

ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



BÀI TẬP VỀ NHÀ BUỔI 6
THIẾT KẾ KIẾN TRÚC

Giáo viên hướng dẫn: ThS. Võ Tấn Khoa

Nhóm thực hiện:

- 1. Nguyễn Nhật Trường - 20522087**
- 2. Lại Chí Thiện - 20520309**
- 3. Lê Thị Phương Vy - 20520355**
- 4. Lê Trương Ngọc Hải - 20520481**
- 5. Trần Văn Lực - 20521587**
- 6. An Trương Tường Vi - 20520856**
- 7. Trần Thị Anh Thư - 20520792**
- 8. Trương Thị Mai Trinh - 20520825**
- 9. Đỗ Thị Ngọc Bích - 19521263**

TP. HỒ CHÍ MINH, 2022

PHÂN TASK CHO TỪNG THÀNH VIÊN

STT	MSSV	HỌ TÊN	TASK
1	20522087	Nguyễn Nhật Trường	Phân task, quản lí tiến độ và hỗ trợ nhóm, Bài tập đồ án – Câu 1
2	20520481	Lê Trương Ngọc Hải	Bài tập đồ án – Câu 2,3
3	20520355	Lê Thị Phương Vy	Bài tập đồ án – Câu 1 – Bài tập lí thuyết câu 10
4	20520309	Lại Chí Thiện	Bài tập đồ án – Câu 4
5	20521587	Trần Văn Lực	Bài tập đồ án – Câu 3, Bài tập lí thuyết – Câu 9
6	20520856	An Trương Tường Vi	Bài tập lí thuyết – Câu 1,2
7	20520792	Trần Thị Anh Thư	Bài tập lí thuyết – Câu 3,4
8	20520825	Trương Thị Mai Trinh	Bài tập lí thuyết – Câu 5,6
9	19521263	Đỗ Thị Ngọc Bích	Bài tập lí thuyết – Câu 7,8

BÀI TẬP LÝ THUYẾT

1. (Vi) - Câu 1: Khi mô tả một hệ thống, hãy giải thích lý do tại sao bạn cần bắt đầu thiết kế kiến trúc hệ thống trước khi đặc tả yêu cầu được hoàn thành.

Trong mô hình quy trình phát triển phần mềm, thiết kế kiến trúc hệ thống nằm trong giai đoạn đầu tiên của hệ thống và làm cơ sở cho mô tả hệ thống.

Rất khó để hình thành các đặc tả yêu cầu cho toàn bộ hệ thống vì sẽ làm tăng độ phức tạp. Việc xác định các thông số kỹ thuật trở nên dễ dàng hơn khi hệ thống được chia thành các hệ thống con.

Cần thiết kế kiến trúc hệ thống trước các đặc tả yêu cầu để cung cấp nền tảng chung về kiến trúc của hệ thống. Kiến trúc này sẽ tác động đáng kể đến các yêu cầu phi chức năng cũng như các yêu cầu chức năng. Dựa trên kiến trúc này, chúng ta có thể xác định các thành phần hệ thống chính, hệ thống con và mối quan hệ giữa chúng, nên khi mô tả sẽ dễ dàng hơn để chỉ định đặc tả nào đi đến hệ thống con nào. Kiến trúc của hệ thống được hoàn thiện giúp hình dung rõ hơn những gì khách hàng thực sự cần đối với hệ thống cũng như cách các nhà phân tích hiểu về hệ thống. Sau đó, đặc tả yêu cầu được phát triển dựa trên kiến trúc hệ thống chính xác hơn.

2. (Vi) - Câu 2: Bạn được yêu cầu chuẩn bị và gửi một bài thuyết trình cho một người quản lý phi kỹ thuật (không biết gì về kỹ thuật) để giải thích cho việc thuê một kiến trúc sư hệ thống cho dự án mới. Hãy viết một danh sách các gạch đầu dòng nêu ra các điểm chính trong bản trình bày của bạn, trong đó giải thích tầm quan trọng của kiến trúc phần mềm.

Các điểm trong bản trình bày nhằm giải thích cho người quản lý phi kỹ thuật về việc thuê một kiến trúc sư hệ thống cho dự án mới:

- Kiến trúc hệ thống liên quan đến việc hiểu cách tổ chức hệ thống và cấu trúc tổng thể của hệ thống được thiết kế.

- Kiến trúc phần mềm rất quan trọng vì nó ảnh hưởng đến hiệu suất, tính mạnh mẽ, khả năng phân phối và khả năng bảo trì của hệ thống.
- Thiết kế kiến trúc là các quy trình sáng tạo mà trong đó, bạn thiết kế một tổ chức hệ thống sẽ đáp ứng các yêu cầu chức năng và phi chức năng của một hệ thống.
- Thiết kế kiến trúc là một quá trình sáng tạo, dựa trên kinh nghiệm của kiến trúc sư hệ thống, và các yêu cầu cụ thể đối với hệ thống.
- Trong quá trình thiết kế kiến trúc, kiến trúc sư hệ thống phải đưa ra một số quyết định về cấu trúc có ảnh hưởng sâu sắc đến hệ thống và quá trình phát triển của nó.
- Dựa trên kiến thức và kinh nghiệm của kiến trúc sư hệ thống, các khía cạnh khác nhau về hệ thống như các mẫu kiến trúc, thiết kế, cách tiếp cận được sử dụng để cấu trúc hệ thống,... sẽ được xem xét.

Vì kiến trúc phần mềm là khuôn khổ cơ bản để cấu trúc hệ thống, nên nó phải được thiết kế cẩn thận, bằng cách xem xét nhiều yếu tố khác nhau. Và việc thuê một kiến trúc sư hệ thống có kinh nghiệm sẽ giúp cho việc thiết kế hệ thống dễ dàng và hoàn thiện hơn.

3. **(Thư) - Câu 3: Hiệu suất và bảo mật có thể tạo ra xung đột trong các yêu cầu phi chức năng trong quá trình thiết kế kiến trúc một hệ thống phần mềm. Hãy viết phân luận để ủng hộ ý kiến trên.**

Xung đột về tính khả dụng và yêu cầu bảo mật

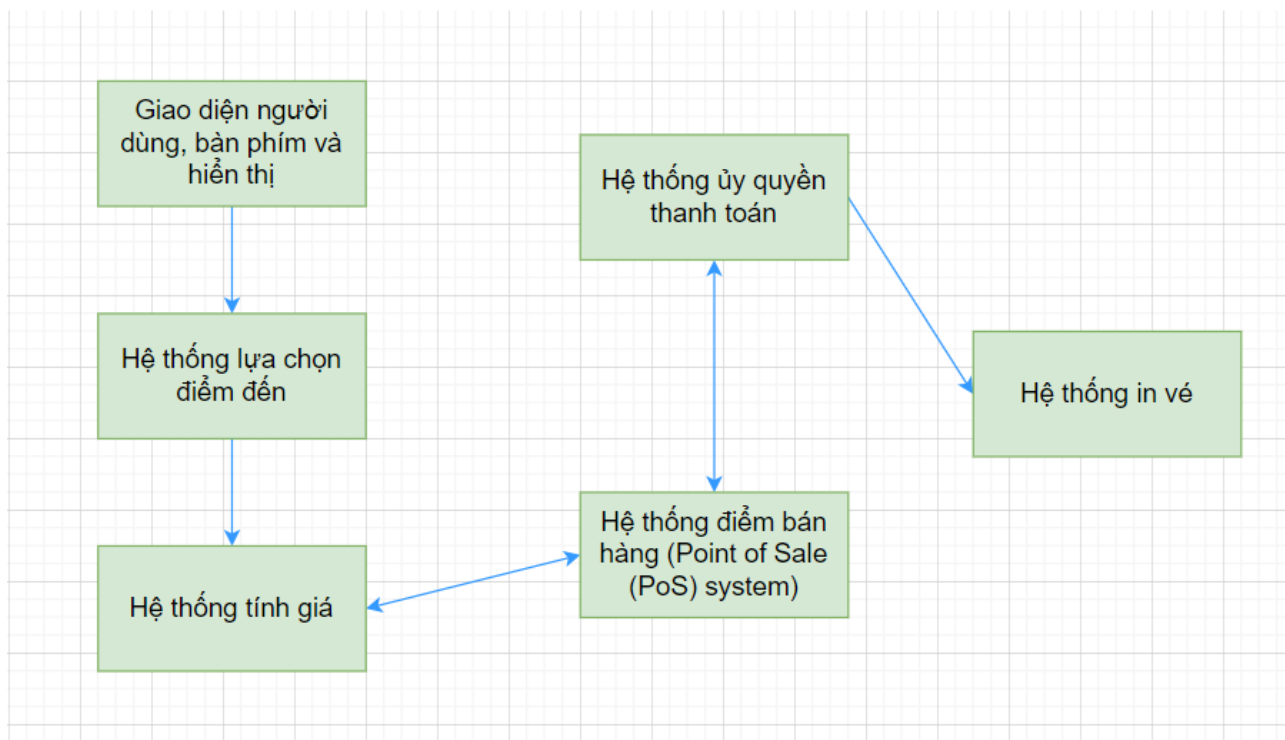
Nói chung, các yêu cầu phi chức năng nhất đối với kiến trúc hệ thống thiết kế như sau:

- Hiệu suất
- Sự an toàn
- Bảo vệ
- Khả dụng
- Khả năng bảo trì

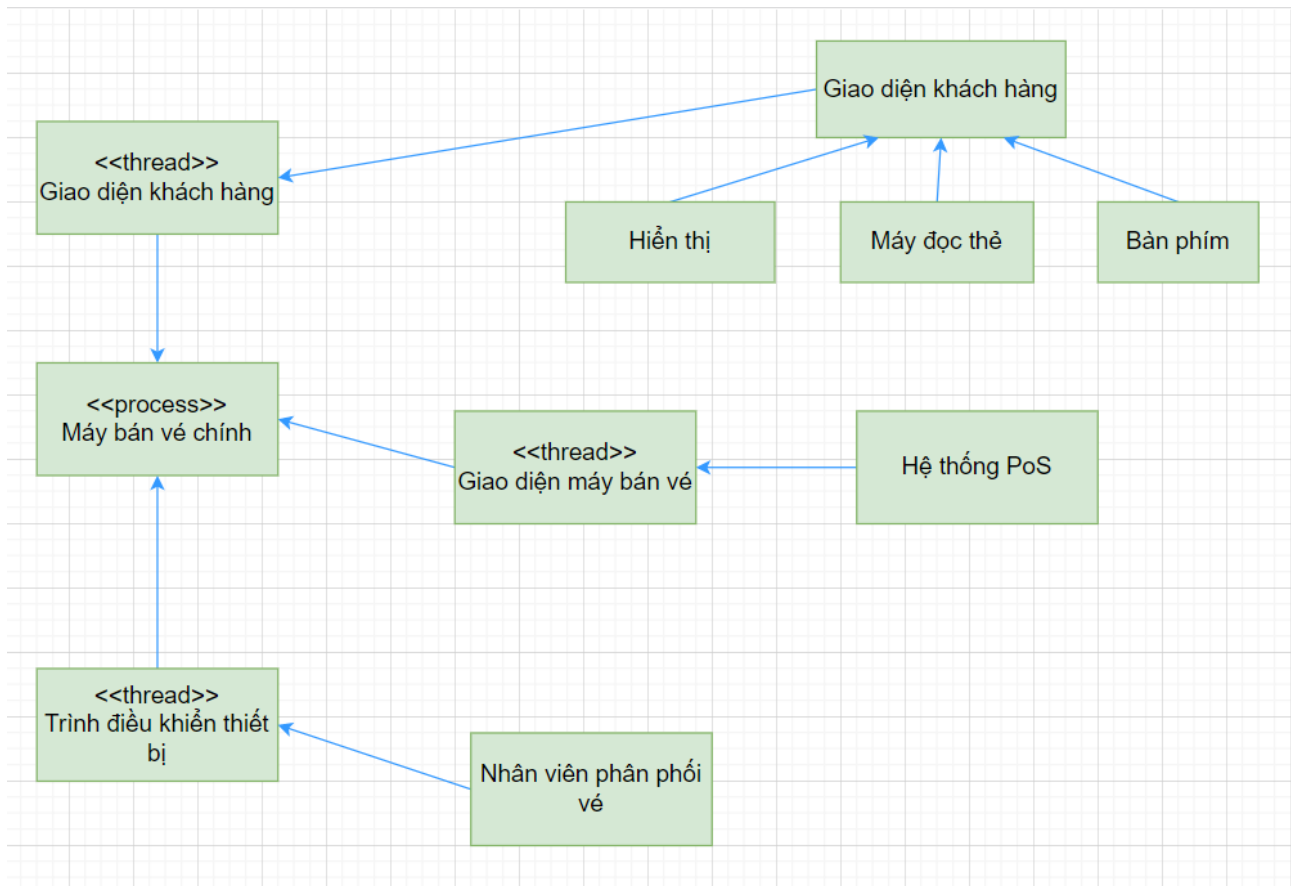
Trong khi thiết kế một kiến trúc, nó có thể phát sinh xung đột trong các trường hợp yêu cầu **bảo mật và tính khả dụng** trong các yêu cầu phi chức năng ở trên.

Lý do như sau:

- Kiến trúc hệ thống làm xáo trộn tính mạnh mẽ, khả năng bảo trì và hiệu suất của hệ thống. Do đó, phong cách và cấu trúc cụ thể được ưu tiên cho một ứng dụng có thể phụ thuộc vào các yêu cầu này.
 - Trong Yêu cầu bảo mật, kiến trúc phải được sử dụng cấu trúc phân lớp trong một yêu cầu bảo mật. Đây là một mức độ xác thực bảo mật đặc biệt được áp dụng cho các lớp này.
 - Trong yêu cầu về tính khả dụng, kiến trúc phải được xem xét để kết thúc các thành phần và để có thể thay đổi và chỉnh sửa các thành phần mà không cản trở hệ thống.
 - Tính bảo mật và tính khả dụng là những yêu cầu quan trọng. Nhưng xung đột kiến trúc chính xảy ra giữa sự sao chép và một số bản sao trong yêu cầu tính khả dụng và xung đột yêu cầu bảo mật xảy ra giữa chuyên biệt hóa và các bản sao tối thiểu. Vì vậy, xung đột thiết kế có thể phát sinh cho dù cấu trúc phân lớp hoặc các thành phần dư thừa được sử dụng khi thiết kế kiến trúc.
4. **(Thư) - Câu 4: Vẽ sơ đồ thể hiện góc nhìn khái niệm và góc nhìn quy trình trong kiến trúc của hệ thống sau:**
- Máy bán vé được sử dụng bởi hành khách tại ga.

