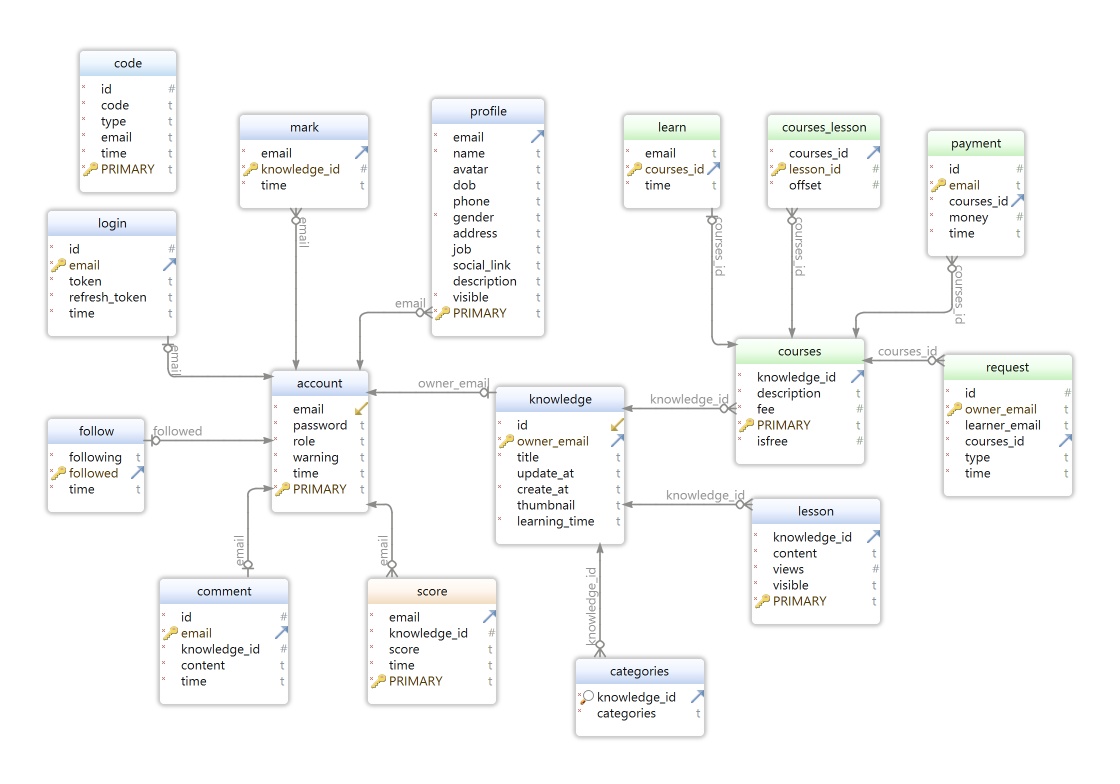
## 4.3. Thiết kế cơ sở dữ liệu

### 4.3.1. Sơ đồ thực thể liên kết:

# 

# 

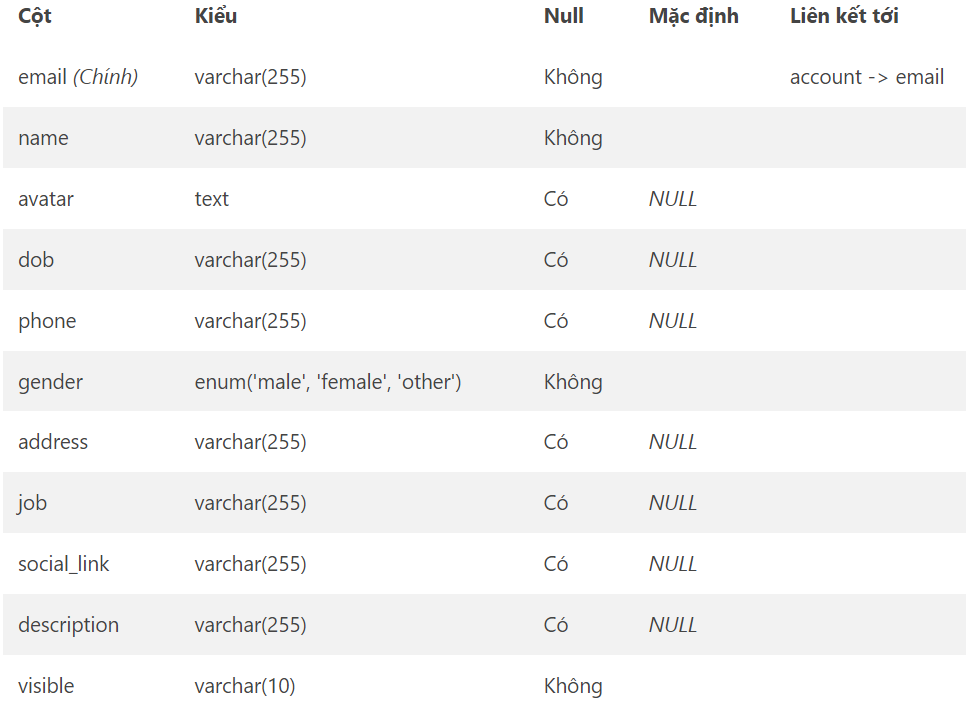
### 4.3.2. Sơ đồ tổng quan cơ sở dữ liệu:



### 

### 4.3.3. Cấu trúc của 1 số bảng trong cơ sở dữ liệu:

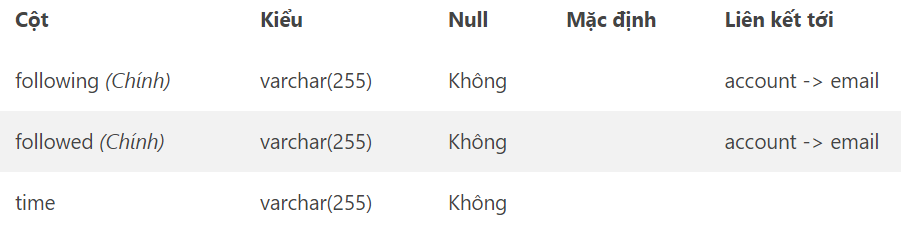
- Bảng profile:



Bảng lưu thông tin về profile với các thông tin cơ bản về một người dùng để người dùng khác có thể hiểu rõ hơn về người dùng này

Trường visible: chuỗi 10 ký tự 0, 1, 2 tương ứng với quyền truy cập của các trường từ trên xuống dưới của bảng

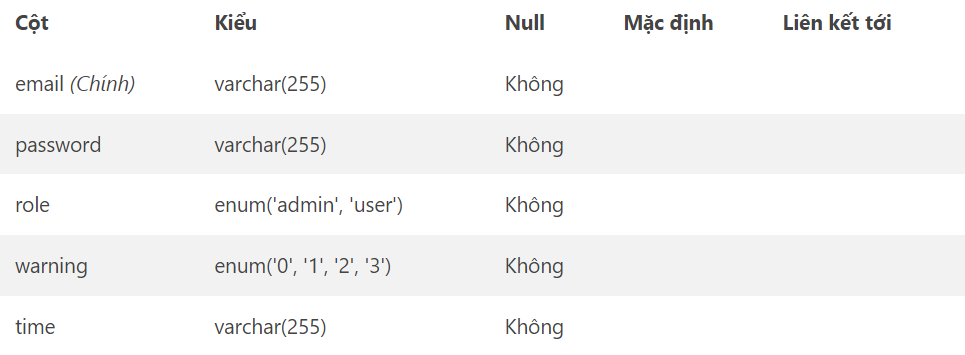
- Bảng theo dõi:



Lưu trữ thông tin theo dõi giữa những người dùng với nhau

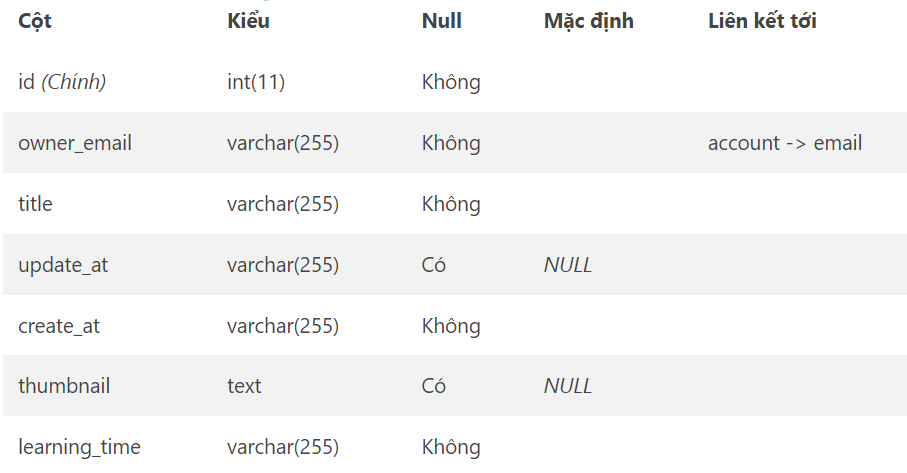
# 

- Bảng account:



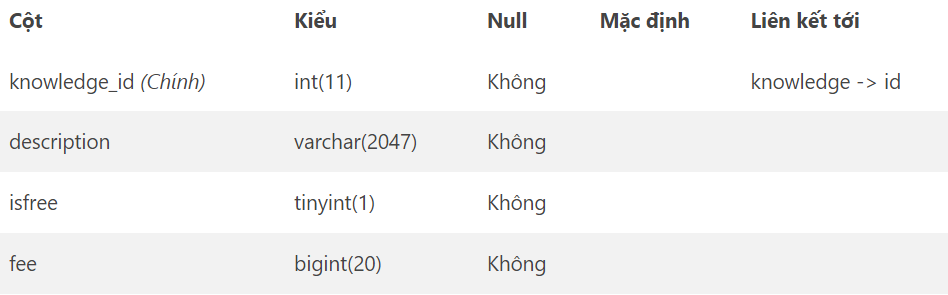
Lưu thông tin về tài khoản của người dùng, vai trò của họ là gì, mức độ cảnh báo ra sao

- Bảng kiến thức:



Đây là bảng kiến thức chung, là cha của các bảng khoá học và bài học, sẽ lưu các thông tin chung mà bất kể kiến thức nào cần có

- Bảng khoá học:



Bảng con của bảng kiến thức, lưu thông tin về một khoá học, giá cả, mô tả trước cho người dùng có cái nhìn tổng quan về khoá học

- Bảng bài học

# 

Bảng con của bảng kiến thức, lưu trữ thông tin về nội dung của một khoá học, đồng thời có thêm trường visible để chủ khoá học có thể tạo các bài học riêng tư hoặc có thể là bài học trong một khoá học có tính phí nào đó

- Bảng mua khoá học:

# 

Lưu thông tin về lịch sử của việc mua bán khoá học của người dùng

- Bảng đánh giá:

# 

Lưu số điểm từ 0 sao đến 5 sao mà mỗi người dùng đã đánh giá cho khoá học, bài học tương ứng.

