

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ - ĐẠI HỌC QUỐC GIA HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO BÀI TẬP LỚN
CÔNG NGHỆ PHẦN MỀM



Đề tài: Ứng dụng web học tiếng Nhật miễn phí

Thành viên:

Cao Thị Phương Anh – Nhóm trưởng

Nguyễn Thị Thanh Thủy

Đặng Thị Thanh Hiền

Phạm Minh Tâm

Giảng viên hướng dẫn:

Thầy Nguyễn Đức Anh

MỤC LỤC

MỤC LỤC	2
MỤC LỤC HÌNH ẢNH	4
MỤC LỤC BẢNG BIỂU	6
LỜI CẢM ƠN.....	7
LỜI CAM ĐOAN.....	8
Chương 1 Đặt vấn đề.....	10
1.1. Bối cảnh bài toán.....	10
1.2. Giải pháp cho bài toán và hạn chế.....	10
1.3. Giải quyết hạn chế.....	10
1.4. Xây dựng và ý nghĩa hệ thống.....	11
1.5. Nội dung báo cáo.....	11
Chương 2 Kiến thức nền tảng	12
2.1. Các khối xây dựng của một ứng dụng web	12
2.2. JavaScript	12
2.3. React.....	12
2.4. Next.js.....	13
2.5. Axios	14
2.6. MySQL.....	14
2.7. Mô hình MVC	14
2.8. Laravel Framework	15
Chương 3 Phân tích và đặc tả yêu cầu	16
3.1 Phân tích yêu cầu.....	16
Thu thập yêu cầu	16
Yêu cầu chức năng	16
Yêu cầu phi chức năng	16
3.2 Phân tích và đặc tả ca sử dụng	16
3.2.1 Ca sử dụng “Đăng ký”.....	18
3.2.2 Ca sử dụng “Đăng nhập”	19
3.2.3 Ca sử dụng “Đăng xuất”.....	19
3.2.4 Ca sử dụng “Xem thông tin”	19
3.2.5 Ca sử dụng “Chỉnh sửa thông tin”.....	20
3.2.6 Ca sử dụng “Xem bài học”.....	20
3.2.7 Ca sử dụng “Làm câu hỏi ôn tập”	20
3.2.8 Ca sử dụng “Thêm flashcard”	21
3.2.9 Ca sử dụng “Xem flashcard”	21
3.2.10 Ca sử dụng “Sửa flashcard”	22
3.2.11 Ca sử dụng “Xoá flashcard”	22

3.2.12	Ca sử dụng “Xem tài khoản người dùng”	23
3.2.13	Ca sử dụng “Xoá tài khoản người dùng”	23
3.3	Mô tả các biểu đồ hoạt động	23
3.3.1	Hoạt động “Quản lý xác thực tài khoản”	23
3.3.2	Hoạt động “Sửa thông tin”	24
3.3.3	Hoạt động “Trả lời câu hỏi ôn tập”	25
3.3.4	Hoạt động “Tạo flashcard”	26
3.3.5	Hoạt động “Sửa flashcard”	27
Chương 4	Phân tích thiết kế và xây dựng	29
hệ thống	29
4.1	Thiết kế kiến trúc chung của hệ thống	29
4.2	Biểu đồ tuần tự các ca sử dụng.....	29
4.3	Thiết kế API	35
4.4	Thiết kế cơ sở dữ liệu	37
4.5	Thiết kế giao diện người dùng.....	38
4.5.1	Giao diện chính khi truy cập vào trang web.....	38
4.5.2	Giao diện đăng ký, đăng nhập	39
4.5.3	Xem thông tin của bản thân, các flashcard đã tạo theo thời gian, các bài học đã hoàn thành	40
4.5.4	Chỉnh sửa thông tin cá nhân	41
4.5.5	Bài học.....	41
4.5.6	Flashcard	43
Chương 5	Cài đặt và kiểm thử	45
5.1	Cài đặt công cụ	45
5.2	Cài đặt dự án.....	45
5.2.1	Cài đặt front-end.....	45
5.2.2	Cài đặt back-end.....	46
5.3	Kiểm thử.....	47
5.3.1	Kiểm thử chức năng	48
5.3.2	Kiểm thử phi chức năng	52
5.4	Tổng kết chương.....	55
Chương 6	Kết luận và định hướng phát triển.....	56
6.1	Kết luận	56
6.2	Định hướng phát triển	56

MỤC LỤC HÌNH ẢNH

1. Hình 2.1: Giao diện người dùng tương tác.	13
2. Hình 2.2: Nextjs cung cấp các khối xây dựng để tạo các ứng dụng web nhanh.	13
3. Hình 2.1. Sơ đồ kiến trúc mô hình MVC.	14
4. Hình 3.1: Các tác nhân hệ thống.....	17
5. Hình 3.2: Biểu đồ lớp của hệ thống.	17
6. Hình 3.3: Biểu đồ use case chung của hệ thống.	18
7. Hình 3.4: Biểu đồ hoạt động quản lý xác thực tài khoản	24
8. Hình 3.5: Biểu đồ hoạt động sửa thông tin.....	25
9. Hình 3.6: Biểu đồ hoạt động trả lời câu hỏi ôn tập.....	26
10. Hình 3.7: Biểu đồ hoạt động tạo flashcard	27
11. Hình 3.8: Biểu đồ hoạt động sửa flashcard	28
12. Hình 4.1: Biểu đồ tuần tự cho hoạt động xem hồ sơ.	29
13. Hình 4.2: Biểu đồ tuần tự của hoạt động đăng nhập.	30
14. Hình 4.3: Biểu đồ tuần tự cho hoạt động xem bài học.	31
15. Hình 4.4: Biểu đồ tuần tự cho hoạt động trả lời câu hỏi	31
16. Hình 4.5: Biểu đồ tuần tự cho hoạt động chỉnh sửa hồ sơ người dùng, admin.	32
17. Hình 4.6: Biểu đồ tuần tự cho hoạt động thêm flashcard.	33
18. Hình 4.7: Biểu đồ tuần tự cho hoạt động chỉnh sửa flashcard.....	33
19. Hình 4.8: Biểu đồ tuần tự cho hoạt động xóa flashcard.	34
20. Hình 4.9: Biểu đồ tuần tự cho hoạt động đăng ký tài khoản người dùng.....	34
21. Hình 4.10: Biểu đồ tuần tự của hoạt động xem tài khoản của người quản trị.	35
22. Hình 4.11: Biểu đồ tuần tự của hoạt động xóa tài khoản của người quản trị.	35
23. Hình 4.12: Sơ đồ dữ liệu quan hệ của cơ sở dữ liệu trên MySQL.	38
24. Hình 4.13: Giao diện chính.....	38
25. Hình 4.14: Giao diện đăng nhập.....	39
26. Hình 4.15: Giao diện đăng ký.....	39
27. Hình 4.16.a: Giao diện profile người dùng.....	40
28. Hình 4.16.b: Giao diện quản trị viên.	40
29. Hình 4.17: Chỉnh sửa thông tin cá nhân.	41
30. Hình 4.18: Giao diện vocabulary.....	41
31. Hình 4.19: Giao diện kanji.	42
32. Hình 4.19: Giao diện grammar.....	42
33. Hình 4.20: Giao diện question.	42
34. Hình 4.21: Xem thẻ flashcard.....	43
35. Hình 4.22: Chỉnh sửa thẻ flashcard.	43
36. Hình 4.23: Thêm thẻ flashcard.	44

37. Hình 5.1: Cấu trúc các module trong front-end.....	46
38. Hình 5.1: Kiến trúc hệ thống website học tiếng Nhật.	47
39. Hình 5.2.1: Giao diện câu hỏi trên máy tính.	52
40. Hình 5.2.2: Giao diện câu hỏi trên máy tính.	53
41. Hình 5.2.3: Giao diện câu hỏi trên máy tính.	53
42. Hình 5.2.4: Giao diện câu hỏi trên máy tính.	54

MỤC LỤC BẢNG BIỂU

Table 1. Bảng mô tả ca sử dụng "Đăng ký"	18
Table 2. Bảng mô tả ca sử dụng "Đăng nhập"	19
Table 3. Bảng mô tả ca sử dụng "Đăng xuất"	19
Table 4. Bảng mô tả ca sử dụng "Xem thông tin"	19
Table 5. Bảng mô tả ca sử dụng chỉnh sửa thông tin.....	20
Table 6. Bảng mô tả ca sử dụng "Xem bài học"	20
Table 7. Bảng mô tả ca sử dụng "Làm câu hỏi ôn tập"	20
Table 8. Bảng mô tả ca sử dụng "Thêm flashcard"	21
Table 9. Bảng mô tả ca sử dụng "Xem flashcard"	21
Table 10. Bảng mô tả ca sử dụng "Sửa flashcard"	22
Table 11. Bảng mô tả ca sử dụng "Xóa flashcard"	22
Table 12. Bảng mô tả ca sử dụng "Xem tài khoản người dùng"	23
Table 13. Bảng mô tả ca sử dụng "Xóa tài khoản người dùng"	23
Table 14. Các giao diện lập trình ứng dụng (API).	36
Table 15. Bảng 5.1: Danh sách các ca kiểm thử thủ công.....	48
Table 16. Bảng 5.2: Ca kiểm thử đăng ký.	49
Table 17. Bảng 5.3: Ca kiểm thử đăng nhập.	49
Table 18. Bảng 5.4: Ca kiểm thử đăng xuất.	50
Table 19. Bảng 5.5: Kiểm thử màn quản lý tài khoản.....	50
Table 20. Bảng 5.6: Kiểm thử màn quản lý tài khoản với quản trị viên.	50
Table 21. Bảng 5.7: Kiểm thử màn bài học.....	51
Table 22. Bảng 5.8: Kiểm thử màn flashcard.....	51
Table 23. Bảng 5.9: Kiểm thử tính nhất quán của giao diện.	54

LỜI CẢM ƠN

Nhóm em xin được gửi lời cảm ơn chân thành tới thầy Nguyễn Đức Anh - giảng viên bộ môn Công nghệ phần mềm đã giảng dạy và hướng dẫn các phần kiến thức liên quan đồng thời góp ý để chúng em tạo ra sản phẩm này.

Website học tiếng Nhật - JAPPER và báo cáo này vẫn còn rất nhiều thiếu sót, vậy nên chúng em rất mong nhận được góp ý, nhận xét của thầy để hoàn thiện hay hướng phát triển thêm cho sản phẩm, nâng cao trình độ, kỹ năng của bản thân. Chúng em xin chân thành cảm ơn!

LỜI CAM ĐOAN

Nhóm chúng em bao gồm Cao Thị Phương Anh - 21020272 (trưởng nhóm) cùng các thành viên Nguyễn Thị Thanh Thủy - 21020411, Phạm Minh Tâm - msv, Đặng Thị Thanh Hiền - msv xin cam đoan rằng dự án Website học tiếng Nhật - JAPPER và báo cáo này không sao chép từ bất kỳ cá nhân hay tổ chức nào khác. Mọi tài liệu tham khảo đều được trích dẫn hợp pháp. Nếu lời cam đoan là sai sự thật thì nhóm chúng em xin nhận trách nhiệm và chịu hình thức kỷ luật theo quy định của thầy và nhà trường.

PHÂN CHIA CÔNG VIỆC

	Công việc	Chi tiết
Cao Thị Phương Anh	<ul style="list-style-type: none"> - Theo dõi tiến độ - Báo cáo - Làm backend 	<ul style="list-style-type: none"> - Báo cáo (kiểm thử, cài đặt) - Backend (Xác thực, phân quyền, API, Controller)
Nguyễn Thị Thanh Thủy	<ul style="list-style-type: none"> - Báo cáo - Làm backend 	<ul style="list-style-type: none"> - Báo cáo (Chương 2: Laravel, Mysql, Mô hình MVC và chương 3) - Vẽ biểu đồ - Dựng database, seed database (Progress, Controller, Question)
Đặng Thị Thanh Hiền	<ul style="list-style-type: none"> - Báo cáo - Làm frontend 	<ul style="list-style-type: none"> - Báo cáo (Chương 1, chương 6, chương 4: Thiết kế giao diện người dung) - Frontend (trang Account, create flashcard)
Phạm Minh Tâm	<ul style="list-style-type: none"> - Báo cáo - Frontend 	<ul style="list-style-type: none"> - Báo cáo (Chương 2: Các khối xây dựng của một ứng dụng web, JavaScript, React, Nextjs, Axios và chương 4) - Vẽ biểu đồ - Frontend (trang Lesson, Flashcard, Admin)

Chương 1 Đặt vấn đề

1.1. Bối cảnh bài toán

Người ta học và sử dụng tiếng Nhật với nhiều mục đích khác nhau để đáp ứng nhu cầu giao tiếp ngày càng lớn trong xã hội, để kiếm công việc tốt, có cơ hội thăng tiến trong công việc với mức lương thu nhập ổn định, hay để giành cơ hội đi du học và làm việc ở nước ngoài. Với GDP lớn thứ ba thế giới, các doanh nghiệp Nhật Bản thành lập tại Việt Nam tương đối nhiều, dân số Nhật Bản đang bị già hóa. Vì vậy, nhu cầu nguồn lao động chất lượng cao lớn, đặc biệt khi các lao động này còn biết thêm Tiếng Nhật thì quả là một điểm cộng sáng giá. Đây cũng chính là điểm nhấn khiến cho họ dễ dàng kiếm được một công việc tốt. Không chỉ vậy, Nhật Bản luôn được coi là một trong các quốc gia tốt nhất trên thế giới với công nghệ tiên tiến và chất lượng giáo dục nằm top đầu.

Học tiếng Nhật không chỉ là học kiến thức mà còn học cả văn hóa Nhật Bản. Văn hóa của họ được thể hiện qua nhiều lĩnh vực đời sống, đặc biệt là tác phong làm việc: đi đứng, ăn nói, chào hỏi. Học Tiếng Nhật sẽ rèn luyện cho các cá nhân tinh thần, văn hóa làm việc chuyên nghiệp và nhận được sự đánh giá cao.

Học tiếng Nhật hội nhập quốc tế, tiếng Nhật là một trong những ngôn ngữ được sử dụng nhiều nhất trên Internet, chỉ sau tiếng Nga và tiếng Đức. Ở Nhật, nhiều thể loại như anime (hoạt hình Nhật Bản) và manga (truyện tranh Nhật Bản) có một lượng lớn người hâm mộ trên khắp thế giới mong muốn nói và hiểu tiếng Nhật để tham gia, trao đổi và kết bạn với các tác giả, họa sĩ và người hâm mộ khác.

1.2. Giải pháp cho bài toán và hạn chế

Để tiếp cận với tiếng Nhật có nhiều cách như đến trung tâm học, mời giảng viên, gia sư đến tận nhà. Nhưng vấn đề lớn đối với người học là ít có thời gian học tiếng vì thời gian đi học, đi làm chiếm nhiều. Quan trọng là việc học ở trung tâm còn tốn thời gian di chuyển và chi phí đắt đỏ. Hơn nữa việc học thủ công còn nhiều hạn chế dù học nhiều nhưng vẫn khó nhớ từ vựng, không áp dụng được những mẫu ngữ pháp vào thực tế, việc học trên sách vở bị hạn chế về không gian và lượng kiến thức cung cấp không cập nhật thường xuyên.

1.3. Giải quyết hạn chế

Để giải quyết các hạn chế nêu trên, dự án này xây dựng hệ thống học tiếng Nhật online miễn phí. Ngoài đọc sách, sử dụng ứng dụng, đi học tại các trung tâm, rất nhiều bạn trẻ khác đang sử dụng rất nhiều website tự học tiếng Nhật để lên trình hằng ngày. Trong khi giáo trình tiếng Nhật bị giới hạn bởi số trang sách, thì các website tự học tiếng Nhật lại có thể chứa tới hàng nghìn kiến thức và ví dụ đa dạng, phù hợp với từng hoàn cảnh. Không chỉ được thường xuyên cập nhật kiến thức và thông tin hữu ích, các website còn giúp bạn nhanh chóng tiếp cận tới những ngữ pháp, từ vựng, hán tự chỉ trong vài cú nhấp chuột. Việc tra cứu từ đó cũng trở nên dễ dàng hơn rất nhiều. Việc học tiếng Nhật qua web còn rất thuận tiện cho phép bạn học mọi lúc mọi nơi. Website tiếng

Nhật được hỗ trợ người học theo dõi tiến độ học, kết quả đạt được. Học từ vựng sẽ được đơn giản hóa bằng cách học theo chủ đề, nhóm từ giúp người học có thể tự hệ thống được từ vựng đã học, không bị nhầm lẫn giữa các từ trong cùng một nhóm. Ngữ pháp được chỉ rõ những trường hợp được sử dụng kèm theo ví dụ cụ thể giúp người học có thể dễ dàng vận dụng vào thực tế. Theo khoa học nghiên cứu, với một lượng kiến thức cần nhớ, thì sau 1 ngày tiếp thu, người học chỉ còn nhớ 35.7% lượng kiến thức và sau 1 tháng, lượng kiến thức chỉ còn khoảng 21% trong não bộ. Vì thế, việc ôn tập lại kiến thức đóng vai trò rất quan trọng trong quá trình ghi nhớ. Do đó, flashcard là công cụ ôn tập rất hiệu quả. Nắm được nhu cầu đó website xây dựng chức năng để người dùng tự tạo flashcard tăng khả năng tự học, ghi nhớ, tự củng cố lại kiến thức còn hổng. Để đáp ứng nhu cầu tự học tiếng Nhật tại nhà đang ngày càng gia tăng, chúng mình sẽ giới thiệu với các bạn website học tiếng Nhật - JAPPER.

1.4. Xây dựng và ý nghĩa hệ thống

Xây dựng hệ thống: hệ thống gồm hai đối tượng là quản trị viên và người học. Học viên có thể đăng ký tạo tài khoản, đăng nhập để bắt đầu học. Các bài học sẽ gồm bốn phần là: vocabulary, kanji, grammar, question. Vocabulary là nhóm các từ vựng theo chủ đề điều này sẽ giúp bạn hệ thống được lượng từ vựng đã học, dễ dàng ghi nhớ và liên tưởng mà không bị nhầm lẫn giữa từ này với từ khác trong cùng một nhóm. Kanji là các thẻ xem nhanh với âm on, âm kun, các từ vựng liên quan đến chữ kanji đó giúp người học hiểu tường tận nhất. Grammar là phần ngữ pháp của bài học được giải thích rõ ràng về các trường hợp sử dụng và các ví dụ cụ thể. Cuối cùng là phần question để người học trả lời câu hỏi củng cố kiến thức sau bài học. Học viên còn có thể tự tạo flashcard để ghi nhớ từ vựng. Các thẻ flashcard sẽ được cập nhật cả ngày tạo và người đọc có thể tự do sửa, xóa thẻ flashcard của mình. Ở trang profile, người học có thể cập nhật lại tên và email, có thể xem lại các flashcard đã tạo vào ngày nào, các bài học đã hoàn thành. Quản trị viên có quyền sử dụng các tài khoản đã được cấp sẵn từ trước để đăng nhập vào hệ thống và sử dụng các chức năng dưới quyền của một quản trị viên. Quản trị viên có thể xem và xóa người dùng.

Ý nghĩa hệ thống: Hệ thống xây dựng tạo ra một môi trường học tập mới mẻ cho người học có đam mê với tiếng Nhật. Với giao diện ưa nhìn, thân thiện với người dùng khiến việc học tiếng Nhật trở nên dễ dàng, hứng thú hơn. Từ vựng và ngữ pháp thông dụng, gần gũi với cuộc sống. Các flashcard được người học tự do tạo, sửa, xóa cập nhật theo ngày kiểm soát được nội dung cần ghi nhớ theo ngày. Mỗi quá trình tạo flashcard sẽ được ghi nhớ lại, giúp bạn có thể tăng khả năng tự học, chú trọng hơn những nội dung chưa nắm rõ.

1.5. Nội dung báo cáo

Phần còn lại của báo cáo được trình bày như sau: Chương 2 trình bày các kiến thức nền tảng bao gồm các phần Reactjs, Next.js, Javascript, Axios, Laravel, Database, Mysql, mô hình MVC. Chương 3 trình bày phân tích và đặc tả yêu cầu. Chương 4 phân tích thiết kế và xây dựng hệ thống. Chương 5 cài đặt và kiểm thử. Cuối cùng, chương 6 kết luận và định hướng phát triển.

Chương 2 Kiến thức nền tảng

2.1. Các khối xây dựng của một ứng dụng web

- User Interface (Giao diện người dùng): cách người dùng sẽ sử dụng và tương tác với ứng dụng.
- Routing (Định tuyến): cách người dùng điều hướng giữa các phần khác nhau trong ứng dụng.
- Data Fetching (Tìm nạp dữ liệu): cách lấy dữ liệu.
- Rendering (Kết xuất): kết xuất nội dung tĩnh hoặc động khi nào và ở đâu.
- Integrations (Tích hợp): cách kết nối với dịch vụ bên thứ ba (CMS, xác thực, thanh toán, v.v.).
- Infrastructure (Cơ sở hạ tầng): nơi bạn triển khai, lưu trữ và chạy mã ứng dụng (Serverless, CDN, Edge, v.v.).
- Performance (Hiệu suất): cách tối ưu hóa ứng dụng cho người dùng cuối.
- Scalability (Khả năng mở rộng): cách ứng dụng thích ứng khi, dữ liệu và lưu lượng truy cập tăng lên.
- Developer Experience (Trải nghiệm của nhà phát triển): kinh nghiệm xây dựng và duy trì ứng dụng.