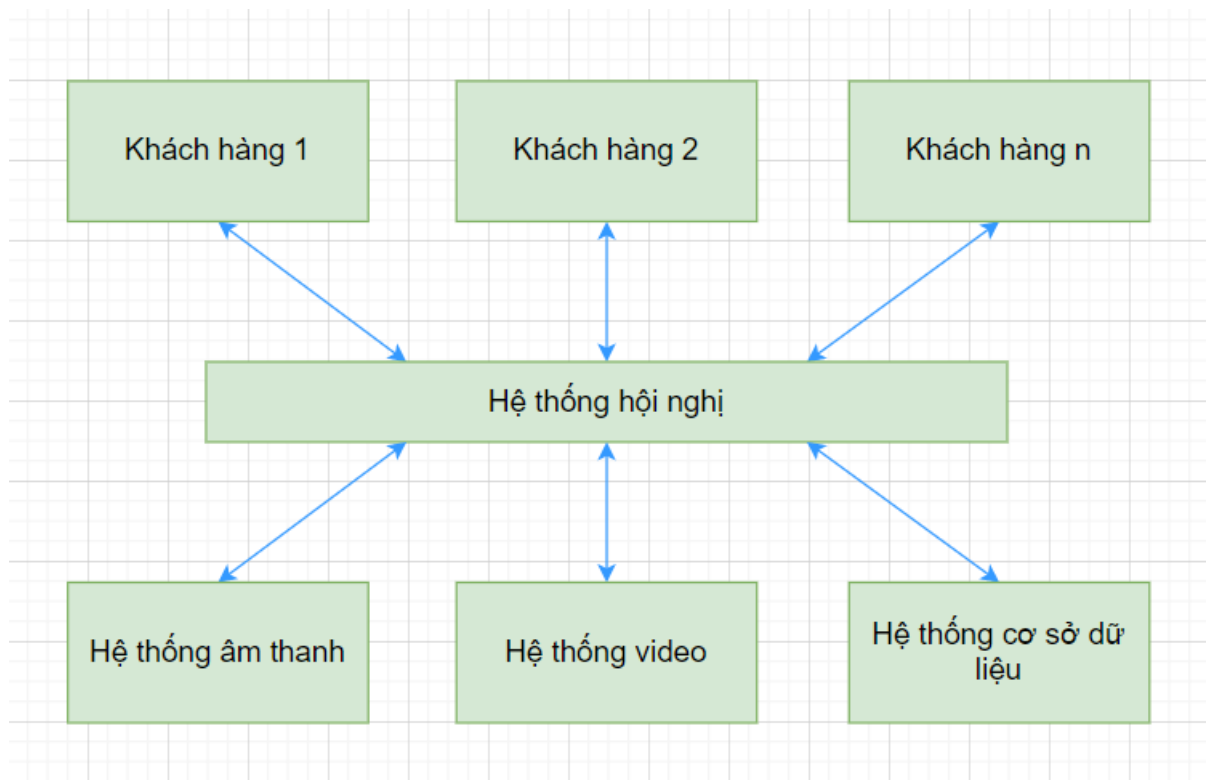


Hình 1. Góc nhìn khái niệm máy bán vé tại ga

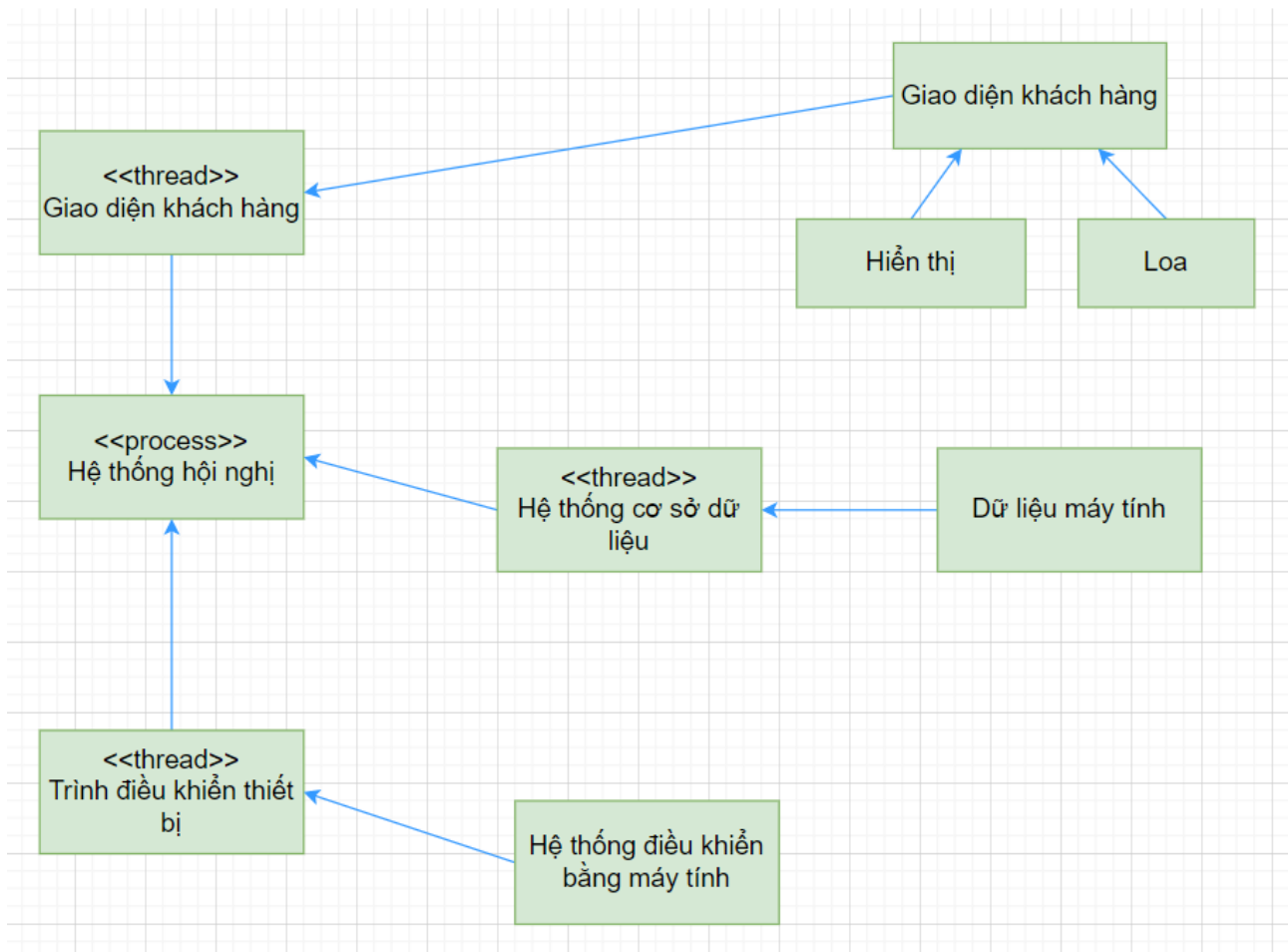


Hình 2. Góc nhìn quy trình máy bán vé tại ga

- Hệ thống hội nghị truyền hình được điều khiển bằng máy tính cho phép nhiều người tham gia cùng lúc, hiển thị dữ liệu video, âm thanh, và dữ liệu máy tính.

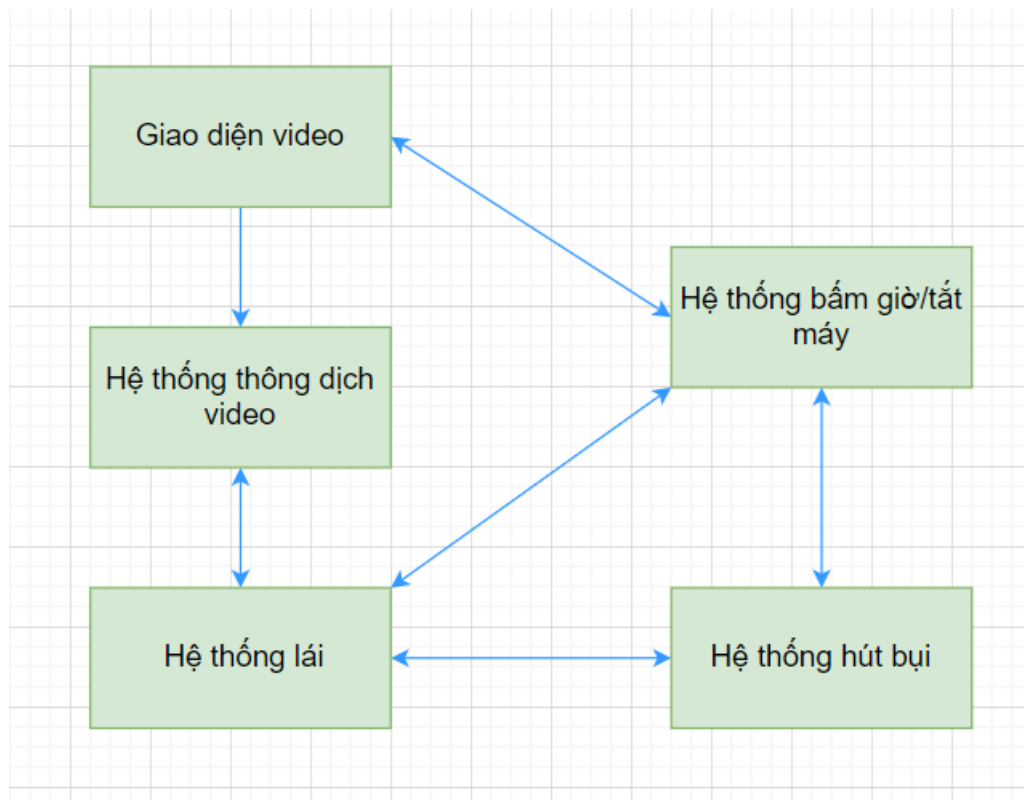


Hình 3. Góc nhìn khái niệm hệ thống hội nghị truyền hình

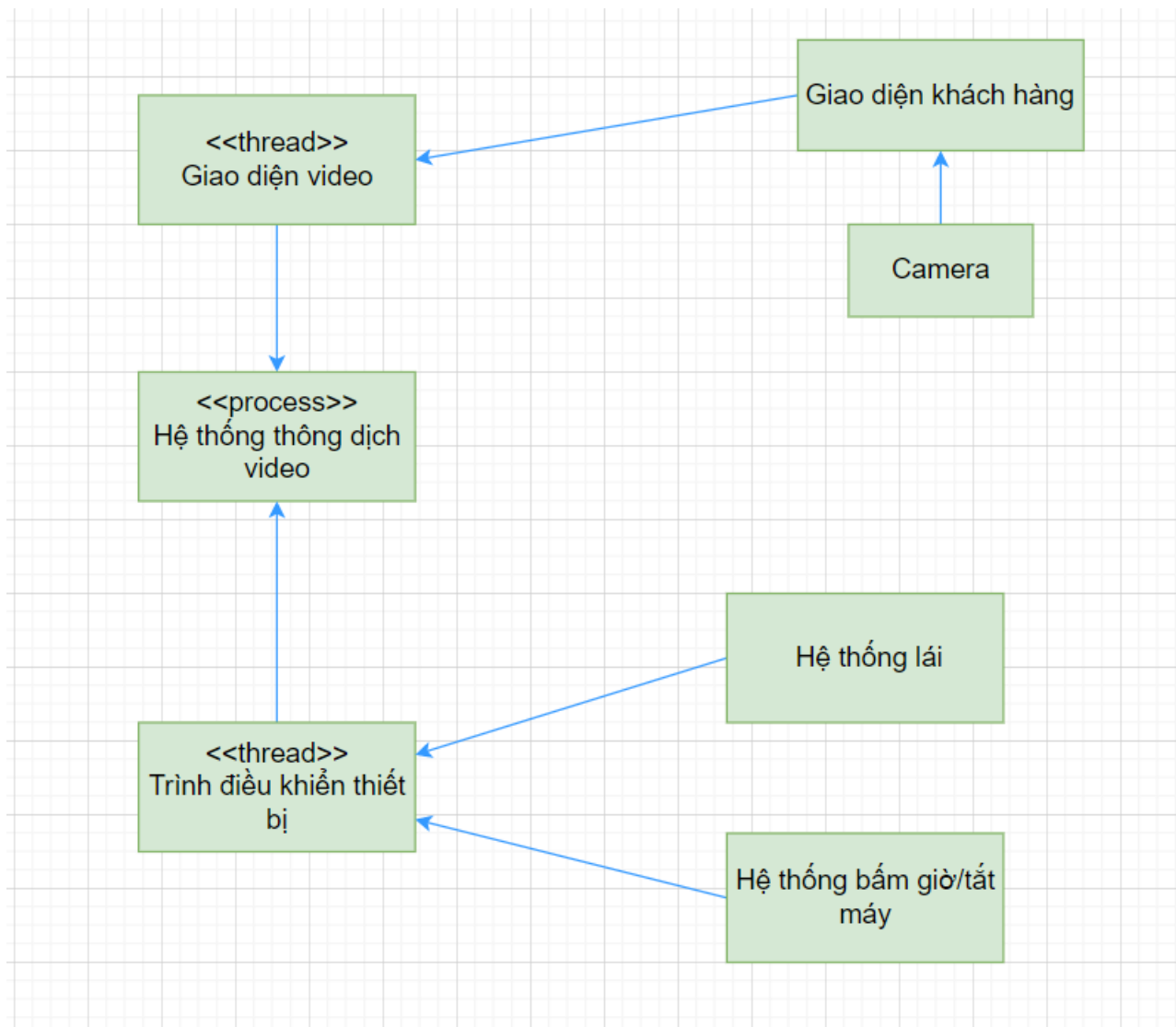


Hình 4. Góc nhìn quy trình hệ thống hội nghị truyền hình

- Robot lau sàn được thiết kế để làm sạch các không gian tương đối thông thoáng như hành lang. Nó phải có khả năng cảm nhận các bức tường và vật cản



