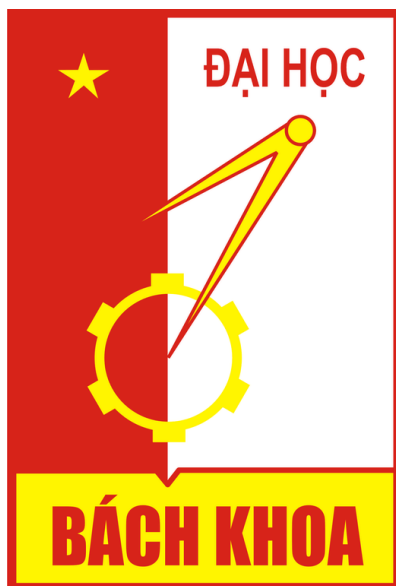


ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



BÁO CÁO CUỐI KỲ

Project: Phần mềm học từ vựng

Môn học: Nghiên cứu tốt nghiệp 2 - IT5022

Giảng viên hướng dẫn : Đỗ Quốc Huy

Sinh viên thực hiện : Đinh Thị Hồng Phúc - 20215118

Lớp : Việt Nhật 03 - K66

Mục lục

I. Khảo sát ứng dụng hiện có	4
1. Mở đầu	4
2. Anki	4
a. Tính năng hiện có	4
b. Ưu điểm	4
c. Nhược điểm	4
3. Quizlet	5
a. Tính năng hiện có	5
b. Ưu điểm	5
c. Nhược điểm	5
4. Kết luận	5
II. Tài liệu đặc tả yêu cầu phần mềm	6
1. Giới thiệu	6
2. Mô tả tổng quan	6
a. Các tác nhân	6
b. Biểu đồ use case tổng quan	6
c. Biểu đồ use case phân rã	8
- Phân rã use case “Manage public collections”	8
- Phân rã use case “Study collections”	8
- Phân rã use case “View statistics”	9
- Phân rã use case “Manage users”	9
- Phân rã use case “Manage notifications”	10
d. Quy trình sử dụng phần mềm	10
3. Đặc tả một số chức năng	11
a. Đặc tả use case UC001 “Login”	11
b. Đặc tả use case UC002 “Create menu”	12
c. Đặc tả use case UC003 “Change password”	13
d. Đặc tả use case UC005 “CRUD collections”	14
III. Một số lớp phân tích	17
1. Sequence Diagram	17
a. Use case “Login”	17
b. Use case “Create menu”	18
c. Use case “CRUD collections”	18
d. Use case “Search public collections”	20

2. Lớp phân tích	20
a. Use case “Login”	20
b. Use case “Create menu”	21
c. Use case “CRUD collections”	21
d. Use case “Search public collections”	22
IV. Thiết kế cơ sở dữ liệu	23
1. Sơ đồ thực thể liên kết ERD	23
2. Sơ đồ quan hệ	23
3. Xây dựng bảng	24
V. Thiết kế mockup	26
1. Figma - web	26
2. Figma - app	31
3. Mẫu thử tương tác độ tin cậy thấp - web	36
VI. Khảo sát ngôn ngữ	37
1. Ngôn ngữ lập trình thường được sử dụng trong lập trình web	37
2. Cơ sở dữ liệu thường được sử dụng	43
3. Kết luận	45
3.1. Sơ đồ thực thể liên kết ERD	45
3.2. Kỹ thuật sử dụng	45
4. Repository Github	46

I. Khảo sát ứng dụng hiện có

1. Mở đầu

- Mong muốn phát triển: Sản phẩm giúp ôn tập, ghi nhớ từ vựng.
- Mục tiêu: Phân tích các sản phẩm hiện có trên thị trường: Anki, Quizlet

2. Anki

Anki là một công cụ mạnh mẽ giúp học và ghi nhớ hầu hết mọi thứ, từ ngoại ngữ, y tế, lịch sử đến toán học và nhiều lĩnh vực khác. Được xây dựng trên hệ thống flashcards, Anki cho phép người dùng tạo ra các bộ thẻ ghi nhớ điện tử. Mỗi thẻ chứa một cặp từ hoặc ngữ cảnh học tập, ví dụ như một từ vựng tiếng Anh và nghĩa tương ứng, hoặc một câu hỏi và câu trả lời.

a. Tính năng hiện có

- Tạo và quản lý flashcard với nhiều chủ đề khác nhau. Tùy chỉnh flashcard: thêm hình ảnh, âm thanh... với nhiều loại thẻ khác nhau (thẻ điền vào chỗ trống, lựa chọn trắc nghiệm....)
- Tùy chỉnh giao diện: thay đổi màu sắc, phong chữ, bố cục...
- Học qua thuật toán Spaced Repetition System (SRS) để lên lịch ôn tập tối ưu dựa trên khả năng ghi nhớ của người dùng.
- Hỗ trợ đa nền tảng: máy tính, thiết bị di động, trình duyệt web.
- Chia sẻ và tải xuống gói thẻ.
- Thống kê quá trình học tập

b. Ưu điểm

- Tối ưu quá trình học tập bằng cách lên lịch ôn tập vào thời điểm tối ưu để tăng khả năng ghi nhớ lâu dài.
- Tự động hóa quá trình ôn tập: Lên lịch tự động, đảm bảo ôn tập hiệu quả.
- Giúp gợi nhớ lại kiến thức, bắt não phải nhớ lại → biết được phần nào còn yếu, đã quên để ôn tập nhiều hơn.
- Đồng bộ trên nhiều nền tảng: máy tính, điện thoại, thiết bị thông minh. Cho phép ôn tập mọi lúc mọi nơi.

c. Nhược điểm

- Nhiều chức năng và chưa phổ biến ở Việt Nam nên người mới tiếp cận khó sử dụng và khiến nhiều người bỏ cuộc.
- Nếu tạo flashcards thủ công sẽ tốn nhiều thời gian công sức nên nhiều người sẽ không sử dụng.
- Kiến thức nắm bắt được còn phụ thuộc vào chất lượng thẻ đã tạo.

- Giao diện đơn giản, không thu hút như các ứng dụng khác.

3. Quizlet

Quizlet là một nền tảng học tập trực tuyến phổ biến được sử dụng để tạo, chia sẻ và ôn tập thẻ ghi nhớ. Đây là một công cụ mạnh mẽ cho việc học và ghi nhớ thông tin thông qua các hoạt động như flashcards, trò chơi trực tuyến, bài kiểm tra và tài liệu học tập.

a. Tính năng hiện có

- Tạo, tùy chỉnh flashcard.
- Chia sẻ, tìm kiếm bộ flashcard từ cộng đồng người dùng.
- Cung cấp các trò chơi trực tuyến (hoạt động nhóm, lớp học...)
- Cung cấp bài kiểm tra giúp ôn tập kiến thức.
- Hỗ trợ đa nền tảng: máy tính, thiết bị di động, trình duyệt web.

b. Ưu điểm

- Ứng dụng di động cho phép người dùng tải bộ flashcard để học offline.
- Đồng bộ trên nhiều nền tảng: máy tính, điện thoại, thiết bị thông minh. Cho phép ôn tập mọi lúc mọi nơi.
- Khi tạo flashcard, ứng dụng gợi ý ý nghĩa (khi nhập từ mới), hình ảnh tương ứng với từ đã nhập. Tích hợp sẵn âm thanh của bộ flashcard.
- Giao diện đơn giản, dễ sử dụng.

c. Nhược điểm

- Các bộ thẻ chỉ đơn giản như một từ vựng tiếng Anh và nghĩa tương ứng.
- Việc ôn tập và ghi nhớ hoàn toàn phụ thuộc vào người dùng, ứng dụng không gợi ý thời gian ôn tập.

4. Kết luận

Xây dựng ứng dụng kết hợp những ưu điểm của Anki và Quizlet: Áp dụng kỹ thuật lặp lại ngắt quãng (giúp ghi nhớ dễ dàng hơn) và xây dựng giao diện dễ sử dụng tránh cho người sử dụng nhàm chán.

II. Tài liệu đặc tả yêu cầu phần mềm

1. Giới thiệu

Trong thời đại hiện nay, việc học các ngôn ngữ khác ngày càng trở nên phổ biến và quan trọng, và phương pháp sử dụng flashcard để học từ vựng đã trở thành một cách thức học tập hiệu quả. Các hệ thống học tập dựa trên flashcard cho phép người học tự tạo flashcard từ vựng phù hợp với mục tiêu học của mình, và cũng tạo điều kiện để hình thành cộng đồng, nơi mà mọi người có thể chia sẻ flashcard và nguồn tài liệu học ngoại ngữ với nhau. Điều này giúp việc học ngoại ngữ trở nên dễ dàng và hiệu quả hơn.

Mục đích của phần mềm nhằm tạo ra hệ thống học từ vựng với các chức năng (function) mà người dùng/vai trò người dùng có thể sử dụng tại thời điểm chạy.

Trước khi đăng nhập, người dùng có thể xem các public collection và standard collection nhưng không thể tương tác hoặc tạo collection riêng. Bất kỳ ai cũng có thể đăng ký, đăng nhập để trải nghiệm đầy đủ tính năng. Sau khi đăng nhập vào hệ thống, người dùng được cấp một không gian lưu trữ cho collection cá nhân, có thể tạo và quản lý collection chứa flashcards theo nhu cầu bản thân. Người dùng có thể học, thêm mới, xem, chỉnh sửa, hoặc xóa flashcard. Có thể thiết lập quyền riêng tư cho collection của mình là public hoặc private. Người dùng cũng có thể xem, tìm kiếm public collection của người khác và sử dụng standard collection của hệ thống, và clone collection đó vào không gian lưu trữ cá nhân.

Quản trị hệ thống đóng vai trò quan trọng trong việc quản lý thông báo và người dùng, đảm bảo an ninh, quyền riêng tư, và bảo trì hệ thống. Họ chịu trách nhiệm cập nhật và duy trì cơ sở hạ tầng kỹ thuật số, cũng như quản lý dữ liệu để bảo đảm nội dung luôn chất lượng và hiện đại.

2. Mô tả tổng quan

a. Các tác nhân

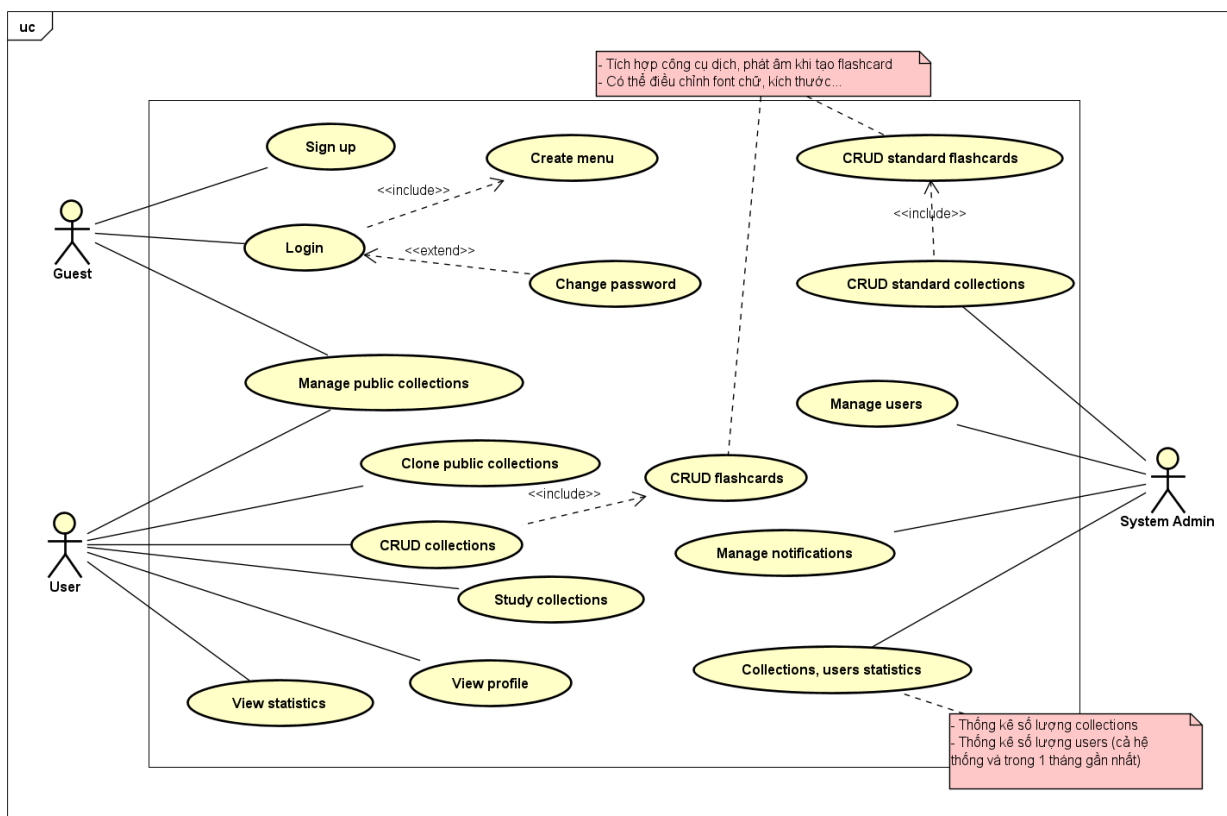
Phần mềm có 3 tác nhân là Guest, User và System Admin.

b. Biểu đồ use case tổng quan

Khi chưa đăng nhập, khách có thể đăng ký tài khoản mới, đăng nhập, yêu cầu thiết lập lại mật khẩu khi quên mật khẩu, và thiết lập lại mật khẩu khi nhận được chỉ dẫn thiết lập qua email. Đồng thời, khách cũng có thể xem các collection đã được public, nhưng không tương tác được với các collection. Khi khách đăng nhập thành công, khách trở thành người dùng, và sử dụng được các chức năng

của hệ thống như tạo collection mới, tạo lớp,... Khi khách yêu cầu thiết lập lại mật khẩu, hệ thống thực hiện tạo token và gửi chỉ dẫn thiết lập lại mật khẩu qua email.