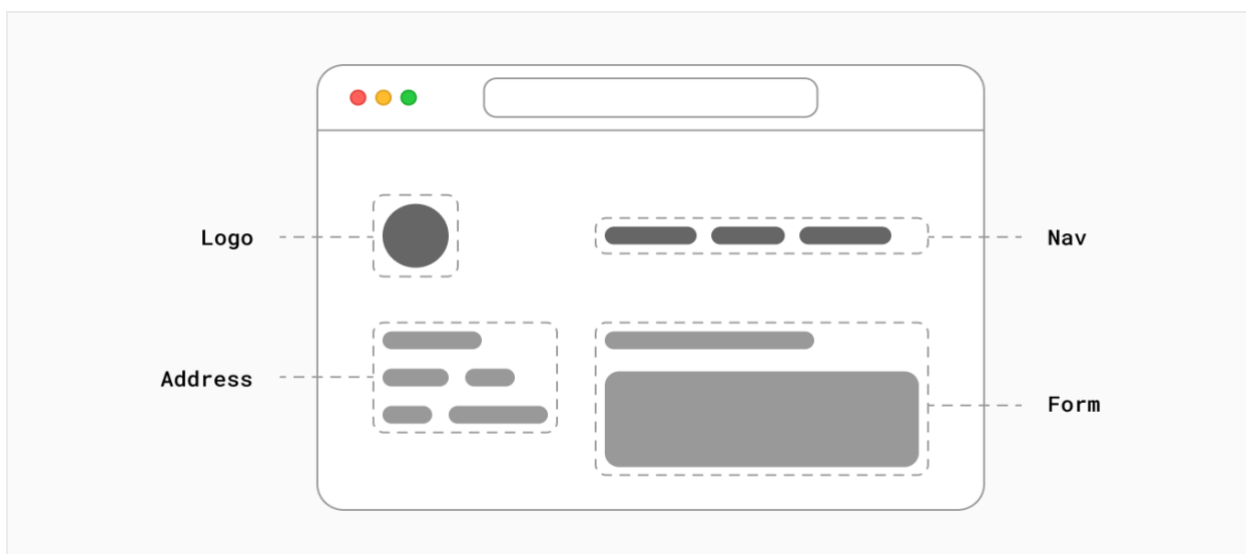
Đối với mỗi phần của ứng dụng, cần quyết định xem xây dựng giải pháp nào hay sử dụng các công cụ khác như các thư viện và framework.

2.2. JavaScript

JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. JavaScript dần được biết đến như một công nghệ phía trình duyệt để làm cho các ứng dụng web linh hoạt hơn. Sử dụng JavaScript, các trình duyệt có thể phản hồi tương tác của người dùng và thay đổi bố cục của nội dung trên trang web. Khi ngôn ngữ này phát triển hoàn thiện, các nhà phát triển JavaScript đã thiết lập các thư viện, framework và cách thức lập trình cũng như bắt đầu sử dụng ngôn ngữ này bên ngoài trình duyệt web. Ngày nay, bạn có thể sử dụng JavaScript để thực hiện hoạt động phát triển cả ở phía máy khách và máy chủ.

2.3. React

React là một thư viện JavaScript để xây dựng giao diện người dùng tương tác - nói đến các yếu tố mà người dùng nhìn thấy và tương tác trên màn hình.

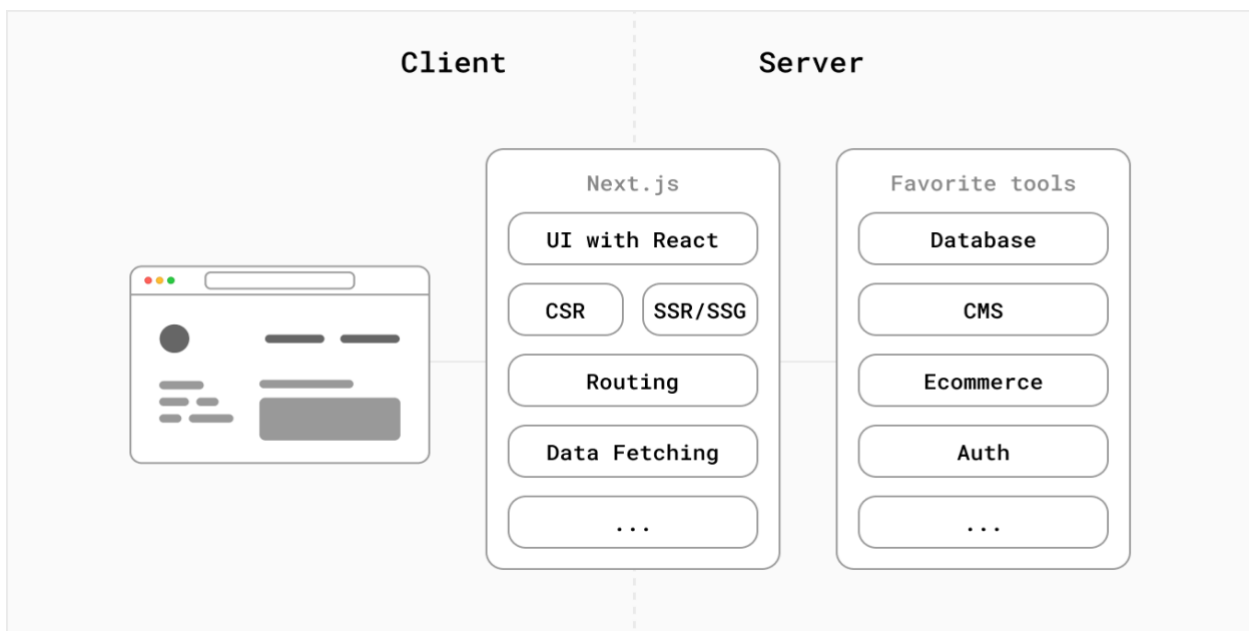


Hình 2.1: Giao diện người dùng tương tác.

React cung cấp các chức năng hữu ích để xây dựng giao diện người dùng. Nó gần như không có vấn đề với các khía cạnh khác của việc xây dựng ứng dụng. Điều này đã dẫn đến một hệ thống các công cụ và giải pháp của bên thứ ba phát triển mạnh mẽ. Tuy nhiên, điều đó cũng có nghĩa là việc xây dựng một ứng dụng React hoàn chỉnh từ đầu đòi hỏi một số nỗ lực. Các nhà phát triển cần dành thời gian để định cấu hình các công cụ và phát triển lại các giải pháp cho các yêu cầu ứng dụng phổ biến.

2.4. Next.js

Next.js là một React framework linh hoạt cung cấp các khối xây dựng để tạo các ứng dụng web nhanh. Next.js xử lý công cụ và cấu hình cần thiết cho React, đồng thời cung cấp cấu trúc, tính năng và tối ưu hóa bổ sung cho ứng dụng.



Hình 2.2: Nextjs cung cấp các khối xây dựng để tạo các ứng dụng web nhanh.

Người lập trình có thể sử dụng React để xây dựng giao diện người dùng của mình, sau đó áp dụng các tính năng Next.js để giải quyết các yêu cầu phổ biến của ứng dụng như định tuyến, tìm nạp dữ liệu, tích hợp đồng thời cải thiện trải nghiệm của nhà phát triển và người dùng cuối.

2.5. Axios

Axios là một thư viện HTTP Client dựa trên Promise dành cho Node.js và trình duyệt. Nó có tính đẳng hình (tức là cùng codebase có thể chạy trong cả trình duyệt và Node.js). Ở phía server thì nó sử dụng native module HTTP trong node.js, còn ở phía client (trình duyệt) thì nó sử dụng XMLHttpRequest.

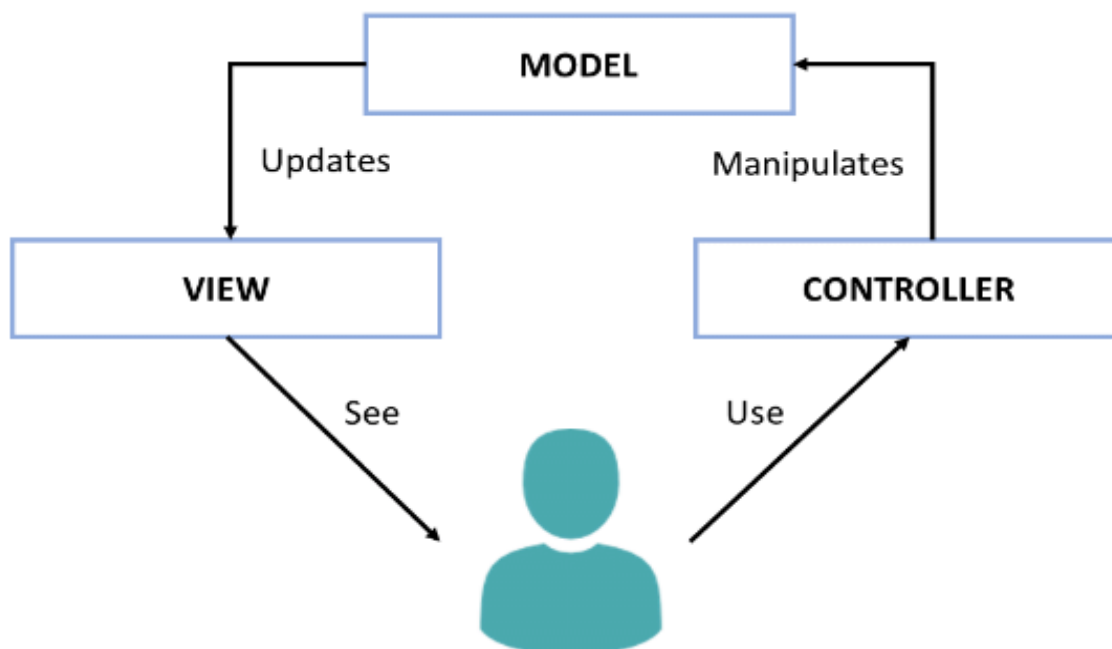
Tính năng:

- Tạo request từ trình duyệt bằng XMLHttpRequest
- Tạo request từ Node.js bằng HTTP
- Hỗ trợ Promise API
- Đón chặn request và response
- Biến đổi dữ liệu request và response
- Bỏ request
- Tự động chuyển đổi cho dữ liệu JSON
- Hỗ trợ phía client bảo vệ chống lại XSRF

2.6. MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System), MySQL lưu trữ và quản lý các dữ liệu dưới dạng bảng và được sử dụng trong các ứng dụng nhằm truy cập, thao tác, và tìm kiếm dữ liệu một cách nhanh chóng và hiệu quả.

2.7. Mô hình MVC



Hình 2.1. Sơ đồ kiến trúc mô hình MVC.

MVC là một kiến trúc phần mềm được sử dụng để phân tách các thành phần của ứng dụng web thành ba phần riêng biệt là: Model, View và Controller. Mỗi thành phần sẽ đảm nhận một vai trò khác nhau:

- Model: đại diện cho các đối tượng và logic xử lý dữ liệu. Model là nơi lưu trữ, truy vấn và xử lý các dữ liệu được lưu trữ trong cơ sở dữ liệu.
- View: đại diện cho phần giao diện người dùng (UI) của ứng dụng. View là nơi hiển thị dữ liệu được truyền từ Model và tương tác với người dùng.
- Controller: đại diện cho các hành động của người dùng và quản lý quá trình xử lý dữ liệu giữa Model và View. Controller tiếp nhận các yêu cầu từ View và gọi các phương thức phù hợp của Model để truy vấn và xử lý dữ liệu, sau đó trả kết quả về View để hiển thị.

2.8. Laravel Framework

Laravel là một framework phát triển ứng dụng web, được sử dụng rộng rãi bởi nó cung cấp nhiều tính năng và công cụ hữu ích trong quá trình phát triển ứng dụng web. Laravel tổ chức mã nguồn dựa trên mô hình MVC giúp tách biệt tầng dữ liệu, xử lý logic và giao diện người dùng, việc phân chia rõ ràng này giúp cho quá trình phát triển và bảo trì ứng dụng trở nên dễ dàng hơn. Ngoài ra, Laravel còn cung cấp các tính năng như: định tuyến (routing), xác thực (authentication), phiên (session), caching, migration giúp cho lập trình viên có thể nhanh chóng hoàn thành các chức năng cơ bản và tập trung phát triển những chức năng mới chuyên biệt hơn cho ứng dụng của mình.

Chương 3 Phân tích và đặc tả yêu cầu

3.1 Phân tích yêu cầu

Thu thập yêu cầu

Ứng dụng của chúng em hướng đến tất cả những người học tiếng Nhật muốn sử dụng giáo trình Mina no Nihongo, phù hợp cho nhiều đối tượng từ học chuyên ngành đến tự học cơ bản. Mục tiêu của ứng dụng là dễ sử dụng, đầy đủ nội dung và tiện lợi cho người học.

Yêu cầu chức năng

Với người dùng:

- Đăng ký, đăng nhập, đăng xuất.
- Xem thông tin: tên người dùng, email, flashcard đã tạo, các bài học đã hoàn thành.
- Chỉnh sửa thông tin: thay đổi tên, email.
- Xem bài học:
 - Xem tổng quan các bài học, có thể click vào từng phần: từ vựng, ngữ pháp, kanji theo từng bài học để xem chi tiết.
 - Trong mục từ vựng: người dùng có thể xem từ vựng, kanji và nghĩa của từ.
 - Trong mục ngữ pháp: người dùng có thể xem: cấu trúc ngữ pháp, giải thích, ví dụ.
 - Trong mục kanji: người dùng có thể xem: kanji, âm On, âm Kun, từ vựng chứa kanji đó.
 - Làm câu hỏi ôn tập: Người dùng có thể trả lời câu hỏi ôn tập theo từng bài và đáp án từng câu. Nếu trả lời đúng hết các câu hỏi người dùng sẽ được tính là đã hoàn thành bài học và hiển thị bài học đó tại trang thông tin cá nhân.
 - Sử dụng flashcard: Người dùng có thể thêm, xem, sửa, xoá flashcard.

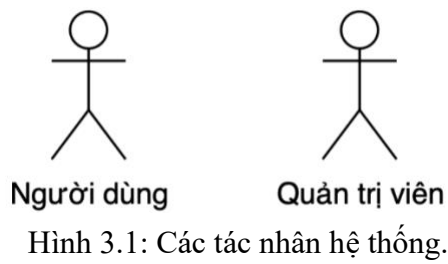
Với quản trị viên:

- Đăng nhập vào hệ thống: Quản trị viên sử dụng các tài khoản đã được cấp sẵn từ trước để đăng nhập vào hệ thống và sử dụng các chức năng dưới quyền của một quản trị viên.
- Quản lý hệ thống: Quản trị viên có thể xem và xoá các tài khoản trong hệ thống.

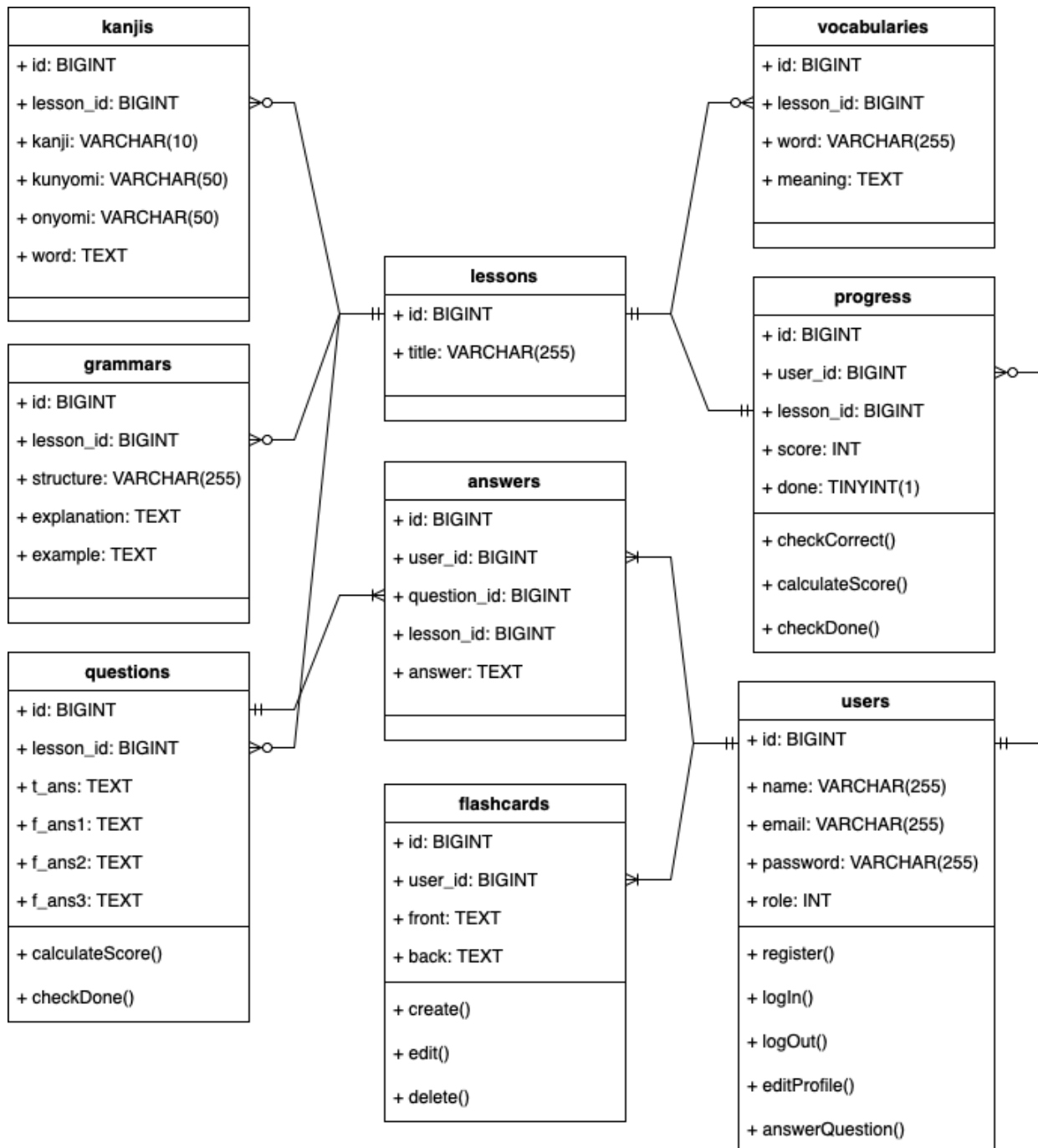
Yêu cầu phi chức năng

Giao diện người dùng đơn giản, đẹp mắt, dễ sử dụng để phù hợp với nhiều đối tượng. Mã nguồn được phân chia rõ ràng thành các module để có thể bổ sung, mở rộng thêm các tính năng mới.

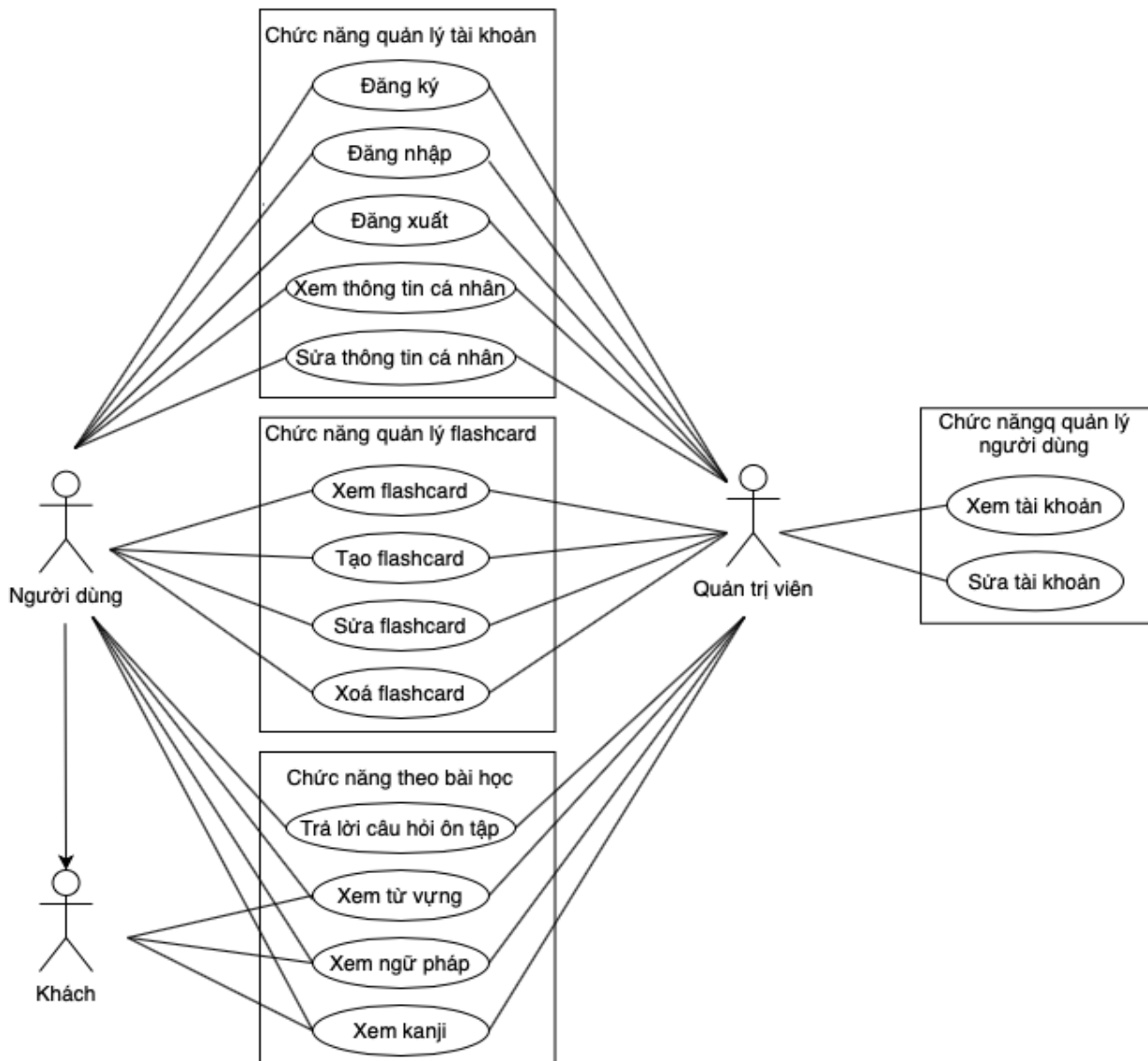
3.2 Phân tích và đặc tả ca sử dụng



Hình 3.1: Các tác nhân hệ thống.



Hình 3.2: Biểu đồ lớp của hệ thống.



Hình 3.3: Biểu đồ use case chung của hệ thống.