# 4.4. Thiết kế API giao tiếp giữa frontend và backend

### 4.4.1. Nhóm API chức năng User

| stt | Tên | Url/Method | Input | Output | Chức năng |
| --- | --- | --- | --- | --- | --- |
| 1 | login | Post api/auth/login | email  password | email  token  refreshToken | Xác thực tên tài khoản và mật khẩu, sau đó trả về token và refresh token |
| 2 | validate token | Post api/auth/validateToken | token | profile & account | Xác thực token cho phiên đăng nhập và trả về object gồm các thông tin về account và profile |
| 3 | refresh token | Post api/auth/refreshToken | refreshToken | token | Refresh token khi token đã bị hết hạn |
| 4 | logout | Post api/auth/logout | token |  | Thực hiện xoá phiên đăng nhập |
| 5 | logout all device | Post api/auth/logoutAll | token |  | Thực hiện xoá tất cả các phiên đăng nhập liên quan đến email này |
| 6 | get register code | Post api/auth/getRegisterCode | email |  | Nhận mã code sau khi đã nhập email và password muốn đăng ký |
| 7 | register | Post api/auth/register | email  password  code |  | Xác thực mã code, tạo tài khoản và yêu cầu người dùng đăng nhập |
| 8 | Change password | Post api/auth/changePassword | token  oldPassword  newPassword | email  token  refreshToken | Đổi mật khẩu và chỉ đăng nhập lại ngay người dùng đã đổi mật khẩu |
| 9 | get forgot password code | Post api/auth/getForgotPasswordCode | email |  | Nhận mã code sau khi đã nhập email và ấn quên mật khẩu |
| 10 | forgot password | Post api/auth/forgotPassword | code  email  newPassword |  | Nhập mã code, mật khẩu mới và thực hiện đổi mật khẩu, logout mọi phiên đăng nhập |
| 11 | cancel account | Post api/auth/cancelAccount | token  password |  | Nhập mật khẩu của tài khoản và xác nhận huỷ |
| 12 | update profile | Put api/profile | token  name\*  avatar\*  dob\*  phone\*  gender\*  address\*  job\*  socialLink\*  description\*  visible\* | name  avatar  dob  phone  gender  address  job  socialLink  description  visible  warning | Update profile cho người dùng |
| 13 | get profile | Get api/profile/:email | token\* | name  avatar  dob  phone  gender  address  job  socialLink  description  visible\* (chỉ cho chính mình)  relation  numfollower, numfollowing, numlesson, numcourses,  warning | Get profile của 1 người, nếu là khách (không có token) thì chỉ hiển thị thông tin public, nếu get của chính mình thì phải trả về visible |
| 14 | add follow | Post api/follow | token  followedEmail |  | Follow 1 ai đó |
| 15 | unfollow | Delete api/follow | token  followedEmail |  | Unfollow 1 ai đó |
| 16 | get list follow | Get api/follow | index (0 hoặc 1)  email  myEmail\* (để lấy relation với mình)  offset\* = 0  length\* = 10 | [  email  name  avatar  followers  relation  ] | Get list follow của user  Index=0: get list followers  Index=1: getlist following |

### 4.4.2. Nhóm API chức năng Người học

| stt | Tên | Url/Method | Input | Output | Chức năng |
| --- | --- | --- | --- | --- | --- |
| 1  90% | free courses register | post api/courses/register/:courseid | token |  | Gửi đăng ký tới khóa học miễn phí |
| 2  90% | courses paying | post api/courses/pay/:courseid | token  password  money |  | Gửi thanh toán đăng ký khóa học tính phí |
| 3  90% | courses request | post api/courses/request/:courseid | token |  | Gửi/hủy yêu cầu tham gia khoá học |
| 4  90% | courses cancel register (leave course) | delete api/courses/register/:courseid | token |  | Huỷ đăng ký khoá học (Rời khóa học) |
| 5  90% | courses accept/deny request | delete api/courses/request/:requestid | token  type: 0/1 |  | Chủ khóa học chấp nhận/ từ chối yêu cầu  0: từ chối  1: chấp nhận |
| 6  90% | courses invite | post api/courses/invite/:email/:courseid | token |  | Chủ khóa học gửi/huỷ lời mời tham gia khoá học tới ai đó |
| 7  90% | courses accept/deny invite | delete api/courses/invite/:requestid | token  type: 0/1 |  | Người học chấp nhận/ từ chối lời mời  0: từ chối  1: chấp nhận |
| 8  90% | get list request | get api/courses/request?coursesid\*= | token  offset\* = 0  length\* = 10 | [(id, email, name, avatar, cid, thumbnail, tittle, time)] | Lấy danh sách yêu cầu tham gia khóa học,  courseid = null: lấy danh sách mình chủ động gửi đi |
| 9  90% | get list invite | get api/courses/invite?courseid\*= | token  offset\* = 0  length\* = 10 | [  (id, email, name, avatar, cid, thumbnail, tittle, time)  ] | Lấy danh sách lời mời tham gia khoá học, nếu courseid=null : lấy danh sách lời mời của mình được nhận  không thì lấy danh sách lời mời của 1 khoá học cụ thể |
| 10  90% | get lesson detail | get api/lesson/detail/:lessonid | token\* | id, owner, title, update\_at, create\_at, thumbnail, learning\_time, content, views, visible, score, ismark, nummark, numcmt, categories | get chi tiết bài học, token = 0: get trong vai trò là khách (guest) |
| 11  90% | get list lesson | get api/lesson/list?email\*=email&courseid\*=courseid | token  offset\* = 0  length\* = 10 | [(id, email, name, title, update\_at, create\_at, thumbnail, learning\_time, views, visible\*-chỉ cho mình xem, score, ismark, nummark, numcmt, categories)] | - của mình: email = cid = 0  - của ng khác: cid=0, em != 0  - của 1 khóa học: cid != 0, email = 0  (tip: email và token không đồng thời != 0) |
| 12  90% | score knowledge | put api/knowledge/score/:knid | token  score |  | cập nhật hoặc thêm điểm chấm cho kiến thức |
| 12  90% | comment knowledge | post api/knowledge/comment/:knid | token  content |  | thêm comment cho kiến thức |
| 13  90% | update comment | patch api/knowledge/comment/:commentid | token  newContent |  | cập nhật comment cho kiến thức |
| 13  90% | delete comment | delete api/knowledge/comment/:commentid | token |  | xoá comment cho kiến thức |
| 14  90% | get list comment | get api/knowledge/comment/:knid | offset\* = 0  length\* = 10 | [(id, email, avatar, name, content, time)] | Lấy danh sách bình luận cho một kiến thức |
| 15  90% | set mark knowledge | post api/knowledge/mark/:knid | token  type: 0/1 |  | Lưu/huỷ lưu mục kiến thức |
| 16  90% | get listmark | get api/knowledge/mark?knid\*=knid | token  offset\* = 0  length\* = 10 | [(email, name, avatar, time, knid, thumbnail, title, owner\_email, is\_course)] | Lấy danh sách kiến thức đã lưu, knid != null: lấy danh sách người đánh dấu khóa học đã cho |
| 17  90% | post image | post api/image | token  image: file | url of image | Đăng ảnh lên hệ thống, lưu ra cloud, lấy lại url |