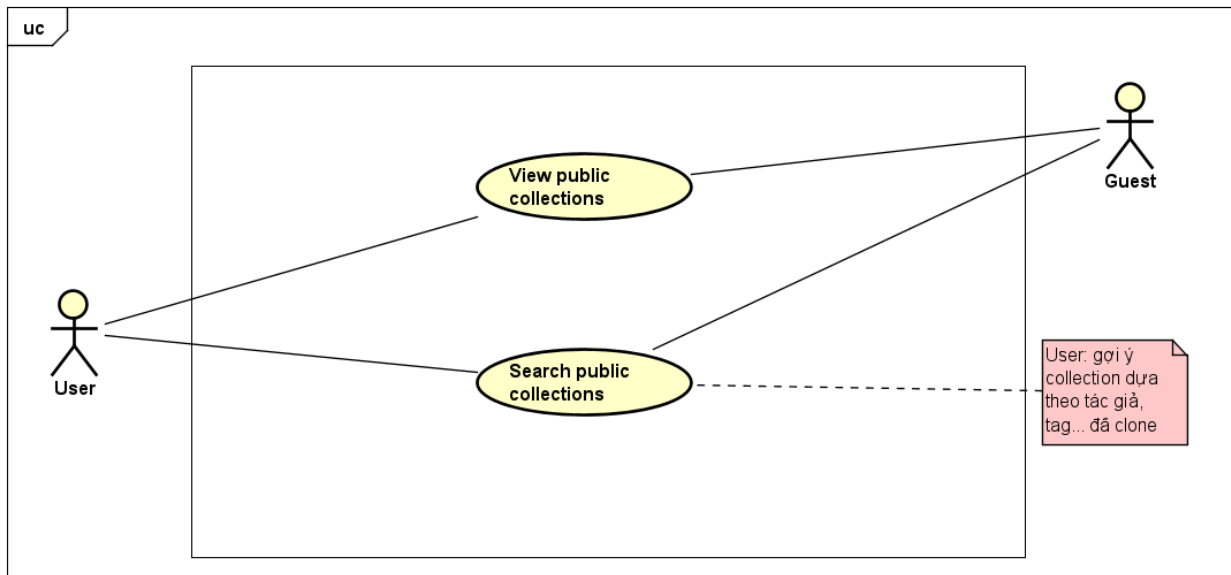
Sau khi đăng nhập, người dùng có thể xem và cập nhật thông tin cá nhân của mình, và có thể thay đổi thay đổi mật khẩu.



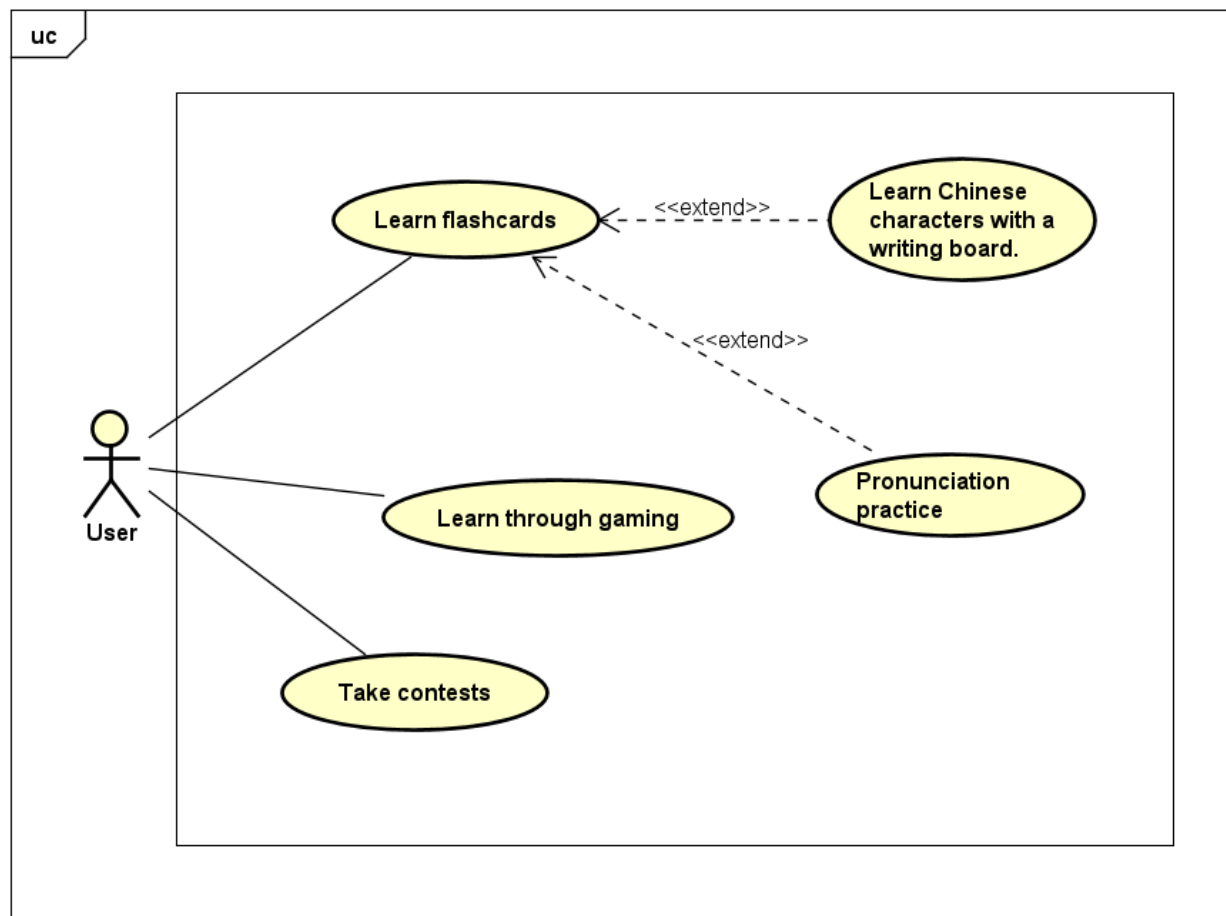
Quản trị viên có thể quản lý người dùng, chức năng và loại chức năng có trong hệ thống. Các use case của quản trị viên trong biểu đồ use case tổng quan này là use case phức hợp của một nhóm các use case. Chi tiết về các use case phức này được đưa ra trong các biểu đồ phân rã ở phần sau.

c. Biểu đồ use case phân rã

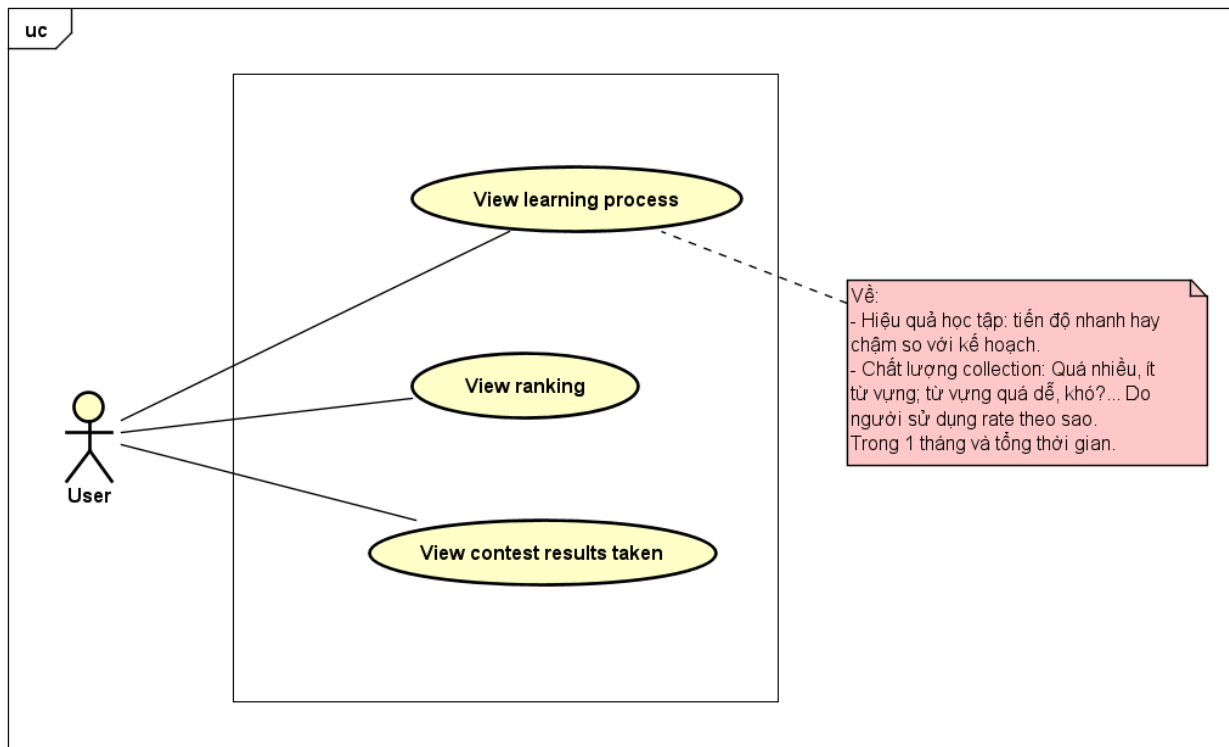
- Phân rã use case “Manage public collections”



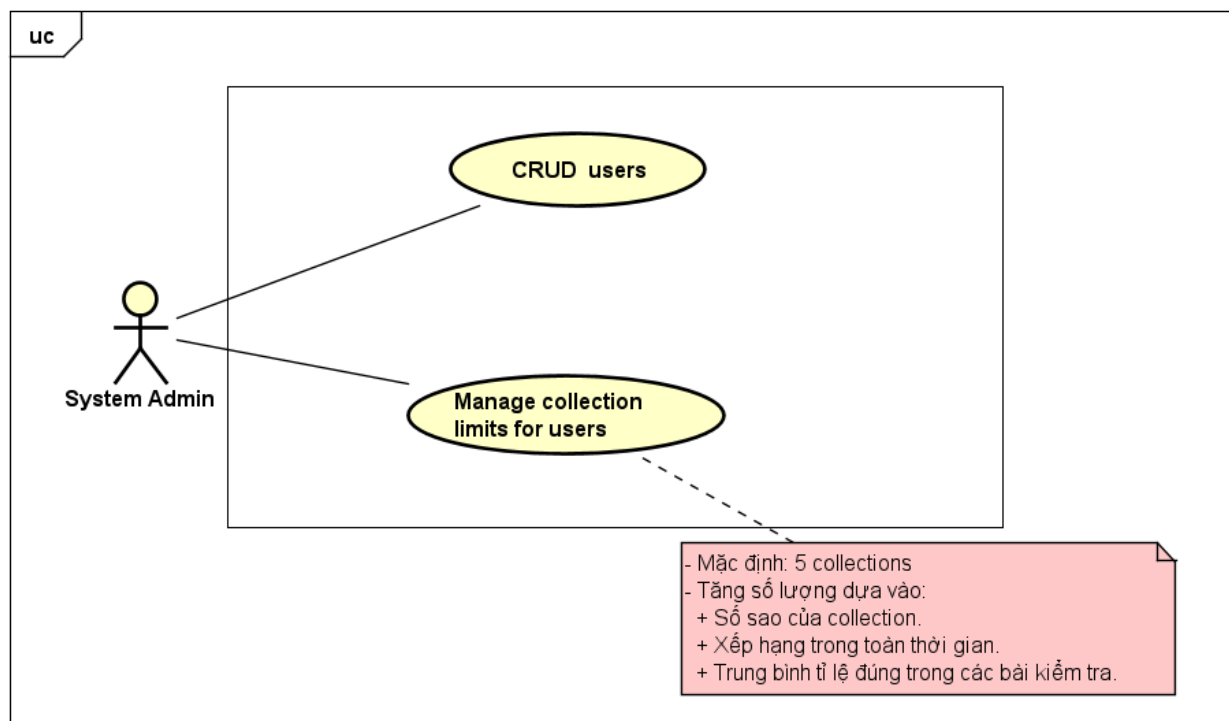
- Phân rã use case “Study collections”



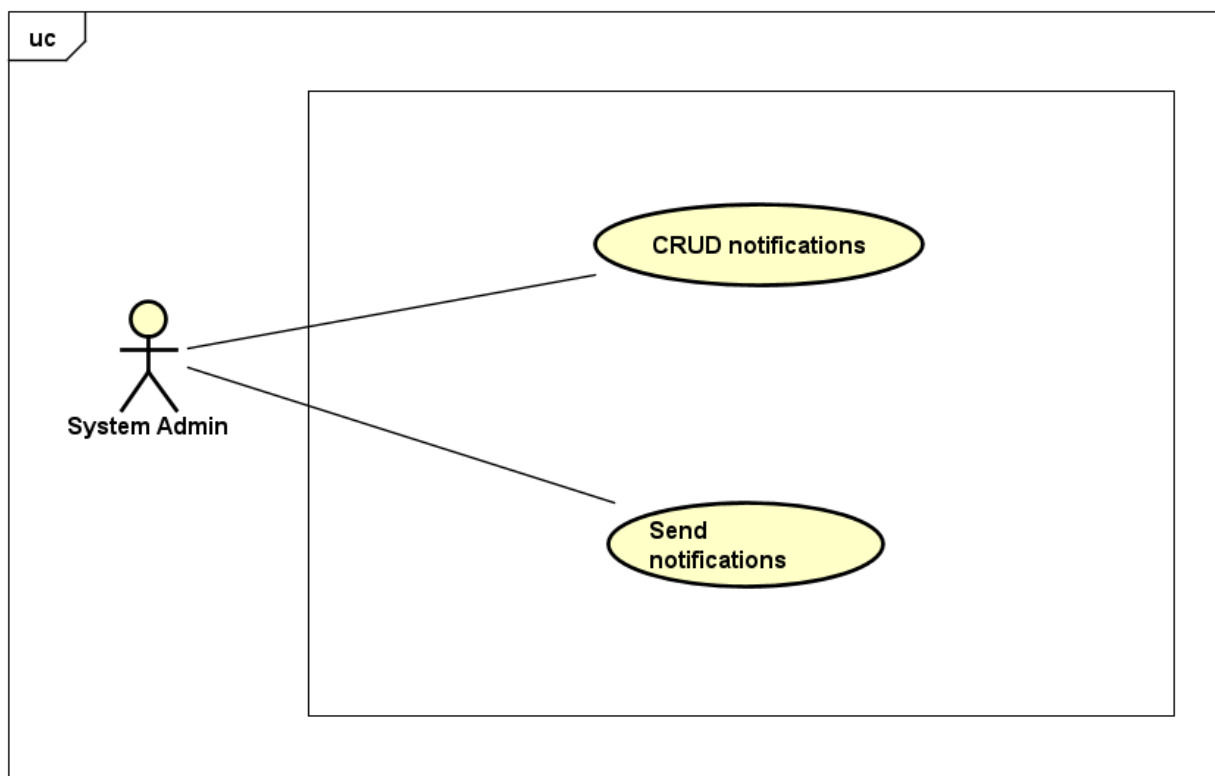
- Phân rã use case “View statistics”