3.2.1 Ca sử dụng “Đăng ký”

Table 1. Bảng mô tả ca sử dụng "Đăng ký"

Mô tả	Người dùng tạo tài khoản mới để sử dụng các tính năng trên hệ thống
Tác nhân	Người dùng
Tiền điều kiện	Người dùng có tài khoản email
Hậu điều kiện	Tài khoản của người dùng được tạo và người dùng được đăng nhập vào hệ thống
Luồng sự kiện chính	Người dùng truy cập và trang đăng ký Người dùng nhập các trường thông tin trên form đăng ký Người dùng gửi biểu mẫu đăng ký

	Hệ thống xử lý yêu cầu, lưu tài khoản vào database và điều hướng sang trang chủ
Luồng sự kiện rẽ nhánh	Nếu thông tin đăng ký không hợp lệ (không đúng hoặc không đủ), hệ thống sẽ báo lỗi và yêu cầu người dùng nhập lại thông tin

3.2.2 Ca sử dụng “Đăng nhập”

Table 2. Bảng mô tả ca sử dụng "Đăng nhập"

Mô tả	Người dùng sử dụng tài khoản đã đăng ký để đăng nhập vào hệ thống, một số tính năng chỉ được cung cấp khi người dùng đã đăng nhập.
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng: Đã đăng ký tài khoản Quản trị viên: Đã được cấp tài khoản với quyền admin
Hậu điều kiện	Đăng nhập thành công vào hệ thống
Luồng sự kiện chính	Người dùng truy cập vào trang đăng nhập Người dùng nhập các trường thông tin trên form đăng nhập Người dùng gửi biểu mẫu đăng nhập Hệ thống xử lý yêu cầu, kiểm tra tài khoản trong database và chuyển hướng sang trang chủ
Luồng sự kiện rẽ nhánh	Nếu người dùng nhập sai thông tin tài khoản, hệ thống sẽ báo đăng nhập không thành công và yêu cầu nhập lại.

3.2.3 Ca sử dụng “Đăng xuất”

Table 3. Bảng mô tả ca sử dụng "Đăng xuất"

Mô tả	Người dùng đăng xuất khỏi hệ thống
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Đã đăng nhập vào hệ thống
Hậu điều kiện	Đăng xuất thành công khỏi hệ thống
Luồng sự kiện chính	Người dùng nhấn vào “Sign out”, hệ thống chuyển hướng đến trang đăng nhập
Luồng sự kiện rẽ nhánh	Không có

3.2.4 Ca sử dụng “Xem thông tin”

Table 4. Bảng mô tả ca sử dụng "Xem thông tin"

Mô tả	Người dùng có thể xem thông tin các nhân bao gồm: tên, email, các flashcard đã tạo, các bài học đã hoàn thành
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Đã đăng nhập vào hệ thống
Hậu điều kiện	Truy cập vào trang cá nhân

Luồng sự kiện chính	Người dùng nhấn “Account” để truy cập vào trang cá nhân, hệ thống hiển thị các thông tin cá nhân
Luồng sự kiện rẽ nhánh	Nếu người dùng chưa đăng nhập, hệ thống sẽ chuyển hướng sang trang đăng nhập.

3.2.5 Ca sử dụng “Chỉnh sửa thông tin”

Table 5. Bảng mô tả ca sử dụng chỉnh sửa thông tin

Mô tả	Thay đổi thông tin tài khoản cá nhân bao gồm: Fullname và Email
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Đã đăng nhập vào hệ thống
Hậu điều kiện	Hiển thị thông tin cá nhân được thay đổi
Luồng sự kiện chính	Người dùng truy cập vào trang cá nhân và nhấn “Edit” Hệ thống hiển thị form Update Profile Người dùng điền các trường thông tin và nhấn “OK” Hệ thống xử lý thông tin và hiển thị trang cá nhân với thông tin đã được thay đổi
Luồng sự kiện rẽ nhánh	Nếu người dùng chưa điền đủ các trường và nhấn “OK”, hệ thống sẽ báo lỗi và yêu cầu nhập đủ

3.2.6 Ca sử dụng “Xem bài học”

Table 6. Bảng mô tả ca sử dụng "Xem bài học"

Mô tả	Xem danh sách các bài học, từng phần của bài học bao gồm: từ vựng, ngữ pháp, kanji.
Tác nhân	Người dùng, quản trị viên, khách
Tiền điều kiện	Không có
Hậu điều kiện	Hiển thị danh sách các bài học và các mục của bài học
Luồng sự kiện chính	Người dùng truy cập vào trang Lesson, hệ thống hiển thị danh sách các bài học Người dùng nhấn “Vocabulary”, hệ thống hiển thị danh sách từ vựng trong bài đó Người dùng nhấn “Grammar”, hệ thống hiển thị danh sách ngữ pháp trong bài đó Người dùng nhấn “Kanji”, hệ thống hiển thị danh sách hán tự trong bài đó
Luồng sự kiện rẽ nhánh	Không có

3.2.7 Ca sử dụng “Làm câu hỏi ôn tập”

Table 7. Bảng mô tả ca sử dụng "Làm câu hỏi ôn tập"

Mô tả	Người dùng có thể trả lời câu hỏi trắc nghiệm liên quan đến nội dung của từng bài
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Hậu điều kiện	Hiển thị điểm và kết quả câu trả lời
Luồng sự kiện chính	Người dùng truy cập vào trang Lesson, hệ thống hiển thị danh sách các bài học, người dùng nhấn “Question” trong một bài học, hệ thống hiển thị câu hỏi trắc nghiệm Người dùng chọn đáp án và nhấn “Submit”, hệ thống trả về kết quả và chuyển sang câu tiếp theo.
Luồng sự kiện rẽ nhánh	Người dùng ấn submit nhiều hơn 1 lần, nếu câu trả lời trước đó đã trả lời đúng, hệ thống không cộng điểm mà trả về thông báo “You have got score for this question”, nếu trước đó trả lời sai thì cộng điểm bình thường

3.2.8 Ca sử dụng “Thêm flashcard”

Table 8. Bảng mô tả ca sử dụng "Thêm flashcard"

Mô tả	Người dùng tạo flashcard bằng cách thêm thông tin mặt trước và mặt sau của flashcard
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Hậu điều kiện	Hiển thị flashcard được tạo và lưu thành công
Luồng sự kiện chính	Người dùng truy cập vào trang Flashcard, nhấn “Create” hoặc biểu tượng dấu cộng. Hệ thống hiển thị form tạo flashcard Người dùng nhấn “Add”. Hệ thống bổ sung thêm ô tạo flashcard. Người dùng điền thông tin flashcard và nhấn “OK”. Hệ thống lưu lại flashcard và hiển thị flashcard vừa tạo
Luồng sự kiện rẽ nhánh	Người dùng không điền thông tin flashcard và nhấn “OK”, hệ thống thông báo thông tin flashcard không được trống

3.2.9 Ca sử dụng “Xem flashcard”

Table 9. Bảng mô tả ca sử dụng "Xem flashcard"

Mô tả	Người dùng xem lần lượt các flashcard đã tạo, click chuột vào flashcard để xem mặt sau của flashcard
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống
Hậu điều kiện	Hiển thị danh sách flashcard đã tạo

Luồng sự kiện chính	<p>Người dùng truy cập vào trang Flashcard, hệ thống hiển thị lần lượt từng flashcard</p> <p>Người dùng nhấn biểu tượng mũi tên, hệ thống chuyển sang flashcard tiếp theo</p> <p>Người dùng nhấn vào flashcard, hệ thống hiển thị mặt còn lại của flashcard</p>
Luồng sự kiện rẽ nhánh	Người dùng chưa đăng nhập và nhấn vào “Flashcard”, hệ thống thông báo người dùng chưa đăng nhập và chuyển đến trang đăng nhập

3.2.10 Ca sử dụng “Sửa flashcard”

Table 10. Bảng mô tả ca sử dụng "Sửa flashcard"

Mô tả	Người dùng sửa flashcard mình đã tạo
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống, đã có flashcard được tạo
Hậu điều kiện	Hiển thị flashcard sau khi đã sửa
Luồng sự kiện chính	<p>Người dùng truy cập vào trang Flashcard, hệ thống hiển thị lần lượt từng flashcard</p> <p>Người dùng nhấn biểu tượng mũi tên để chuyển đến flashcard muốn sửa và nhấn “Edit”, hệ thống hiện form sửa flashcard</p> <p>Người dùng điền các trường trong form và nhấn “Save”, hệ thống xử lý thông tin và hiển thị flashcard đã sửa</p>
Luồng sự kiện rẽ nhánh	Người dùng không điền thông tin flashcard và nhấn “Save”, hệ thống thông báo thông tin flashcard không được trống

3.2.11 Ca sử dụng “Xóa flashcard”

Table 11. Bảng mô tả ca sử dụng "Xóa flashcard"

Mô tả	Người dùng xóa flashcard mình đã tạo
Tác nhân	Người dùng, quản trị viên
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống, đã có flashcard được tạo
Hậu điều kiện	Flashcard bị xóa khỏi danh sách
Luồng sự kiện chính	<p>Người dùng truy cập vào trang Flashcard, hệ thống hiển thị lần lượt từng flashcard</p> <p>Người dùng nhấn biểu tượng mũi tên để chuyển đến flashcard muốn xóa và nhấn “Delete”. Hệ thống xóa flashcard khỏi danh sách và trả về danh sách flashcard sau khi xóa</p>
Luồng sự kiện rẽ nhánh	Không có

3.2.12 Ca sử dụng “Xem tài khoản người dùng”

Table 12. Bảng mô tả ca sử dụng "Xem tài khoản người dùng"

Mô tả	Quản trị viên có thể xem được thông tin về tất cả các tài khoản trong hệ thống: Fullname, Email, Role (Admin: 1 và User: 0), Date created
Tác nhân	Quản trị viên
Tiền điều kiện	Đã đăng nhập vào hệ thống bằng tài khoản có quyền admin
Hậu điều kiện	Hiển thị sách tất cả các tài khoản trong hệ thống
Luồng sự kiện chính	Truy cập vào trang Account, hệ thống hiển thị danh sách tất cả các tài khoản trong hệ thống
Luồng sự kiện rẽ nhánh	Không có

3.2.13 Ca sử dụng “Xóa tài khoản người dùng”

Table 13. Bảng mô tả ca sử dụng "Xóa tài khoản người dùng"

Mô tả	Xóa tài khoản người dùng khỏi hệ thống
Tác nhân	Quản trị viên
Tiền điều kiện	Đã đăng nhập vào hệ thống bằng tài khoản có quyền admin, đã có người dùng đăng ký tài khoản
Hậu điều kiện	Tài khoản người dùng bị xóa khỏi danh sách
Luồng sự kiện chính	Quản trị viên truy cập vào trang Account, hệ thống hiển thị danh sách tất cả các tài khoản trong hệ thống Quản trị viên nhấn nút “Delete”, hệ thống xóa tài khoản người dùng
Luồng sự kiện rẽ nhánh	Không có

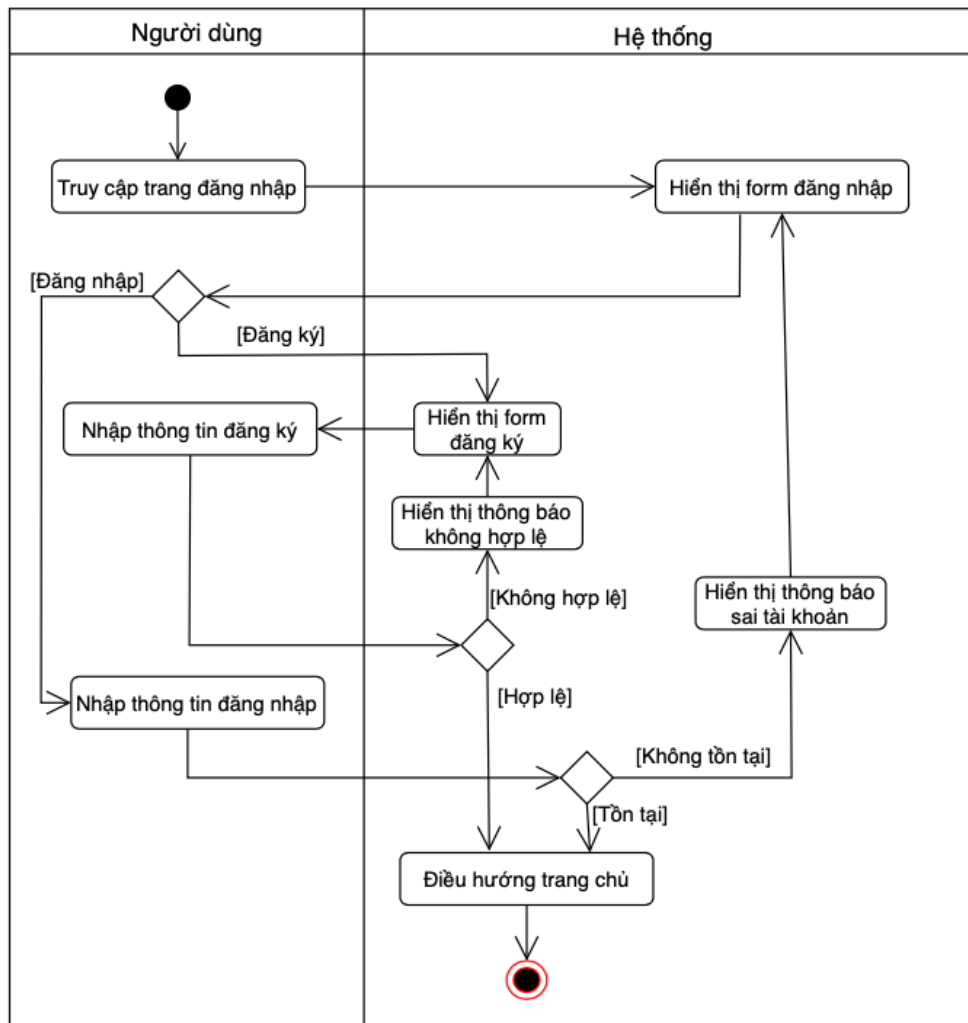
3.3 Mô tả các biểu đồ hoạt động

3.3.1 Hoạt động “Quản lý xác thực tài khoản”

Đối tượng: người dùng, quản trị viên.

Mô tả hoạt động: Người dùng truy cập trang đăng nhập, hệ thống hiển thị form đăng nhập.

- Nếu đã có tài khoản thì thực hiện nhập thông tin và đăng nhập, hệ thống sẽ kiểm tra tài khoản có tồn tại hay không. Nếu có sẽ chuyển hướng đến trang chủ, nếu không trở về trang đăng nhập.
- Nếu chưa có tài khoản thì thực hiện đăng ký, hệ thống sẽ hiển thị form đăng ký, người dùng nhập thông tin. Nếu thông tin hợp lệ, hệ thống sẽ chuyển hướng đến trang chủ, nếu thông tin không hợp lệ, hệ thống sẽ trở về trang đăng ký.

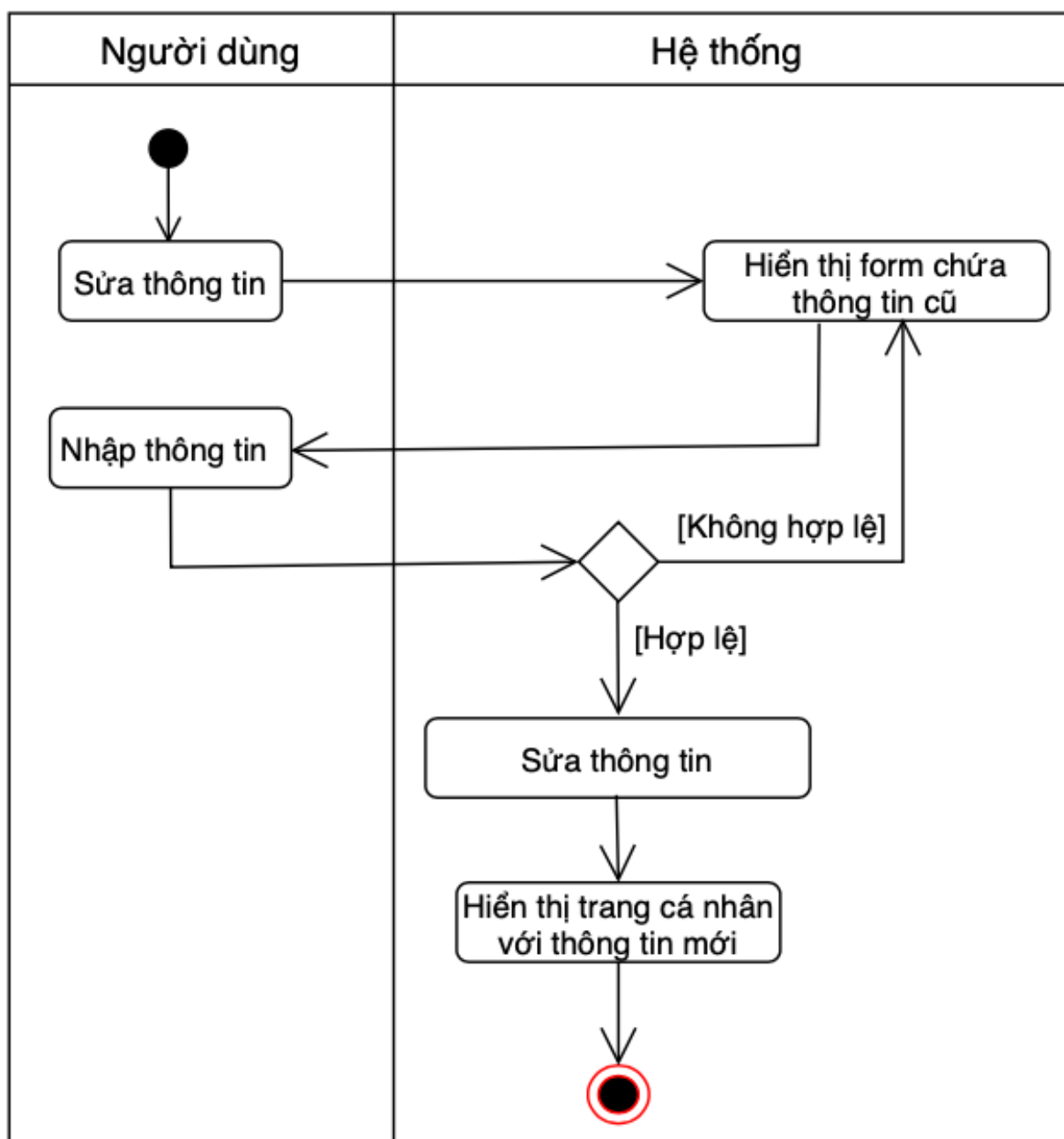


Hình 3.4: Biểu đồ hoạt động quản lý xác thực tài khoản

3.3.2 Hoạt động “Sửa thông tin”

Đối tượng: người dùng, quản trị viên.

Mô tả hoạt động: Ở trang cá nhân, người dùng nhấn “Edit” và điền vào các thông tin rồi nhấn “OK”. Hệ thống sẽ kiểm tra các trường có rỗng hay không, nếu rỗng thì yêu cầu người dùng nhập đủ, ngược lại sẽ xử lý thông tin và hiển thị trang cá nhân sau khi đã sửa.

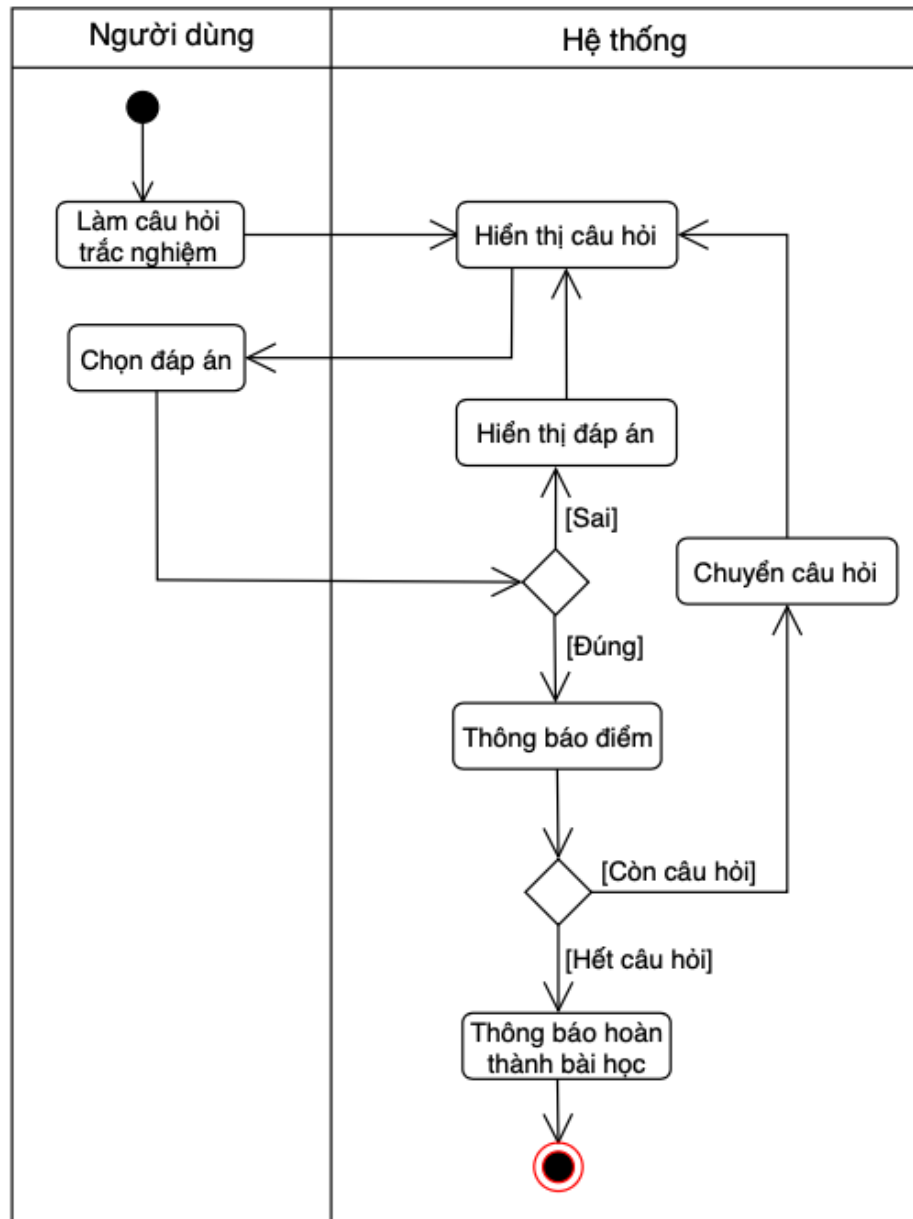


Hình 3.5: Biểu đồ hoạt động sửa thông tin

3.3.3 Hoạt động “Trả lời câu hỏi ôn tập”

Đối tượng: người dùng, quản trị viên.

