

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA ĐA PHƯƠNG TIỆN



BÁO CÁO
THỰC TẬP CHUYÊN SÂU

TÊN ĐỀ TÀI

Thiết kế Backend cho hệ thống quản lý ký túc xá

Đơn vị thực tập	: Viện CDiT
Người hướng dẫn	: Trần Minh Hiếu
Giảng viên hướng dẫn	: Nguyễn Thị Thanh Tâm
Sinh viên thực hiện	: Tô Hải Đăng

Hà Nội, 2025

Mục lục

1 Giới thiệu đơn vị thực tập	4
1.1 Giới thiệu chung về CDIT	4
1.2 Chức năng, nhiệm vụ	4
1.2.1 Đào tạo và phát triển nhân lực	4
1.2.2 Nghiên cứu khoa học và phát triển công nghệ	4
1.2.3 Chuyển giao và tư vấn công nghệ	4
1.2.4 Hợp tác quốc tế và trong nước	4
1.3 Cơ cấu tổ chức	5
1.4 Các lĩnh vực hoạt động	5
2 Giới thiệu hệ thống	7
2.1 Mục tiêu hệ thống:.....	7
2.2 Phạm vi hệ thống:.....	7
2.3 Tổng quan chức năng:.....	8
3 Phân tích thiết kế hệ thống.....	9
3.1 Sơ đồ UseCase.....	9
3.1.1 Quản lý người dùng	9
3.1.2 Quản lý thông tin sinh viên.....	10
3.1.3 Quản lý dữ liệu ký túc xá.....	10
3.2 Phân tích nghiệp vụ	11
4 Thiết kế hệ thống	12
4.1 Kiến trúc tổng quan	12
4.2 Công nghệ & công cụ	13
4.3 Thiết kế cơ sở dữ liệu	15
4.3.1 Sơ đồ lớp thực thể.....	15
4.3.2 Kết quả cơ sở dữ liệu	16
4.4 Lưu đồ tuần tự	17
4.4.1 Quản lý người dùng	19
4.4.2 Quản lý hợp đồng	22
5 Kết luận	23

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn đến Khoa Đa phương tiện của Học Viện Công nghệ Bưu Chính Viễn Thông đã tạo điều kiện cho chúng em có cơ hội thực tập và học hỏi tại Trung tâm Công nghệ Thông tin CDIT. Nhờ có thời gian thực tập tại đây mà em đã học hỏi được nhiều điều cũng như đã trưởng thành hơn.

Đặc biệt, em xin chân thành cảm ơn cô Nguyễn Thị Thanh Tâm đã luôn tận tình giảng dạy, hướng dẫn và hỗ trợ chúng em trong suốt quá trình học tập và thực tập. Những kiến thức và kỹ năng mà cô truyền đạt đã giúp em rất nhiều trong việc hoàn thành tốt đợt thực tập này.

Em cũng xin gửi lời cảm ơn chân thành đến thầy Hoàng, anh Trần Minh Hiếu thuộc CDIT. Sự hướng dẫn, hỗ trợ và chia sẻ kinh nghiệm quý báu của thầy và anh đã giúp em không chỉ hoàn thành tốt nhiệm vụ được giao mà còn hiểu thêm về môi trường làm việc thực tế, tích lũy được nhiều kiến thức và kỹ năng mới. Một lần nữa, em xin chân thành cảm ơn và kính chúc cô Tâm cùng Viện CDIT sức khỏe, hạnh phúc và thành công.

Hà Nội, tháng 5 năm 2025

1 Giới thiệu đơn vị thực tập

1.1 Giới thiệu chung về CDIT

Viện công nghệ Thông tin và Truyền thông CDIT là tổ chức đào tạo, nghiên cứu trực thuộc Học viện Công nghệ Bưu chính Viễn thông, có nhiệm vụ: đào tạo, nghiên cứu khoa học và chuyển giao công nghệ trong lĩnh vực Công nghệ thông tin và Truyền thông.

1.2 Chức năng, nhiệm vụ

1.2.1 Đào tạo và phát triển nhân lực

- Đào tạo các trình độ từ đại học đến sau đại học (thạc sĩ, tiến sĩ) trong lĩnh vực Công nghệ Thông tin và Truyền thông.
- Đào tạo chuyên gia, bồi dưỡng nâng cao nghiệp vụ và huấn luyện nhân lực theo nhu cầu xã hội.

1.2.2 Nghiên cứu khoa học và phát triển công nghệ

- Nghiên cứu, phát triển và ứng dụng khoa học kỹ thuật.
- Thực hiện sản xuất thử nghiệm các sản phẩm phần cứng, phần mềm, giải pháp công nghệ thông tin và truyền thông.

1.2.3 Chuyển giao và tư vấn công nghệ

- Tư vấn và chuyển giao công nghệ, cung cấp các dịch vụ khoa học kỹ thuật.

1.2.4 Hợp tác quốc tế và trong nước

- Thực hiện các hoạt động hợp tác trong đào tạo, nghiên cứu và chuyển giao công nghệ với các đối tác trong và ngoài nước theo quy định của pháp luật và Học viện Công nghệ BCVT.

1.3 Cơ cấu tổ chức



1.4 Các lĩnh vực hoạt động

1. Nghiên cứu khoa học công nghệ.
 - CDiT tập trung nghiên cứu các vấn đề lý luận cơ bản, nghiên cứu đón đầu và nghiên cứu ứng dụng trong lĩnh vực công nghệ thông tin và truyền thông. Các lĩnh vực nghiên cứu chính bao gồm:
 - **Công nghệ Thông tin và Truyền thông (ICT):** Nghiên cứu các công nghệ truyền thông, mạng lưới, cơ sở dữ liệu, lập trình, trí tuệ nhân tạo và các ứng dụng của công nghệ thông tin.
 - **Đa phương tiện (Multimedia):** Nghiên cứu các công nghệ xử lý hình ảnh, âm thanh, video, thực tế ảo và thực tế tăng cường, các ứng dụng của đa phương tiện trong giáo dục, giải trí, truyền thông.
 - **An ninh và An toàn thông tin (Information Security):** Nghiên cứu các biện pháp bảo vệ thông tin khỏi các cuộc tấn công, bảo mật dữ liệu, bảo vệ hệ thống mạng, quản lý rủi ro an ninh thông tin.
 - 2. Đào tạo và bồi dưỡng nhân lực.
 - CDiT đào tạo các ngành công nghệ thông tin, kỹ thuật phần mềm, kỹ thuật mạng, an ninh mạng, đa phương tiện, kỹ thuật điện tử và viễn thông. Bên cạnh đó, CDiT còn tổ chức các khóa

bồi dưỡng ngắn hạn và dài hạn cho cán bộ, công chức và người lao động trong lĩnh vực công nghệ thông tin.

3. Phát triển, triển khai công nghệ và sản phẩm.
 - CDIT phát triển các sản phẩm phần mềm, ứng dụng di động, hệ thống quản lý, hệ thống thông tin và các giải pháp công nghệ khác cho các doanh nghiệp và tổ chức.
4. Sản xuất phần mềm và thiết bị.
 - CDiT sản xuất các sản phẩm phần mềm và thiết bị công nghệ thông tin, bao gồm hệ thống phần mềm quản lý, phần mềm ứng dụng, thiết bị mạng, thiết bị viễn thông.
5. Tiếp nhận và chuyển giao công nghệ.
 - CDiT tiếp nhận các công nghệ mới từ các đối tác trong và ngoài nước, sau đó nghiên cứu và chuyển giao cho các doanh nghiệp, tổ chức và cộng đồng.