- Phân rã use case “Manage users”



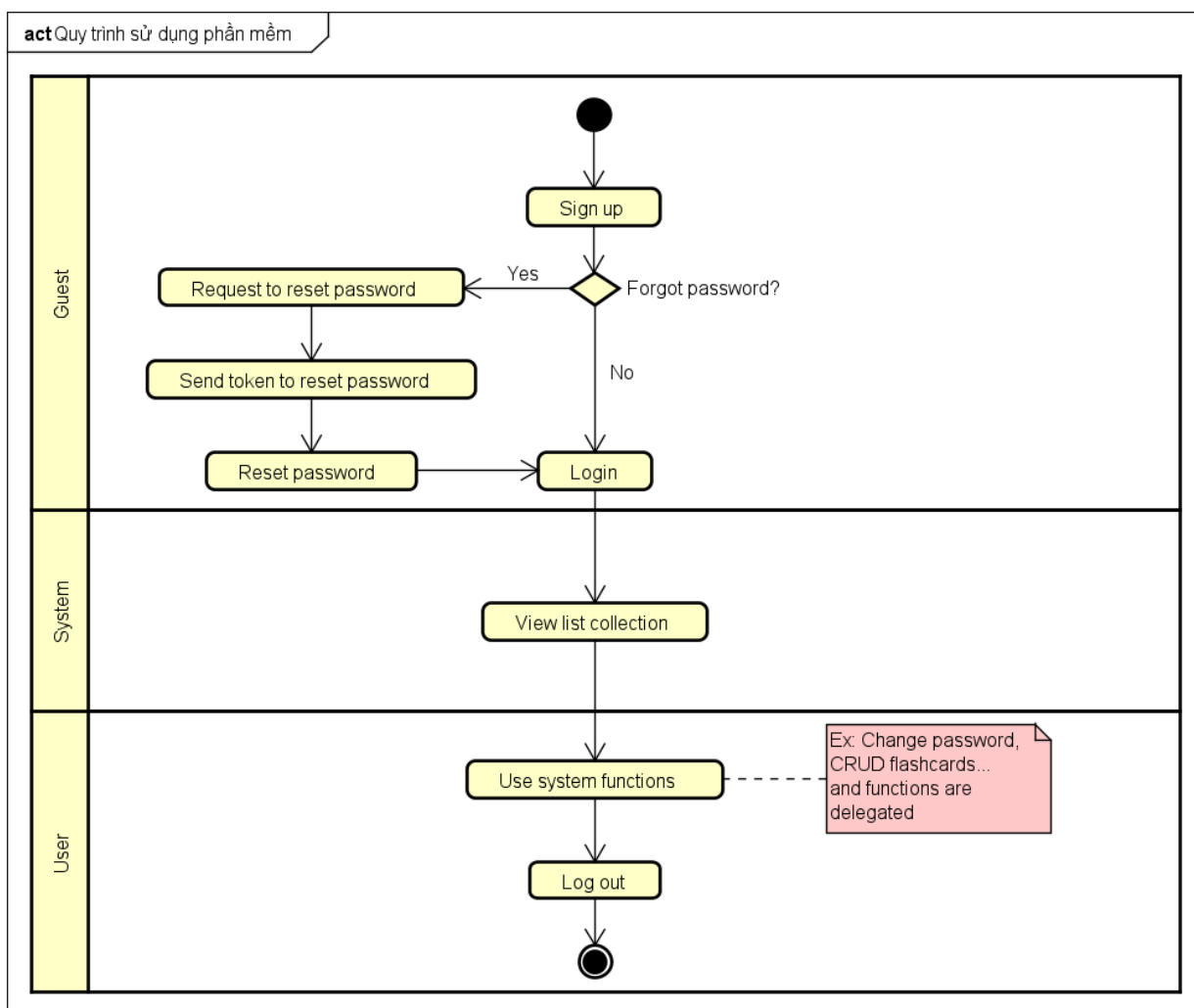
- Phân rã use case “Manage notifications”



d. Quy trình sử dụng phần mềm

Khách có thể đăng ký để tạo ra tài khoản cho mình. Sau đó có thể đăng nhập để sử dụng các chức năng của phần mềm. Nếu khách quên mật khẩu, khách có thể yêu cầu hệ thống cho phép mình thiết lập lại mật khẩu. Lúc này, hệ thống sẽ gửi token trong liên kết kèm gửi qua email đã đăng ký. Khách có thể vào liên kết đó để thực hiện việc thiết lập lại mật khẩu.

Sau khi đăng nhập thành công vào hệ thống, người dùng có thể sử dụng các chức năng như Xem và cập nhật thông tin cá nhân của mình, Thay đổi mật khẩu, và các chức năng được người quản trị cấp phép cho các nhóm người dùng mà họ tham gia.



3. Đặc tả một số chức năng

Chi tiết về các use case được đưa ra trong phần 2 được đặc tả đầy đủ trong [file \(GR2 - SRS\)](#).

a. Đặc tả use case UC001 “Login”

Mã Use case	UC001	Tên Use case	Login
Tác nhân	Guest		
Tiền điều kiện	Không		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1.	Guest	chọn chức năng <i>Login</i>
	2.	Hệ thống	hiển thị giao diện đăng nhập
	3.	Guest	nhập email và mật khẩu (mô tả phía dưới *)

	4.	Guest	yêu cầu đăng nhập
	5.	Hệ thống	kiểm tra xem khách đã nhập các trường bắt buộc nhập hay chưa
	6.	Hệ thống	kiểm tra email và mật khẩu có hợp lệ do khách nhập trong hệ thống hay không
	7.	Hệ thống	gọi use case <i>Study collections</i> kèm email của khách đăng nhập
Luồng sự kiện thay thế			
	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu khách nhập thiếu
	7a.	Hệ thống	thông báo lỗi: Email và/hoặc mật khẩu chưa đúng nếu không tìm thấy email và mật khẩu trong hệ thống
	7b.	Hệ thống	thông báo lỗi: Tài khoản bị khoá, nếu email/mật khẩu đúng như tài khoản đang bị admin khoá.
	7c1 .	Hệ thống	gọi use case <i>Change password</i> nếu đúng email và mật khẩu; nhưng người dùng được đánh dấu là cần thay đổi mật khẩu
	7c2 .	Hệ thống	gọi use case <i>Study collections</i> kèm email của khách đăng nhập
Hậu điều kiện	Không		

* Dữ liệu đầu vào của thông tin cá nhân gồm các trường dữ liệu sau:

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1.	Email		Có		h.anh@gmail.com
2.	Mật khẩu		Có		ToiLa12#\$

b. Đặc tả use case UC002 “Create menu”

Mã Use case	UC002	Tên Use case	Create menu
Tác nhân	Hệ thống		
Tiền điều kiện	Guest đăng nhập thành công với vai trò User		

Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1.	Hệ thống	gọi tới use case <i>Create menu</i> kèm email của người dùng đăng nhập
	2.	Hệ thống	kiểm tra người dùng có sở hữu collection nào không
	3.	Hệ thống	lấy danh sách các collection theo tag mà người dùng sở hữu
	4.	Hệ thống	hiển thị danh sách các collection đó (như mô tả *)
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	3a.	Hệ thống	thông báo: Không có collection nào nếu trả về danh sách rỗng
Hậu điều kiện	Không		

* Dữ liệu đầu ra khi hiển thị danh sách collection:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	STT	Số thứ tự		1
2.	Tên collection			TOEIC Writing Essential 1
3.	Collection tag			#toEIC, #english
4.	Số flashcard mới tạo			30
5.	Số flashcard đến hạn			20

c. Đặc tả use case UC003 “Change password”

Mã Use case	UC003	Tên Use case	Change password
Tác nhân	User		
Tiền điều kiện	Guest đăng nhập thành công với vai trò User		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1.	User	chọn chức năng <i>Change password</i>
	2.	Hệ thống	hiển thị form thay đổi mật khẩu
	3.	User	nhập các thông tin (mô tả phía dưới *)
	4.	User	yêu cầu thay đổi mật khẩu
	5.	Hệ thống	kiểm tra các trường bắt buộc đã được nhập hay chưa

	6.	Hệ thống	kiểm tra mật khẩu mới có khác mật khẩu cũ không
	7.	Hệ thống	kiểm tra mật khẩu có đủ mức độ an toàn hay không
	8.	Hệ thống	kiểm tra mật khẩu nhập lại và mật khẩu có trùng nhau hay không
	9.	Hệ thống	cập nhật mật khẩu mới trong cơ sở dữ liệu và thông báo thay đổi mật khẩu thành công
Luồng sự kiện thay thế			
	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	thông báo lỗi: Cần nhập các trường bắt buộc nếu người dùng nhập thiếu
	7a.	Hệ thống	thông báo lỗi: Mật khẩu mới và mật khẩu cũ là giống nhau nếu hai mật khẩu giống nhau
	8a.	Hệ thống	thông báo lỗi: Mật khẩu cần đảm bảo độ an toàn nếu mật khẩu không đảm bảo độ an toàn được quy định bởi hệ thống (có ít nhất 8 ký tự, trong đó có cả chữ cái thường và chữ cái hoa, ít nhất 1 chữ số và 1 ký tự đặc biệt)
	9a.	Hệ thống	thông báo lỗi: mật khẩu xác nhận không trùng với mật khẩu nếu hai mật khẩu không trùng nhau
Hậu điều kiện			
Không			

* Dữ liệu đầu vào của thông tin cá nhân gồm các trường dữ liệu sau:

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1.	Mật khẩu cũ		Có		ToiLa12#\$
2.	Mật khẩu mới		Có	Ít nhất 8 ký tự, gồm cả chữ cái hoa và thường, ít nhất 1 chữ số và ký tự đặc biệt	Sausau12!
3.	Xác nhận mật khẩu		Có	Trùng với mật khẩu mới	ToiLa12#\$

d. Đặc tả use case UC005 “CRUD collections”

Mã Use case	UC005	Tên Use case	CRUD collections
Tác nhân	User		
Tiền điều kiện	Guest đăng nhập thành công với vai trò User		
<u>Xem (R):</u>			

	STT	Thực hiện bởi	Hành động
Luồng sự kiện chính	1.	Hệ thống	gọi tới use case <i>Create menu</i> kèm email của khách đăng nhập
	2.	Hệ thống	lấy danh sách tất cả collection
	3.	Hệ thống	hiển thị giao diện Xem danh sách collection gồm: - Phía trên: Danh sách collection để người dùng có thể lọc collection theo tag - Trung tâm: Danh sách collection (mô tả phía dưới *)
Luồng sự kiện thay thế	5a1	User	lựa chọn một hoặc nhiều tag
	5a2	Hệ thống	lấy các collection thuộc (các) tag đã chọn, hiển thị lên giao diện

Sửa:

	STT	Thực hiện bởi	Hành động
Luồng sự kiện chính	1.	User	chọn một collection và yêu cầu sửa
	2.	Hệ thống	lấy thông tin chi tiết của collection và hiển thị thông tin cũ của collection trên giao diện sửa collection
	3.	User	chỉnh sửa các thông tin collection (mô tả phía dưới **) và yêu cầu sửa
	4.	Hệ thống	kiểm tra các trường bắt buộc nhập
	5.	Hệ thống	cập nhật các thông tin cần chỉnh sửa và thông báo chỉnh sửa thành công.
Luồng sự kiện thay thế	5a	Hệ thống	thông báo lỗi: chưa nhập đủ các trường cần thiết nếu người dùng nhập thiếu trường

Xoá:

	STT	Thực hiện bởi	Hành động
Luồng sự kiện chính	1.	User	chọn một collection và yêu cầu xoá
	2.	Hệ thống	hiển thị thông báo yêu cầu người dùng xác nhận việc xoá
	3.	User	xác nhận xoá collection
	4.	Hệ thống	xoá collection và thông báo xoá thành công
Luồng sự kiện thay thế	4a	Hệ thống	kết thúc use case nếu người dùng xác nhận không xoá

Thêm:

	STT	Thực hiện bởi	Hành động
Luồng sự kiện chính	1.	User	yêu cầu thêm collection mới
	2.	Hệ thống	hiển thị giao diện thêm collection

	3.	User	nhập các thông tin collection (mô tả phía dưới **) và yêu cầu thêm
	4.	Hệ thống	kiểm tra các trường bắt buộc nhập
	5.	Hệ thống	thêm collection mới và thông báo thành công.
Luồng sự kiện thay thế	5a	Hệ thống	thông báo lỗi: chưa nhập đủ các trường cần thiết nếu người dùng nhập thiếu trường
Hậu điều kiện	Không		

* Dữ liệu đầu ra khi hiển thị danh sách collection:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	STT	Số thứ tự		1
2.	Tên collection			TOEIC Writing Essential 1
3.	Collection tag			#toEIC, #english
4.	Ngày tạo	Chỉ chọn trong DatePicker	dd/mm/yyyy	12/12/2005