### 4.4.3. Nhóm API chức năng Chủ khóa học

| stt | Tên | Url/Method | Input | Output | Chức năng |
| --- | --- | --- | --- | --- | --- |
| 1  90% | create courses | post api/courses/ | token  title, thumbnail\*, learning\_time, description, isfree, fee, categories | cid, create\_at, update\_at,  title, thumbnail, learning\_time, description, isfree, fee, score, ismark, numark, numcmt, categories, | tạo khoá học |
| 2  90% | update course | patch api/courses/:cid | token, cid,  title\*, thumbnail\*, learning\_time\*, description\*, fee\*, categories\* | cid, create\_at, update\_at,  title, thumbnail, learning\_time, description, isfree, fee, score, ismark, numark, numcmt, categories, | cập nhật khoá học (không có isfree vì không được sửa) |
| 3  90% | delete course | delete api/courses/:cid | token |  | xoá khoá học |
| 4  90% | get list course | get api/courses/list?email= | token\*  offset\* = 0  length\* = 10 | [(cid, thumbnail, owner\_email, name, create\_at, update\_at, title, fee, isfree, categories, nummark, numcmt, numlesson, ismark, score, numlearner, relevant)] | Lấy danh sách khóa học của mình (email = 0) hoặc của người khác,  relevant: 0: not relevant, 1: requesting, 2: invited, 3: registered, 4: owner |
| 4.5  90% | get list registered course | get api/courses/list-registered/ | token  offset\* = 0  length\* = 10 | như trên | Lấy DS các khóa học mà mình đang học |
| 5  90% | get course details | get api/courses/detail/:cid | token\* | cid, thumbnail, email, name, create\_at, update\_at, title, description, fee, isfree, categories, nummark, numcmt, numlesson, ismark, score, numlearner, relevant, list-lesson |  |
| 6  90% | add lesson | post api/lesson | token, title, thumbnail\*, categories, learning\_time, content, visible | id, email, name, title, update\_at, create\_at, thumbnail, learning\_time, content, views, visible, score, ismark, nummark, numcmt, categories |  |
| 7  90% | add lesson to course | post api/courses/lesson/:cid/:lessonid | token,  offset |  |  |
| 8  90% | delete lesson from course | delete api/courses/lesson/:cid/:lessonid | token |  |  |
| 8.5  90% | update lesson in courses | patch api/courses/lesson/:cid/:lessonid | token, offset |  |  |
| 9  90% | update lesson | patch api/lesson/:lessonid | token, title\*, thumbnail\*, categories\*, learning\_time\*, content\*, visible\* |  |  |
| 10  90% | delete lesson | delete api/lesson/:lessonid | token |  |  |
| 11  90% | get course members | get api/courses/members/:cid | token\*  offset\* = 0  length\* = 10 | [(email, name, avatar, time)] |  |