2 Giới thiệu hệ thống

- Hệ thống quản lý ký túc xá được xây dựng nhằm mục tiêu hỗ trợ ban quản lý và sinh viên trong việc quản lý và vận hành ký túc xá một cách hiệu quả và minh bạch. Cụ thể:

2.1 Mục tiêu hệ thống:

- **Tự động hóa quy trình quản lý:** Giảm thiểu thao tác thủ công như đăng ký phòng, theo dõi lịch trực, ghi nhận sự cố và báo cáo.
- **Quản lý dữ liệu:** Lưu trữ và quản lý tập trung toàn bộ thông tin về sinh viên, phòng ở, khu vực, thiết bị, ca trực và các báo cáo phát sinh.
- **Nâng cao hiệu suất làm việc:** Giúp cán bộ quản lý nhanh chóng tra cứu, cập nhật thông tin và thực hiện thống kê, báo cáo theo yêu cầu.
- **Hỗ trợ sinh viên:** Sinh viên có thể tra cứu thông tin, đăng ký phòng, đăng ký, gia hạn hợp đồng xem các báo cáo của sinh viên.

2.2 Phạm vi hệ thống:

- Hệ thống được xây dựng nhằm phục vụ cho công tác quản lý và vận hành ký túc xá, trong đó tập trung vào các nghiệp vụ diễn ra thường xuyên như: quản lý sinh viên, quản lý hợp đồng, phân phòng, trực, báo cáo ca trực
- Đối tượng sử dụng:
 - **Admin (quản trị viên):**
 - Quản lý toàn bộ hệ thống và người dùng.
 - **Quản lý Ký túc xá:**
 - Quản lý phòng, lịch trực, hợp đồng của ký túc xá.
 - **Cán bộ trực:**
 - Thực hiện điểm danh theo ca trực.
 - Gửi báo cáo về ca trực
 - **Sinh viên:**
 - Xem thông tin cá nhân, hợp đồng, các báo cáo về sinh viên.
- Phạm vi chức năng:
 - **Xác thực và phân quyền người dùng:** Đăng nhập, đăng xuất, cấp token truy cập (JWT).

- **Quản lý dữ liệu:** Sinh viên, người dùng, phòng ở, thiết bị, ca trực.
- **Điểm danh và báo cáo ca trực:** Ghi nhận trạng thái có mặt, lưu báo cáo kèm ảnh minh họa.
- **Xử lý file upload:** Lưu trữ hình ảnh báo cáo hoặc thông tin sinh viên.
- **API RESTful:** Cung cấp API cho frontend hoặc hệ thống khác có thể tương tác.
- **Bảo mật và vận hành hệ thống:** Bao gồm xác thực, log, chống truy cập trái phép.
- Công nghệ sử dụng:
 - **Ngôn ngữ lập trình:** JavaScript
 - **Framework:** Express.js
 - **Cơ sở dữ liệu:** MySQL, sử dụng ORM(Sequelize)
 - **Xác thực:** JSON Web Token (JWT).
 - **Bảo mật:** bcrypt (mã hóa mật khẩu), CORS

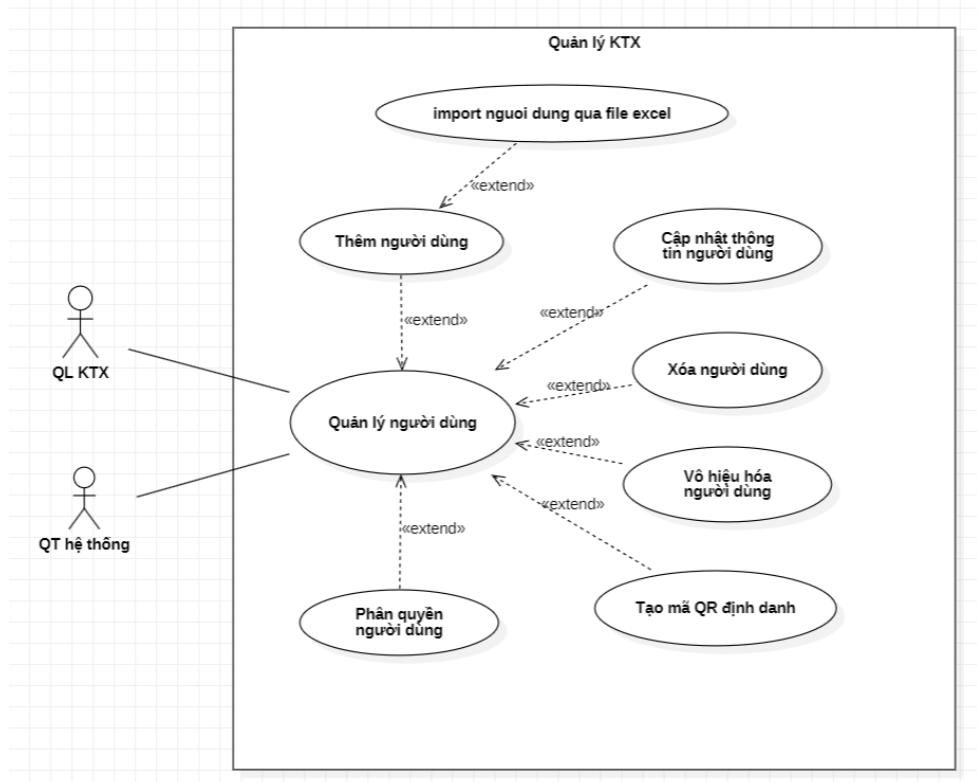
2.3 Tổng quan chức năng:

- Quản lý người dùng
 - Đăng ký, đăng nhập, phân quyền (Quản trị hệ thống, Quản lý KTX, Trực ca, Sinh viên)
 - Quản lý thông tin cá nhân
 - Quản lý quyền người dùng
- Quản lý sinh viên:
 - Đăng ký nội trú, quản lý danh sách sinh viên
 - Theo dõi lịch sử ở, chuyển phòng, hợp đồng.
 - Ghi nhận lỗi và theo dõi lỗi vi phạm
- Quản lý khu vực và phòng Ở:
 - Quản lý khu vực, tòa nhà, tầng, phòng
 - Quản lý chi tiết thông tin phòng