Mô tả hoạt động: Người dùng truy cập vào trang Question, hệ thống hiển thị lần lượt từng câu hỏi, người dùng có thể chọn đáp án hoặc chuyển câu. Nếu chọn đáp án và đáp án là đúng thì người dùng được cộng điểm và chuyển sang câu tiếp theo, nếu sai thì hệ thống sẽ hiển thị đáp án. Khi hoàn thành hết các câu hỏi, người dùng sẽ được thông báo là đã hoàn thành bài học.

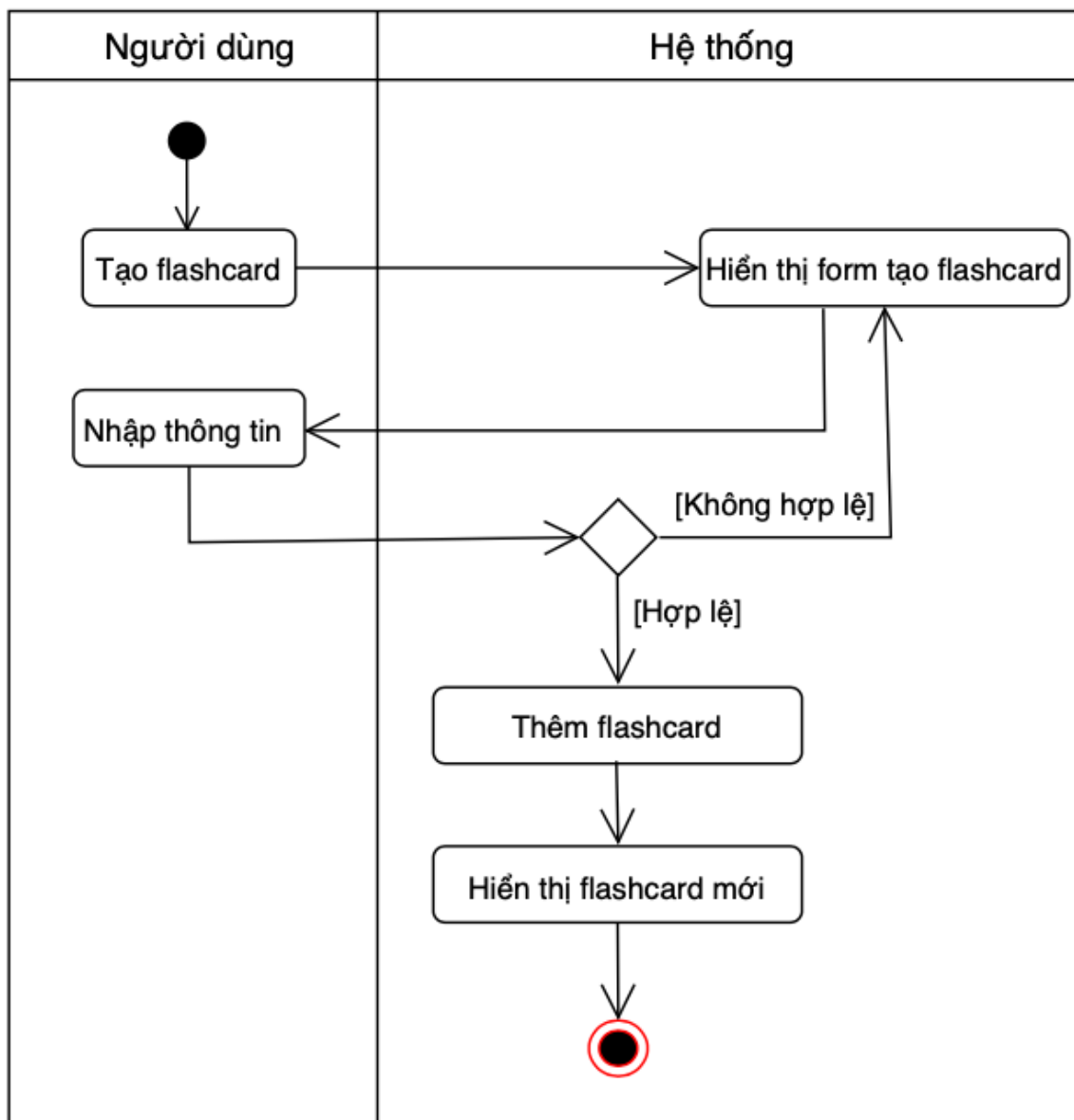


Hình 3.6: Biểu đồ hoạt động trả lời câu hỏi ôn tập

3.3.4 Hoạt động “Tạo flashcard”

Đối tượng: người dùng, quản trị viên.

Mô tả hoạt động: Người dùng truy cập vào trang Flashcard, hệ thống hiển thị flashcard, người dùng nhấn “Create” và nhập thông tin để tạo flashcard, nếu thông tin hợp lệ hệ thống sẽ lưu và hiển thị flashcard mới, nếu thông tin không hợp lệ, hệ thống sẽ báo lỗi và yêu cầu nhập lại.

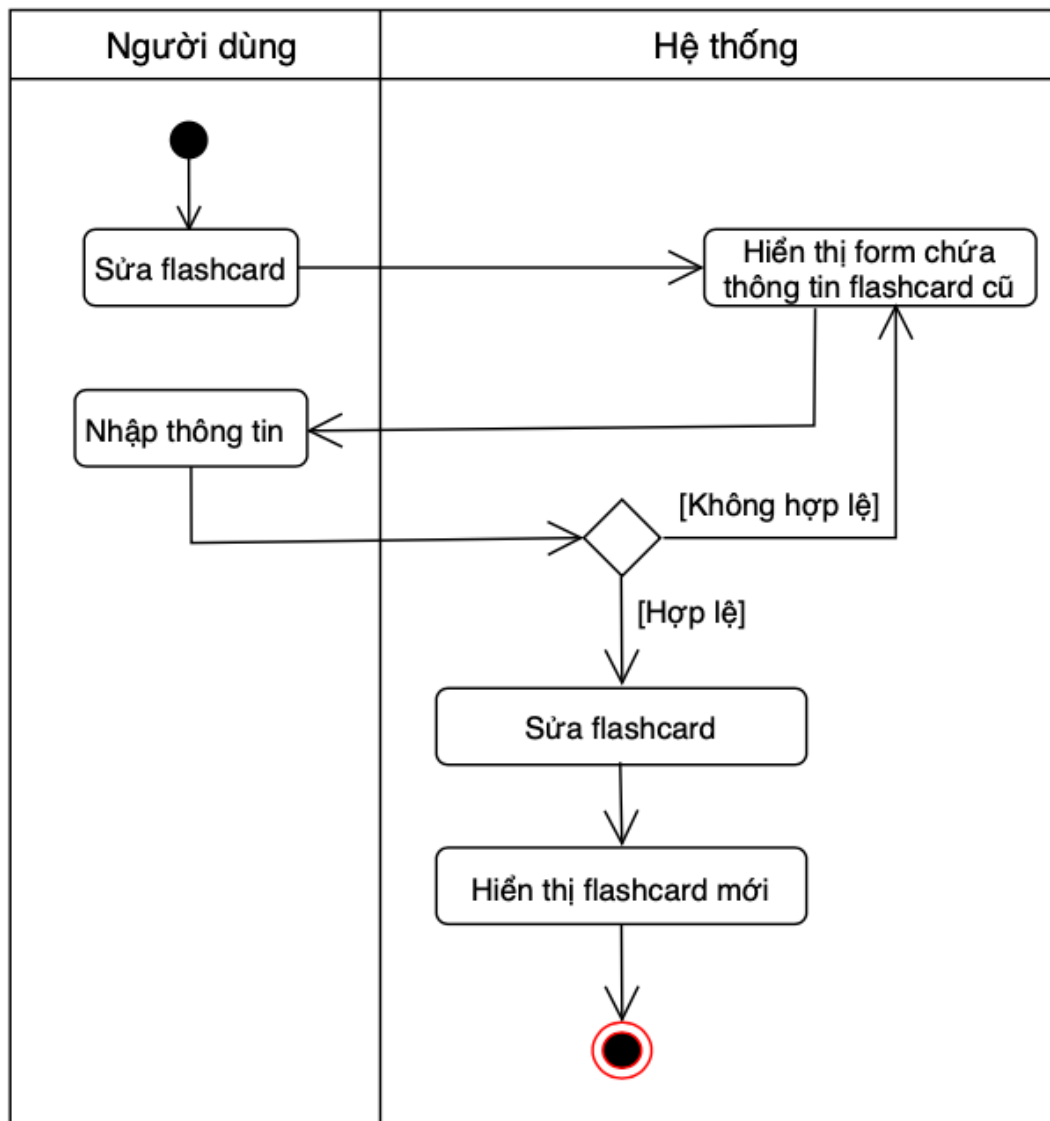


Hình 3.7: Biểu đồ hoạt động tạo flashcard

3.3.5 Hoạt động “Sửa flashcard”

Đối tượng: người dùng, quản trị viên.

Mô tả hoạt động: Tại flashcard muốn sửa, người dùng nhấn “Edit”, hệ thống sẽ hiện ra form chứa thông tin của flashcard hiện tại. Người dùng nhập thông tin để tạo flashcard, nếu thông tin hợp lệ hệ thống sẽ lưu và hiển thị flashcard mới, nếu thông tin không hợp lệ, hệ thống sẽ hiển thị lại form flashcard yêu cầu người dùng nhập lại.



Hình 3.8: Biểu đồ hoạt động sửa flashcard

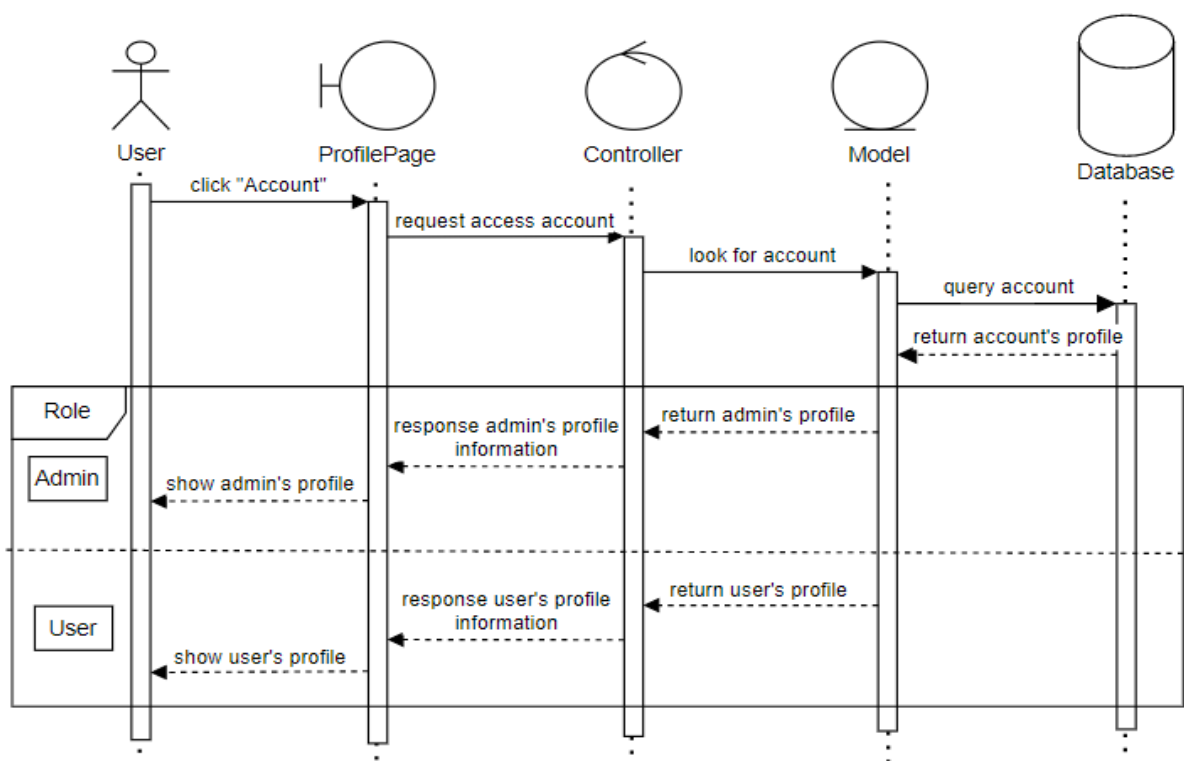
Chương 4 Phân tích thiết kế và xây dựng hệ thống

4.1 Thiết kế kiến trúc chung của hệ thống

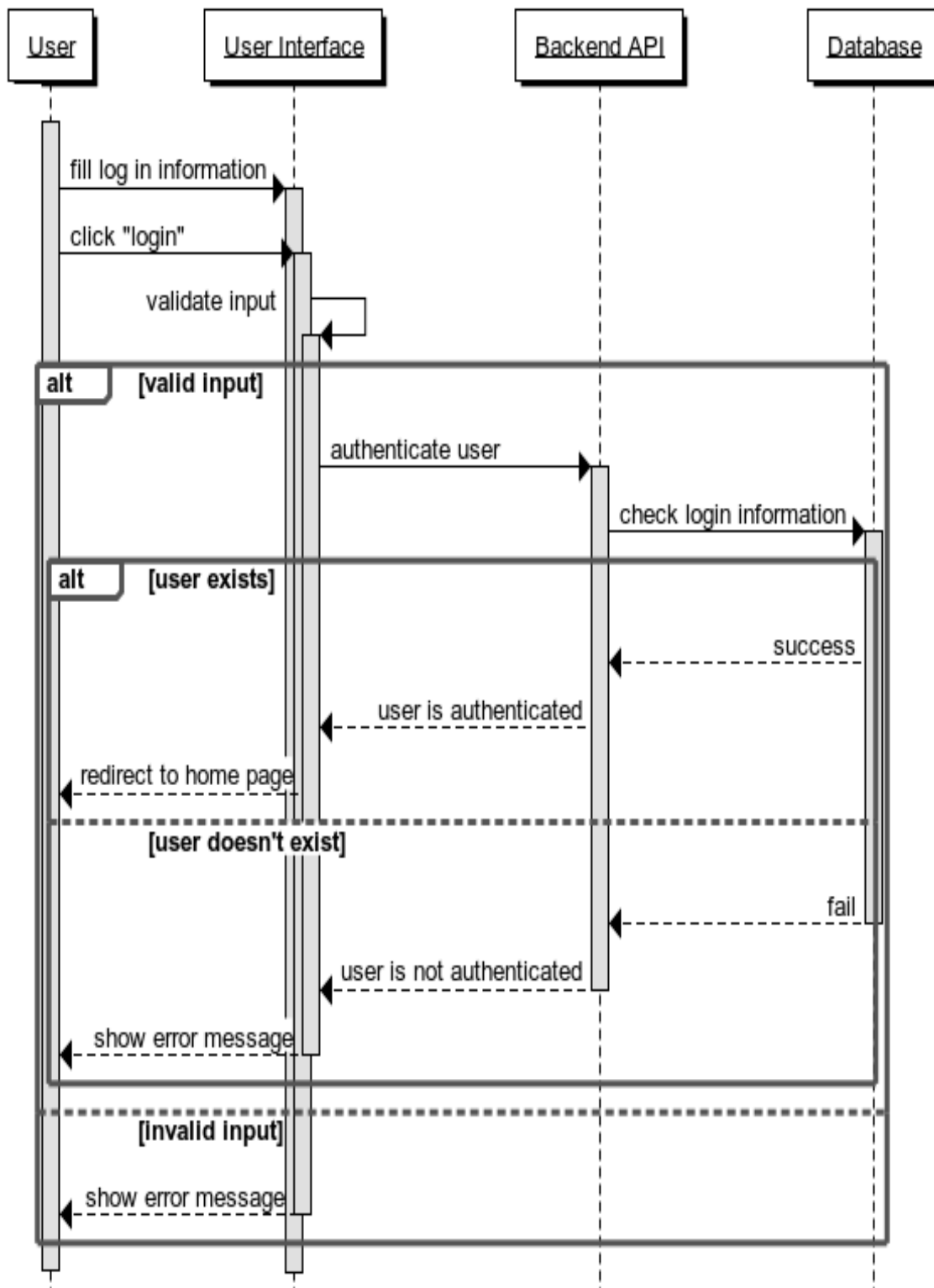
Các thành phần của hệ thống bao gồm máy chủ, cơ sở dữ liệu, ứng dụng web và giao diện người dùng. Kiến trúc chung của hệ thống được xây dựng theo mô hình MVC (Model – View – Controller). Trong đó View được xây dựng từ các file JavaScript trong Nextjs, Model được xây dựng từ các file .php trong laravel và Controller đóng vai trò nhận yêu cầu của người dùng và trả về View dữ liệu tương ứng. Trong Model tương tác với cơ sở dữ liệu bằng hệ quản trị cơ sở dữ liệu MySQL. Cơ sở dữ liệu được thiết kế với các bảng để lưu trữ thông tin về người dùng và người quản trị, bài học, flashcard, quá trình và câu trả lời. Hơn nữa ứng dụng web còn phân quyền cho người dùng và người quản trị, và sử dụng các token khác nhau cho mỗi phiên đăng nhập để bảo mật dữ liệu người dùng.

4.2 Biểu đồ tuần tự các ca sử dụng

Biểu đồ tuần tự (sequence diagram) là một công cụ hữu ích trong phát triển phần mềm để hiển thị các tương tác giữa các đối tượng trong hệ thống. Nó cung cấp một cái nhìn tổng thể về luồng xử lý của ứng dụng và giúp định nghĩa các use case một cách rõ ràng hơn. Biểu đồ tuần tự có thể minh họa cho các nhà phát triển một cách trực quan về cách các thành phần của hệ thống tương tác với nhau. Dưới đây là các biểu đồ tuần tự cho từng ca sử dụng:

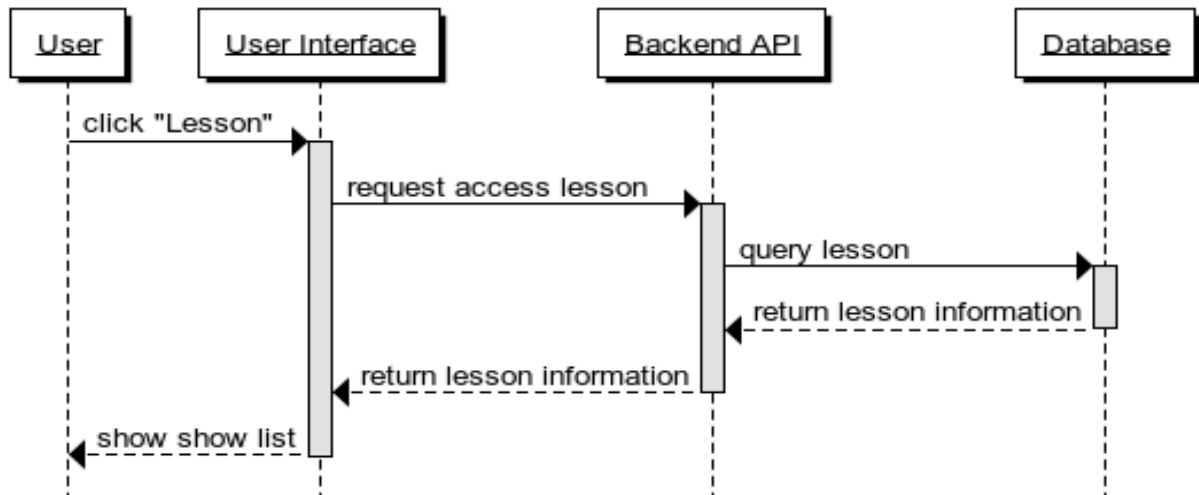


Hình 4.1: Biểu đồ tuần tự cho hoạt động xem hồ sơ.

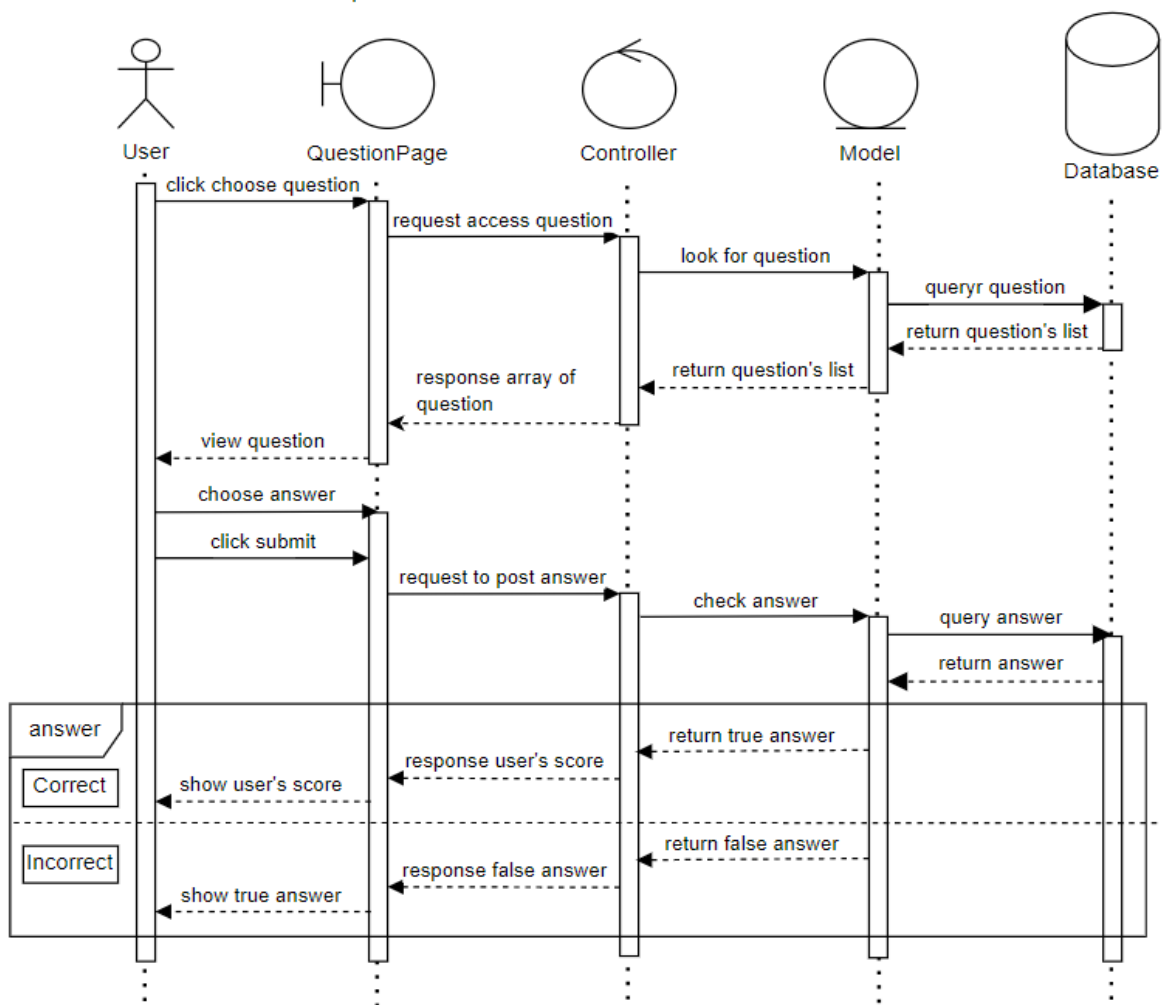


Hình 4.2: Biểu đồ tuần tự của hoạt động đăng nhập.