### 4.4.4. Nhóm API chức năng Tìm kiếm

| stt | name | method / url | input | output | chức năng |
| --- | --- | --- | --- | --- | --- |
| 1 | search account | get api/search/account/:key | token\* offset\* = 0  length\* = 10 | [(email, name, avatar, numFollowers,  numFollowing,  numcourses, numlesson, relation)] |  |
| 2 | search courses | get api/search/courses/:key | token\*  categories\*  offset\* = 0  length\* = 10 | [(knowledge\_id, thumbnail, owner\_email, name, create\_at, update\_at, title, fee, isfree, categories, nummark, numcmt, numlesson, ismark, score, numlearner, isregistered)] |  |
| 3 | search lesson | get api/search/lesson/:key | categories\*  offset\* = 0  length\* = 10 | [(id, email, name, title, update\_at, create\_at, thumbnail, learning\_time, views, visible\*-chỉ cho mình xem, score, ismark, nummark, numcmt, categories)] |  |

### 4.4.5. Nhóm API chức năng Admin

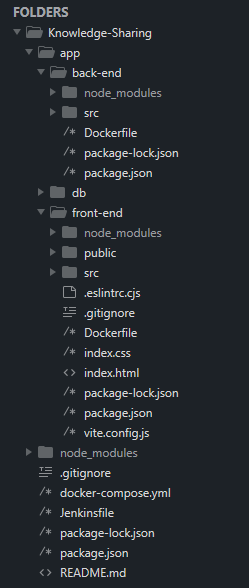
| stt | name | method / url | input | output | chức năng |
| --- | --- | --- | --- | --- | --- |
| 1  90% | limit/pardon acc | patch api/admin/limit | token  emai: email  warning: 0/1/2/3 | email, warning, name, avatar | 0: normal  1: only learner  2: only user  3: consider delete |
| 2  90% | get list acc | get api/admin/account | token  offset\*  length\* | [(email, warning, name, avatar)] |  |
| 3 | delete account | delete api/admin/account | token  email: email |  | only delete account in 3-level-limition, and after 7 days |
| 4  90% | delete courses | delete api/admin/course | token  coursesid: |  |  |
| 5  90% | delete lesson | delete api/admin/lesson | token  lessonid: |  |  |
| 6  90% | add admin | post api/admin/add-admin | token  email  password  name |  |  |

# 

# 5. Cài đặt và triển khai hệ thống

5.1. Triển khai thành các thành phần

Hệ thống được chia làm 3 thành phần chính là Database, Back-end và Front-end. Cấu trúc thư mục của ứng dụng như hình dưới đây:



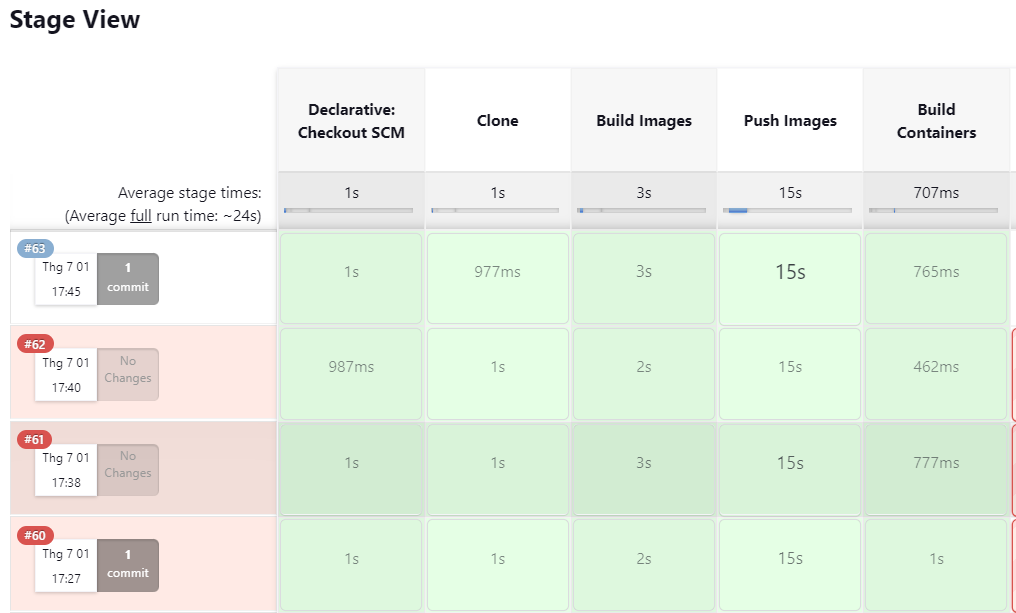
Database được triển khai trên server của db4free.net - 1 dịch vụ cơ sở dữ liệu MySQL miễn phí và là môi trường phù hợp cho học tập và phát triển và thử nghiệm. Back-end của hệ thống sẽ tự động kết nối và thực hiện truy vấn đến cơ sở dữ liệu.