3 Phân tích thiết kế hệ thống

3.1 Sơ đồ UseCase

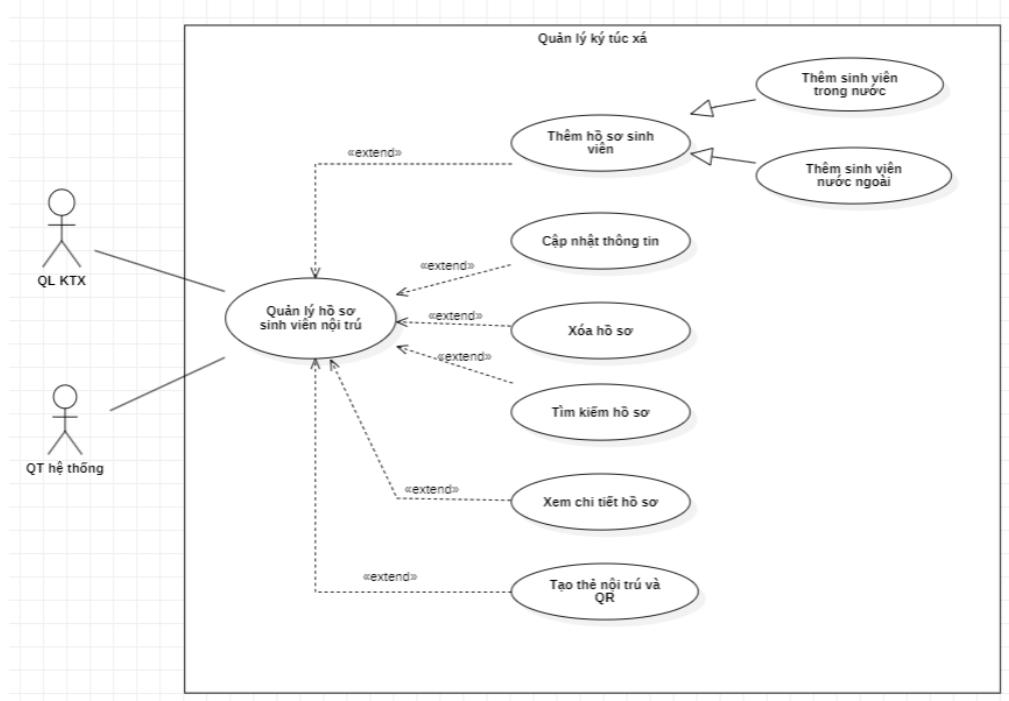
3.1.1 Quản lý người dùng



Mô tả use case

- Thêm người dùng: Cho phép nhập thủ công người dùng mới vào hệ thống.
- Import người dùng qua file Excel: Cho phép thêm nhiều người dùng từ file Excel.
- Cập nhật thông tin người dùng: Cho phép sửa đổi thông tin cá nhân người dùng.
- Xoá người dùng: Cho phép xoá hoàn toàn tài khoản khỏi hệ thống.
- Vô hiệu hoá người dùng: Cho phép khoá tài khoản, không cho đăng nhập hệ thống.
- Tạo mã QR định danh: Tạo mã QR cá nhân phục vụ điểm danh hoặc nhận diện.
- Phân quyền người dùng: Cho phép gán vai trò cho người dùng

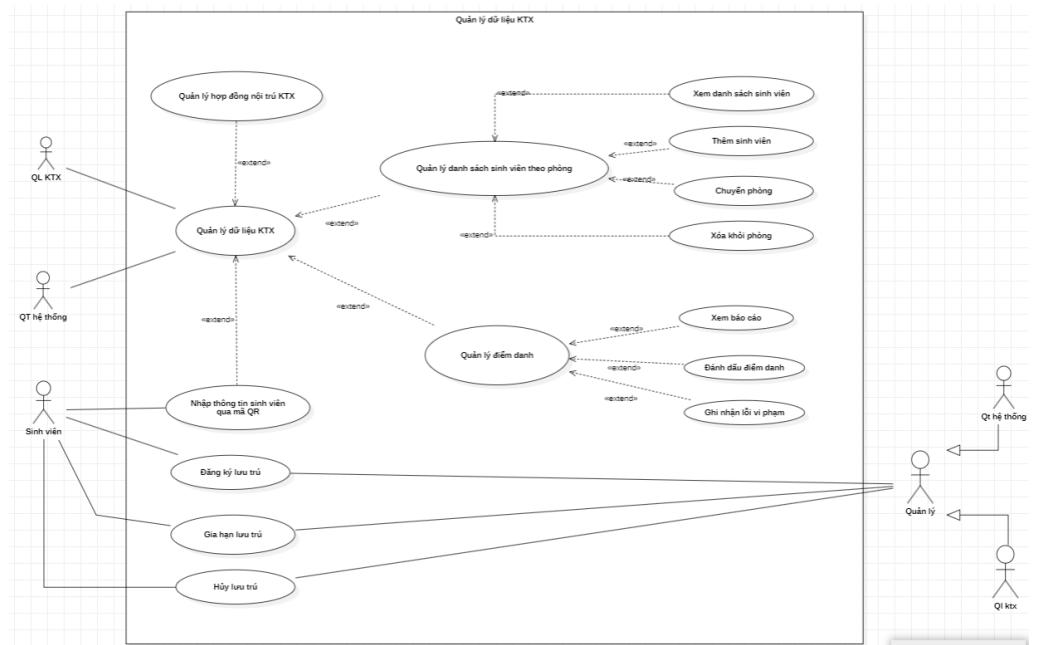
3.1.2 Quản lý thông tin sinh viên



Mô tả use case

- Thêm hồ sơ sinh viên: Cho phép tạo mới hồ sơ sinh viên.
- Cập nhật thông tin: Cho phép sửa thông tin sinh viên.
- Xoá hồ sơ: Cho phép xoá hồ sơ sinh viên khi cần.
- Tìm kiếm hồ sơ: Cho phép tra cứu hồ sơ sinh viên.
- Xem chi tiết hồ sơ: Cho phép xem thông tin sinh viên.

3.1.3 Quản lý dữ liệu ký túc xá



Mô tả use case

- Quản lý dữ liệu KTX: Cho phép quản lý hợp đồng, danh sách sinh viên theo phòng, điểm danh,...
- Quản lý hợp đồng nội trú KTX: Cho phép tạo, xem và lưu trữ hợp đồng ký túc xá.
- Quản lý danh sách sinh viên theo phòng: Xem, thêm, xoá hoặc chuyển phòng cho sinh viên.
- Quản lý điểm danh: Cho phép đánh dấu sinh viên vắng/mặt, tổng hợp báo cáo, ghi nhận lỗi vi phạm
- Nhập thông tin sinh viên qua mã QR: Quét mã QR để điều nhanh thông tin.
- Đăng ký lưu trú: Cho phép tạo đơn đăng ký ở ký túc xá.
- Gia hạn lưu trú: Cho phép tạo đơn gia hạn thêm thời gian ở.
- Huỷ lưu trú: Cho phép tạo đơn kết thúc hợp đồng ở.