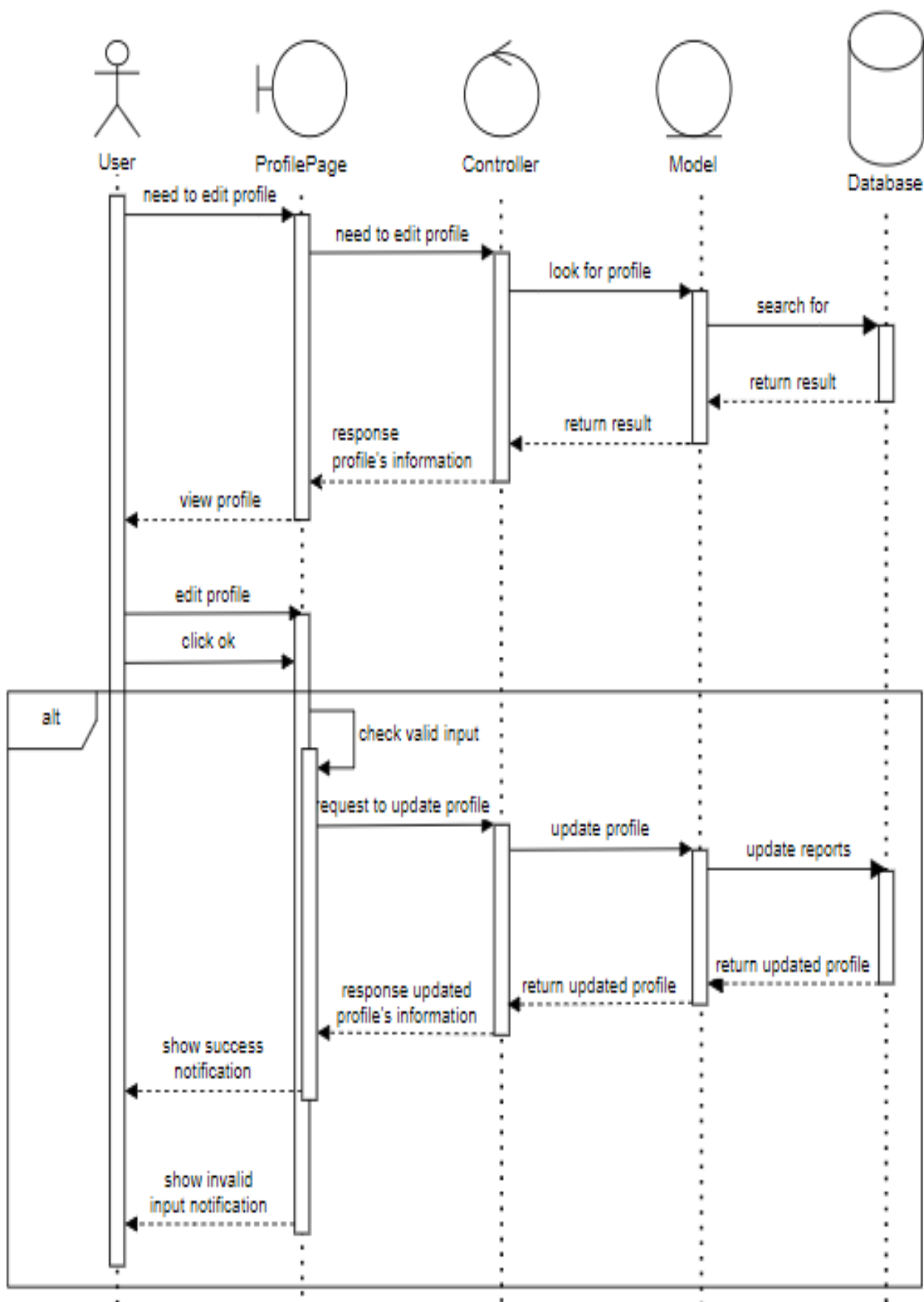
View Lesson Sequence



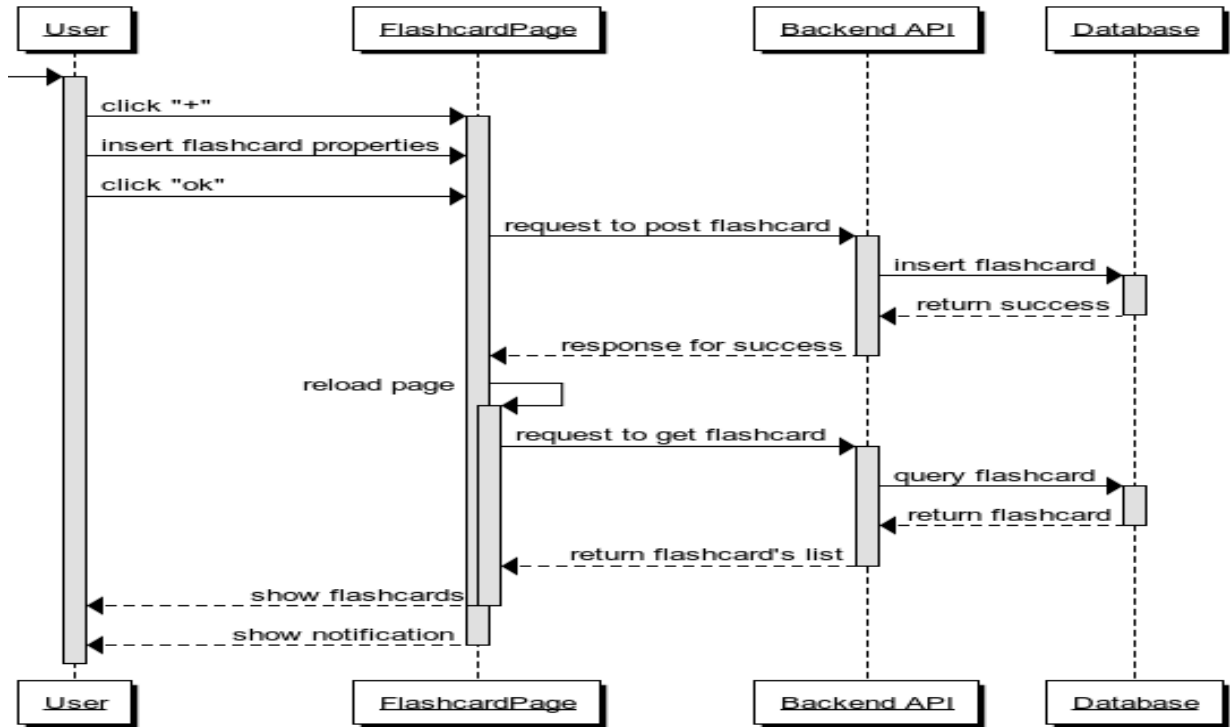
Hình 4.3: Biểu đồ tuần tự cho hoạt động xem bài học.



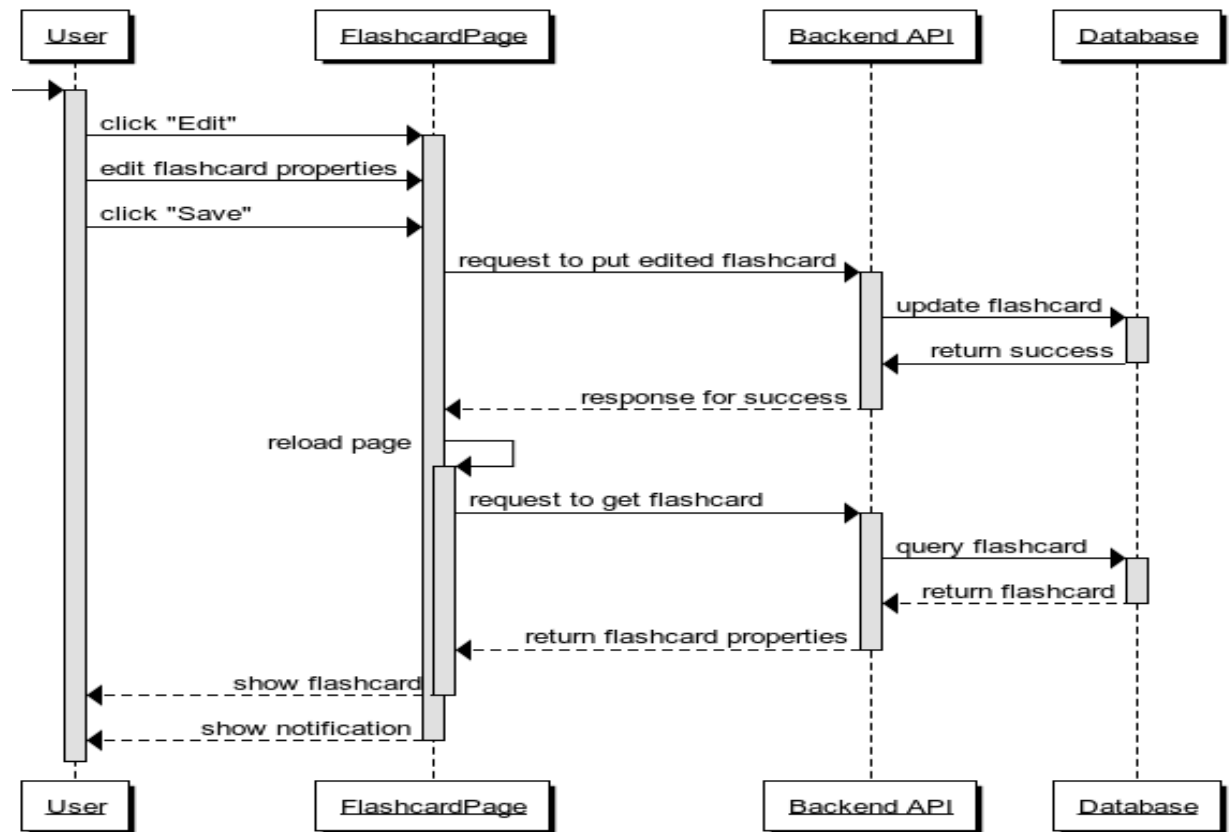
Hình 4.4: Biểu đồ tuần tự cho hoạt động trả lời câu hỏi



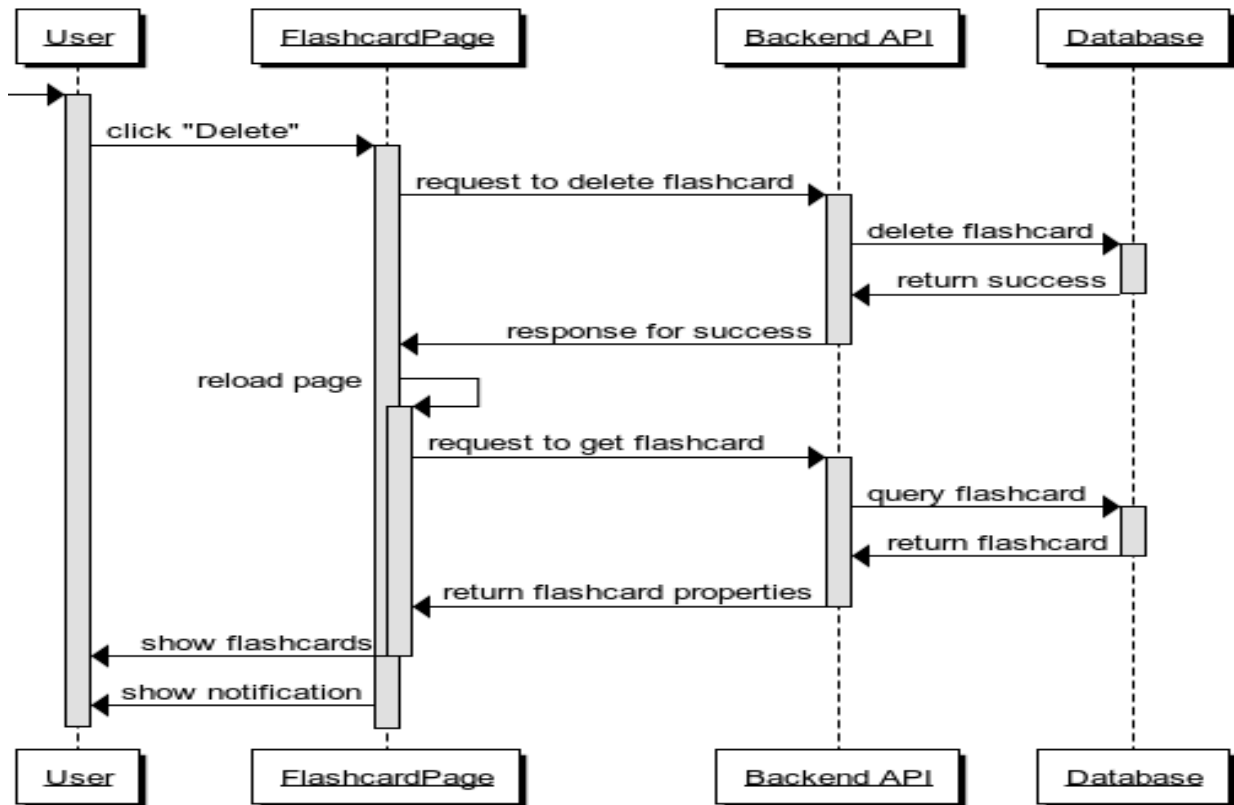
Hình 4.5: Biểu đồ tuần tự cho hoạt động chỉnh sửa hồ sơ người dùng, admin.



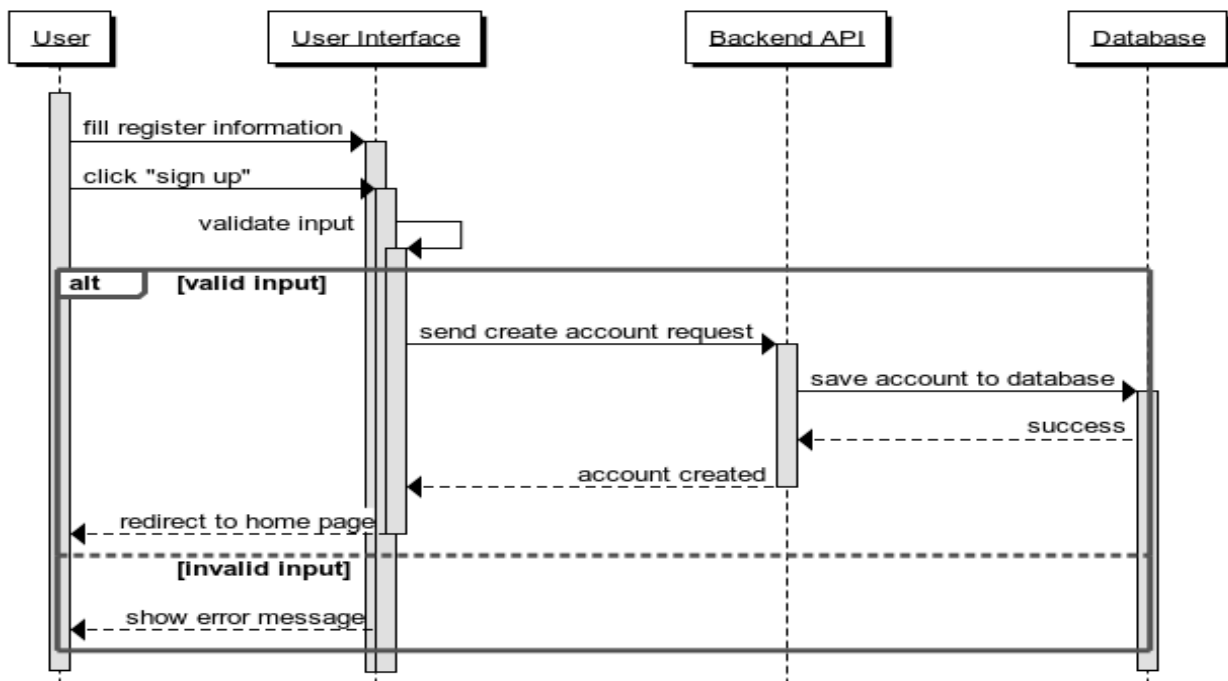
Hình 4.6: Biểu đồ tuần tự cho hoạt động thêm flashcard.



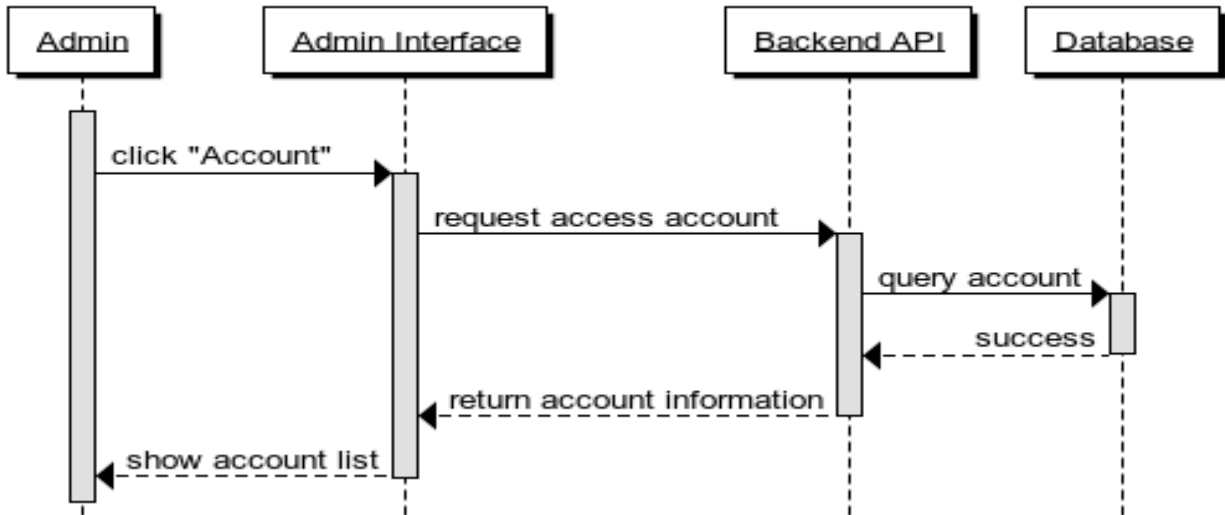
Hình 4.7: Biểu đồ tuần tự cho hoạt động chỉnh sửa flashcard.



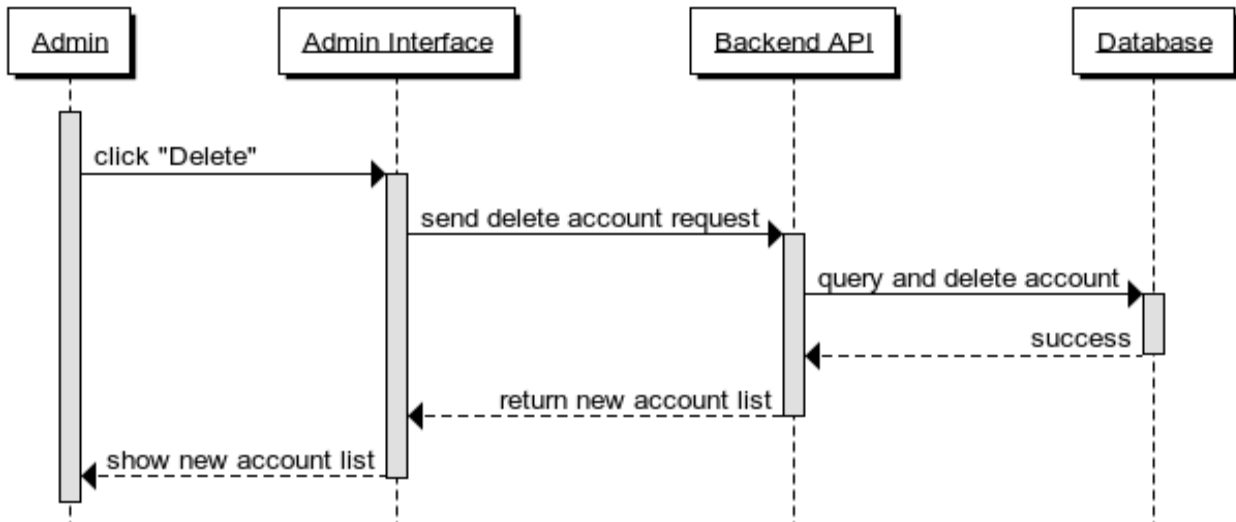
Hình 4.8: Biểu đồ tuần tự cho hoạt động xóa flashcard.



Hình 4.9: Biểu đồ tuần tự cho hoạt động đăng ký tài khoản người dùng.



Hình 4.10: Biểu đồ tuần tự của hoạt động xem tài khoản của người quản trị.



Hình 4.11: Biểu đồ tuần tự của hoạt động xóa tài khoản của người quản trị.

4.3 Thiết kế API

Ở phần xác thực cung cấp 3 giao diện lập trình ứng dụng bao gồm đăng ký, đăng nhập và đăng xuất, đều sử dụng phương thức HTTP POST hỗ trợ sử dụng HTTP Request gửi thông tin đến back-end.

Trong quản lý tài khoản cung cấp 2 giao diện lập trình ứng dụng gồm lấy, sửa thông tin cá nhân người dùng, sử dụng phương thức HTTP GET, HTTP PUT tương ứng cho từng phần để lấy và gửi thông tin đến back-end.