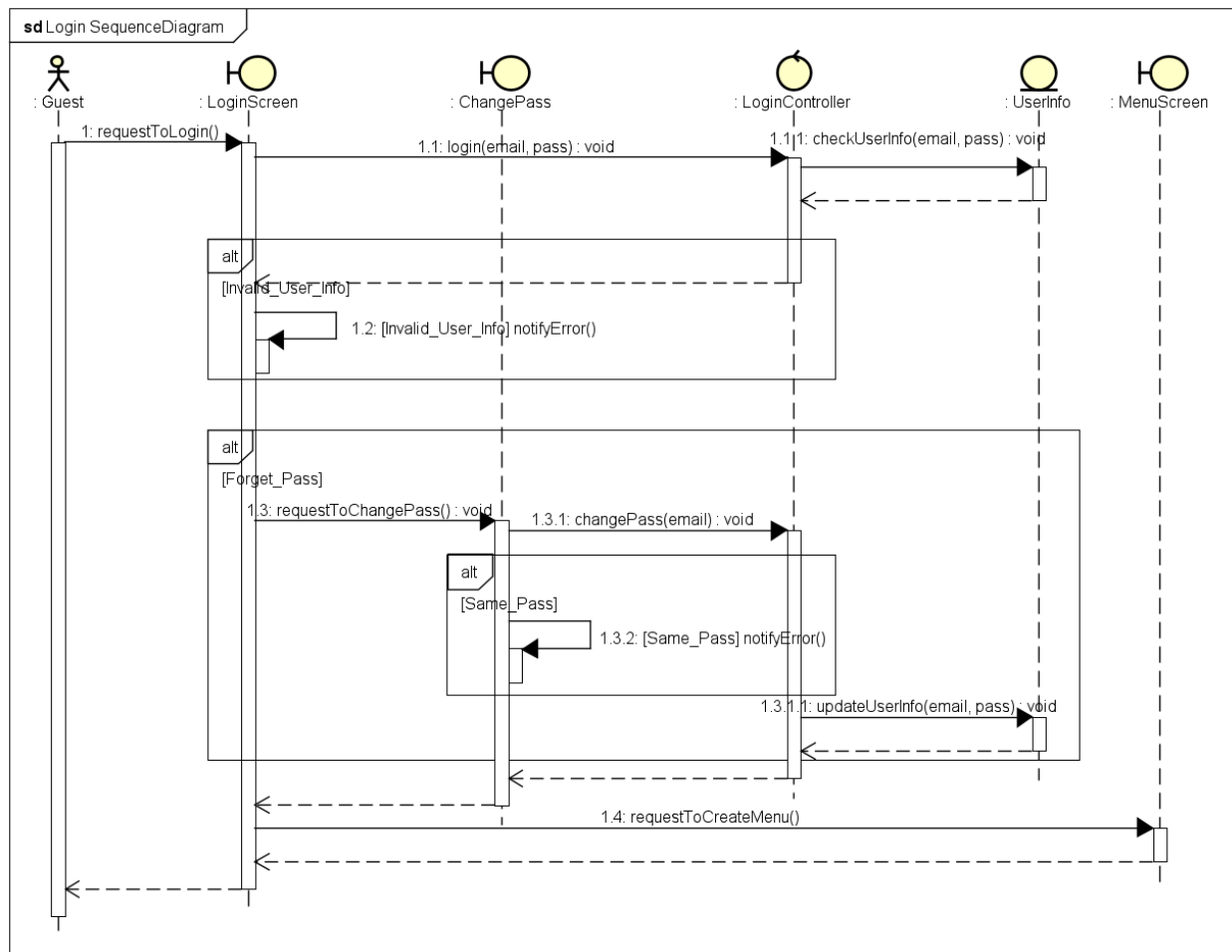
** Dữ liệu đầu vào khi thêm/sửa collection:

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1.	Tên collection		Có		TOEIC Writing Essential 1
2.	Collection tag		Có		#toEIC, #english
3.	Collection privacy	Lựa chọn Public hoặc Private	Có		Lựa chọn Public
4.	Mô tả		Không		Từ vựng TOEIC

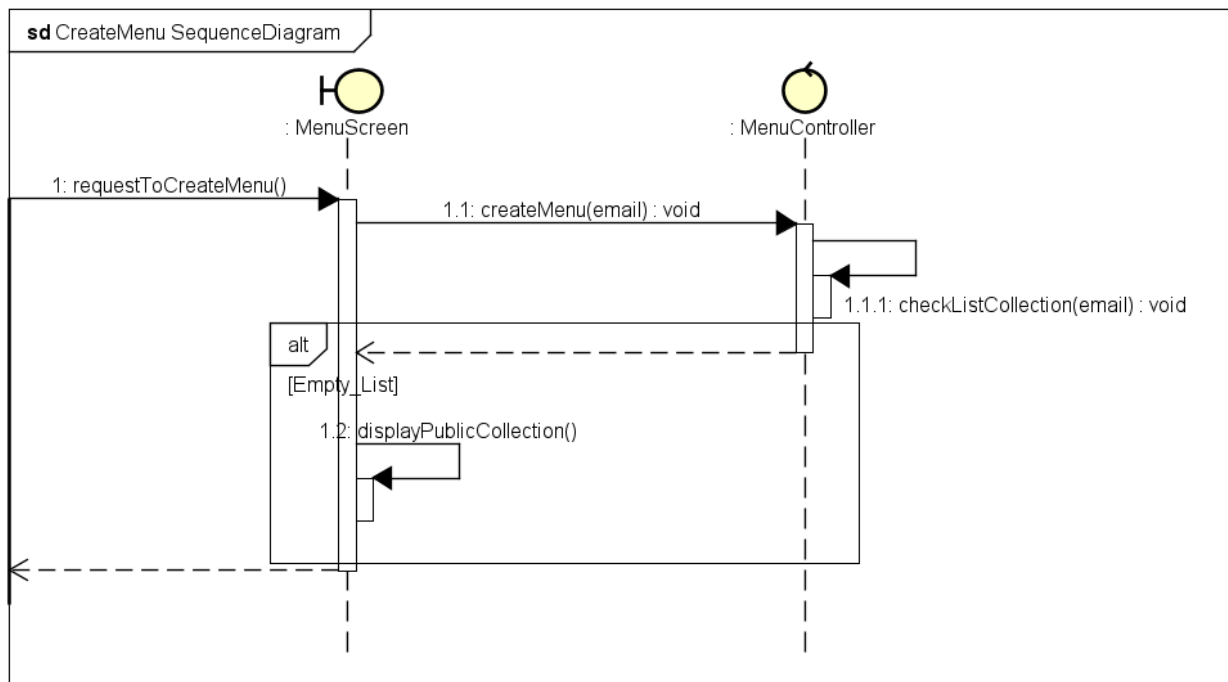
III. Một số lớp phân tích

1. Sequence Diagram

a. Use case “Login”

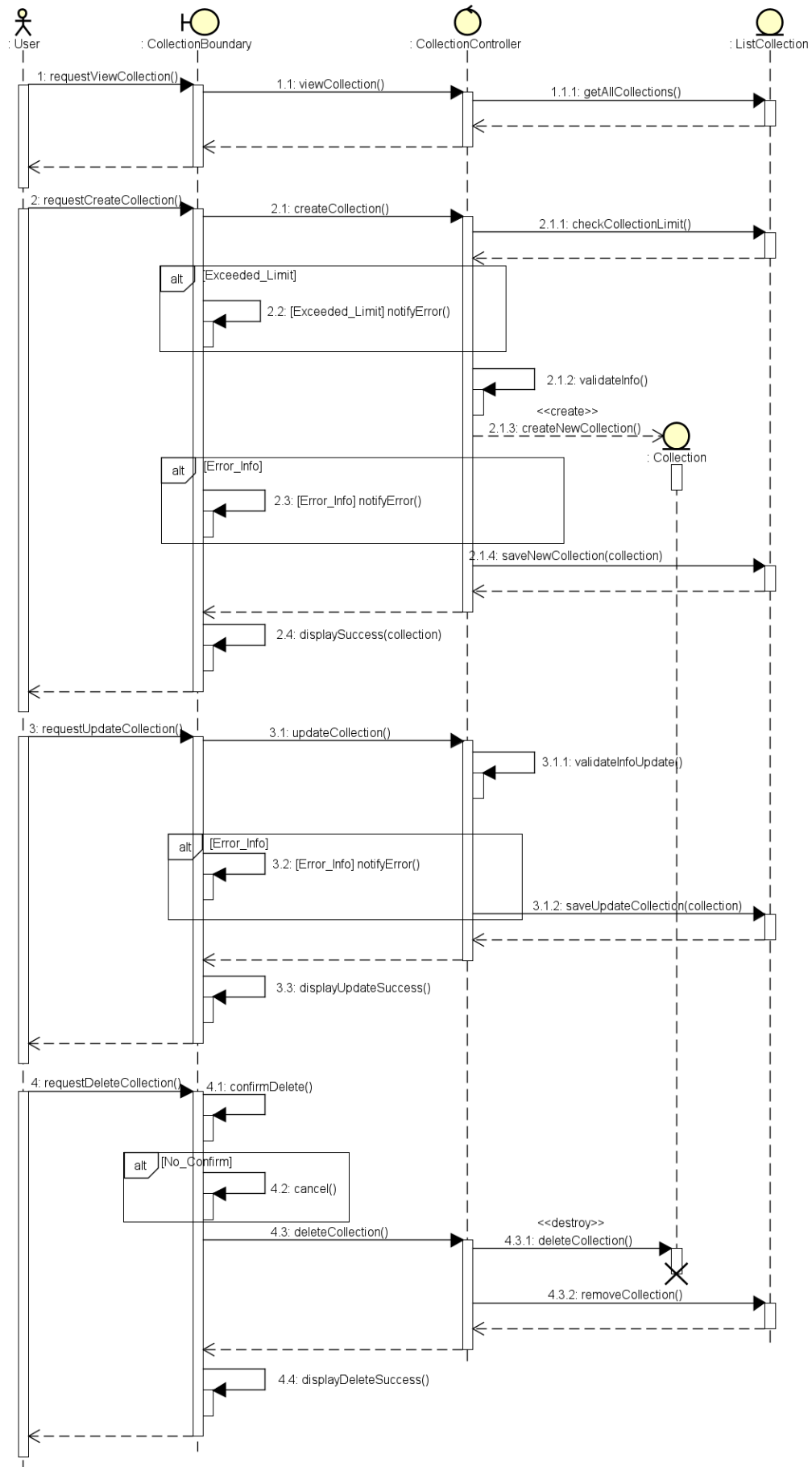


b. Use case “Create menu”

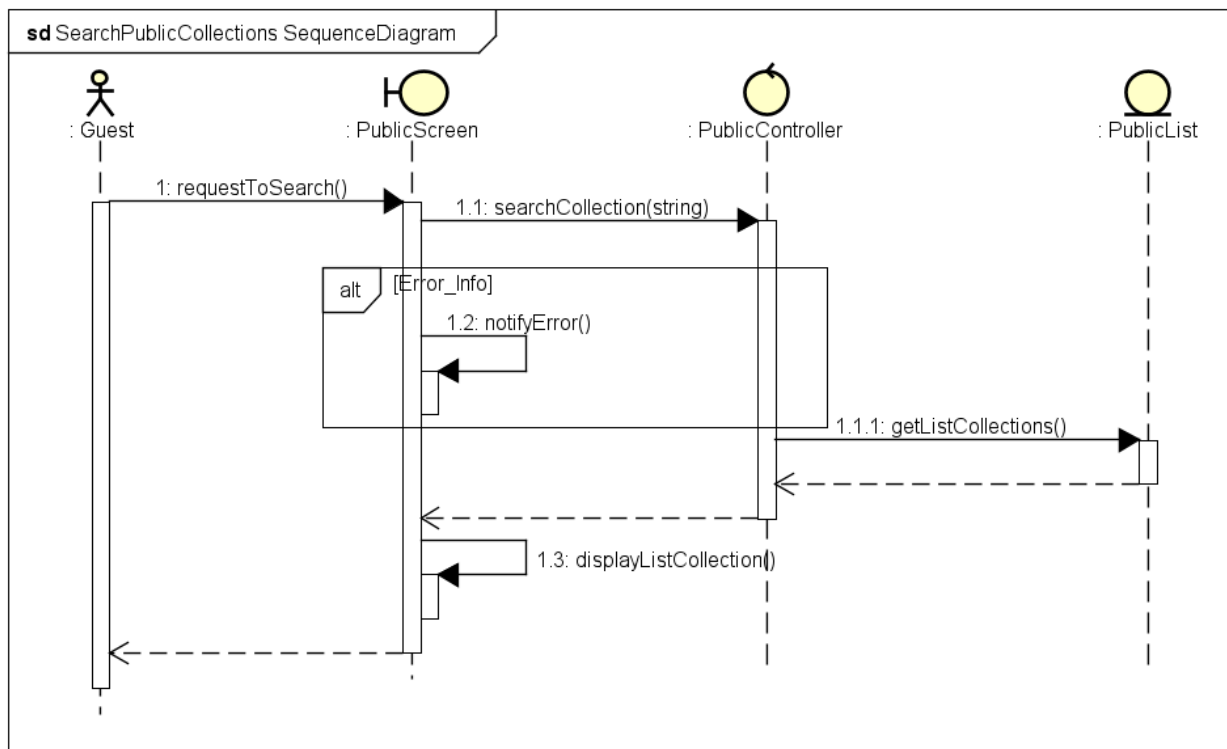


c. Use case “CRUD collections”

sd CRUD collections SequenceDiagram

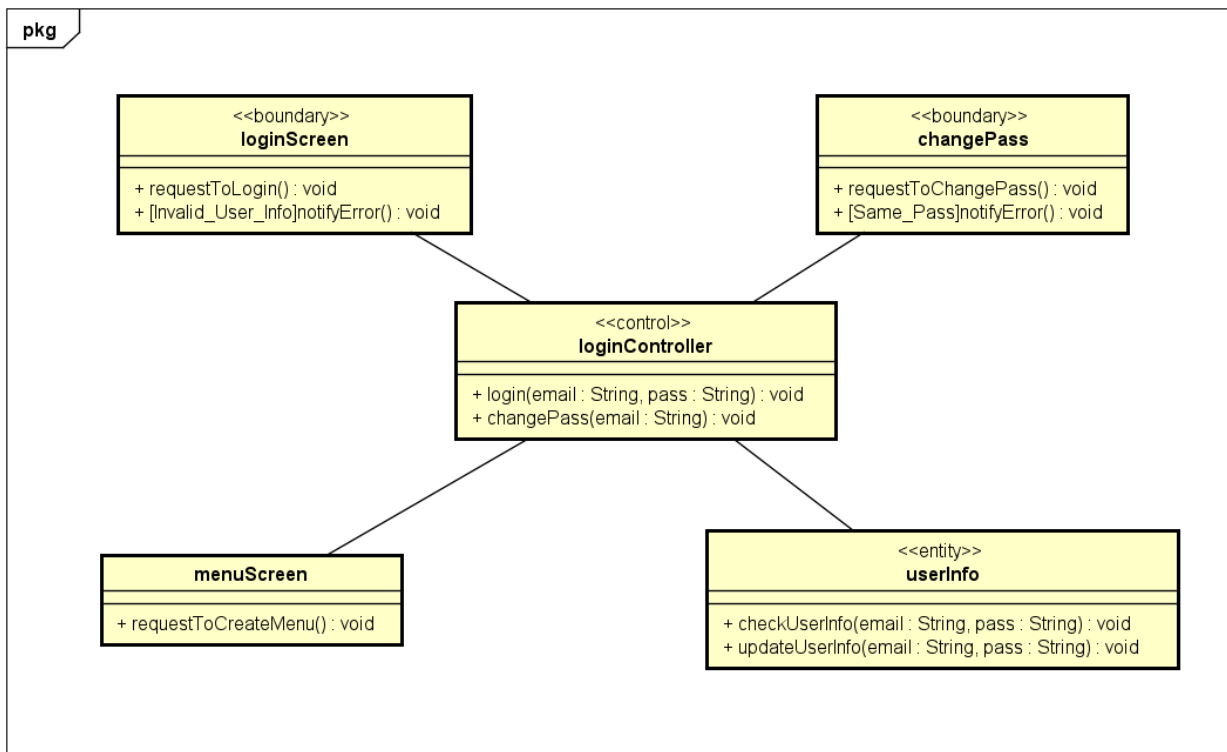


d. Use case “Search public collections”