Hình 5. Góc nhìn khái niệm robot lau sàn



Hình 6. Góc nhìn quy trình robot lau sàn

5. **(Trình)** - Câu 5: Một hệ thống phần mềm sẽ được xây dựng để cho phép máy bay không người lái tự động chặn thả gia súc trong các trang trại. Những máy bay không người lái này có thể được điều khiển từ xa bởi người điều khiển. Giải thích cách nhiều mẫu kiến trúc có thể phù hợp với nhau để giúp xây dựng loại hệ thống này.

Trong khoa học máy tính, hệ thống phần mềm đề cập đến các bộ mã có thể hỗ trợ phát triển chức năng. Chúng cho phép một loạt các tính năng, chẳng hạn như chặn gia súc bằng máy bay không người lái.

Người ta có thể sử dụng các mẫu kiến trúc khác nhau như model-view-controller (MVC) để tạo ra một chương trình. Điều này có thể giúp người vận hành dễ dàng sử dụng và chăn nuôi gia súc một cách hiệu quả. Ngoài ra, còn có một cơ sở dữ liệu với mô hình dữ liệu chứa thông tin về máy bay không người lái và cả gia súc. Mô hình sẽ quản lý các hành vi và dữ liệu cơ bản của ứng dụng. Điều này có thể giúp người vận hành dễ dàng sử dụng và quản lý các hành vi và dữ liệu cơ bản của ứng dụng.

Nó có thể phản hồi các yêu cầu cung cấp thông tin, phản hồi các hướng dẫn để thay đổi trạng thái của thông tin và thậm chí thông báo cho những người quan sát trong các hệ thống hướng sự kiện khi thông tin thay đổi. Đây có thể là cơ sở dữ liệu hoặc bất kỳ số lượng cấu trúc dữ liệu hoặc hệ thống lưu trữ nào.

Trong ngắn hạn, nó là dữ liệu và quản lý dữ liệu của ứng dụng. Nó có thể phản hồi các yêu cầu cung cấp thông tin, phản hồi các hướng dẫn để thay đổi trạng thái của thông tin và thậm chí thông báo cho những người quan sát trong các hệ thống hướng sự kiện khi thông tin thay đổi. Đây có thể là cơ sở dữ liệu hoặc bất kỳ số lượng cấu trúc dữ liệu hoặc hệ thống lưu trữ nào. Trong ngắn hạn, nó là dữ liệu và quản lý dữ liệu của ứng dụng. Ngoài ra, chế độ xem cung cấp hiệu quả phần tử giao diện người dùng của ứng dụng. Nó sẽ hiển thị dữ liệu từ mô hình thành một biểu mẫu phù hợp với giao diện người dùng. Bộ điều khiển nhận đầu vào của người dùng và thực hiện các cuộc gọi đến các đối tượng mô hình và khung nhìn để thực hiện các hành động thích hợp.

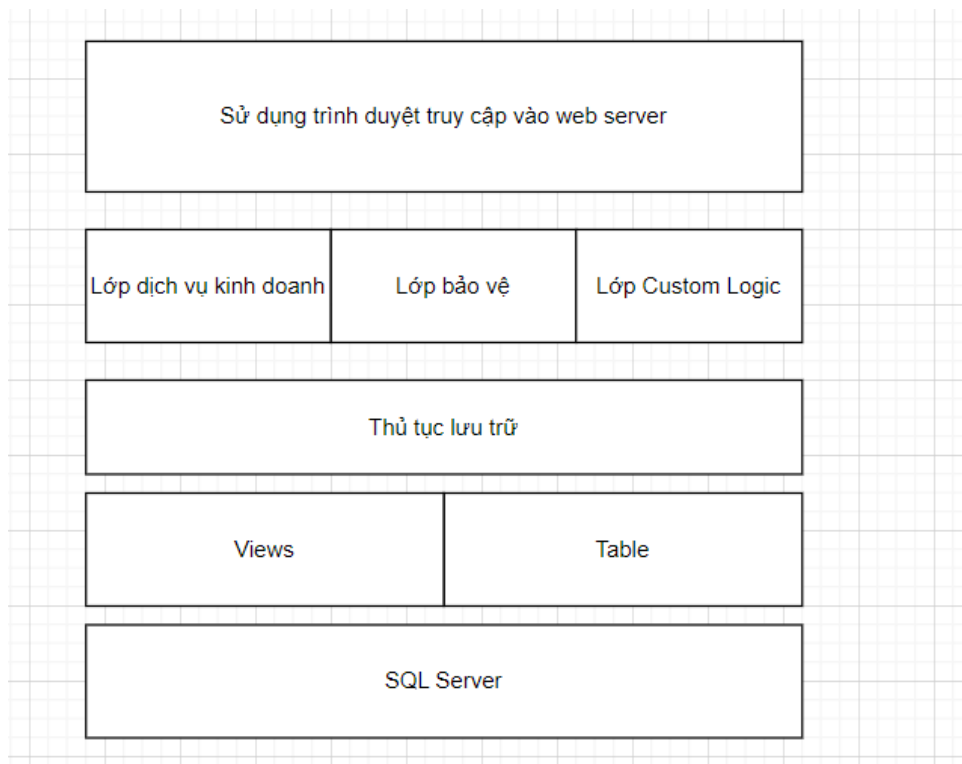
Nói chung, ba thành phần này kết hợp với nhau để tạo ra ba thành phần cơ bản của MVC.

6. **(Trình) - Câu 6: Đề xuất kiến trúc cho một hệ thống (như iTunes) được sử dụng để bán và phân phối nhạc trên internet. Những mẫu kiến trúc nào là cơ sở cho kiến trúc đề xuất của bạn?**

Mô hình client-server là mô hình kiến trúc thích hợp cho loại hệ thống như vậy. Trong kiến trúc client-server, chức năng của hệ thống được tổ chức thành các dịch vụ, với mỗi dịch vụ được phân phối từ một máy chủ riêng biệt. Khách hàng là người sử dụng các dịch vụ và truy cập vào máy chủ để sử dụng. iTunes lưu trữ tất cả các bản nhạc mà họ bán trong một cơ sở dữ liệu. Ở đó, khách hàng có thể tìm kiếm các bản nhạc này theo tên nghệ sĩ, thể loại...

thông qua giao diện dựa trên web. Ngoài ra, các bản nhạc có thể được tải xuống và trả tiền tương ứng. Sau đó, máy chủ quản lý việc đặt hàng nhạc cũng thông qua giao diện dựa trên web. Vì thế, mô hình kiến trúc làm cơ sở cho điều này là mô hình client-sever.

7. **(Bích)** - Câu 7: Một hệ thống thông tin sẽ được phát triển để duy trì thông tin về các tài sản thuộc sở hữu của một công ty tiện ích như tòa nhà, phương tiện và thiết bị. Dự kiến rằng điều này sẽ được cập nhật bởi các nhân viên làm việc trong lĩnh vực này bằng thiết bị di động khi có thông tin tài sản mới. Công ty có một số cơ sở dữ liệu tài sản hiện có cần được tổng hợp qua hệ thống này. Hãy thiết kế kiến trúc phân lớp cho hệ thống quản lý tài sản này dựa trên kiến trúc hệ thống thông tin chung được thể hiện trong hình 6.18.