Quản lý tài khoản với quản trị viên cung cấp 2 giao diện lập trình ứng dụng là lấy danh sách thông tin của tất cả tài khoản người dùng và xóa tài khoản người dùng, sử dụng phương thức HTTP GET, HTTP DELETE ứng với mỗi phần để lấy và xóa thông tin qua back-end.

Đối với phần bài học, cung cấp 6 giao diện lập trình ứng dụng bao gồm đưa ra danh sách các phần trong một bài học (từ vựng, Kanji, ngữ pháp, câu hỏi), đưa ra danh sách toàn bộ bài học, tất cả đều sử dụng phương thức HTTP GET, chỉ có duy nhất giao diện lập trình ứng dụng trả lời câu hỏi là sử dụng phương thức HTTP POST hỗ trợ sử dụng HTTP Request gửi đáp án người dùng đến back-end.

Cuối cùng, phần flashcard gồm 4 giao diện lập trình ứng dụng, giao diện lập trình ứng dụng lấy danh sách flashcard sử dụng phương thức HTTP GET, giao diện lập trình ứng dụng tạo mới flashcard sử dụng phương thức HTTP POST, giao diện lập trình ứng dụng sửa thông tin flashcard sử dụng phương thức HTTP PUT, và giao diện lập trình ứng dụng xóa flashcard đã được tạo trước đó sử dụng phương thức HTTP DELETE.

Table 14. Các giao diện lập trình ứng dụng (API).

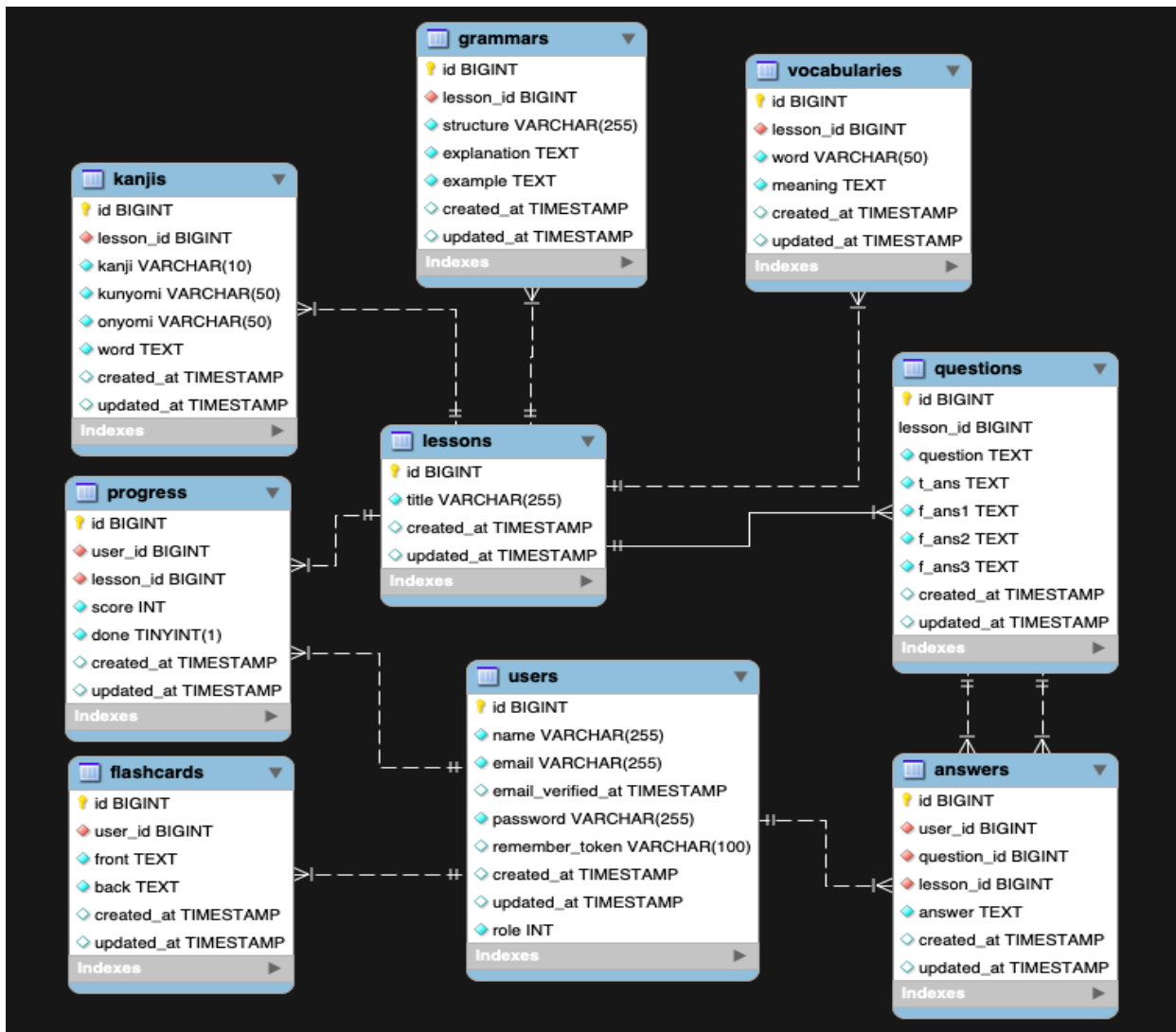
Nhóm	Đường dẫn	Mục đích
Xác thực	/api/register	Đăng ký
	/api/login	Đăng nhập
	/api/logout	Đăng xuất
Quản lý tài khoản	/api/profile	Lấy, sửa thông tin cá nhân người dùng
Quản lý tài khoản (quản trị viên)	/api/admin/accounts	Lấy danh sách thông tin của tất cả tài khoản
	/api/admin/account/{id}	Xoá tài khoản người dùng
Bài học	/api/lessons	Lấy danh sách tất cả bài học
	/api/lesson/{id}/vocabulary	Lấy danh sách từ vựng của một bài
	/api/lesson/{id}/kanji	Lấy danh sách Kanji của một bài
	/api/lesson/{id}/grammar	Lấy danh sách ngữ pháp của một bài

	/api/lesson/{id}/question/{question_id}	Lấy, trả lời từng câu hỏi một của bài
Flashcard	/api/flashcards	Lấy danh sách toàn bộ flashcard, tạo mới flashcard
	/api/flashcard/{id}	Cập nhật (xóa) flashcard đã được tạo trước đó

4.4 Thiết kế cơ sở dữ liệu

Cơ sở dữ liệu gồm các đối tượng chính: người dùng (users), thẻ ghi nhớ (flashcard), bài học (lessons), mỗi bài học bao gồm các phần từ vựng (vocabularies), ngữ pháp (grammars), kanji (kanjis), câu hỏi (questions). Từ đó xây dựng được cơ sở dữ liệu như dưới đây:

- Bảng users lưu các thông tin liên quan đến người dùng: mã định danh người dùng (id), tên (name), tài khoản email (email), mật khẩu (password).
- Bảng lessons lưu các bài học: mã định danh bài học (id), tên bài (title).
- Bảng vocabularies lưu các từ vựng theo bài: mã định danh từ vựng (id), bài học chứa từ vựng đó (lesson_id), từ tiếng Nhật (word), nghĩa tiếng Anh (meaning).
- Bảng grammars lưu ngữ pháp theo bài: mã định danh cấu trúc ngữ pháp (id), bài học chứa ngữ pháp đó (lesson_id), giải thích (explanation), ví dụ (example).
- Bảng kanjis lưu hán tự theo bài: mã định danh hán tự (id), bài học chứa hán tự đó (lesson_id), hán tự (kanji), âm kun (kunyomi), âm on (onyomi), từ vựng chứa hán tự đó (word).
- Bảng questions lưu câu hỏi trắc nghiệm theo bài: mã định danh câu hỏi (id), bài học chứa câu hỏi đó (lesson_id), câu hỏi (question), đáp án đúng (t_ans), 3 đáp án sai (f_ans1, f_ans2, f_ans3).
- Bảng answers lưu lại câu trả lời của người dùng: mã định danh câu trả lời (id), mã định danh người dùng (user_id), mã định danh câu hỏi (question_id), mã định danh bài học chứa câu hỏi đó (lesson_id), câu trả lời (answer).
- Bảng progress lưu lại quá trình học tập của người dùng theo bài: mã định danh quá trình (id), mã định danh người dùng (user_id), mã định danh bài học chứa câu hỏi đó (lesson_id), điểm của bài học tính số câu hỏi trắc nghiệm trả lời đúng (score), trạng thái hoàn thành (done).
- Bảng flashcards lưu lại các flashcard mà người dùng tạo ra: mã định danh flashcard (id), mã định danh người dùng (user_id), nội dung mặt trước (front), nội dung mặt sau (back).



Hình 4.12: Sơ đồ dữ liệu quan hệ của cơ sở dữ liệu trên MySQL.

4.5 Thiết kế giao diện người dùng

4.5.1 Giao diện chính khi truy cập vào trang web

Japper

- Lesson
- Flashcard
- Account

Lesson 1

はじめまして

VocabularyKanjiGrammarQuestion

Lesson 2

これから お世話に なります

VocabularyKanjiGrammarQuestion

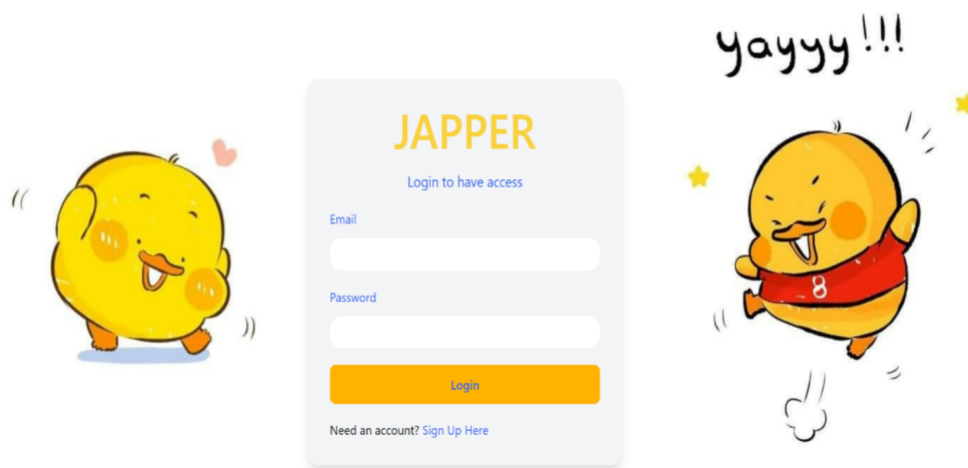
Lesson 3

これを ください

Hình 4.13: Giao diện chính.

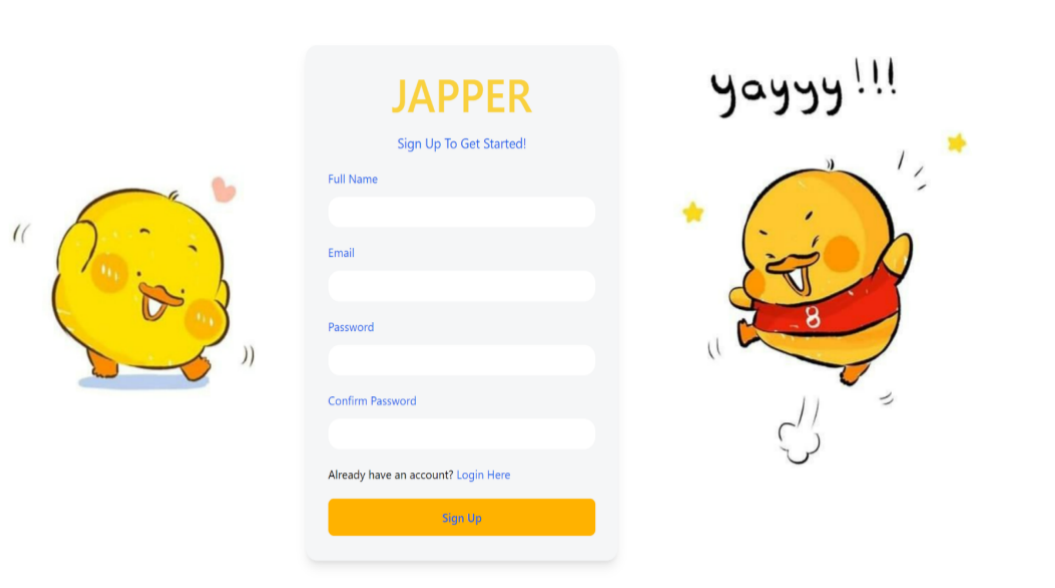
Giao diện chính gồm 2 phần là thanh điều hướng và các bài học. Thanh điều hướng gồm 3 button: Lesson, Flashcard, Account. Button “Lesson” điều hướng đến các bài học. Button “Flashcard” để điều hướng đến xem, sửa, xóa, tạo flashcard. Button “Account” điều hướng người dùng đến trang đăng nhập, nếu chưa có tài khoản thì đến trang đăng ký. Sau khi đăng nhập/ đăng ký button “Account” sẽ dẫn người dùng đến trang profile người dùng hoặc trang quản lý hệ thống trong trường hợp quản trị viên đăng nhập với tài khoản được cấp từ trước.

4.5.2 Giao diện đăng ký, đăng nhập



Hình 4.14: Giao diện đăng nhập.

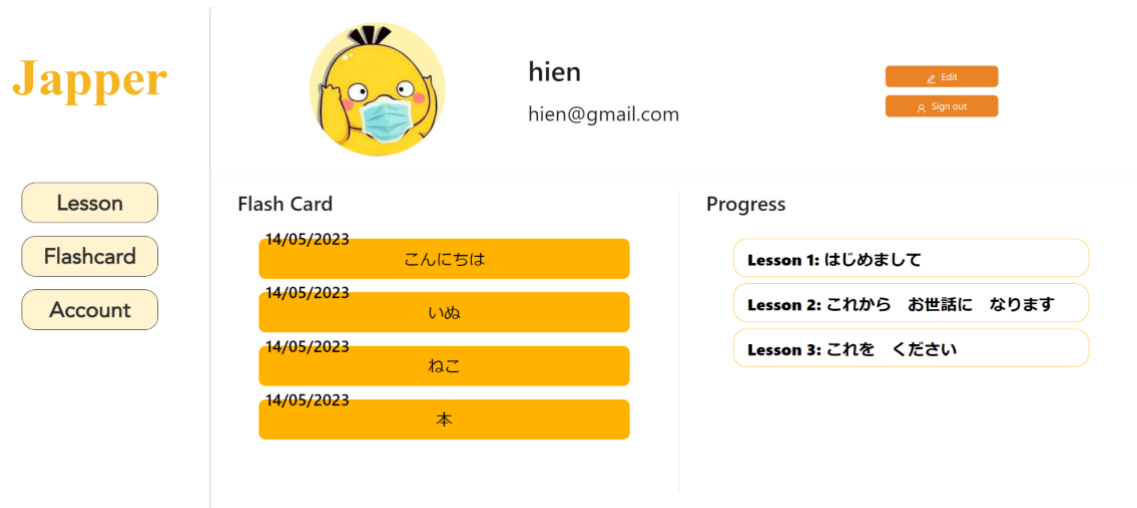
Khi ấn button “Account” tại trang chủ sẽ điều hướng người dùng đến trang đăng nhập. Tại đây người dùng sẽ điền email, password đã đăng ký để đăng nhập tài khoản. Button “Japper” điều hướng người học về trang chủ. Nếu chưa có tài khoản thì nhấp vào “Sign up here” để điều hướng đến trang đăng ký. Người dùng tạo tài khoản bằng cách nhập vào các trường nhập liệu. Nhấp “sign up” để lưu tài khoản.



Hình 4.15: Giao diện đăng ký.

4.5.3 Xem thông tin của bản thân, các flashcard đã tạo theo thời gian, các bài học đã hoàn thành

Với người dùng:

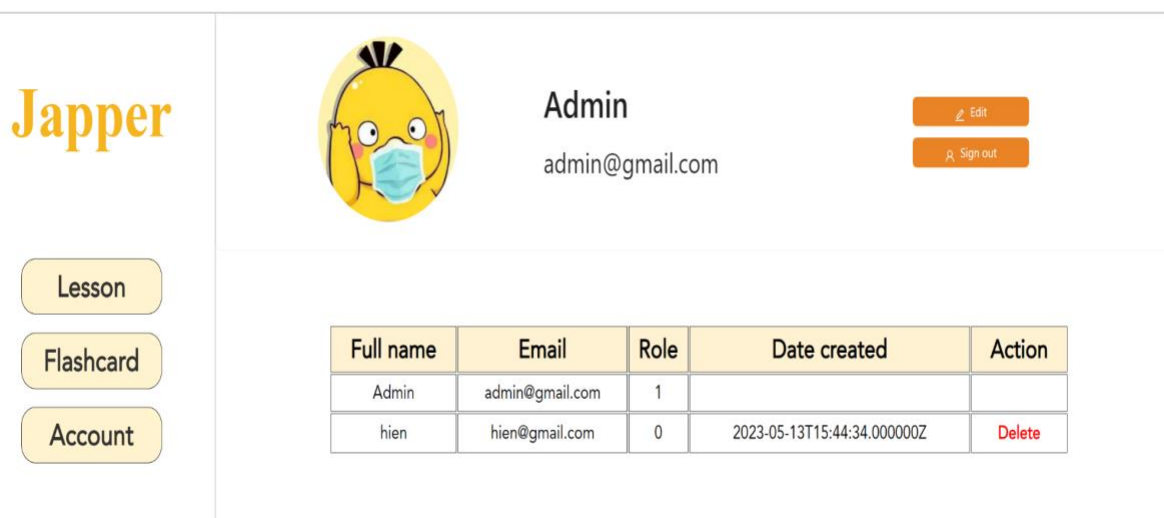


Hình 4.16.a: Giao diện profile người dùng.

- Button “edit” để chỉnh sửa thông tin cá nhân: thay đổi tên, email.
- Button “Sign out” để đăng xuất.

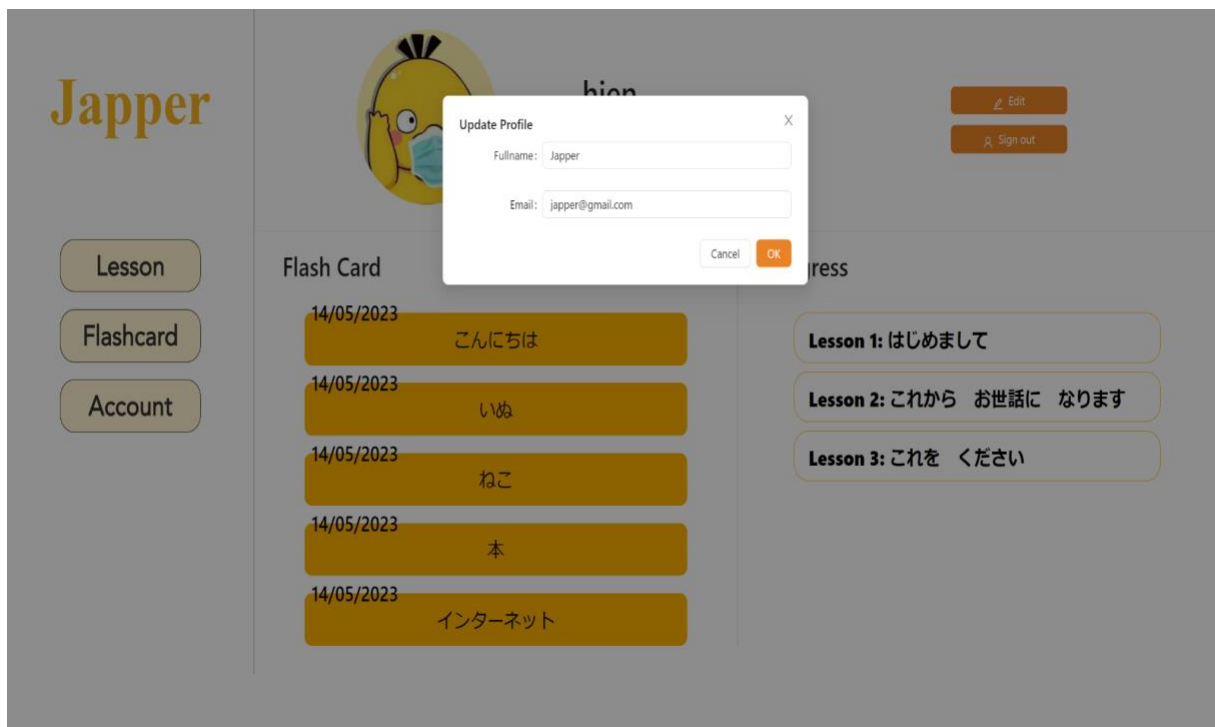
Với quản trị viên:

- Đăng nhập vào hệ thống: Quản trị viên sử dụng các tài khoản đã được cấp sẵn từ trước để đăng nhập vào hệ thống và sử dụng các chức năng dưới quyền của một quản trị viên.
- Sau khi đăng nhập, quản trị viên có thể xem và xóa các tài khoản trong hệ thống.



Hình 4.16.b: Giao diện quản trị viên.

4.5.4 Chỉnh sửa thông tin cá nhân



Hình 4.17: Chỉnh sửa thông tin cá nhân.

Sau khi thay đổi thông tin cá nhân tại các trường full name, email. Người dùng nhấn “Ok” để lưu hoặc “cancel” để thoát.

4.5.5 Bài học

Mỗi bài học sẽ có 4 phần: vocabulary, kanji, grammar, question.

Japper

Lesson
Flashcard
Account

Lesson 1: はじめまして

Vocabulary	Meaning
わたし	I
わたしたち	we
あなた	you
あのひと	that person
あのかた	(polite) that person
せんせい	teacher
がくせい	student
かいしゃいん	company employee
ぎんこういん	bank employee
いしや	medical doctor

Hình 4.18: Giao diện vocabulary.

Trong mục từ vựng: người dùng có thể xem từ vựng, kanji và nghĩa của từ.

Lesson 1: はじめまして

私	Kun: わたし, わたくし On: シ Word: 私
---	------------------------------------

Hình 4.19: Giao diện kanji.

Trong mục kanji: người dùng có thể xem kanji, âm On, âm Kun, từ vựng chứa kanji đó.