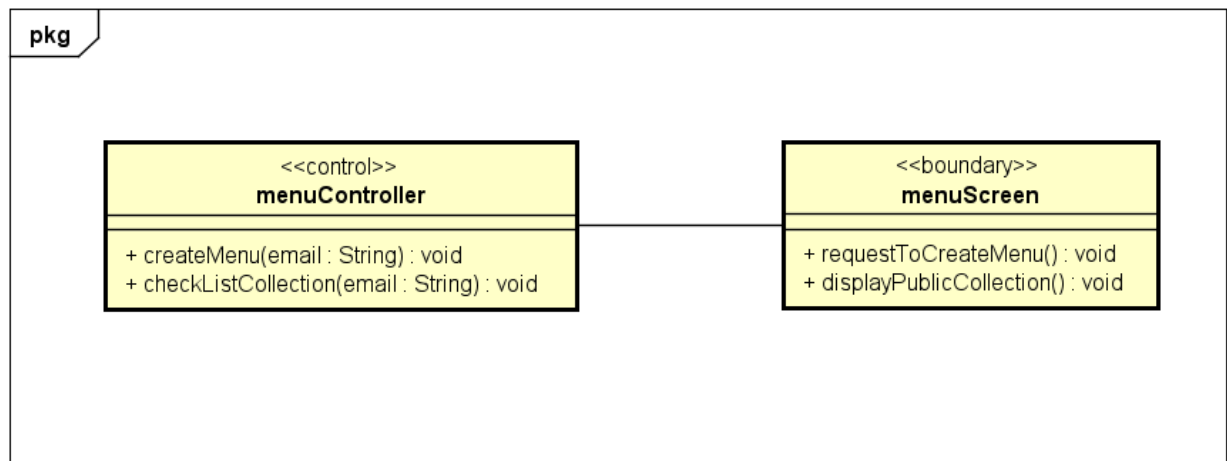


2. Lớp phân tích

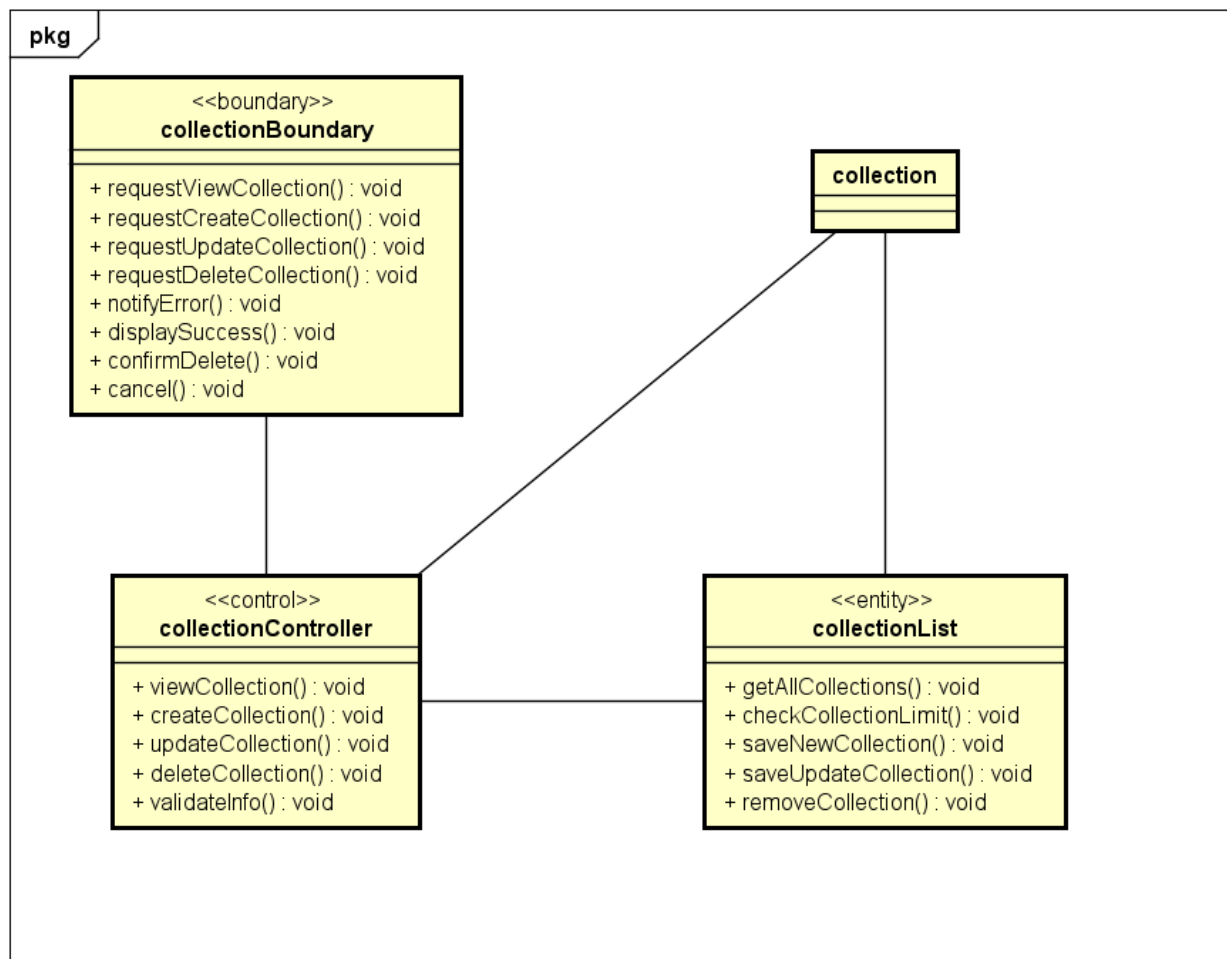
a. Use case “Login”



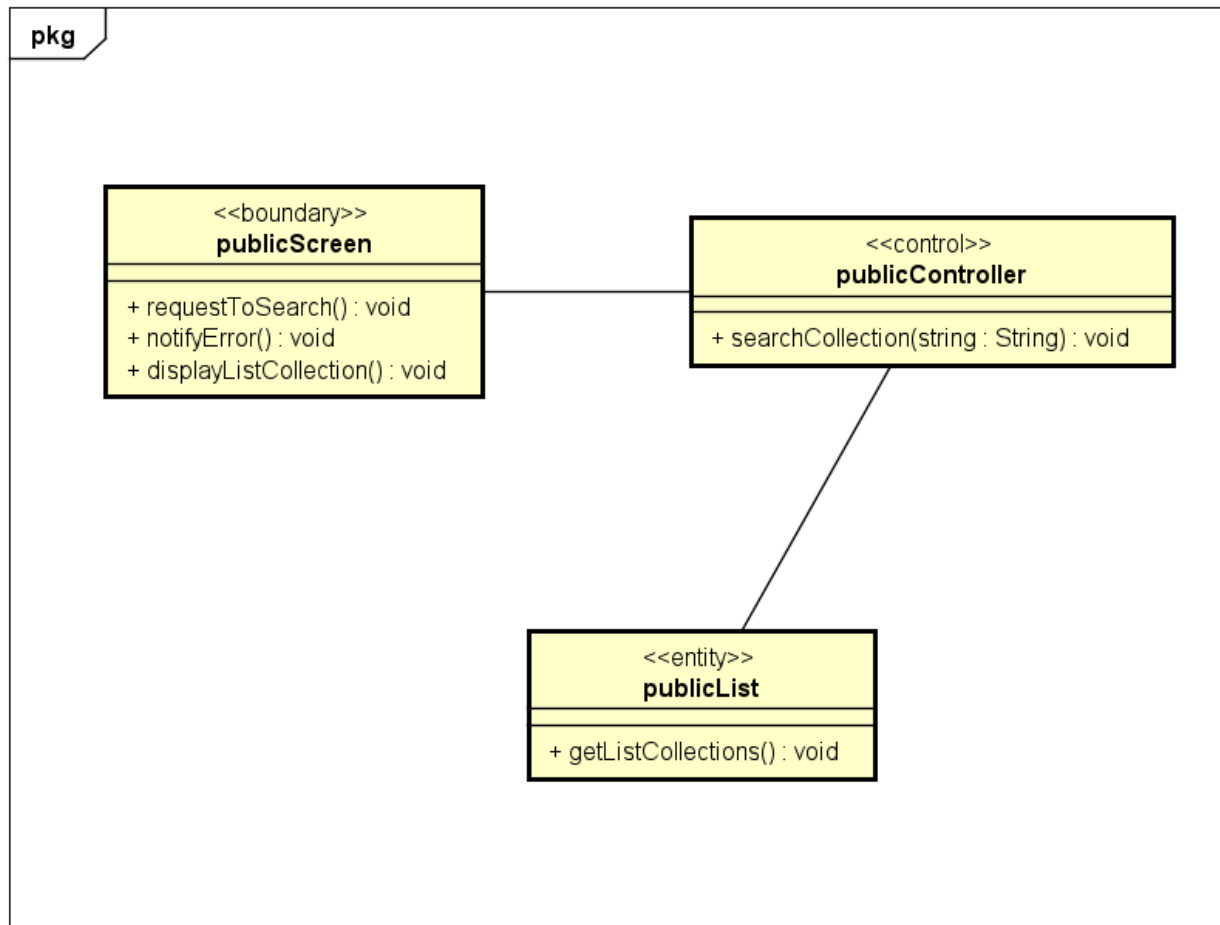
b. Use case “Create menu”



c. Use case “CRUD collections”

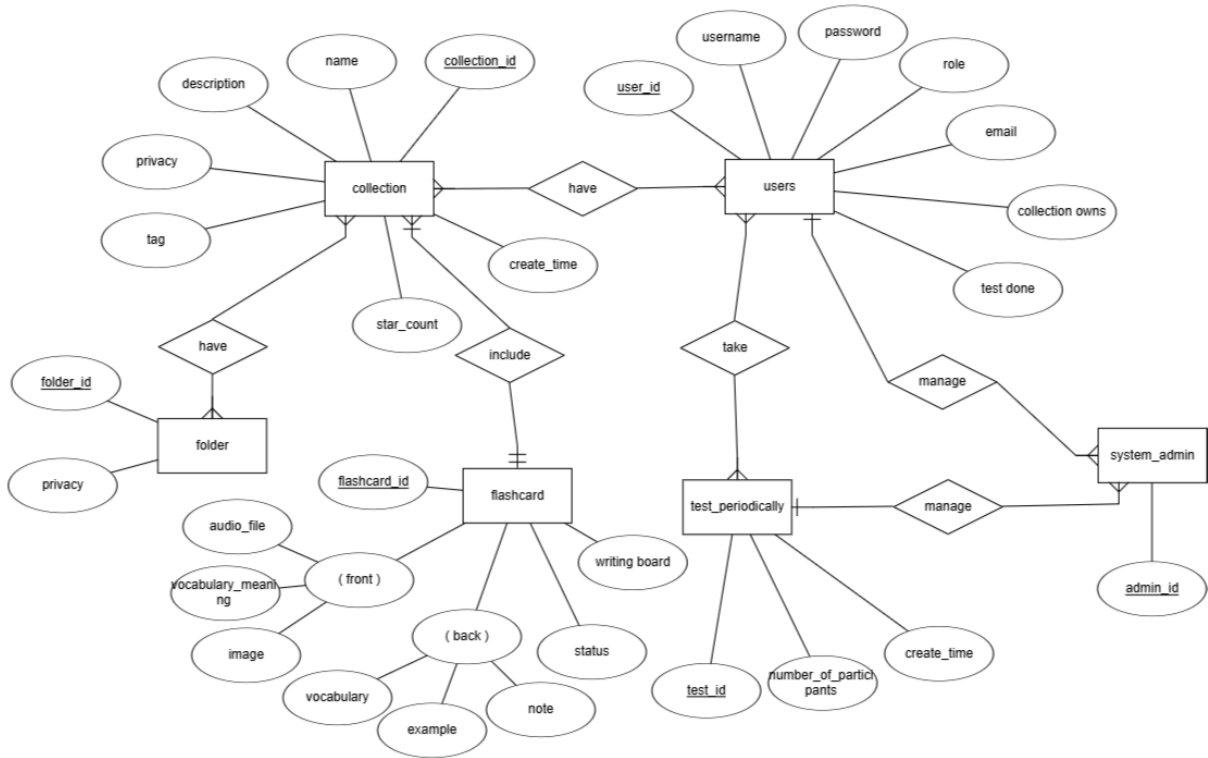


d. Use case “Search public collections”

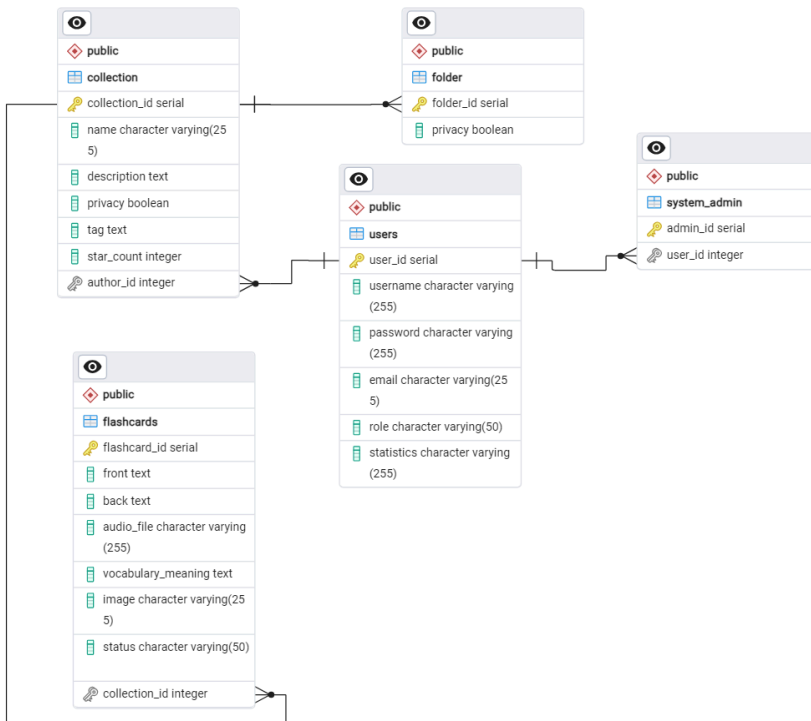


IV. Thiết kế cơ sở dữ liệu

1. Sơ đồ thực thể liên kết ERD



2. Sơ đồ quan hệ



3. Xây dựng bảng

1. Thuộc tính bảng **users**:

+ user_id (serial) (PK)	ID người dùng
+ username (character)	Tên người dùng
+ password (integer)	Mật khẩu
+ email (character)	Email của người dùng
+ role (character)	Vai trò của người dùng (guest, user)
+ statistics (character)	Thống kê quá trình học của người dùng