3.2 Phân tích nghiệp vụ

- Quản lý người dùng:
 - Hệ thống quản lý thông tin người dùng(Quản trị, Quản lý ký túc xá, Cán bộ trực, Sinh viên)
 - Chức năng chính:
 - Cấp tài khoản cho người dùng
 - Thêm,sửa,xoá người dùng
 - Phân quyền người dùng
- Quản lý sinh viên:
 - Quản lý thông tin liên quan đến sinh viên
 - Chức năng:
 - Sửa, xoá thông tin sinh viên
 - Nhập thông tin sinh viên thông qua file excel
- Quản lý hợp đồng:
 - Quản lý thông tin hợp đồng KTX
 - Chức năng:
 - Tạo mới hợp đồng (Đăng ký , gia hạn , hủy)
 - Cập nhật hợp đồng
 - Xem thông tin hợp đồng
- Quản lý vai trò người dùng:
 - Chức năng:
 - Thêm, sửa xoá vai trò người dùng

- Quản lý lỗi vi phạm của sinh viên:
 - Quản lý lỗi vi phạm trong ca trực
 - Chức năng:
 - Tạo lỗi vi phạm
 - Lấy danh sách lỗi vi phạm theo sinh viên

4 Thiết kế hệ thống

4.1 Kiến trúc tổng quan

Hệ thống quản lý ký túc xá được tổ chức theo kiến trúc **RESTful API** theo hướng phân tầng (MVC + Service Layer) giúp tách biệt rõ ràng trách nhiệm giữa các thành phần. Cấu trúc thư mục thể hiện như sau:

```
ptit_dormitory_be
├── config
├── constants
├── controller
├── middleware
├── models
├── node_modules
├── routes
├── services
├── swagger
├── templates
└── upload
    ├── .env
    └── .gitignore
    └── .prettierrc
```

- controller/ : Xử lý logic điều hướng yêu cầu từ client, gọi đến services

- services/ : Chứa logic xử lý nghiệp vụ chính, tương tác với models và DB
- models/ : Khai báo cấu trúc bảng và quan hệ giữa các bảng trong CSDL (Sequelize)
- routes/ : Định nghĩa các endpoint API và ánh xạ đến các hàm controller
- config/ : Cấu hình hệ thống như Sequelize (CSDL), môi trường,...
- middleware/ : Middleware như xác thực JWT, xử lý lỗi,...
- constants/ : Lưu các hằng số cố định như enum, role, trạng thái,...
- utils/ : Hàm tiện ích (helper) dùng lại nhiều nơi
- upload/ : Quản lý file upload từ người dùng (hợp đồng, ảnh,...)

4.2 Công nghệ & công cụ

Hệ thống backend được xây dựng dựa trên các công nghệ mã nguồn mở phổ biến, hiện đại nhằm đảm bảo hiệu suất, khả năng mở rộng và dễ dàng phát triển. Các công nghệ chính bao gồm:

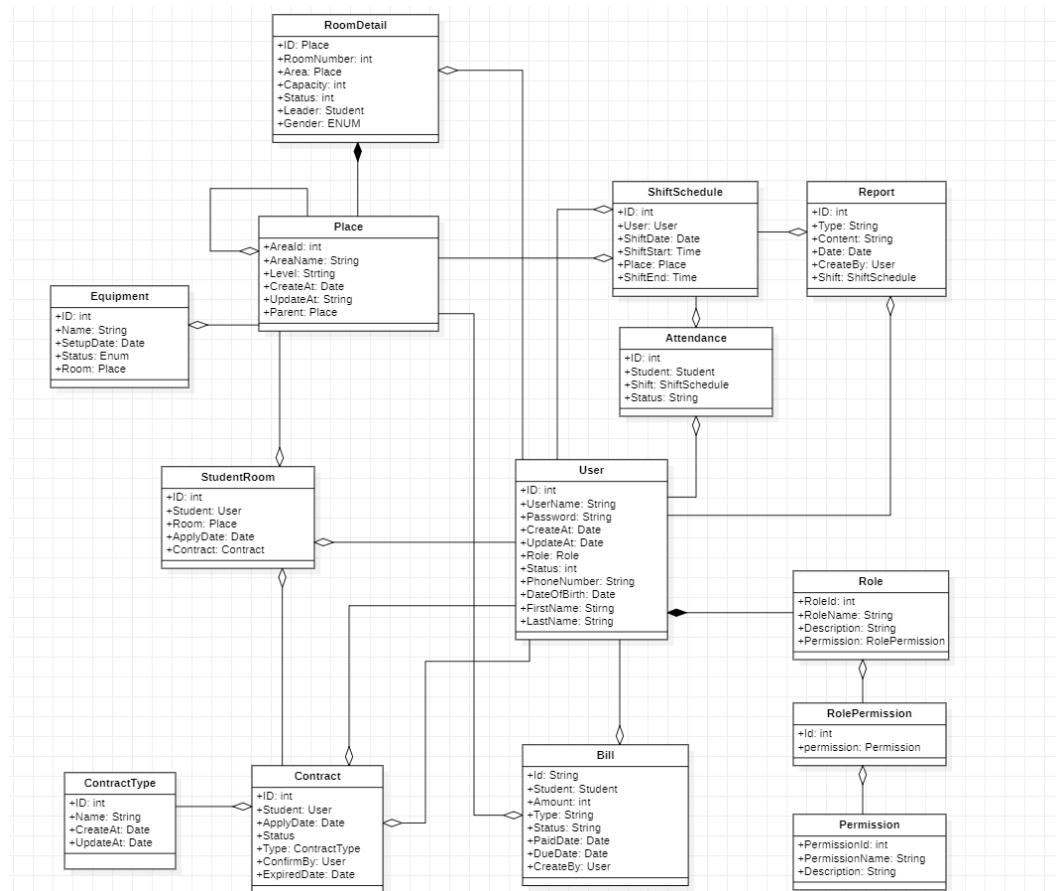
- 1) Node.js: là một môi trường chạy JavaScript phía server, cho phép xây dựng các ứng dụng web hiệu năng cao, xử lý bất đồng bộ và khả năng mở rộng tốt. Node.js được sử dụng làm nền tảng cho backend của hệ thống.
- 2) Express.js: một framework web nhẹ và linh hoạt cho Node.js, cung cấp các công cụ để xây dựng API và xử lý yêu cầu HTTP. Với cú pháp đơn giản, Express.js hỗ trợ định tuyến, middleware và tích hợp dễ dàng với các công cụ khác, giúp phát triển ứng dụng web nhanh chóng và hiệu quả.
- 3) Sequelize ORM: là một ORM (Object-Relational Mapping) cho Node.js, dùng để làm việc với các cơ sở dữ liệu quan hệ như MySQL. Sequelize cung cấp cách truy vấn cơ sở dữ liệu bằng cú pháp JavaScript, hỗ trợ ánh xạ đối tượng, quản lý quan hệ và tạo truy vấn phức tạp mà không cần viết SQL thủ công, giúp tăng tốc độ phát triển và bảo trì..
- 4) MySQL: MySQL là hệ quản trị cơ sở dữ liệu quan hệ, lưu trữ toàn bộ dữ liệu của hệ thống như thông tin sinh viên, phòng ở, lịch trực, báo cáo sự cố... MySQL đảm bảo lưu trữ an toàn, truy vấn nhanh và hỗ trợ các thao tác phức tạp.
- 5) Visual Studio Code : là trình soạn thảo được sử dụng trong dự án. Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn (source code editor) miễn phí, nhẹ, mượt mà.

nguồn mở do Microsoft phát triển. VS Code rất phổ biến trong giới lập trình vì sự linh hoạt, tốc độ, và hệ sinh thái mở rộng mạnh mẽ