Lesson 1: はじめまして

Practice は	Explanation: The particle は indicates that the word before it is the topic of the sentence. You select a noun you want to talk about, add は to show that it is the topic and give a statement about the topic. Example: わたしは マイク ミラー です。
Practice です	Explanation: Nouns used with です work as predicates. です indicates judgement or assertion. Example: わたしは エンジニア です。
Practice も	Explanation: も is added after a topic instead of は when the statement about the topic is the same as the previous topic. Example: ミラーさんは かいしゃいん です。グブタさんも かいしゃいん です。

Hình 4.19: Giao diện grammar.

Trong mục ngữ pháp: người dùng có thể xem cấu trúc ngữ pháp, giải thích, ví dụ.

Lesson 1: はじめまして

Question 1:
わたし means:

Choose the correct answer:

A: I

B: you

C: teacher

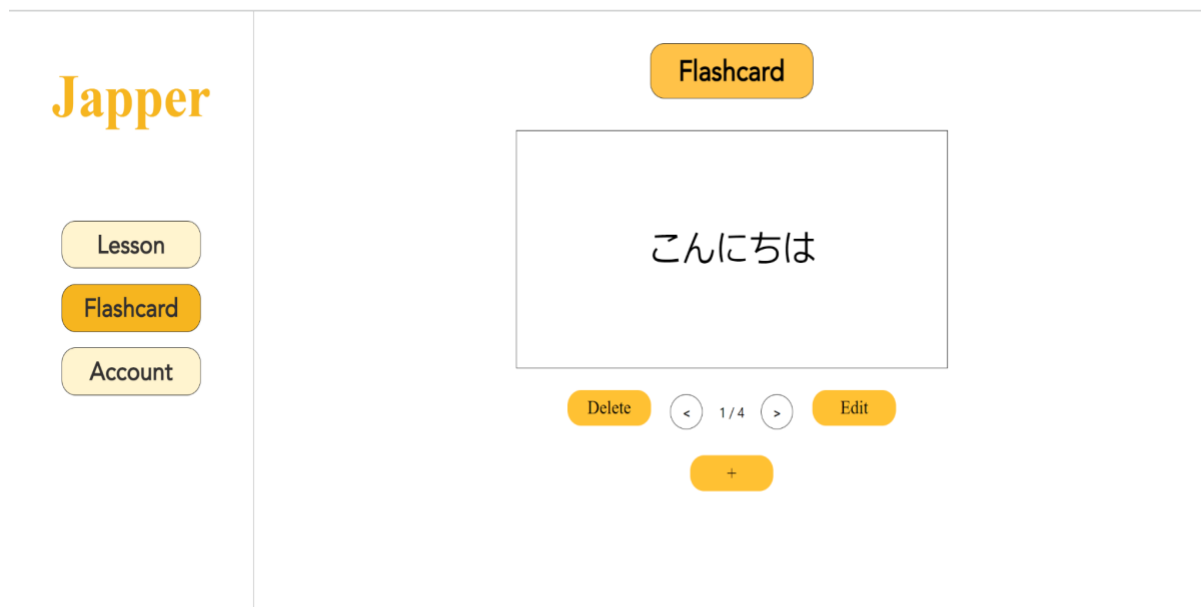
D: student

Submit

Hình 4.20: Giao diện question.

Người học nhấn đáp án muốn chọn và ấn submit để biết đúng hay sai. Nếu sai thì phải trả lời lại đến khi đúng thì sẽ được tự động chuyển sang câu tiếp theo. Khi trả lời hết câu hỏi thì trong trang profile người học sẽ được ghi nhận là hoàn thành bài học đó.

4.5.6 Flashcard



Hình 4.21: Xem thẻ flashcard.

Người học có thể thêm, xem, sửa, xóa flashcard.

Người học muốn xem thẻ flashcard đã tạo trước tiên nhấn vào button flashcard. Thẻ flashcard hiện ra, để xem hai mặt chỉ cần di chuột qua thẻ.

Button “Delete” dùng để xóa thẻ.

Button “Edit” để chỉnh sửa thẻ.

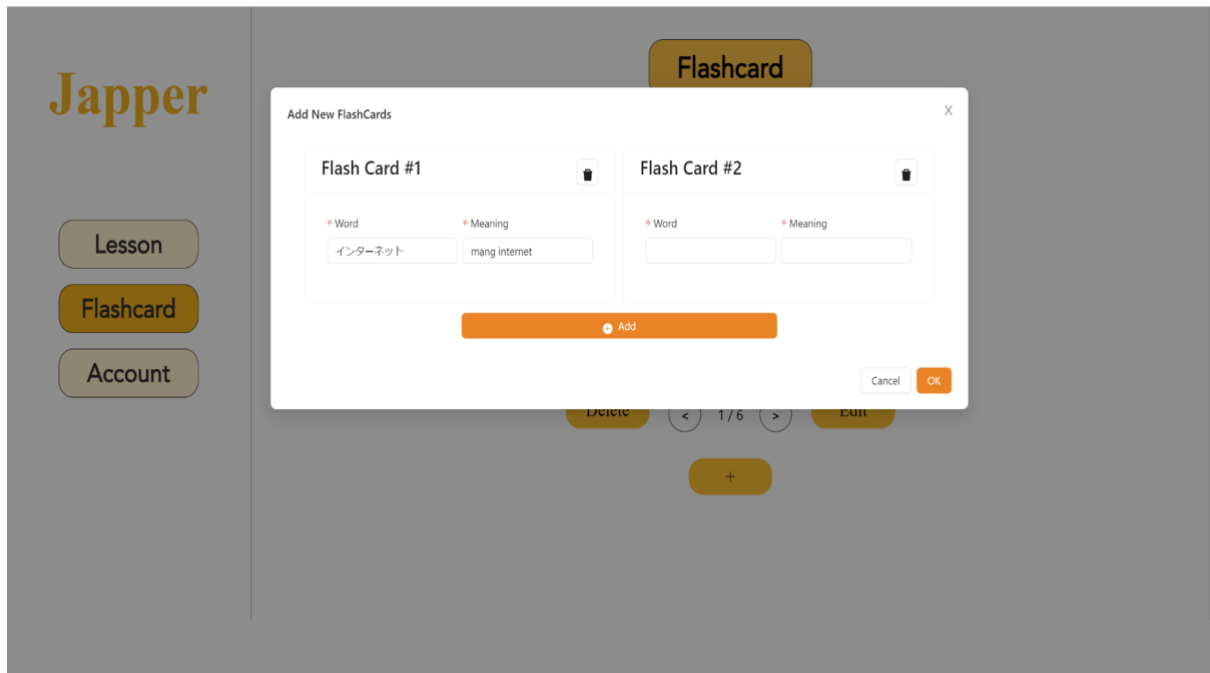
Button “+” để tạo thêm thẻ mới.



Hình 4.22: Chỉnh sửa thẻ flashcard.

Button “Save” để lưu sau khi chỉnh sửa.

Button “Cancel” trở lại sau quá trình chỉnh sửa.



Hình 4.23: Thêm thẻ flashcard.

Nhập dữ liệu về flashcard muốn tạo vào trường word và meaning.

Button “Add” để tạo thêm nhiều hơn một flashcard.

Button “Ok” để kết thúc tạo và lưu flashcard muốn tạo.

Chương 5 Cài đặt và kiểm thử

Dựa vào phân tích từ chi tiết đến tổng quát ở các chương trước, chương này sẽ trình bày về cách cài đặt, kiểm thử website.

5.1 Cài đặt công cụ

Website học tiếng Nhật được phát triển dựa trên mô hình MVC được triển khai với phần front-end sử dụng framework Next.js và phần back-end sử dụng framework Laravel, trong đó front-end và back-end giao tiếp với nhau thông qua chuẩn RESTful API.

Website được phát triển bằng công cụ hỗ trợ lập trình VSCode (Visual Studio Code). Website được chạy trực tiếp trên localhost nhằm tối ưu hoá thời gian phát triển, sửa lỗi mã nguồn. Ngoài ra, sử dụng phần mềm quản lý mã nguồn Github để lưu trữ, theo dõi lịch sử thay đổi của tất cả các file trong dự án, hơn nữa còn có thể triển khai và kiểm thử ở nhiều nền tảng khác nhau như Window, Linux, MacOS.

5.2 Cài đặt dự án

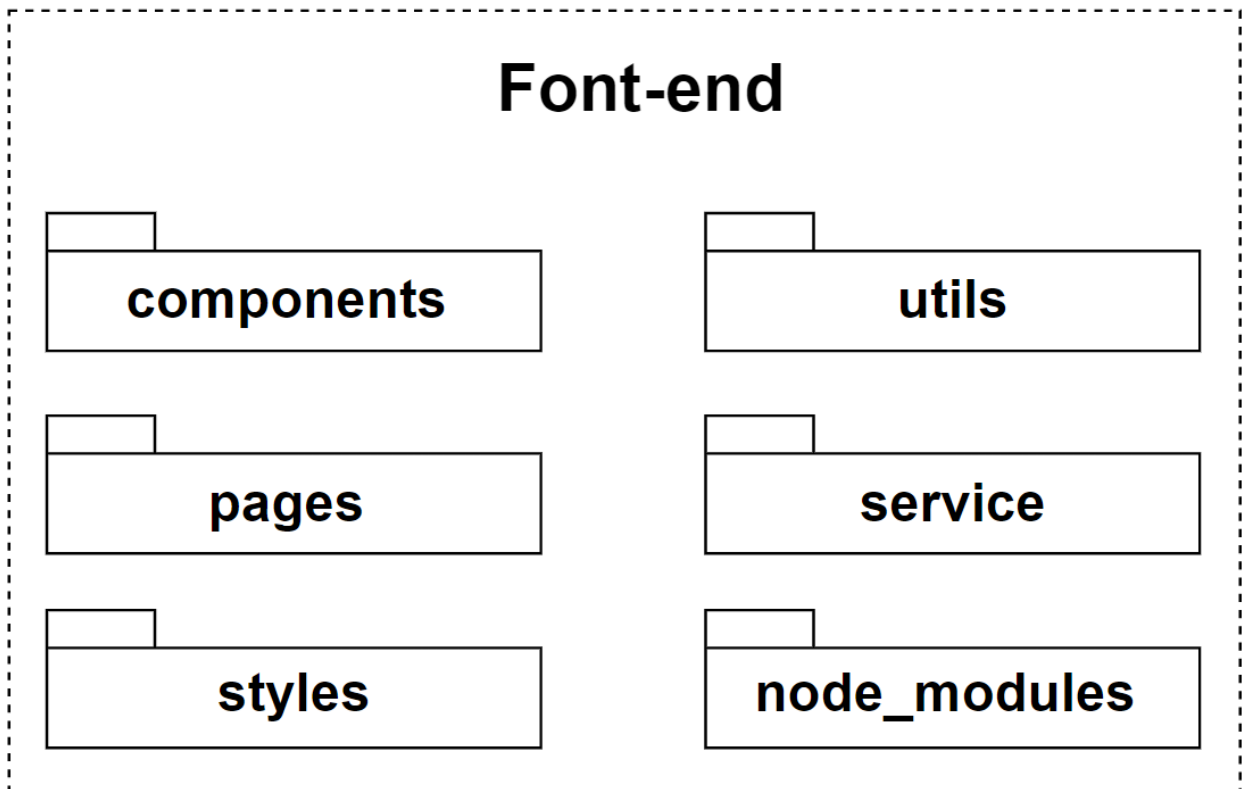
5.2.1 Cài đặt front-end

Phía front-end được cài đặt sử dụng framework Next.js theo mô hình ứng dụng đơn trang SPA (Single - Page Application). Các thành phần trong giao diện được tách thành các module, nhằm tái sử dụng lại code và tải các thành phần mà không cần tải lại toàn bộ trang web.

Mô tả cấu trúc module được cài đặt trong ứng dụng. Module `node_modules` bao gồm các thư viện được cài đặt trong ứng dụng, trong đó bao gồm các thư viện và framework chính thư viện `nextjs`, `axios`, `react`, `antd`, `fortawesome`. Thư viện `antd` là một framework UI (User Interface) được xây dựng dành cho React, cung cấp các thành phần giao diện cho ứng dụng web như `button`, `input`, `modal`, `table`..., ngoài ra bộ icon của thư viện `fortawesome` cũng được dùng để thiết kế giao diện web. `Axios` là một thư viện HTTP client hỗ trợ trong việc tìm nạp dữ liệu. `Nextjs` và `react` được sử dụng để xây dựng web người dùng tương tác hỗ trợ hiệu năng và cải thiện trải nghiệm người dùng.

Modules components bao gồm các khối xây dựng do người lập trình tự định nghĩa để xây dựng trang web. Modules pages là nơi lưu trữ các tập mã nguồn dùng để xây dựng giao diện của trang web. Các dữ liệu được lấy từ bên ngoài như hình ảnh, font chữ được lưu trong module `public` và module `styles` là các file css để tạo trang web sinh động. Module `utils` xử lý các token được lưu để kết nối api và file service để lấy dữ liệu hiển thị cho trang trang hồ sơ.

Hơn nữa trong module pages chứa file `index.js` là file chính mặc định của `nextjs` khi chạy ứng dụng, nó bao các thành phần đầu tiên của ứng dụng.



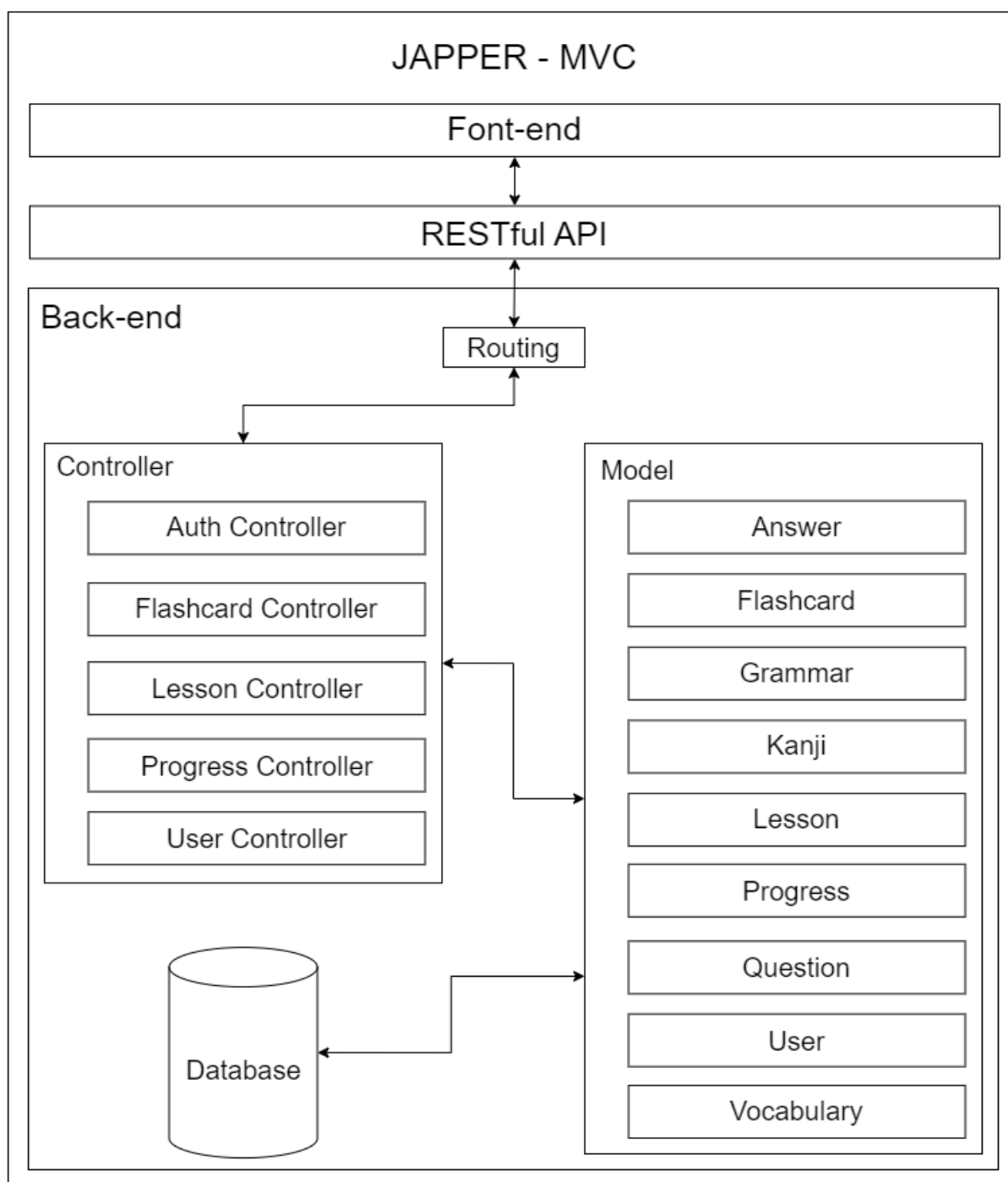
Hình 5.1: Cấu trúc các module trong front-end

5.2.2 Cài đặt back-end

Phần back-end được cài đặt sử dụng framework Laravel theo mô hình MVC (Model - View - Controller) đơn giản, phân tách độc lập thành các phần riêng biệt tránh việc xảy ra xung đột trên hệ thống.

Cụ thể, có tất cả năm controller bao gồm Auth Controller, Flashcard Controller, Lesson Controller, Progress Controller, User Controller. Trong đó Auth Controller có chức năng xử lý các hoạt động liên quan đến người dùng như đăng nhập, đăng ký và đăng xuất. Flashcard Controller có nhiệm vụ xử lý CRUD (create, read, update, delete) flashcard. Lesson Controller có nhiệm vụ quản lý các phần liên quan đến bài học như từ vựng, kanji, ngữ pháp và câu hỏi. Progress Controller có nhiệm vụ xử lý các hoạt động của người dùng khi trả lời các câu hỏi và quá trình học tập của người dùng. User Controller có chức năng quản lý tài khoản người dùng (xem, sửa thông tin cá nhân), quản lý riêng cho người dùng admin (xem, xóa tài khoản của tất cả người dùng).

Có 9 model bao gồm Answer, Flashcard, Grammar, Kanji, Lesson, Progress, Question, User, Vocabulary thao tác trực tiếp với cơ sở dữ liệu, giúp các hoạt động xem, truy xuất hoặc xử lý dữ liệu trong ứng dụng diễn ra nhanh chóng, dễ dàng, thuận tiện hơn.



Hình 5.1: Kiến trúc hệ thống website học tiếng Nhật.

5.3 Kiểm thử

Trong quy trình xây dựng và phát triển ứng dụng, kiểm thử có vai trò vô cùng quan trọng. Việc thực hiện các ca kiểm thử tuy không thể bao quát được toàn bộ các lỗi, không thể khẳng định được hệ thống hoàn toàn không có lỗi, thế nhưng vẫn cần có giai đoạn kiểm thử để đảm bảo tối ưu hoá được hệ thống, số lượng cũng như ảnh hưởng của lỗi đảm bảo ở mức thấp nhất, không ảnh hưởng quá lớn đến hệ thống.

Pha kiểm thử hoạt động xuyên suốt quá trình xây dựng và phát triển ứng dụng, sử dụng trực tiếp giao diện front-end để kiểm thử cả API và giao diện. Dùng kịch bản kiểm thử, kết quả để tổng kết và đánh giá hoạt động các chức năng của các ca kiểm thử.

Môi trường kiểm thử:

- Trình duyệt Microsoft Edge trên hệ điều hành Windows.
- Trình duyệt Google Chrome trên hệ điều hành Windows, Ubuntu.
- Trình duyệt Firefox trên hệ điều hành Ubuntu.

5.3.1 Kiểm thử chức năng

5.3.1.1 Kế hoạch kiểm thử

Bảng 5.1: Danh sách các ca kiểm thử thủ công.

STT	Nhóm chức năng	Các ca kiểm thử
1	Xác thực tài khoản	1. Đăng ký 2. Đăng nhập 3. Đăng xuất
2	Quản lý tài khoản	1. Xem thông tin tài khoản 2. Sửa thông tin tài khoản
3	Quản lý tài khoản (với quản trị viên)	1. Xem thông tin tài khoản người dùng 2. Xoá tài khoản người dùng
4	Bài học	1. Hiện thị danh sách bài học 2. Hiện thị từ vựng cho từng bài học 3. Hiện thị ngữ pháp cho từng bài học 4. Hiện thị kanji cho từng bài học 5. Hiện thị câu hỏi cho từng bài học 6. Xử lý dữ liệu trả lời câu hỏi của người dùng
5	Flashcard	1. Xem toàn bộ flashcard 2. Tạo mới flashcard 3. Sửa flashcard đã tạo 4. Xoá flashcard đã tạo

5.3.1.2 Kịch bản kiểm thử

Các chức năng ở bảng 5.1 được tiến hành kiểm thử từ tháng 05/2023. Các ca kiểm thử không sử dụng công cụ hỗ trợ nào, quá trình kiểm thử được diễn ra một cách thủ công qua các kịch bản được xây dựng sẵn, mỗi ca kiểm thử đều có nhiều kịch bản khác nhau cho cả kiểm thử tích cực và kiểm thử tiêu cực, toàn bộ kết quả của các ca kiểm thử đều giống như kỳ vọng của kịch bản.

Bảng 5.2: Ca kiểm thử đăng ký.

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Sai định dạng mail	{ “Full name”: “test”, “Email”: “test”, “Password” : “12345678”, “Confirm Password” : “12345678” }	Hiện ra thông báo “Please include an ‘@’ in the email address. ‘test’ is missing an ‘@’.” ngay dưới ô nhập email.	Thành công
2	Biểu mẫu chính xác	{ “Full name”: “test”, “Email”: “test@gmail.com”, “Password” : “12345678”, “Confirm Password” : “12345678” }	Xác thực người dùng, điều hướng qua trang thông tin cá nhân của người dùng.	Thành công

Bảng 5.3: Ca kiểm thử đăng nhập.

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Điền thiếu trường	{ “Email”: “”, “Password” : “12345678” }	Hiện ra thông báo màu đỏ “Password is required” ngay dưới ô điền dữ liệu các trường.	Thành công
2	Mật khẩu ngắn hơn 8 ký tự	{ “Email”: “test@gmail.com”, “Password” : “12345” }	Hiện ra thông báo màu đỏ “Password must be more than 8 characters” ngay dưới ô nhập mật khẩu.	Thành công
3	Biểu mẫu chính xác	Điền đúng email và mật khẩu của tài khoản	Xác thực người dùng, điều hướng qua trang thông tin cá nhân của người dùng.	Thành công

Bảng 5.4: Ca kiểm thử đăng xuất.