2. Thuộc tính bảng **collection**:

+ collection_id (serial) (PK)	ID của collection
+ name (character)	Tên collection
+ description (text)	Mô tả về collection
+ privacy (boolean)	Collection công khai hay riêng tư
+ tag (text)	Tag của collection
+ star_count (int)	Số sao collection được đánh giá

3. Thuộc tính bảng **flashcard**:

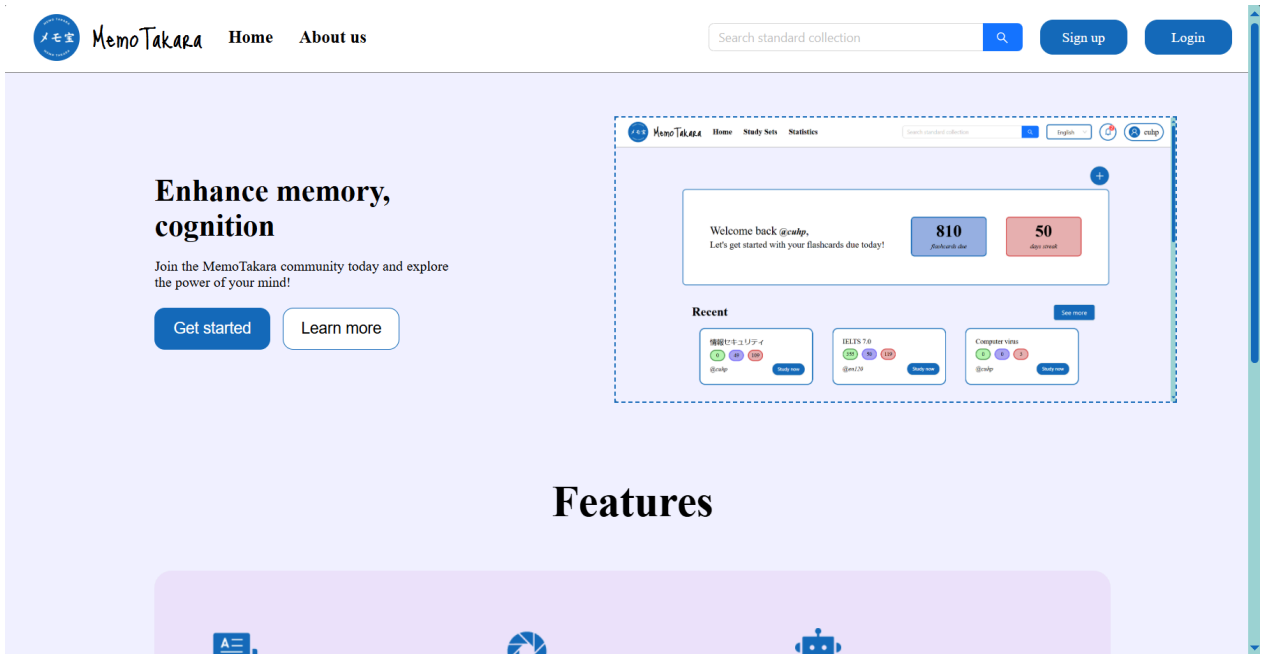
+ flashcard_id (serial) (PK)	ID của flashcard
+ front (text)	Mặt trước của thẻ
+ back (text)	Mặt sau của thẻ
+ audio_file (character)	File phát âm của từ vựng
+ vocabulary_meaning (character)	Nghĩa của từ vựng
+ image (character)	Hình ảnh miêu tả từ vựng

- + status (character) Trạng thái của từ vựng (long_term, short_term, forgotten, known, favorite)
4. Thuộc tính bảng **system_admin**:
- + admin_id (serial) ID của người quản trị hệ thống (PK)
5. Thuộc tính bảng **folder**:
- + folder_id (serial) ID của folder (PK)
 - + privacy (boolean) Folder công khai hay riêng tư

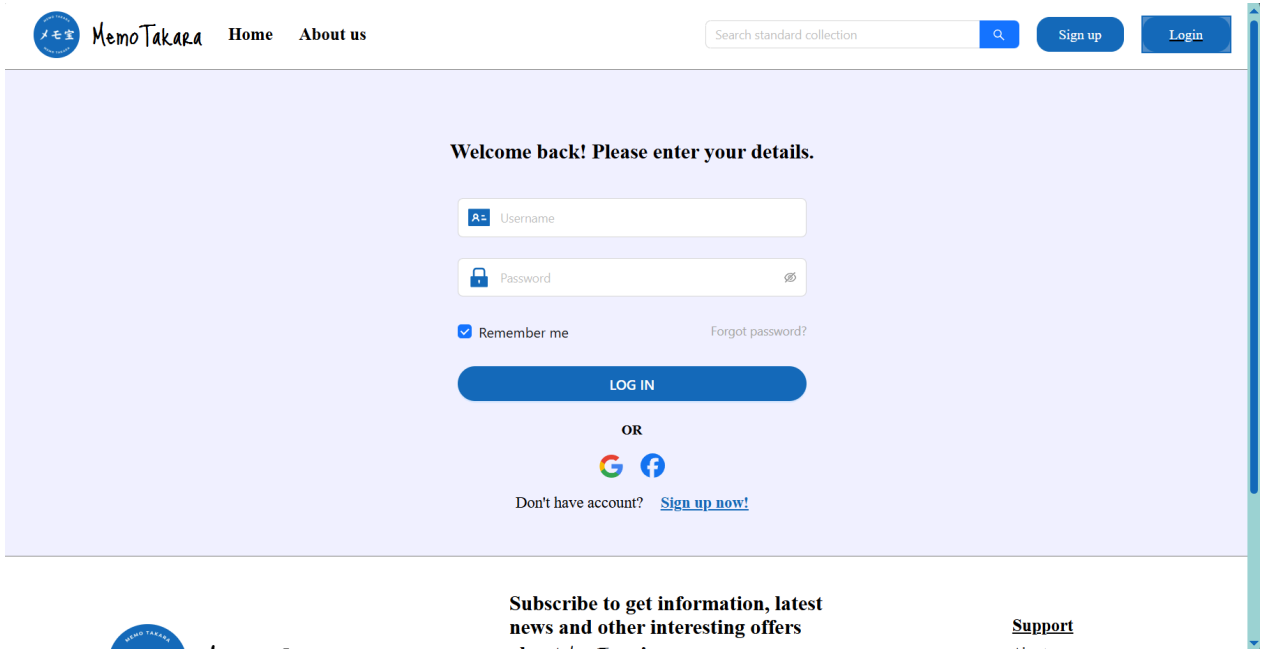
V. Thiết kế mockup

1. Figma - web

- Landing page



- Sign up (Login)



- Dashboard



Welcome back @cuhp293,
Let's get started with your flashcards due today!

810

flashcards due

50

days streak

Recent

[See more](#)

情報セキュリティ

0 49 109

@cuhp

[Study now](#)

IELTS 7.0

355 50 119

@en120

[Study now](#)


Computer virus


0 0 3


@cuhp


[Study now](#)

- Statistics


[Home](#)
[Study sets](#)
[Statistics](#)

 English



 cuhp

Study History

- Learned 100 flashcards of HSK 4
2024-05-10
- Created 情報セキュリティ
2024-05-10
- Re-learned 50 flashcards of HSK 4
2024-05-05
- Learned 100 flashcards of HSK 4
2024-05-01
- Learned 20 flashcards of TOEIC 500+
2024-04-23
- Clone HSK 4
2024-04-23



Calendar

< 2023 >





Subscribe to get information, latest news
and other interesting offers about MemoTakara

Support


- [About us](#)
- [Terms and Condition](#)
- [Help/FAQ](#)

© Copyright cuhp293 2024







- Settings


 MemoTakara

HomeStudy setsStatistics

Search standard collection


English



 cuhp

Settings

Account



+ Import image

Email:
Change your email

User name:
Change your user name

Unlink your Google account

Unlink Google

Unlink your Facebook account

Unlink Facebook

Delete your account
(This will delete all your data and cannot be undone)

Delete

Password

Current password:

New password:

Re-type new password:

Forgot password?

Notifications

Study reminders
(Reminds you to review at the right "golden time" to memorize vocabulary longer)

On

Progress reports

Week

- Study Sets - detail

MemoTakara

Home

Study sets

Statistics

Search standard collection

English

cuhp

Computer virus

60 cards

@cuhp

edit

match

flashcard

writing

learn & test

Get a hint

transmission

Click to flip the card

—

1 / 3

✓

Cards in this set (3)

transmission

lây nhiễm, lan truyền

☆

soft-transmitting function

chức năng tự lây nhiễm

☆

latent function

chức năng tiềm ẩn

☆

MemoTakara

Subscribe to get information, latest news and other interesting offers about MemoTakara

enter your email

Subscribe

Support

About us

Terms and Condition

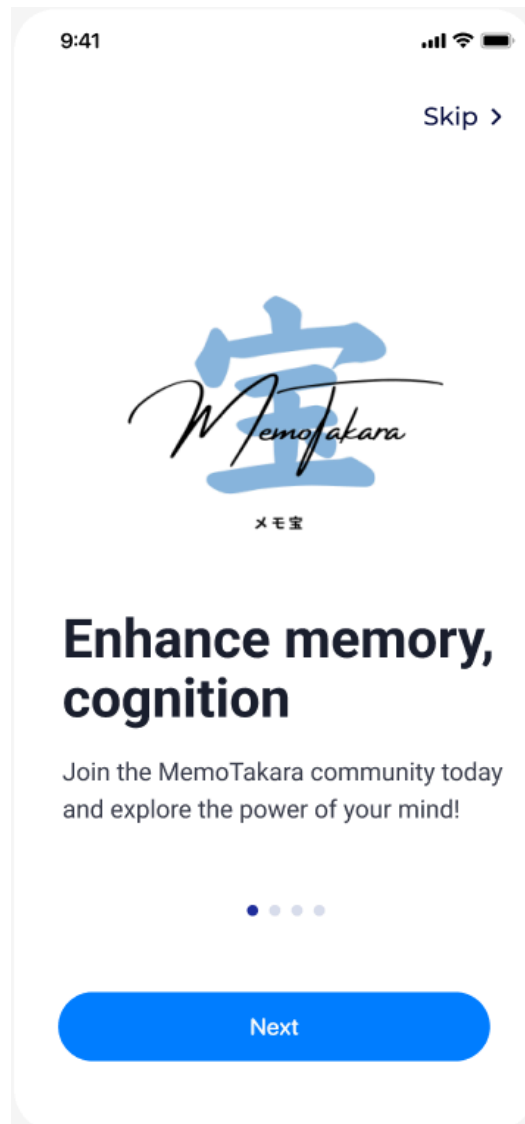
Help/FAQ

© Copyright cuhp293 2024

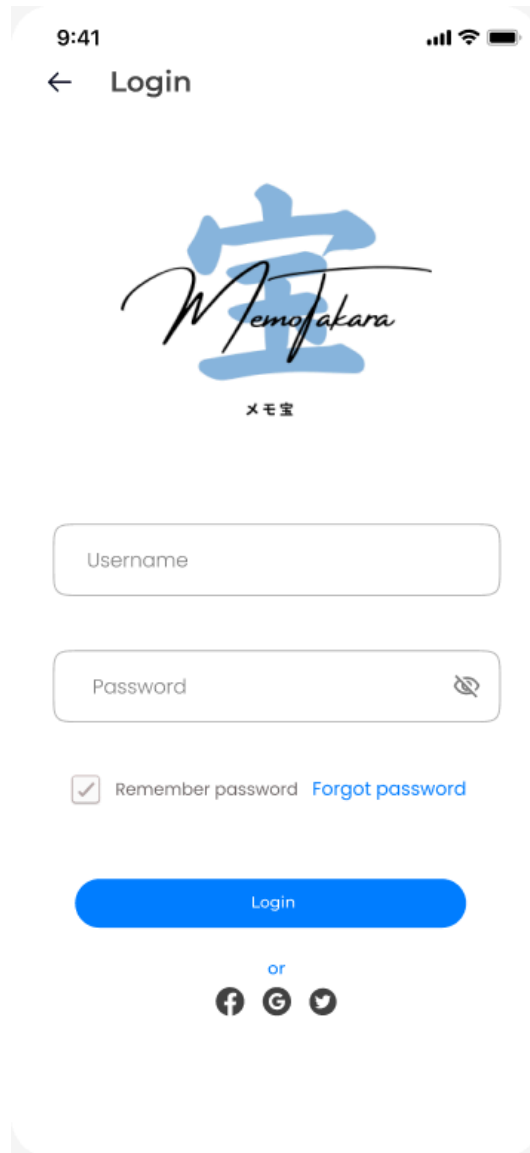
30

2. [Figma - app](#)

- Landing page




- Sign up (Login)




The image shows a mobile application interface for logging in. At the top, the status bar displays the time 9:41, signal strength, Wi-Fi, and battery icons. Below the status bar is a navigation bar with a back arrow and the text "Login". The main content area features the Memofakara logo, which consists of a blue stylized character "室" (Muro) with the word "Memofakara" written in a cursive script across it, and the text "メモ室" (Memo Muro) below. Below the logo are two input fields: "Username" and "Password". The "Password" field has a toggle icon on the right. Below the input fields is a checkbox labeled "Remember password" and a link "Forgot password". A blue "Login" button is positioned below the checkbox. Below the button is the word "or" and three social media icons: Facebook, Google+, and Twitter.