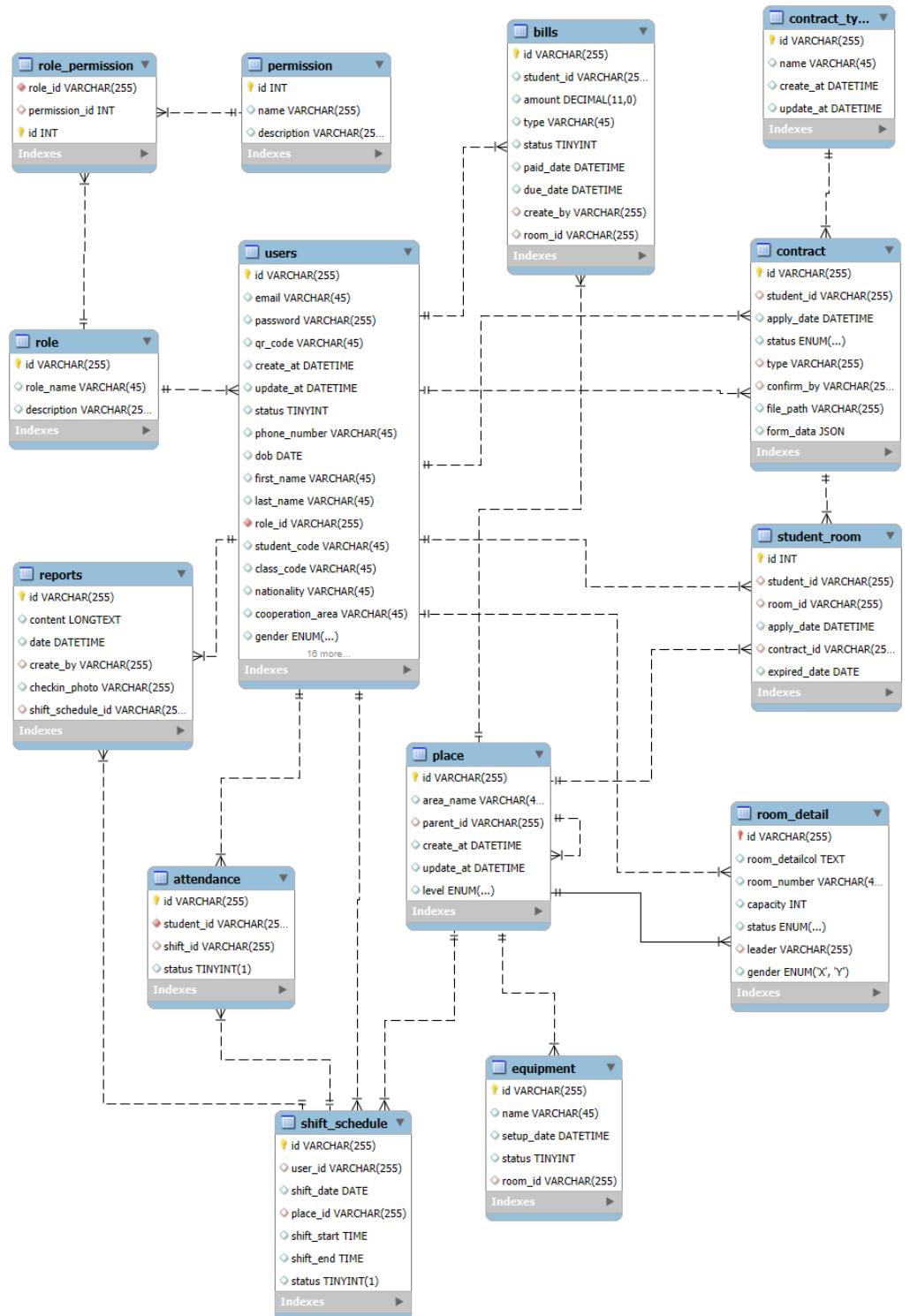
Việc sử dụng kết hợp Node.js, Express.js, Sequelize và MySQL để xây dựng backend mang lại nhiều ưu điểm rõ rệt. Node.js với cơ chế non-blocking I/O giúp xử lý tốt các tác vụ bất đồng bộ và phù hợp với các chức năng real-time như giám sát cảm biến hay điều khiển thiết bị. Express.js là framework đơn giản nhưng mạnh mẽ, cho phép xây dựng hệ thống API RESTful nhanh chóng và dễ mở rộng, rất phù hợp với các chức năng phức tạp như quản lý sinh viên, phòng ờ, hợp đồng hay cảnh báo. Sequelize đóng vai trò là ORM giúp thao tác cơ sở dữ liệu một cách dễ dàng và rõ ràng thông qua JavaScript, đồng thời hỗ trợ các quan hệ dữ liệu chặt chẽ giữa các bảng như sinh viên, phòng, hợp đồng,... một cách hiệu quả. MySQL là hệ quản trị cơ sở dữ liệu quan hệ phổ biến, ổn định và có khả năng xử lý tốt các dữ liệu có cấu trúc rõ ràng – rất phù hợp với bài toán quản lý ký túc xá. Ngoài ra, việc sử dụng chung ngôn ngữ JavaScript cho cả backend và frontend giúp tăng tính đồng bộ, dễ học, dễ bảo trì và phát triển về sau.