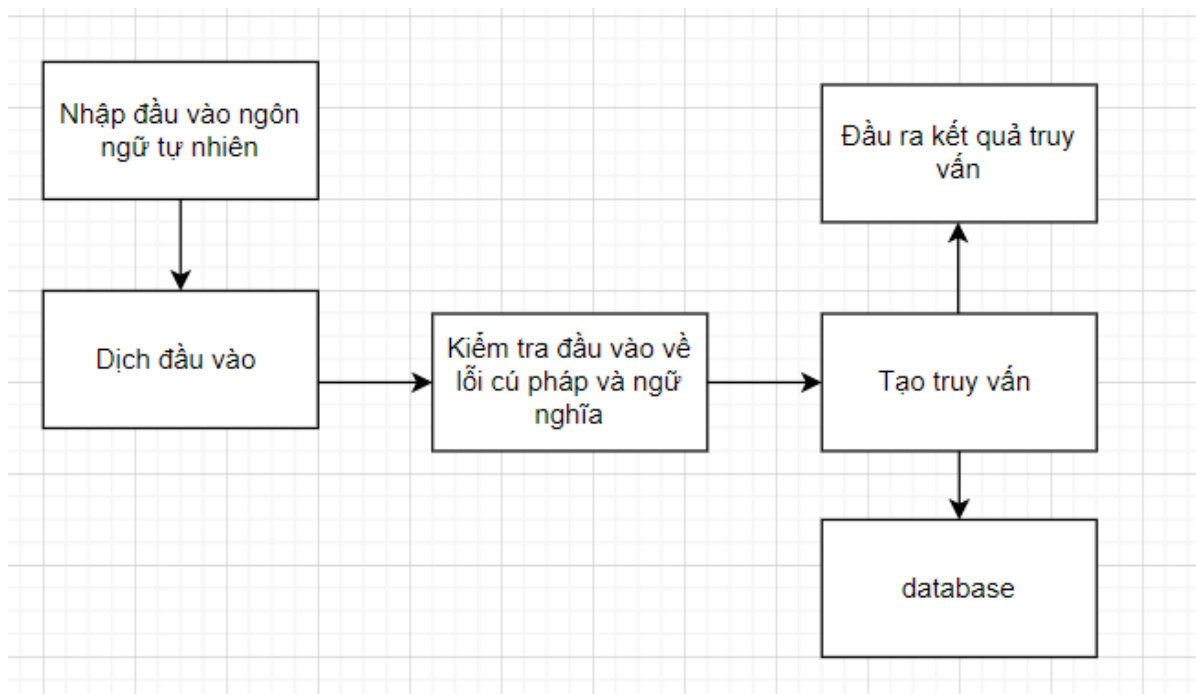
Hình 7: Kiến trúc phân lớp của hệ thống quản lý tài sản

Có ba loại lớp cho kiến trúc của hệ thống quản lý tài sản:

- Lớp giao diện người dùng: người dùng có thể truy cập các dịch vụ web bằng trình duyệt

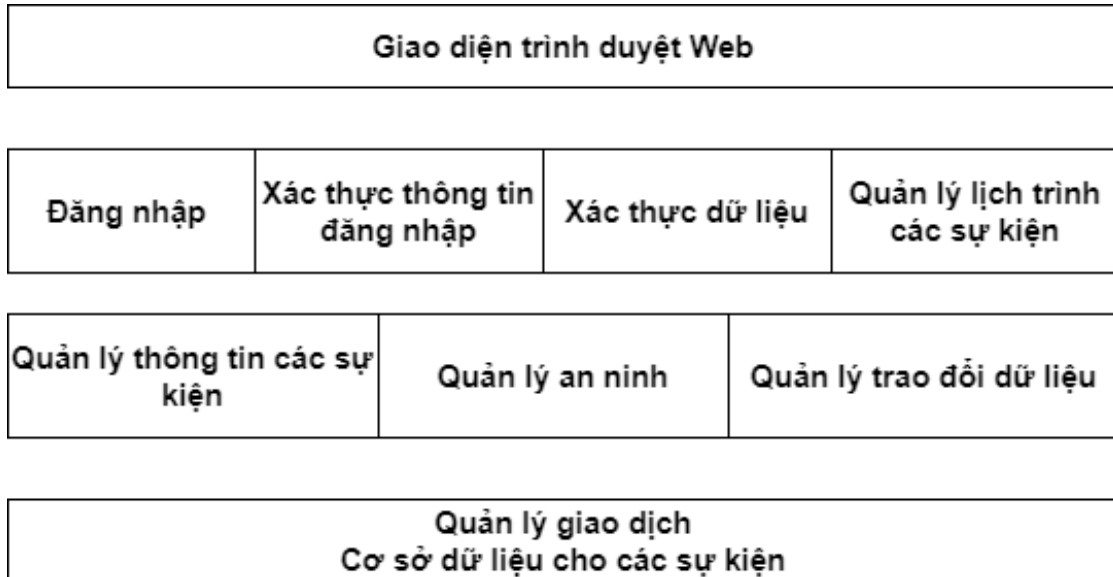
- Lớp logic nghiệp vụ: người dùng có một số tùy chọn. Người dùng chuyển đến lớp logic nghiệp vụ và truy cập dữ liệu trên lớp truy cập dữ liệu
- Lớp cơ sở dữ liệu: người dùng thấy các thủ tục được lưu trữ, các dạng xem khác nhau, bảng và máy chủ SQL

8. **(Bích)** - Câu 8: Sử dụng mô hình chung (tổng quát) của một hệ thống xử lý ngôn ngữ, để thiết kế kiến trúc lưu trữ của một hệ thống chấp nhận các lệnh ngôn ngữ tự nhiên và chuyển chúng thành các truy vấn cơ sở dữ liệu bằng một ngôn ngữ khác (ví dụ SQL)



Hình 8: Kiến trúc lưu trữ của hệ thống chấp nhận các lệnh ngôn ngữ tự nhiên và chuyển chúng thành các truy vấn csdl.

9. **(Lực)** - Câu 9: Sử dụng mô hình cơ bản của hệ thống thông tin, như hình 6.18, đề xuất các thành phần có thể là một phần của hệ thống thông tin cho phép người dùng xem các sự kiện phòng vé, vé có sẵn và giá, cuối cùng là sự kiện mua vé.



Lớp đầu tiên chứa giao diện người dùng trên web.

Lớp thứ hai bao gồm các thành phần đăng nhập và quản lý lịch trình cho các sự kiện.

Lớp thứ ba thực hiện bảo mật, duy trì thông tin.

Lớp thứ tư chứa DataBase của các sự kiện phòng vé, vé có sẵn và giá, mua vé và quản lý các giao dịch.

10. (Vy) - Câu 10: Câu hỏi đặt ra là có nên có một nghề riêng là nghề “kiến trúc sư phần mềm” có vai trò là làm việc chuyên sâu với khách hàng để thiết kế kiến trúc hệ thống phần mềm không? Sau đó, một công ty phần mềm riêng biệt với người ở trên sẽ triển khai hệ thống. Những khó khăn khi thành lập một nghề như vậy có thể là gì?

Không nên có một nghề riêng là nghề “kiến trúc sư phần mềm” có vai trò là làm việc chuyên sâu với khách hàng để thiết kế kiến trúc hệ thống vì các lập trình viên là những người hiểu rõ những gì cần làm với phần mềm:

- Thiết kế kiến trúc liên quan đến việc hiểu cách tổ chức hệ thống phần mềm và thiết kế cấu trúc tổng thể của hệ thống đó.
- Tồn tại mối liên hệ quan trọng giữa thiết kế và kỹ thuật yêu cầu, vì nó xác định các thành phần cấu trúc chính trong một hệ thống và mối quan hệ giữa chúng, các lập trình viên có thể gặp khó khăn để hiểu kiến trúc phần mềm nếu họ không phải là những người làm ra nó.
- Thiết kế kiến trúc phần mềm là một việc yêu cầu sự sáng tạo, không có một công thức riêng cho quá trình này và nó phụ thuộc vào loại hệ thống đang phát triển, kiến thức, kinh nghiệm của kỹ sư phát triển và các yêu cầu cụ thể dành cho hệ thống, nếu để 1 người riêng biệt thiết kế kiến trúc, nó có thể không đáp ứng tất cả các yêu cầu kỹ thuật.
- Trong quá trình thiết kế kiến trúc, những người thiết kế hệ thống đưa ra các quyết định ảnh hưởng sâu sắc đến hệ thống và năng suất phát triển của nó.
- Các hệ thống trong cùng một tên miền thường có kiến trúc tương tự nhau, khi thiết kế hệ thống cần phải quyết định có thể tái sử dụng những phần nào trong hệ thống.
- Đôi khi cần làm lại hoặc thiết kế lại do không tương thích hoặc thay đổi kỹ thuật yêu cầu.

BÀI TẬP ĐỒ ÁN

1. **(Vy + Trường) - Câu 1:** Dựa trên các câu hỏi ở phần 1 hãy đưa ra quyết định thiết kế kiến trúc cho đồ án của nhóm.

Nhóm 3 quyết định sử dụng kiến trúc phân lớp (layered architecture) cho đồ án vì:

- Vì việc thiết kế không quá khó, đồng thời các thành viên trong nhóm cũng dễ tiếp cận được với mẫu thiết kế này.
- Các chức năng, thành phần của ứng dụng được tổ chức rõ ràng, cụ thể để dễ dàng nhận biết được các tầng của hệ thống có và thực hiện những gì.
- Dễ dàng xây dựng các hệ thống hay chức năng mới cho ứng dụng.

2. **(Thiện) - Câu 2:** Đảm bảo các nguyên tắc khi thiết kế kiến trúc hệ thống.

Hiệu suất – Performance

- Các chức năng của hệ thống AInE phải được load chậm nhất là 1 giây.

Bảo mật – Security

- Chỉ có người dùng được xem thông tin tài khoản của mình.
- Chỉ những người phát triển phần mềm mới có thể truy cập vào cơ sở dữ liệu của hệ thống.

An toàn – Safety

- Tụi em chưa suy nghĩ ra nguyên tắc này.