9:41

← Login


メモ室




Username

Password 

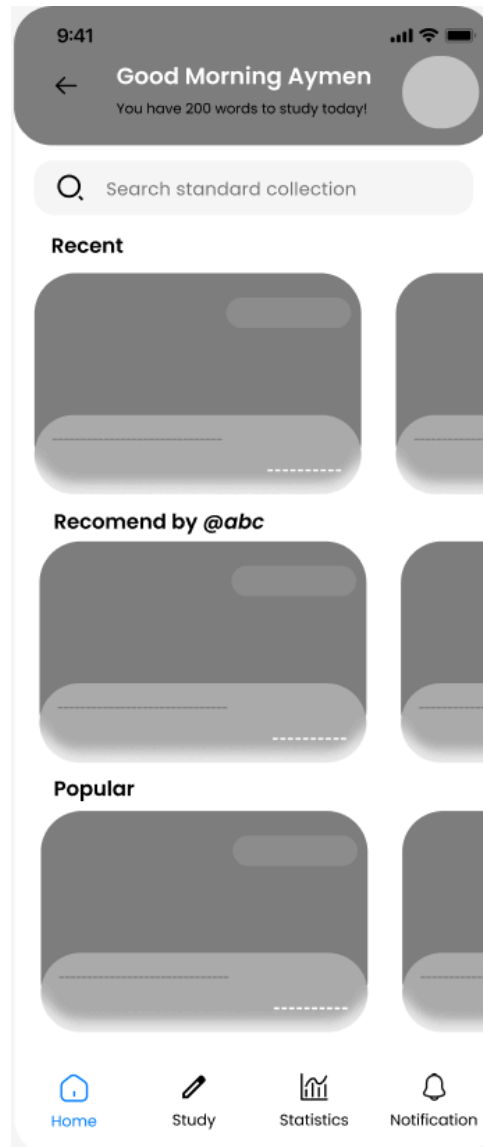
☒ Remember password [Forgot password](#)

Login

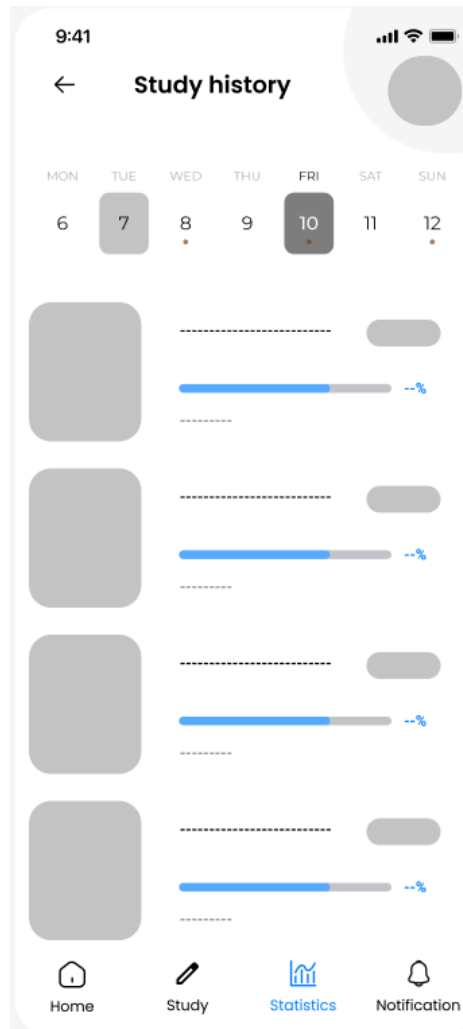
or

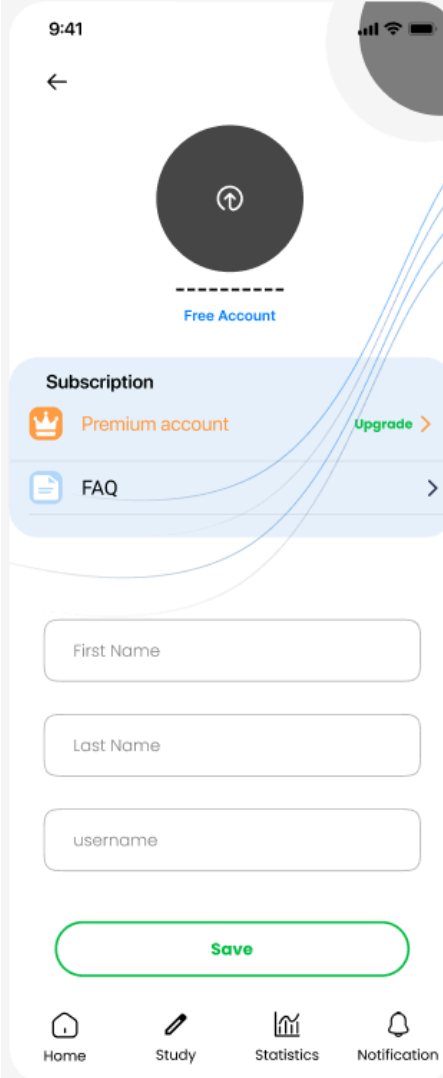
- Dashboard



- Statistics



- Settings





9:41

←

Free Account

Subscription

 Premium account [Upgrade >](#)

 FAQ >

First Name

Last Name

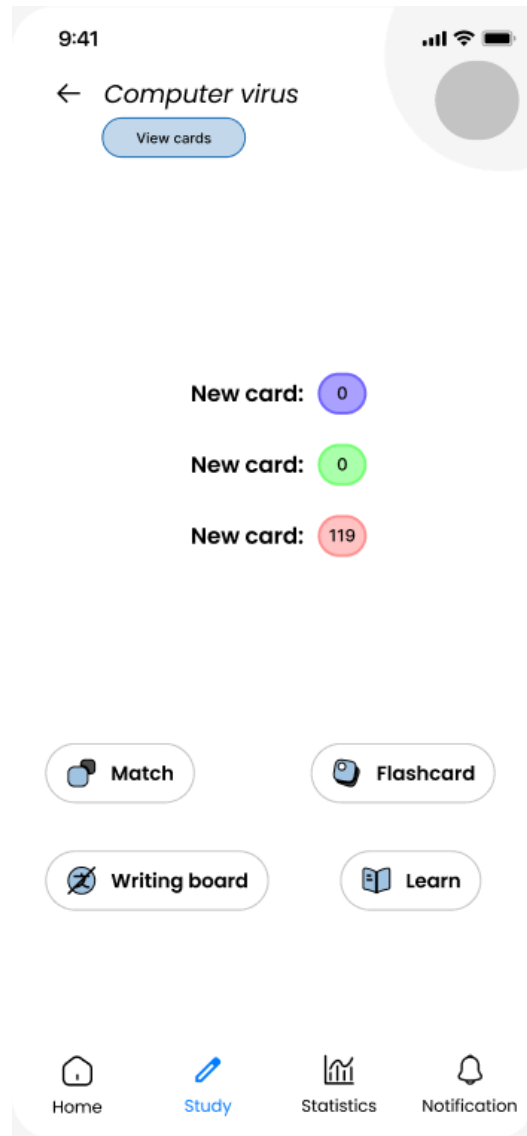
username

[Save](#)

Home Study Statistics Notification

The image shows a mobile application settings screen. At the top, the status bar displays the time 9:41 and signal/battery icons. Below the status bar is a back arrow. A circular profile picture placeholder with a refresh icon is shown, followed by the text 'Free Account'. A blue rounded rectangle contains the 'Subscription' section, which includes a 'Premium account' option with a crown icon and an 'Upgrade >' link, and an 'FAQ' option with a document icon and a right arrow. Below this are three text input fields labeled 'First Name', 'Last Name', and 'username'. A green 'Save' button is positioned below the input fields. At the bottom, a navigation bar contains four icons: a house for 'Home', a pencil for 'Study', a bar chart for 'Statistics', and a bell for 'Notification'.

- Study Sets - detail



3. Mẫu thử tương tác độ tin cậy thấp - web

VI. Khảo sát ngôn ngữ

1. Ngôn ngữ lập trình thường được sử dụng trong lập trình web¹

Ngôn ngữ lập trình (Framework)	Mô tả	Ưu điểm	Nhược điểm
Python (Django)	<ul style="list-style-type: none">- Sử dụng mô hình MVT (Model View Template) thay vì MVC (Model View Controller). Mô hình này kết hợp HTML và Django Templage Language. Điều này cho phép Django tập trung vào quản lý và xử lý dữ liệu một cách hiệu quả, đồng thời phân chia rõ ràng giữa logic của ứng dụng và giao diện người dùng.- Các template của Django bao gồm HTML tích hợp sẵn, định tuyến URL, ORM, và quản lý phiên làm việc. Với bảng quản trị độc đáo, Django cung cấp một lợi thế so với các framework khác như Laravel, đặc biệt là trong việc quản lý.	<ul style="list-style-type: none">- Chu kỳ phát triển nhanh hơn do mã ngắn gọn và dễ đọc, nguyên tắc DRY: có thể tái sử dụng code hiện có.- Đảm bảo an ninh bằng cách ngăn chặn các lỗ hổng bảo mật thường gặp như: Cross-site request forgery, SQL injection.- Cài đặt dễ dàng thông qua dòng lệnh và hỗ trợ bởi tài liệu chi tiết, cập nhật thường xuyên.- Các thư viện như numpy, pandas... thúc đẩy sự phát triển trong khoa học dữ liệu và học máy.- Django cung cấp các tính năng như quản lý đăng nhập, quyền truy cập, bảo mật, và xử lý form, giúp tăng tốc độ và đơn giản hóa	<ul style="list-style-type: none">- Django không thích hợp với các dự án nhỏ do yêu cầu lượng code nhiều.- Đặc tính monolithic của nó có thể làm phức tạp việc mở rộng và quản lý dự án.- Django gặp hạn chế trong việc xử lý đồng thời nhiều request.- Tiêu thụ bộ nhớ cao, không áp dụng cho các tác vụ đòi hỏi nhiều bộ nhớ.

¹ [13 best languages for web development in 2024](#)

		quá trình phát triển web.	
PHP (Laravel)	<ul style="list-style-type: none"> - Mô hình MVC được áp dụng để tương tác với cơ sở dữ liệu thông qua Eloquent ORM, giúp thao tác dữ liệu bằng cách sử dụng cú pháp giống như ngôn ngữ lập trình, không cần viết truy vấn SQL. - Tích hợp sẵn các chức năng bảo mật như xác thực người dùng, phân quyền, và chống lại các cuộc tấn công CSRF. - Công cụ migration và seed được cung cấp để quản lý cấu trúc và dữ liệu mẫu của cơ sở dữ liệu, với khả năng chỉnh sửa và tạo dữ liệu mẫu một cách linh hoạt, hỗ trợ quá trình phát triển và kiểm thử. Kết nối với dịch vụ thư điện tử thông qua nền tảng đám mây hoặc hệ thống cục bộ sử dụng API đơn giản từ thư viện SwiftMailer. 	<ul style="list-style-type: none"> - Đảm bảo an ninh thông tin mức độ cao, và đã tích hợp sẵn các hệ thống lưu trữ đệm như Memcached và Redis. - Laravel có cộng đồng lớn và nhiều tài nguyên hỗ trợ. - Blade template engine của Laravel mạnh mẽ và dễ sử dụng, giúp tạo giao diện người dùng nhanh chóng. - Laravel hỗ trợ Artisan - công cụ dòng lệnh giúp quản lý các tác vụ lặp lại dễ dàng. 	<ul style="list-style-type: none"> - Việc quản lý gặp trở ngại vì sự đa dạng của các phiên bản và sự không tương thích giữa chúng. - Tốc độ tải trang cũng bị ảnh hưởng do mã nguồn Laravel có phần nặng. - Laravel có thể phức tạp đối với người mới bắt đầu do số lượng tính năng và công cụ đi kèm. - Cấu hình và quản lý mối quan hệ giữa các module trong ứng dụng lớn có thể gặp nhiều khó khăn.
Java (Spring)	<ul style="list-style-type: none"> - Spring Framework được thiết kế theo các nguyên lý Dependency Injection và Aspect Oriented Programming, bao gồm nhiều module hỗ trợ mục tiêu đa dạng như kiểm thử, quản lý core, data access, integration và web. - Tương thích với nhiều công nghệ như ORM, logging, JEE và các thư viện lịch trình như Quartz và JDK Timer. - Spring MVC, một phần của Spring Framework, chủ yếu 	<ul style="list-style-type: none"> - Tích hợp trọn vẹn trong hệ sinh thái Spring: Spring Boot, Spring MVC,... - Cấu hình tự động toàn bộ các thành phần cần thiết cho một ứng dụng Spring sẵn sàng cho sản xuất. - Spring Boot cung cấp một hệ sinh thái hoàn chỉnh với khả năng cấu hình 	<ul style="list-style-type: none"> - Quản lý nhiều phiên bản và module đòi hỏi nhiều thời gian. - Việc cấu hình có thể trở nên phức tạp do việc định nghĩa bean và quản lý các mối quan hệ phụ thuộc. - Spring có thể tốn nhiều tài nguyên và đòi hỏi một lượng lớn bộ nhớ,

	<p>được sử dụng để tạo ứng dụng web theo mô hình MVC.</p> <ul style="list-style-type: none"> - Spring Boot, cũng là một phần của Spring Framework, giúp phát triển ứng dụng độc lập một cách nhanh chóng và hiệu quả với ít cấu hình hơn. Truy cập nhanh chóng vào các cơ sở dữ liệu và dịch vụ hàng đợi như MySQL, Oracle... 	<p>tự động và môi trường phát triển nhanh chóng, giúp các nhà phát triển tiết kiệm thời gian.</p> <ul style="list-style-type: none"> - Spring Security giúp bảo vệ ứng dụng web với các tính năng bảo mật tích hợp sẵn. 	<p>đặc biệt trong các ứng dụng lớn.</p> <ul style="list-style-type: none"> - Việc học và làm quen với Spring có thể mất nhiều thời gian do độ phức tạp của nó.
C# (.NET)	<ul style="list-style-type: none"> - .NET là một framework của Microsoft hỗ trợ nhiều ngôn ngữ lập trình, trong đó C# là ngôn ngữ chính. - Sử dụng mô hình MVC, .NET giúp phát triển ứng dụng mạnh mẽ và có khả năng mở rộng. - Hỗ trợ tích hợp với nhiều dịch vụ và công cụ của Microsoft như Azure, SQL Server và Active Directory. - Được sử dụng rộng rãi trong các ứng dụng doanh nghiệp và web, đặc biệt trong môi trường Windows. 	<ul style="list-style-type: none"> - Hiệu suất cao, bảo mật tốt và hỗ trợ mạnh mẽ từ Microsoft. - Công cụ phát triển và IDE như Visual Studio cung cấp nhiều tiện ích và tính năng mạnh mẽ cho lập trình viên. - .NET Core (phiên bản mã nguồn mở của .NET) hỗ trợ đa nền tảng, cho phép phát triển ứng dụng trên Windows, macOS và Linux. - .NET có thư viện phong phú và hỗ trợ tốt cho phát triển ứng dụng web, desktop và mobile. 	<ul style="list-style-type: none"> - Chi phí cao cho giấy phép và dịch vụ của Microsoft. - Khả năng tùy biến thấp so với một số framework khác. - Đôi khi các bản cập nhật và thay đổi trong .NET có thể gây ra sự không tương thích và yêu cầu điều chỉnh mã nguồn.
Ruby (Ruby on Rails - RoR)	<ul style="list-style-type: none"> - Sử dụng mô hình MVC, RoR là một framework mạnh mẽ và dễ sử dụng để phát triển ứng dụng web. - RoR nổi tiếng với nguyên tắc “Convention over Configuration” (Ưu tiên quy ước hơn cấu hình) và “Don’t 	<ul style="list-style-type: none"> - Tốc độ phát triển nhanh, cộng đồng hỗ trợ mạnh mẽ và nhiều thư viện phong phú. - Cung cấp các công cụ và tính năng giúp kiểm thử và 	<ul style="list-style-type: none"> - Hiệu suất không cao bằng các framework khác như Django hay Spring, đặc biệt là với các ứng dụng lớn. - RoR có thể trở