Front-end và Back-end được xây dựng và triển khai thành 2 thư mục chính trong kiến trúc thư mục ứng dụng. Mỗi thành phần Front-end và Back-end sẽ được build thành các container để cùng chạy trên host. Trong mỗi thư mục ứng dụng của mỗi thành phần có một File Dockerfile dùng để biên dịch thành các image. Ở ngoài thư mục gốc của dự án chứa 2 file đặc biệt: file docker-compose.yml dùng để xây dựng và chạy các container của dự án dựa vào các image đã được đẩy lên docker registry trong quá trình thực thi CI/CD pipeline; file Jenkinsfile định nghĩa các stage trong quá trình thực hiện CI/CD.

5.2. Triển khai CI/CD pipeline

Nhóm sẽ triển khai CI/CD của hệ thống trên Jenkins với các bước thực hiện như sau:

* Cài đặt và chạy Jenkins trên ubuntu
* Xây dựng Dockerfile để build cho Back-end và Front-end
* Xây dựng docker-compose cho Back-end và Front-end
* Để đảm bảo với các thay đổi khi có commit code trên nhánh main trên Github, cài đặt webhook trên Github của dự án kết nối tới Jenkins để thực hiện lắng nghe các sự kiện push và pull request, cấu hình Jenkins của dự án, cài đặt các plugins cần thiết, tạo pipeline cho dự án trên jenkins
* Luồng CI/CD: pull code từ nhánh main của dự án về -> thực hiện build images cho Back-end và Front-end -> Push các images lên dockerhub của nhóm -> xây dựng các container từ file docker-compose đã cấu hình cho dự án
* Do chưa tìm được server miễn phí cho hệ thống nên nhóm sẽ thực hiện publish domain front-end của dự án với dịch vụ của Serveo để minh hoạ, cho người dùng sử dụng dự án trên các máy tính từ xa và một máy laptop minh hoạ máy chủ sẽ chạy phần Back-end và Front-end của hệ thống



Ảnh trên giao diện Jenkins pipeline

Để cài đặt và sử dụng hệ thống, người dùng có thể pull images back-end và front-end từ dockerhub (nhungthisope123/frontend và nhungthisope123/backend) của nhóm và tải file docker-compose từ github

(<https://github.com/tranphuc8a/Knowledge-Sharing>)

sau đó docker-compose up và trải nghiệm thử hệ thống

# KẾT LUẬN

Dự án xây dựng trang web chia sẻ kiến thức là một công cụ quan trọng để khuyến khích và hỗ trợ việc chia sẻ kiến thức trong cộng đồng. Trang web này sẽ mang lại lợi ích rất lớn cho người dùng bằng cách cung cấp một nền tảng tương tác và tiếp cận thông tin đa dạng từ nhiều nguồn. Đồng thời, việc phân tích và thiết kế cẩn thận sẽ đảm bảo tính bảo mật và hiệu quả của trang web. Với sự phát triển và định hướng đúng, trang web chia sẻ kiến thức sẽ trở thành một nguồn tài nguyên quý giá và một cộng đồng sáng tạo và học hỏi.

# TÀI LIỆU THAM KHẢO