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Thao tác với nút logout để đăng xuất.		Đăng xuất, không thể truy cập các trang yêu cầu có tài khoản.	Thành công

Table 19. Bảng 5.5: Kiểm thử màn quản lý tài khoản

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Mở trang cá nhân xem thông tin tài khoản người dùng.		Hiển thị thông tin người dùng (tên, email), danh sách các flashcard người dùng đã tạo, danh sách bài học đã hoàn thành.	Thành công
2	Mở trang cá nhân xem thông tin tài khoản khi chưa đăng nhập		Điều hướng người dùng đến trang đăng nhập.	Thành công
3	Sửa thông tin với biểu mẫu chính xác	{ "Fullname": "test", "Email": "test@gmail.com" }	Thông tin người dùng được cập nhật.	Thành công
4	Sửa thông tin khi sai định dạng mail	{ "Fullname": "test", "Email": "test" }	Báo lỗi "Email is invalid".	Thành công

Bảng 5.6: Kiểm thử màn quản lý tài khoản với quản trị viên.

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Xem thông tin tài khoản người dùng.		Hiển thị thông tin người dùng (tên, email, ngày đăng ký).	Thành công
2	Xoá tài khoản người dùng	Quản trị viên sử dụng chức năng xoá bằng cách ấn vào ô "delete" cạnh dữ liệu người dùng	Xoá tài khoản, thông tin người dùng, hiển thị danh sách tài khoản, thông tin người dùng còn lại.	Thành công

Bảng 5.7: Kiểm thử màn bài học.

STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Xem tổng quan danh sách các bài học	Người dùng truy cập trang Lesson.	Hiển thị toàn bộ các bài học có trong cơ sở dữ liệu.	Thành công
2	Xem từ vựng của từng bài	Người dùng truy cập phần Vocabulary của bất kì bài học nào.	Hiển thị bảng danh sách từ vựng có trong bài.	Thành công
3	Xem ngữ pháp của từng bài	Người dùng truy cập phần Grammar của bất kì bài học nào.	Hiển thị danh sách ngữ pháp của bài.	Thành công
4	Xem Kanji của từng bài	Người dùng truy cập phần Kanji của bất kì bài học nào.	Hiển thị từng từ một, mỗi từ kèm theo cách đọc Kun, On và từ vựng đi kèm.	Thành công
5	Làm bài tập vận dụng của từng bài	Người dùng truy cập phần Lesson trả lời câu hỏi.	Hiển thị lần lượt từng câu hỏi theo điều hướng của người dùng.	Thành công
6	Người dùng trả lời câu hỏi đã trả lời đúng.	Người dùng gửi câu trả lời của câu hỏi đã được tính điểm.	Hiển thị thông báo “You have got score for this question!”.	Thành công
7	Người dùng trả lời không chính xác	Câu trả lời sai	Hiển thị đáp án đúng.	Thành công
8	Người dùng trả lời chính xác	Câu trả lời đúng	Hiển thị thông báo “Your answer is correct”. Cập nhật điểm người dùng và thông báo điểm “Your score: 2”.	Thành công

Bảng 5.8: Kiểm thử màn flashcard.

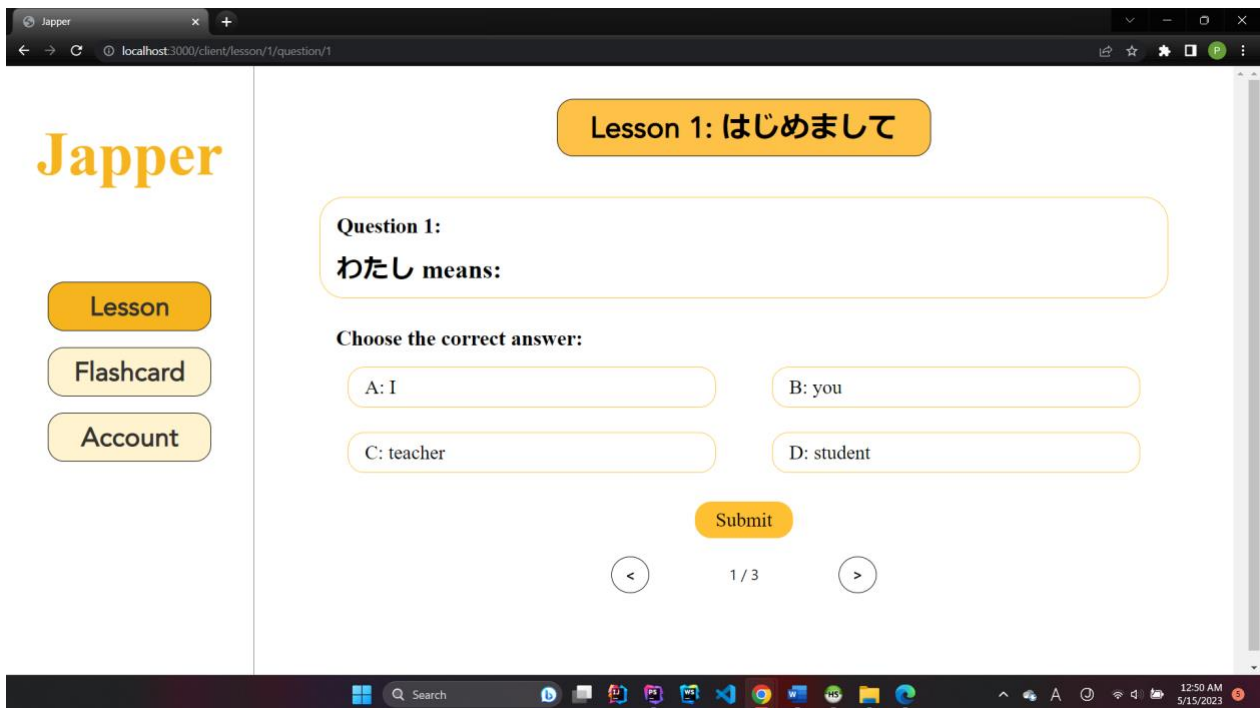
STT	Kịch bản kiểm thử	Dữ liệu kiểm thử	Kết quả mong đợi	Trạng thái
1	Xem flashcard	Người dùng truy cập trang Flashcard.	Hiển thị lần lượt từng flashcard theo điều hướng của người dùng.	Thành công

2	Tạo mới flashcard	{ “Word”: “私”, “Meaning”: “I” }	Hiển thị flashcard mới tạo trên giao diện cùng thông báo tạo thành công.	Thành công
3	Sửa flashcard đã tạo	{ “Word”: “女の子”, “Meaning”: “girl” }	Hiển thị flashcard đã sửa.	Thành công
4	Xoá flashcard đã tạo	Người dùng chọn xoá flashcard ở giao diện.	Hiển thị danh sách flashcard không chứa flashcard đã xoá.	Thành công

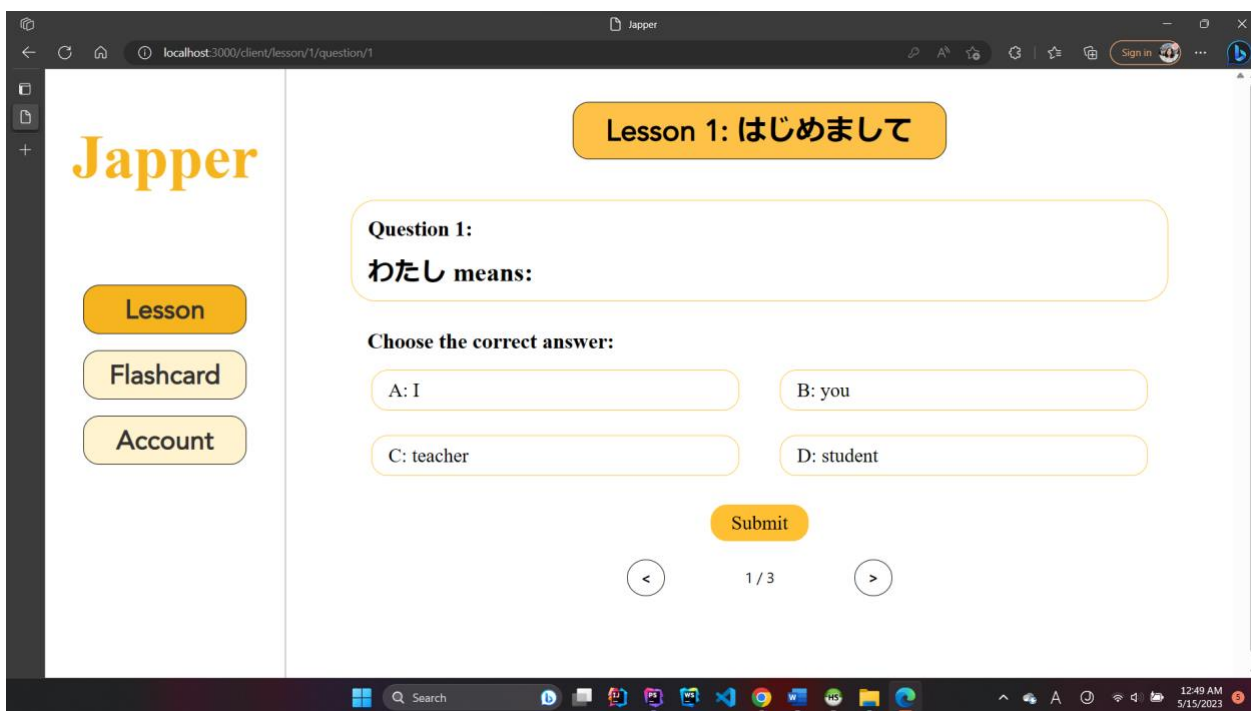
5.3.2 Kiểm thử phi chức năng

Ngoài kiểm thử chức năng thì kiểm thử phi chức năng cũng góp phần quan trọng trong việc xây dựng và phát triển ứng dụng, đánh giá mức độ hoàn thiện của sản phẩm.

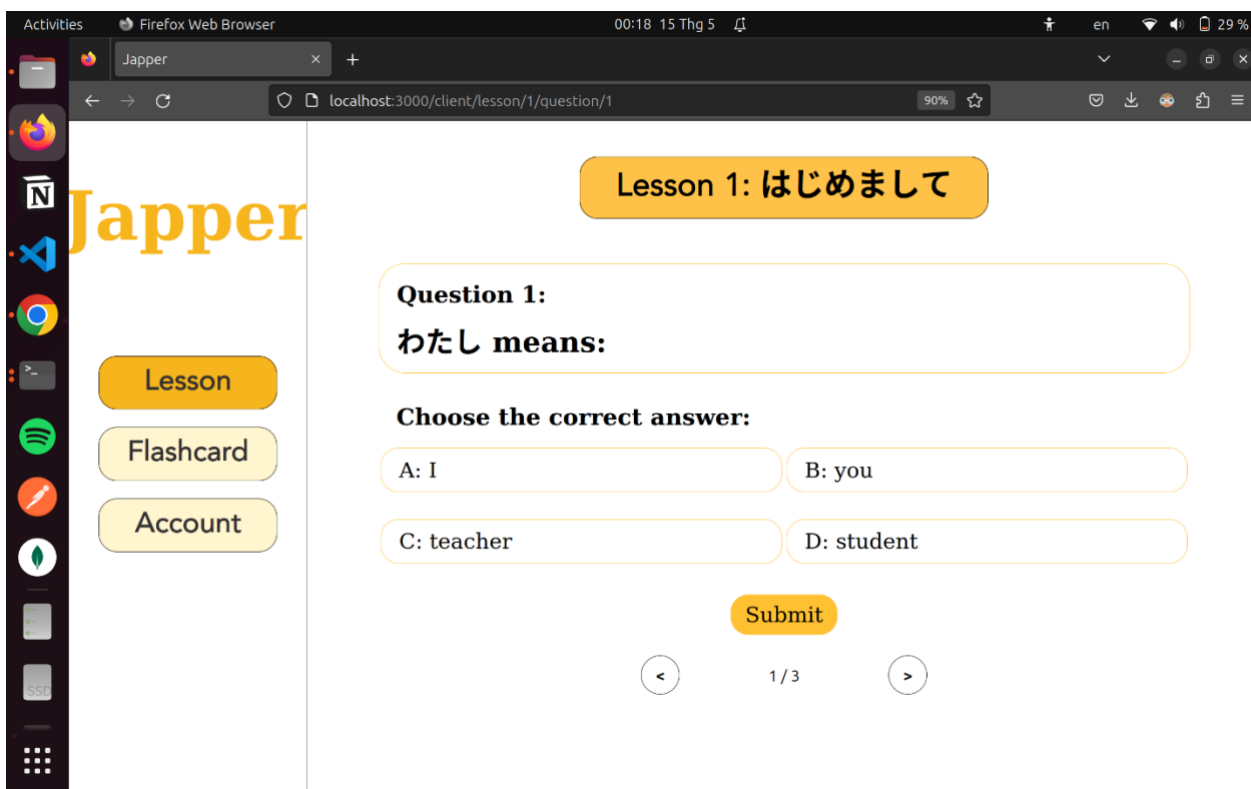
- **Kiểm thử hiệu suất:** Giao diện ứng dụng được xây dựng đơn giản, trực quan, dễ tương tác và sử dụng. Có sự hài hoà, đồng nhất về bố cục, màu sắc, chủ đề chung. Các trang liên kết với nhau, có thanh điều hướng luôn nằm ở bên trái giúp dễ dàng truy cập vào các trang khác nhau. Hơn nữa, ứng dụng còn đảm bảo sự nhất quán khi sử dụng trên các trình duyệt, hệ điều hành khác nhau.



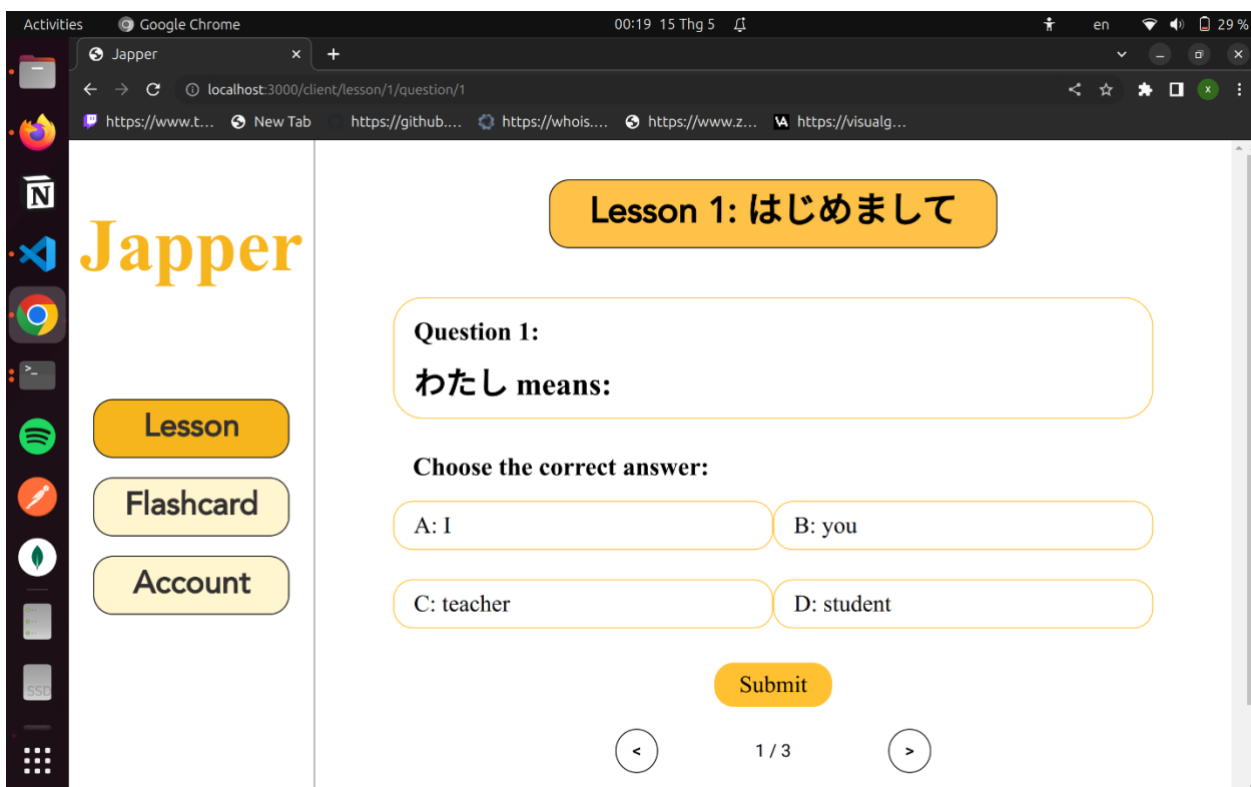
Hình 5.2.1: Giao diện câu hỏi trên máy tính.



Hình 5.2.2: Giao diện câu hỏi trên máy tính.



Hình 5.2.3: Giao diện câu hỏi trên máy tính.



Hình 5.2.4: Giao diện câu hỏi trên máy tính.

Bảng 5.9: Kiểm thử tính nhất quán của giao diện.

STT	Kịch bản kiểm thử	Kết quả
1	<ul style="list-style-type: none"> - Giao diện câu hỏi trên máy tính - Hệ điều hành: Windows - Trình duyệt: Google Chrome - Kích thước máy: 1920px x1080px 	Hình 5.2.1
2	<ul style="list-style-type: none"> - Giao diện câu hỏi trên máy tính - Hệ điều hành: Windows - Trình duyệt: Microsoft Edge - Kích thước máy: 1920px x1080px 	Hình 5.2.2
3	<ul style="list-style-type: none"> - Giao diện câu hỏi trên máy tính - Hệ điều hành: Ubuntu - Trình duyệt: Firefox - Kích thước máy: 1920px x1080px 	Hình 5.2.3
4	<ul style="list-style-type: none"> - Giao diện câu hỏi trên máy tính - Hệ điều hành: Ubuntu - Trình duyệt: Google Chrome - Kích thước máy: 1920px x1080px 	Hình 5.2.4

- **Kiểm thử bảo trì:** Dữ liệu có thể thay đổi qua người dùng quản trị viên, góp phần nâng cao khả năng quản lý dữ liệu (phục hồi, tránh mất mát và cải tiến dữ liệu). Ngoài ra, mã nguồn được lưu ở Github có thể quản lý được các phiên bản giúp việc phục hồi, cải tiến ứng dụng tốt hơn.

5.4 Tổng kết chương

Thông qua cài đặt, kiểm thử chức năng và kiểm thử phi chức năng, có thể thấy sản phẩm được phát triển tương đối hoàn thiện, đầy đủ tính năng, hỗ trợ đa trình duyệt, đa nền tảng. Các yêu cầu chức năng và phi chức năng hầu hết được đáp ứng, bên cạnh đó vẫn cần cải thiện, phát triển thêm nhiều từ giao diện, tối ưu mã nguồn,... để sản phẩm ngày càng hoàn thiện hơn.

Chương 6 Kết luận và định hướng phát triển

6.1 Kết luận

Xây dựng một trang website học tiếng Nhật không phải là một vấn đề mới nhưng rất cần thiết trong thực tiễn. Khi mà nhu cầu học tiếng Nhật của Việt Nam tăng cao đi đôi với thời đại công nghệ thông tin lên ngôi chỉ cần một chiếc laptop có kết nối internet là có thể linh hoạt học tiếng Nhật mọi lúc mọi nơi vừa tiết kiệm chi phí, thời gian di chuyển, tăng khả năng tự học. Vì vậy việc xây dựng website học tiếng Nhật và quản lý website càng trở thành nhu cầu thiết yếu. Qua quá trình xây dựng website học tiếng Nhật, chúng em đã tích lũy được các kiến thức về các khái niệm cơ bản trong công nghệ phần mềm, quá trình phát triển phần mềm, phân tích và đặc tả yêu cầu, mô hình hóa hệ thống phần mềm, thiết kế kiến trúc, thiết kế giao diện người dùng, thiết kế chi tiết và cài đặt mã nguồn, kiểm thử phần mềm, quản lý dự án. Bên cạnh những lý thuyết cần phải có, nó còn phải đòi hỏi những kỹ năng như có khả năng tạo các tài liệu và chế tác phần mềm, có khả năng sử dụng các công cụ hỗ trợ phát triển phần mềm, học được các kỹ năng làm việc nhóm và khả năng vận dụng các kỹ thuật được học vào thực hành.

Công nghệ sử dụng:

- Backend: Laravel
- Frontend: Reactjs, Next js, Javascript, Axios
- Database: Mysql

6.2 Định hướng phát triển

Trong dự án này, chúng em đã hoàn thành được các yêu cầu đã đặt ra như: hiển thị bài học từ vựng, kanji, ngữ pháp, làm câu hỏi trắc nghiệm để ôn tập, tạo flashcard, quản lý tài khoản, giao diện thân thiện với người dùng.

Ngoài ra, do sự hạn chế về mặt thời gian và trình độ nên website chúng em xây dựng được mới chỉ đáp ứng được những yêu cầu cơ bản, nếu đưa vào thực tế phải có thêm định hướng phát triển sau này về nội dung, giao diện và nền tảng:

- Nội dung bài học cần bổ sung thêm để đáp ứng đủ nhu cầu và trình độ của nhiều đối tượng người học.
- Có phần nghe phát âm từ, nghe hiểu hội thoại, có các bài thi thử của các chứng chỉ tiếng Nhật như JLPT,...
- Có từ điển mở để tra cứu nghĩa, ngữ pháp, kanji của một từ, dịch nghĩa câu nhật - việt/ việt - nhật.
- Cung cấp và giới thiệu các thông tin về văn hóa, đời sống, đất nước và con người Nhật Bản.
- Thiết kế giao diện đẹp hơn, nhiều hình ảnh minh họa màu sắc, thú vị giúp kích thích tinh thần của người học.

- Quản lý thành viên vip, cho phép thành viên được hưởng một số quyền lợi đặc biệt khác.
- Người dùng có thể tự cài đặt hình ảnh đại diện cho bản thân.
- Có thể đăng nhập bằng nhiều cách như qua google, facebook,....Tạo xác thực email khi người dùng quên mật khẩu.
- Phát triển trên nền tảng mobile - nền tảng phổ biến, được ưa thích hiện nay.