Khả dụng – Availability

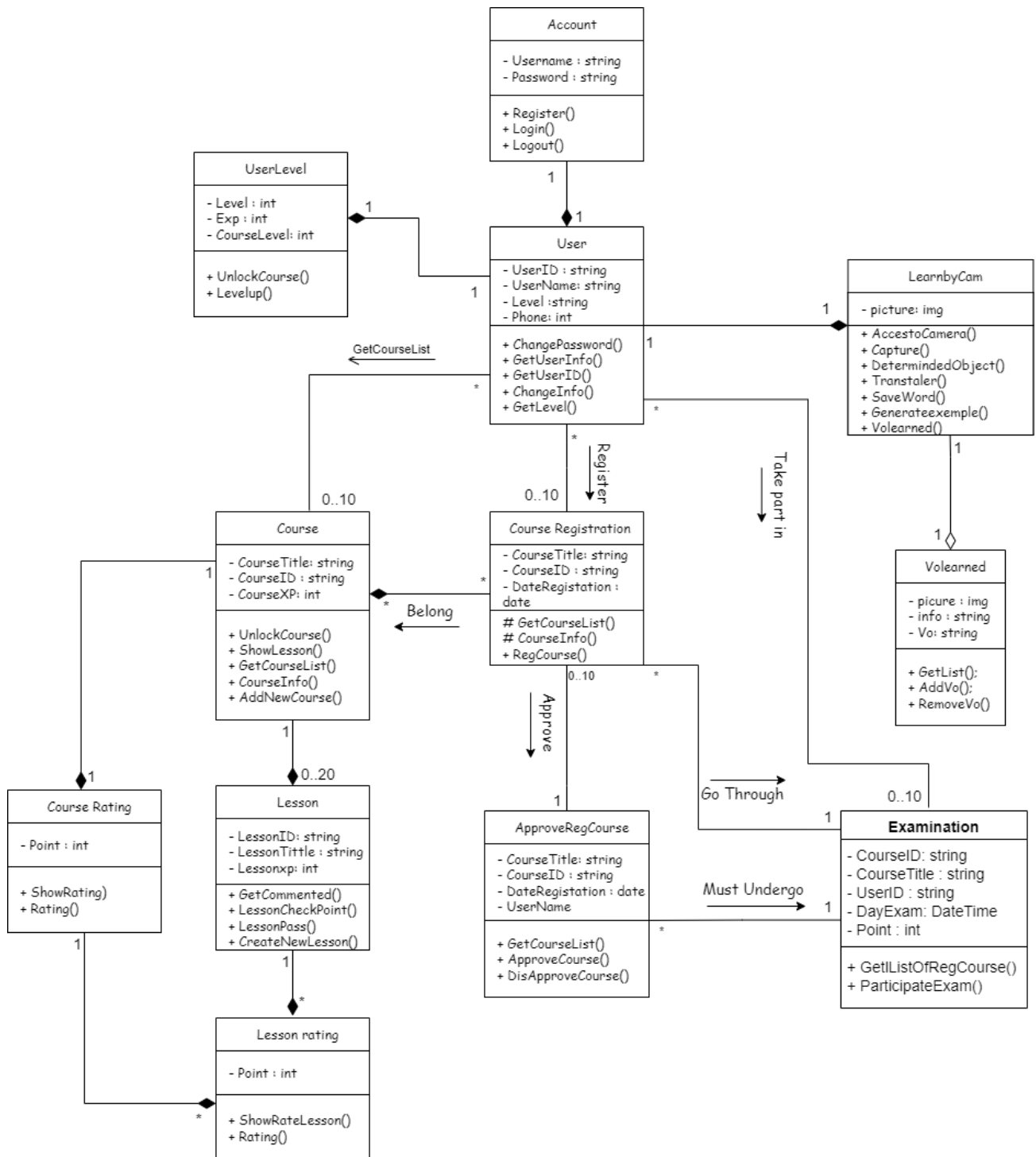
- Khi đã đủ yêu cầu để mở khóa tính năng mà các tính năng đó chưa được mở khóa thì vấn đề sẽ được giải quyết trong vòng 1 ngày.

Khả năng bảo trì – Maintainability

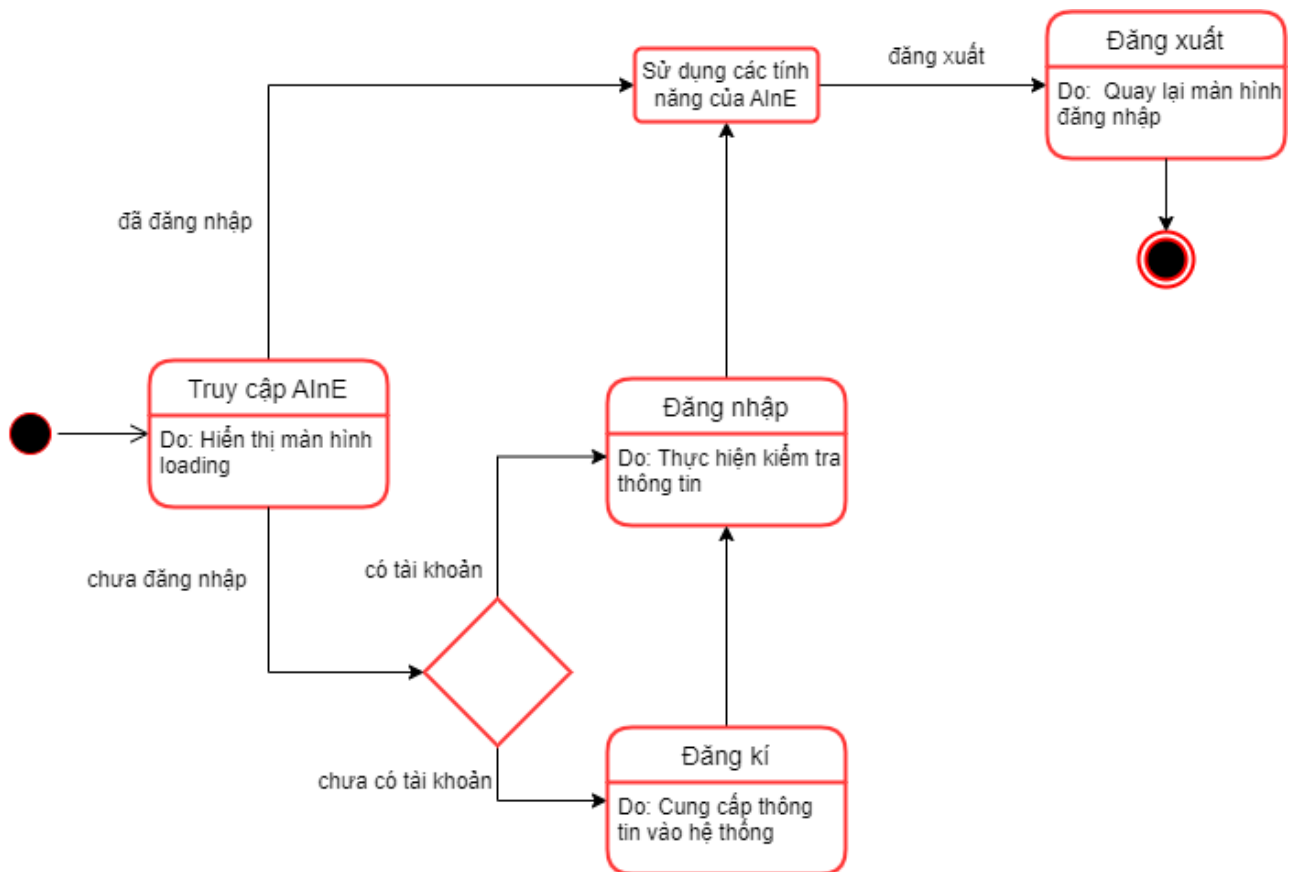
- Nếu có bất kỳ lỗi nào trong quá trình sử dụng thì các lỗi đó sẽ được bảo trì và sửa chữa ngay khi được thông báo.