4.3 Thiết kế cơ sở dữ liệu

4.3.1 Sơ đồ lớp thực thể

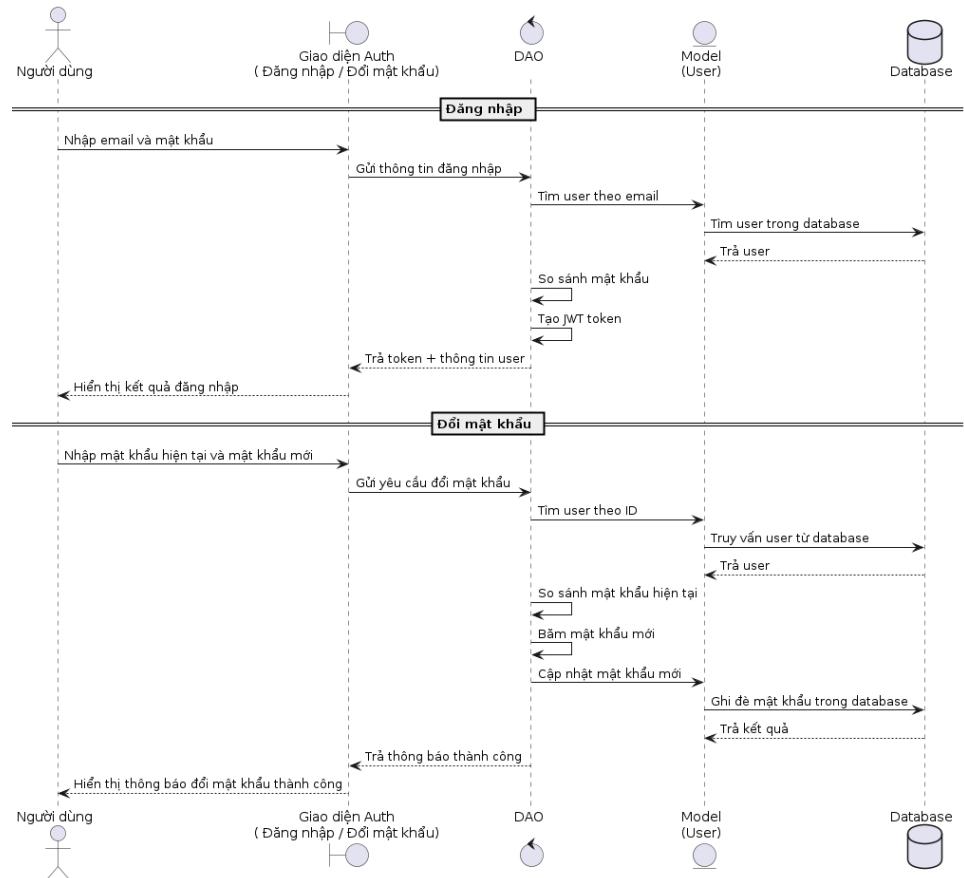


4.3.2 Kết quả cơ sở dữ liệu



4.4 Lưu đồ tuần tự

- Đăng nhập, quyền Đăng nhập & Đổi mật khẩu

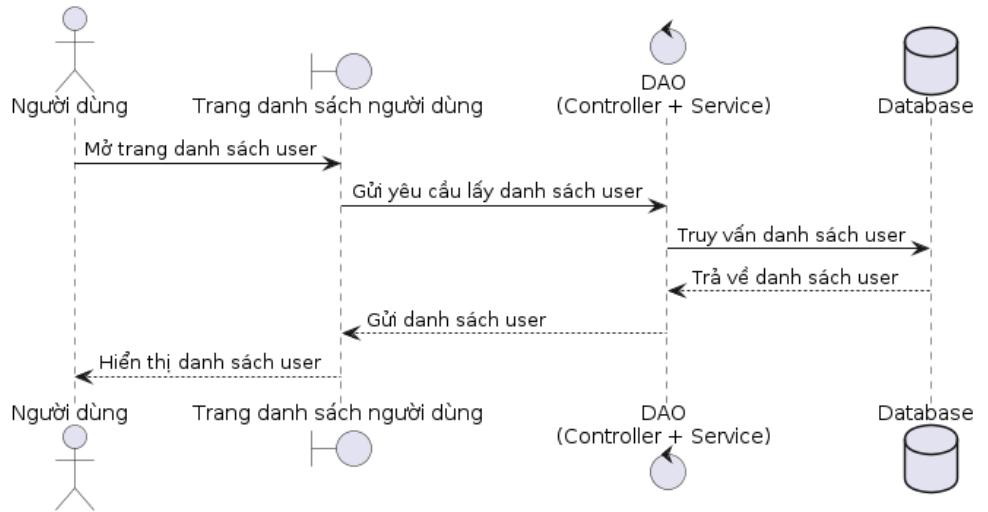


- Quản lý vai trò

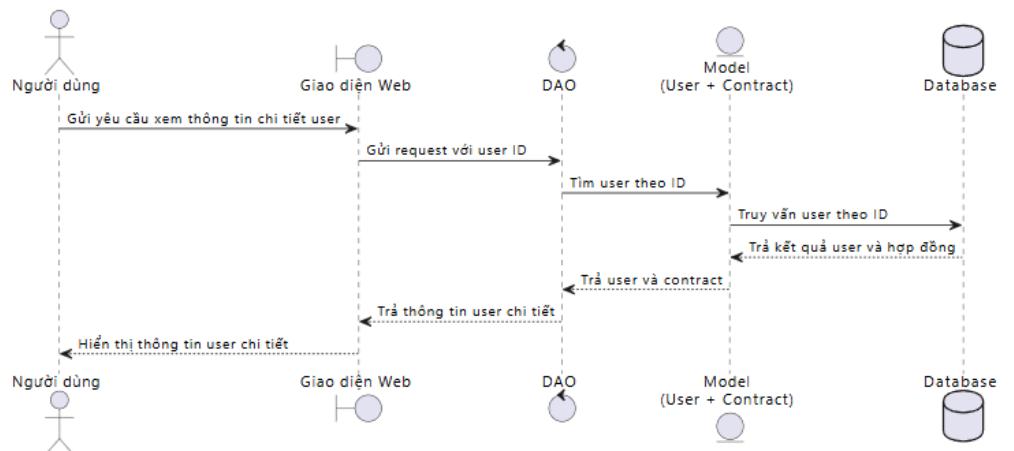


4.4.1 Quản lý người dùng

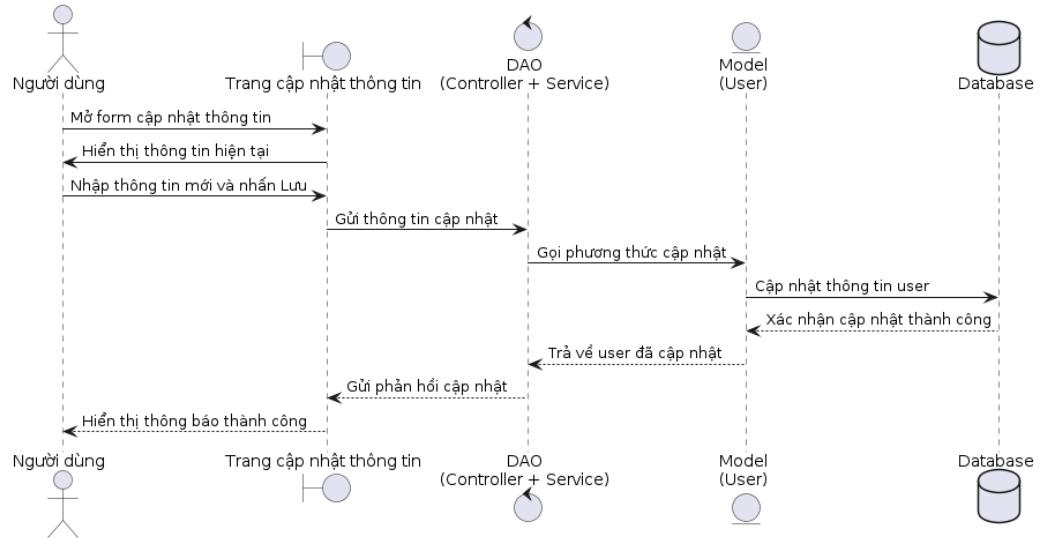
- Lấy danh sách người dùng



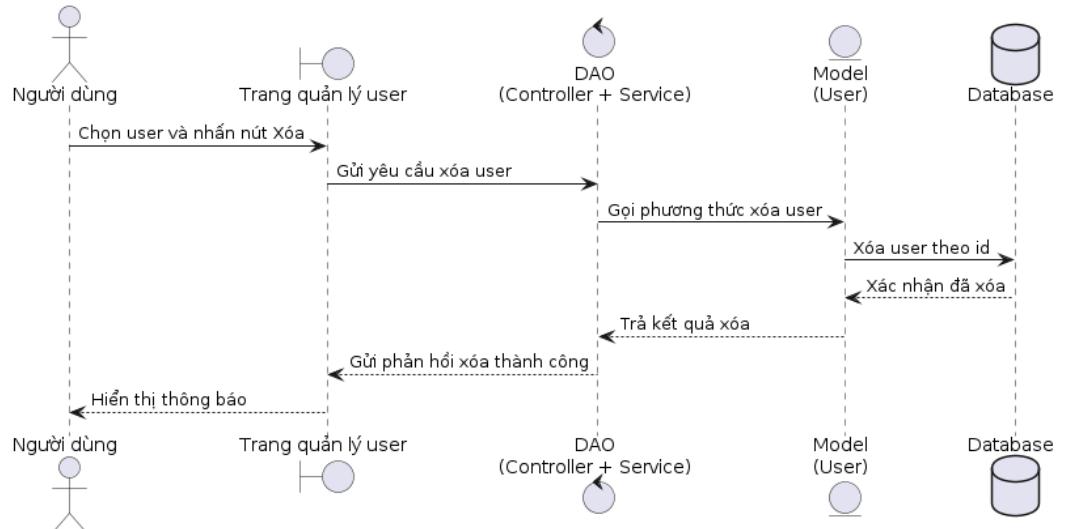
- Lấy ra chi tiết



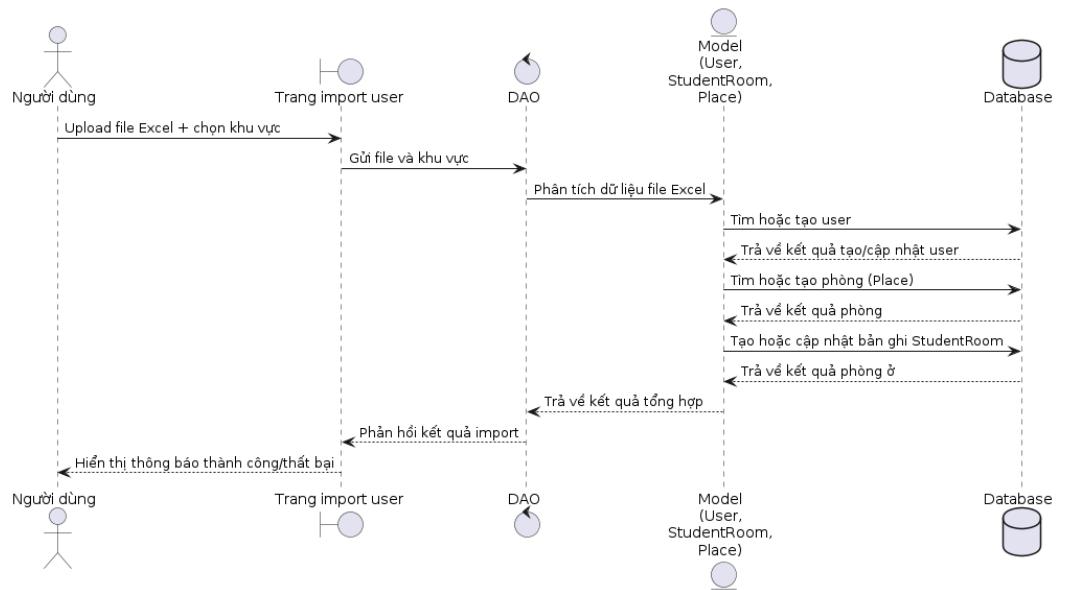
- Cập nhật thông tin người dùng



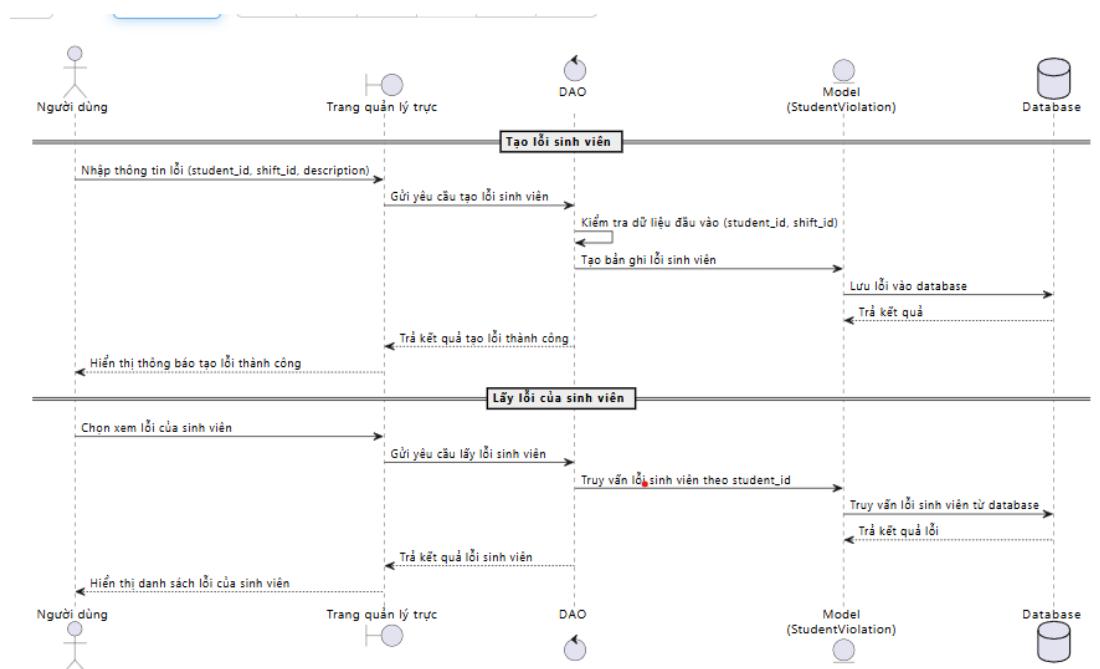
- Xóa người dùng



- Import dữ liệu sinh viên từ file excel

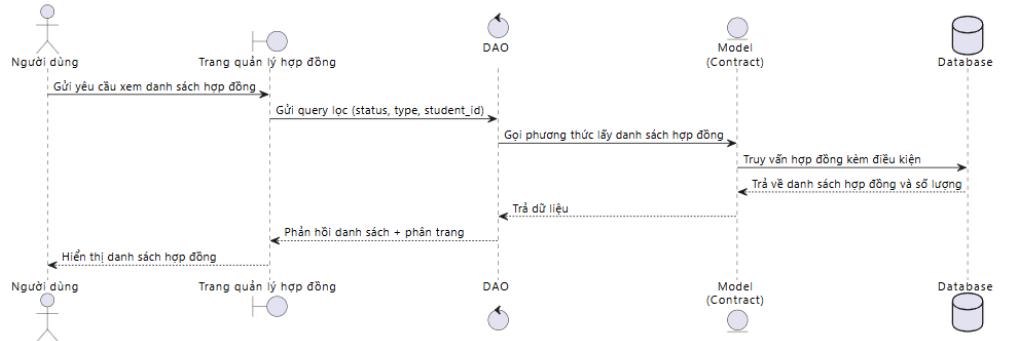


- Quản lý lỗi vi phạm của sinh viên

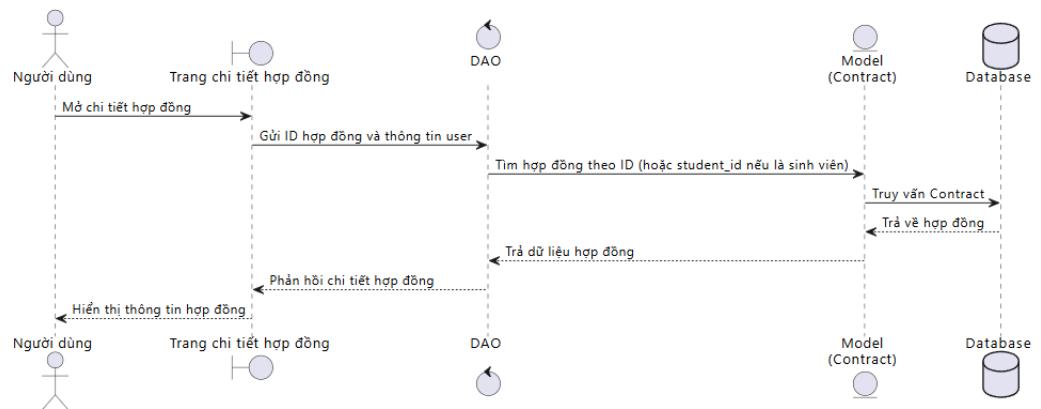