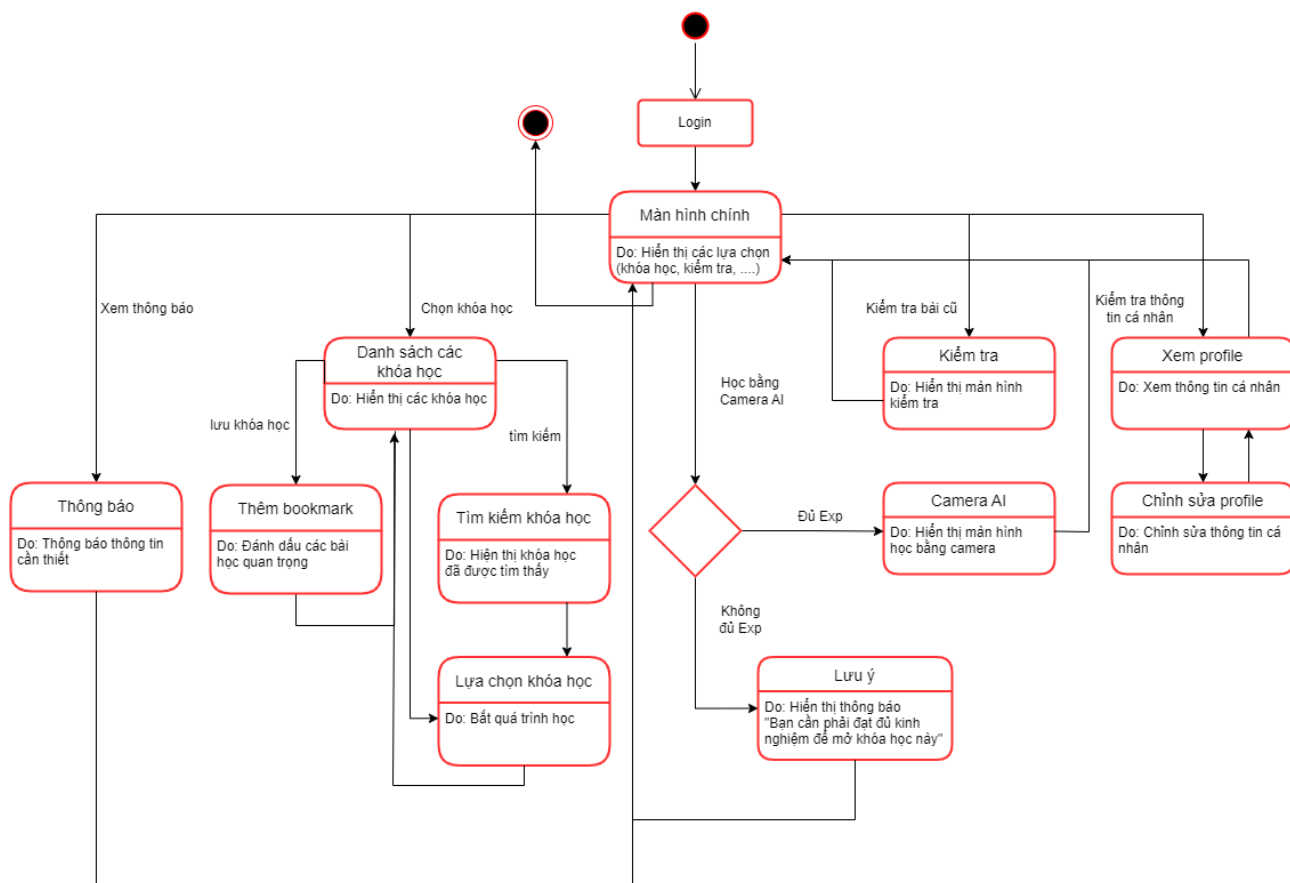
3. (Lực + Hải) - Câu 3: Sử dụng UML để thiết kế kiến trúc hệ thống theo 4+1 góc nhìn được đề cập trong bài.

Góc nhìn **Logical View** sẽ liên quan đến chức năng của hệ thống và người dùng cuối, hai sơ đồ đại diện cho góc nhìn Logical View của hệ thống AInE là **Class Diagrams** và **State Diagrams**:



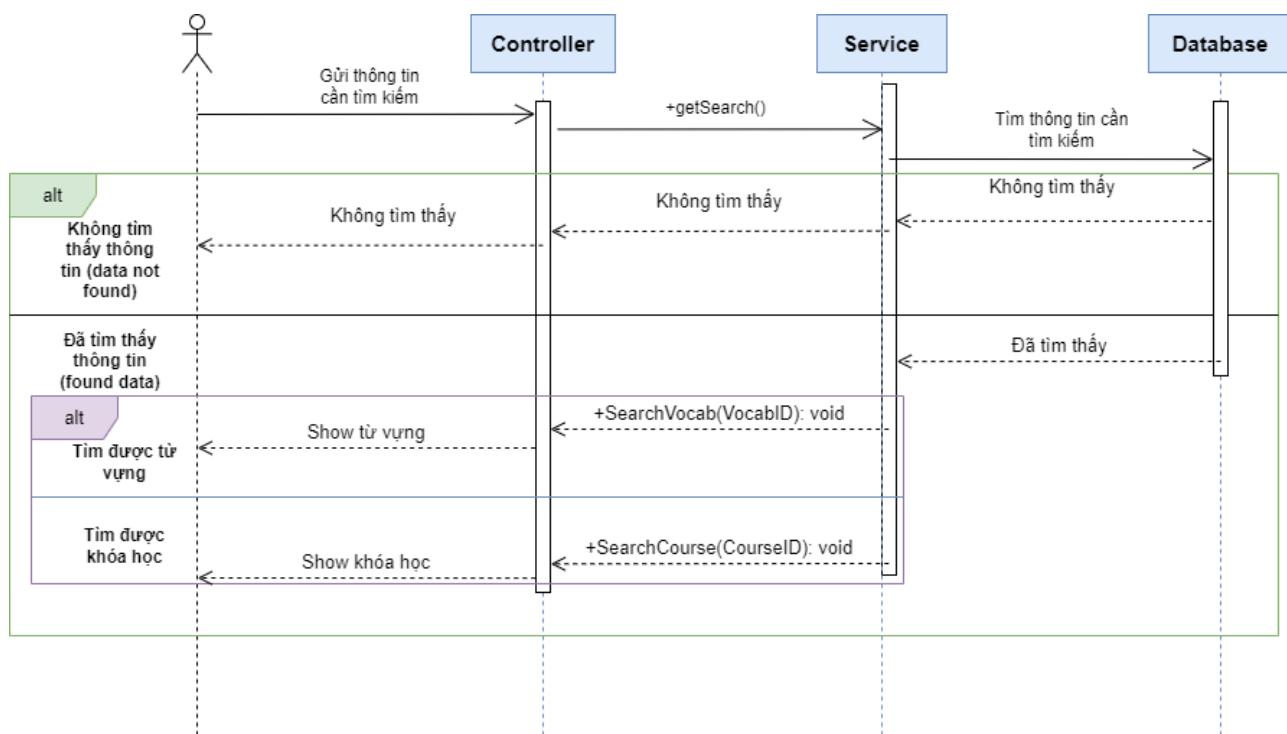
Hình 1: Sơ đồ lớp (Class Diagrams) của hệ thống.



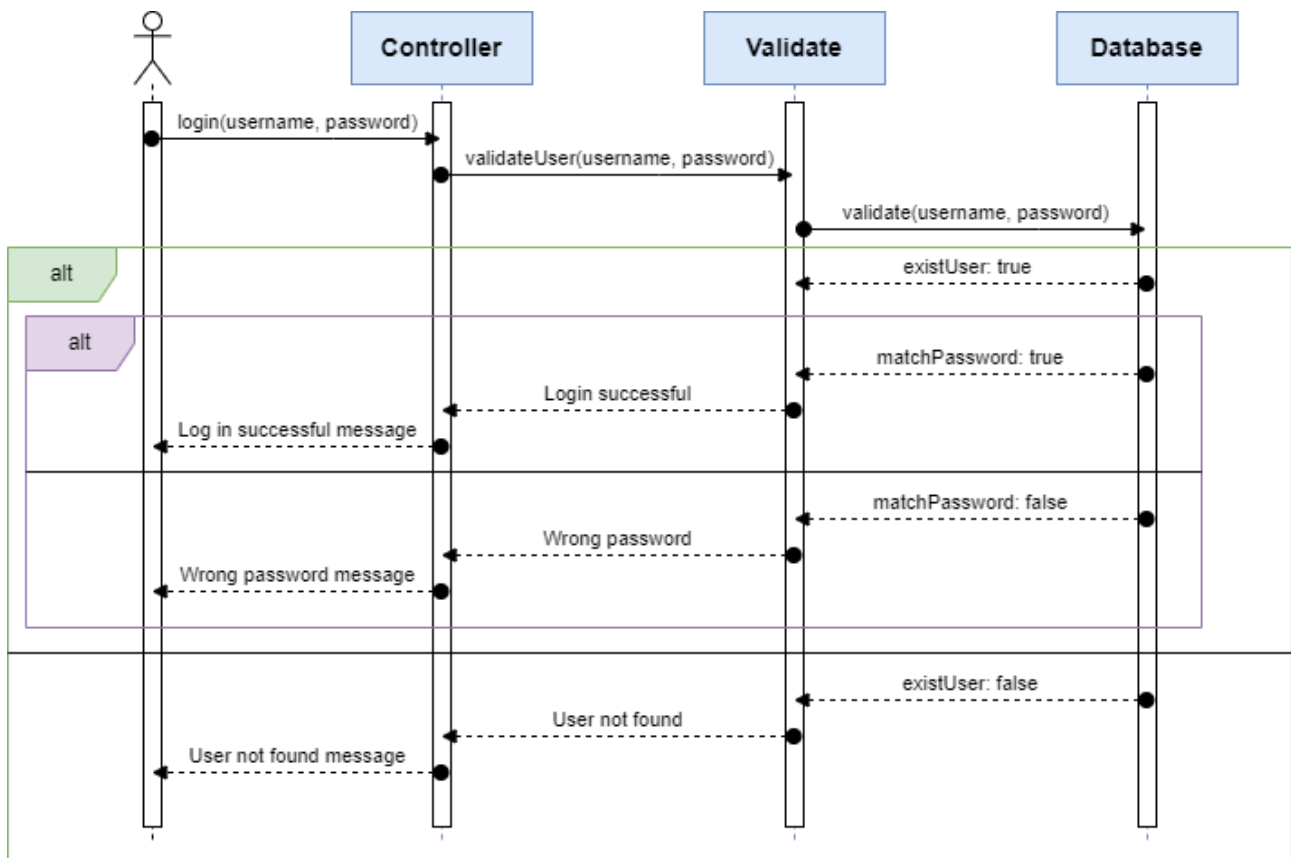


Hình 2: Sơ đồ trạng thái của hệ thống

Góc nhìn **Process View** sẽ hiển thị sự tương tác giữa các quy trình trong hệ thống khi nó đang chạy, sơ đồ đại diện cho góc nhìn Process View của hệ thống AInE là **Sequence Diagrams**