4.4.2 Quản lý hợp đồng

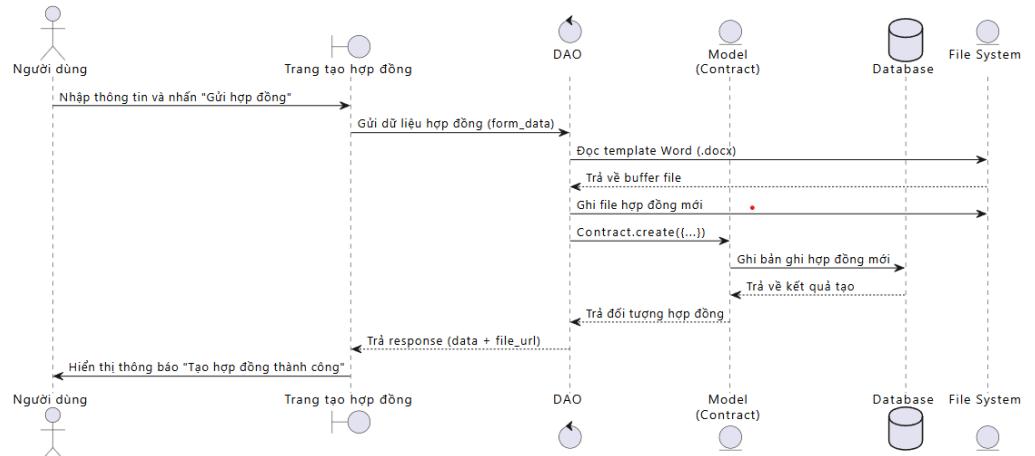
- Lấy danh sách hợp đồng



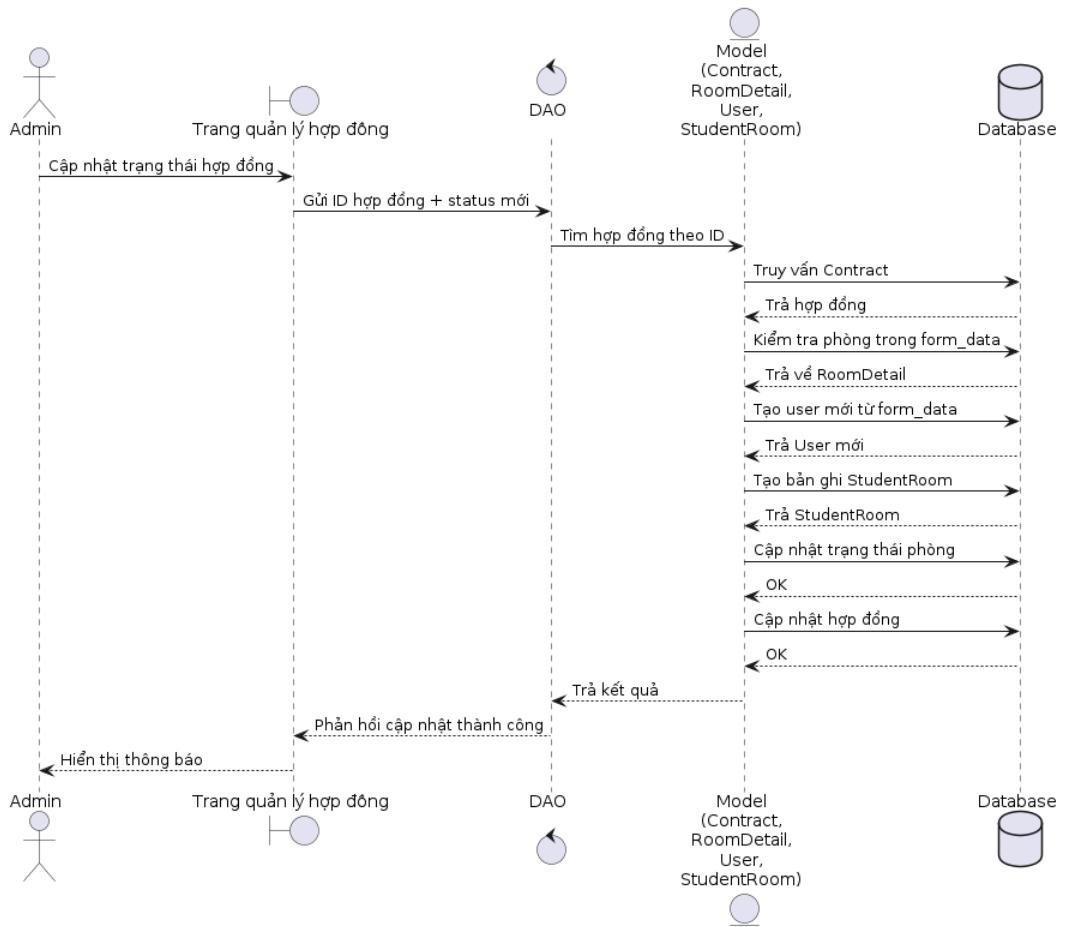
- Chi tiết hợp đồng



- Tạo mới hợp đồng



- Cập nhật thông tin hợp đồng



5 Kết luận

- Trong quá trình thực hiện xây dựng hệ thống backend cho **Website Quản lý Ký túc xá**, em đã có cơ hội tiếp cận và ứng dụng các kiến thức về kiến trúc phần mềm, thiết kế hệ thống theo mô hình phân tầng (MVC + Service Layer), cũng như triển khai thực tế các công nghệ hiện đại như **Node.js**, **Express.js**, **Sequelize ORM**, và **MySQL**.
- Thông qua quá trình phân tích yêu cầu, thiết kế kiến trúc, xây dựng cơ sở dữ liệu và hiện thực các chức năng chính như quản lý khu vực, phòng ở, lịch trực, điểm danh và báo cáo, hệ thống đã đảm bảo được các tiêu chí quan trọng: **phân quyền rõ ràng**, **dữ liệu nhất quán**, **luồng xử lý minh bạch** và **mở rộng linh hoạt**.

- Bên cạnh đó, việc áp dụng sơ đồ Use Case, sơ đồ lớp thực thể và sơ đồ tuần tự đã giúp hệ thống được thiết kế bài bản, dễ hiểu, dễ bảo trì và phục vụ tốt cho việc mở rộng các tính năng trong tương lai.
- Quá trình làm báo cáo và triển khai hệ thống là một trải nghiệm thực tiễn quý báu, giúp em nâng cao kỹ năng lập trình backend, tư duy hệ thống và khả năng làm việc theo mô hình dự án chuyên nghiệp. Em xin chân thành cảm ơn sự hướng dẫn tận tình của giảng viên, các anh chị tại đơn vị thực tập và nhà trường đã tạo điều kiện để em hoàn thành báo cáo này.