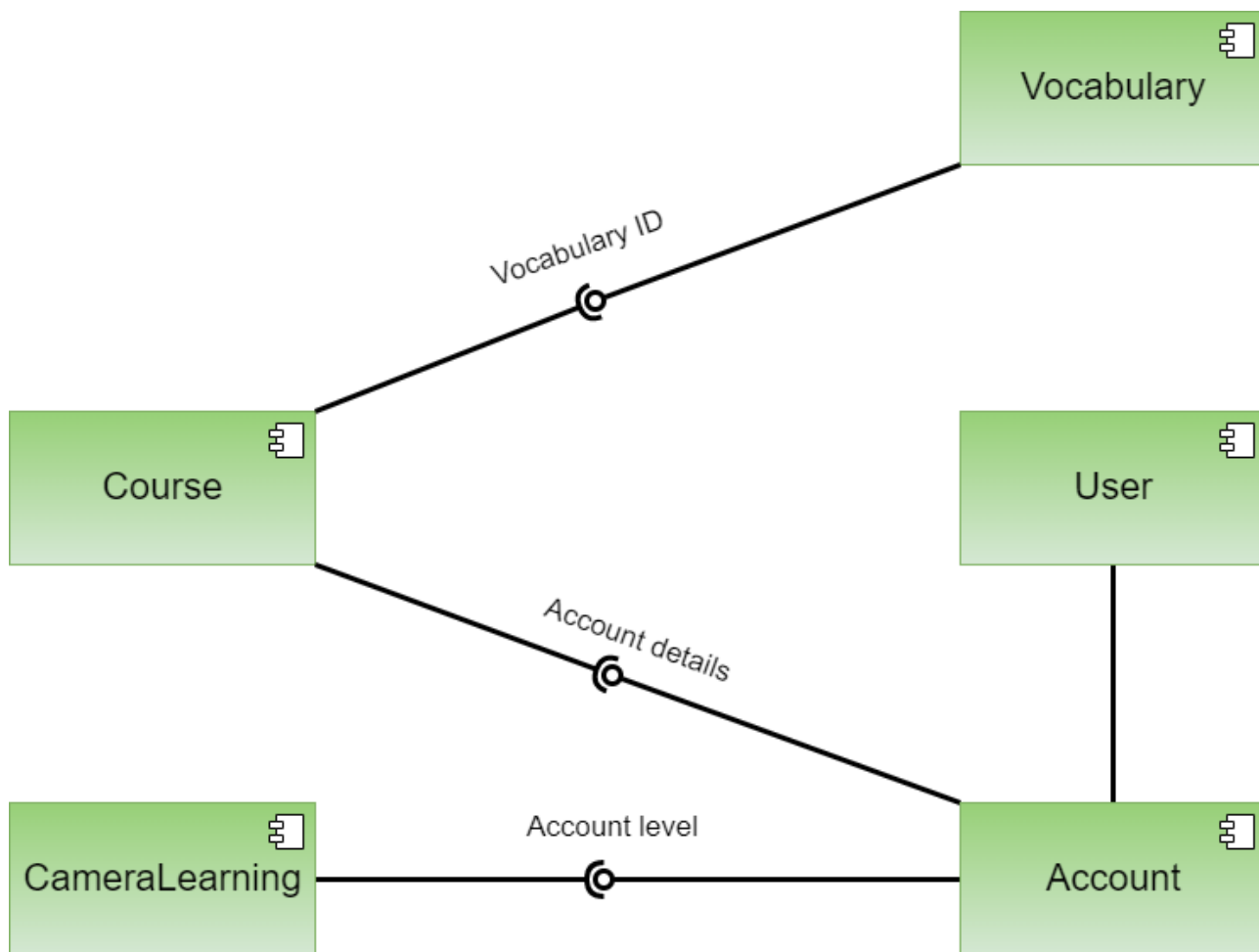
Hình 3: Sơ đồ trình tự (sequence diagrams) của hệ thống search.



Hình 4: Sơ đồ trình tự (sequence diagrams) của hệ thống đăng nhập.

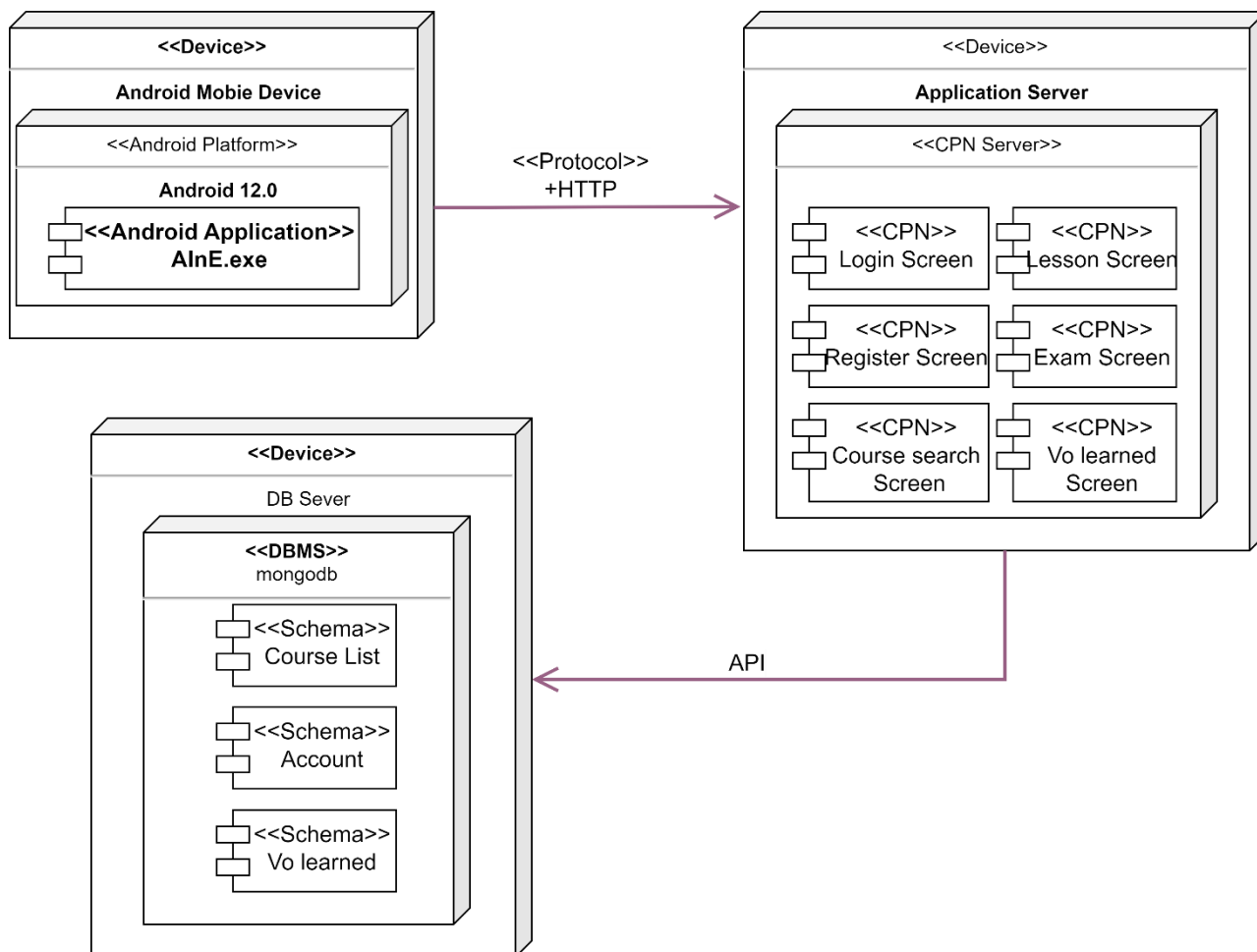
Trong quá trình làm đồ án nhóm tụi em sẽ tiếp tục cập nhật thêm các sơ đồ trình tự (sequence diagrams) của các hệ thống khác của ứng dụng.

Góc nhìn **Development View** sẽ hiển thị các phân rã phần mềm để phát triển, sơ đồ đại diện cho góc nhìn Development View của hệ thống AInE là **Component Diagrams**.



Hình 5: Component Diagrams của hệ thống.

Góc nhìn **Physical View** sẽ hiển thị phần cứng hệ thống và cách các thành phần của phần mềm được phân phối trên các bộ xử lý của hệ thống, sơ đồ đại diện cho góc nhìn; Physical View của hệ thống AInE là **Deployment Diagrams**.



Hình 6: Deployment Diagrams của hệ thống.

4. **(Thiện)** - Câu 4: Tùy thuộc theo đề án, có thể sử dụng lại các mẫu kiến trúc được đề cập trong bài học (MVC, Layer, Kho lưu trữ, Client-Server) và sau đó xây dựng | điều chỉnh phù hợp cho phần mềm của nhóm.

User interface	AlnE app	
Configuration layer	Identity management	
Application layer	Courses management Level management User information mangagement	Rating Camera Exercise Checkpoint saving
Utility layer	Authentication User storage Image storage	AI model storage Search

Hình 7: Kiến trúc Player của hệ thống.

-----**Hết**-----