	<p>Repeat Yourself” (Không lặp lại chính mình), giúp tăng tốc độ phát triển và giảm thiểu lỗi.</p>	<p>triển khai dễ dàng.</p> <ul style="list-style-type: none"> - RoR có các tính năng tích hợp sẵn cho quản lý cơ sở dữ liệu, kiểm thử và bảo mật, giúp giảm bớt khối lượng công việc cho các nhà phát triển. - Rails cũng có hệ sinh thái phong phú với nhiều gem (thư viện mở rộng) giúp giải quyết hầu hết các yêu cầu phát triển. 	<p>nên chậm và khó bảo trì khi ứng dụng phát triển lớn và phức tạp.</p> <ul style="list-style-type: none"> - Rails cần một lượng lớn bộ nhớ và tài nguyên CPU, điều này có thể là vấn đề đối với các ứng dụng có lượng truy cập lớn.
Swift (iOS Development)	<ul style="list-style-type: none"> - Swift là ngôn ngữ lập trình được Apple phát triển dành cho phát triển ứng dụng iOS, macOS, watchOS và tvOS. - Swift kết hợp cú pháp hiện đại và an toàn với hiệu suất nhanh và tính linh hoạt cao, giúp phát triển ứng dụng trên các nền tảng của Apple hiệu quả và dễ dàng. 	<ul style="list-style-type: none"> - Cú pháp hiện đại, dễ học và sử dụng. - Tích hợp chặt chẽ với các công cụ và nền tảng của Apple, cung cấp hiệu suất cao và khả năng tối ưu hóa tốt. - Swift có tính an toàn cao với việc quản lý bộ nhớ tự động và kiểm soát lỗi chặt chẽ. 	<ul style="list-style-type: none"> - Chỉ sử dụng được cho các nền tảng của Apple, hạn chế khả năng phát triển ứng dụng đa nền tảng. - Cộng đồng và tài liệu chưa phong phú như một số ngôn ngữ khác.
Kotlin (Android Development)	<ul style="list-style-type: none"> - Kotlin là ngôn ngữ lập trình được Google chọn làm ngôn ngữ chính cho phát triển ứng dụng Android. - Kotlin tương thích hoàn toàn với Java và cải tiến nhiều vấn đề của Java, có thể sử dụng cùng với mã Java trong cùng một dự án. 	<ul style="list-style-type: none"> - Cú pháp ngắn gọn, an toàn và dễ bảo trì. - Tương thích hoàn toàn với Java, có thể sử dụng cùng với các thư viện và công cụ hiện có của Java. - Hỗ trợ mạnh mẽ từ Google và cộng đồng phát triển Android. - Tính an toàn cao với kiểm soát null 	<ul style="list-style-type: none"> - Dù tương thích với Java, nhưng việc chuyển đổi từ Java sang Kotlin có thể gặp một số khó khăn nhất định. - Cộng đồng và tài liệu chưa phong phú như Java.

		chặt chẽ, giảm thiểu lỗi runtime.	
Flutter	<ul style="list-style-type: none"> - Flutter là một UI toolkit mã nguồn mở, hỗ trợ phát triển giao diện đẹp, mượt mà và có tính nhất quán cao trên các nền tảng iOS, Android, web, và desktop. - Sử dụng widgets để xây dựng giao diện, mọi thứ trong Flutter đều được thiết kế dưới dạng widget. - Hoạt động dựa trên rendering engine riêng (Skia), không phụ thuộc vào native UI components. 	<ul style="list-style-type: none"> - Cross-platform: Một codebase duy nhất, chạy được trên iOS, Android, web, và desktop. - Hiệu suất cao: Sử dụng engine Skia để vẽ giao diện, mang lại trải nghiệm mượt mà tương đương ứng dụng native. - Widget phong phú: Cung cấp nhiều widgets tùy chỉnh, dễ dàng tạo giao diện đẹp và nhất quán. - Cộng đồng lớn: Tài liệu chi tiết và hỗ trợ từ cộng đồng phát triển rộng rãi. 	<ul style="list-style-type: none"> - Kích thước ứng dụng lớn: Ứng dụng Flutter thường có kích thước ban đầu khá lớn so với các ứng dụng native. - Thiếu thư viện native: Một số tính năng đặc thù của hệ điều hành cần viết plugin hoặc tích hợp với native code. - Hỗ trợ web và desktop chưa hoàn thiện: Mặc dù hỗ trợ đa nền tảng, nhưng các tính năng trên web và desktop chưa ổn định như mobile.
HTML/CSS	HTML và CSS kết hợp với nhau tạo nên giao diện và trình bày của các trang web trên Internet.	<ul style="list-style-type: none"> - Cơ bản và dễ học, là nền tảng của mọi trang web. - Được hỗ trợ rộng rãi bởi tất cả các trình duyệt web. - Cho phép tạo ra các trang web đẹp mắt và tương tác cao khi kết hợp với JavaScript. 	<ul style="list-style-type: none"> - Không có khả năng tạo logic phức tạp, cần kết hợp với các ngôn ngữ lập trình khác để tạo ra các ứng dụng web hoàn chỉnh. - CSS có thể phức tạp và khó quản lý trong các dự án lớn nếu không có cấu trúc tốt.
Javascript (ReactJS, VueJS, ExpressJS...)	<ul style="list-style-type: none"> - npm là hệ thống quản lý gói phần mềm cho JavaScript, đồng thời là kho phần mềm lớn nhất thế giới với hơn 350,000 packages. - ReactJS sử dụng model component-based, các 	<ul style="list-style-type: none"> - Thực thi trên trình duyệt, giao tiếp với DOM, và xây dựng các ứng dụng web đa chức năng, tương tác cao. 	<ul style="list-style-type: none"> - Thiếu khả năng xử lý đồng thời và đa nhiệm. - Dễ bị tấn công bằng cách chèn mã độc.

	<p>component là những thành phần UI được tái sử dụng và mỗi component có trạng thái độc lập và không cần tương tác trực tiếp với DOM nhờ vào virtual DOM, cung cấp luồng dữ liệu đơn hướng, tùy chọn cú pháp JSX, cung cấp công cụ dòng lệnh create-react-app để khởi tạo dự án React Js mà không yêu cầu cấu hình.</p> <ul style="list-style-type: none"> - VueJS là sự kết hợp giữa Angular và React, hỗ trợ animated transitions, cho phép người dùng di chuyển giữa các trang web mà không cần tải lại trang. - ExpressJS là một framework dành cho NodeJS, một môi trường chạy JavaScript phía máy chủ, hỗ trợ giao thức HTTP và middleware, giúp xây dựng API mạnh mẽ và dễ dàng định tuyến dựa trên phương thức HTTP và URL. 	<ul style="list-style-type: none"> - Có khả năng đáp ứng tức thì thông qua việc định tuyến và sử dụng virtual DOM. - Cung cấp một môi trường phát triển đầy đủ tính năng. 	<ul style="list-style-type: none"> - Có những giới hạn với một số loại trình duyệt và cần được điều chỉnh cụ thể cho từng loại thiết bị và trình duyệt để đảm bảo tính nhất quán trong hiển thị. - Có thể khó quản lý khi dự án lớn và phức tạp.
Perl	<ul style="list-style-type: none"> - Perl là ngôn ngữ lập trình bậc cao, linh hoạt và mạnh mẽ, thường được sử dụng trong quản trị hệ thống, xử lý văn bản và phát triển web. - Perl hỗ trợ cả lập trình hướng đối tượng và lập trình thủ tục, với cú pháp linh hoạt và phong phú. 	<ul style="list-style-type: none"> - Rất mạnh trong xử lý văn bản và chuỗi, có thể tích hợp dễ dàng với nhiều hệ thống và cơ sở dữ liệu. - Cộng đồng lớn và nhiều tài liệu hỗ trợ. - Có khả năng xử lý biểu thức chính quy mạnh mẽ, giúp thao tác và xử lý văn bản hiệu quả. - Có thể sử dụng cho nhiều mục đích khác nhau từ scripting, automation, đến web development và data processing. 	<ul style="list-style-type: none"> - Cú pháp phức tạp và khó đọc. - Hiệu suất không cao bằng các ngôn ngữ hiện đại hơn như Python hoặc Ruby. - Có thể trở nên khó duy trì và mở rộng khi dự án lớn và phức tạp.

Go (GoLang)	<ul style="list-style-type: none"> - Go là ngôn ngữ lập trình mã nguồn mở do Google phát triển, được thiết kế hiệu quả, dễ đọc và dễ bảo trì. - Go thường được sử dụng trong các hệ thống phân tán, dịch vụ web và ứng dụng cần hiệu suất cao và đồng thời. - Go hỗ trợ lập trình hướng đối tượng, có cú pháp đơn giản và hệ thống quản lý gói mạnh mẽ. 	<ul style="list-style-type: none"> - Hiệu suất cao, hỗ trợ đồng thời tốt với goroutines. - Cú pháp đơn giản, dễ đọc và dễ bảo trì. - Quản lý bộ nhớ tự động và có nhiều công cụ hỗ trợ phát triển. - Go có khả năng biên dịch nhanh giúp triển khai ứng dụng dễ dàng. 	<ul style="list-style-type: none"> - Thiếu nhiều thư viện và framework so với các ngôn ngữ lập trình khác. - Không hỗ trợ tính năng generics, điều này có thể làm hạn chế trong một số trường hợp. - Có thể không phù hợp cho các ứng dụng cần GUI hoặc các ứng dụng đòi hỏi tính năng phức tạp và đa nhiệm cao
-------------	--	---	--

2. Cơ sở dữ liệu thường được sử dụng²

Cơ sở dữ liệu	Mô tả	Ưu điểm	Nhược điểm
Oracle	<ul style="list-style-type: none"> - Là hệ quản trị cơ sở dữ liệu quan hệ mạnh mẽ, được sử dụng rộng rãi trong các doanh nghiệp lớn. - Hỗ trợ nhiều tính năng nâng cao như phân vùng, nhân bản dữ liệu và bảo mật cao. 	<ul style="list-style-type: none"> - Hiệu suất cao và ổn định, đặc biệt với các hệ thống lớn. - Hỗ trợ nhiều tính năng bảo mật và quản lý người dùng. - Khả năng mở rộng tốt cho các ứng dụng lớn và phức tạp. 	<ul style="list-style-type: none"> - Chi phí bản quyền cao. - Quá trình cài đặt và cấu hình phức tạp. - Yêu cầu phần cứng mạnh và có thể khó khăn trong việc quản lý. - Đòi hỏi kiến thức chuyên sâu về cơ sở dữ liệu để sử dụng hiệu quả.
MySQL	<ul style="list-style-type: none"> - Là một hệ quản trị cơ sở dữ liệu mã nguồn mở, phổ biến trong các ứng dụng web và dịch vụ trực tuyến. - Hỗ trợ cả lưu trữ theo dạng bảng (MyISAM) và giao dịch (InnoDB). 	<ul style="list-style-type: none"> - Dễ dàng cài đặt và sử dụng. Miễn phí và mã nguồn mở. - Hiệu suất tốt cho các ứng dụng web. - Hỗ trợ nhiều công cụ quản lý và giao 	<ul style="list-style-type: none"> - Hạn chế trong việc hỗ trợ các tính năng cao cấp so với các RDBMS khác như Oracle. - Hiệu suất có thể giảm khi quy mô

² [Most Popular Databases in 2024](#)

		diện người dùng. - Cộng đồng lớn và nhiều tài liệu hỗ trợ.	dữ liệu lớn. - Các tính năng bảo mật và quản lý người dùng không mạnh mẽ bằng Oracle.
Microsoft SQL Server	- Là hệ quản trị cơ sở dữ liệu quan hệ của Microsoft, được sử dụng rộng rãi trong các ứng dụng doanh nghiệp và dịch vụ trực tuyến. - Hỗ trợ tích hợp tốt với các sản phẩm khác của Microsoft.	- Tích hợp tốt với hệ sinh thái Microsoft, đặc biệt là các sản phẩm như .NET và Azure. - Hỗ trợ các công cụ quản lý và phát triển mạnh mẽ. - Bảo mật cao và khả năng phục hồi dữ liệu tốt. - Hiệu suất tốt cho các ứng dụng doanh nghiệp.	- Chi phí bản quyền cao. - Chỉ chạy trên hệ điều hành Windows, giới hạn sự linh hoạt. - Yêu cầu phần cứng mạnh và tài nguyên hệ thống lớn. - Quá trình cài đặt và cấu hình có thể phức tạp.
PostgreSQL	Là một hệ quản trị cơ sở dữ liệu mã nguồn mở, mạnh mẽ và linh hoạt, hỗ trợ nhiều kiểu dữ liệu và tính năng cao cấp như JSON, XML và các hàm người dùng tùy chỉnh.	- Miễn phí và mã nguồn mở. - Hỗ trợ nhiều tính năng nâng cao và kiểu dữ liệu đa dạng. - Tính toàn vẹn dữ liệu và bảo mật cao. - Khả năng mở rộng tốt và hiệu suất cao. - Cộng đồng lớn và nhiều tài liệu hỗ trợ.	- Quá trình cài đặt và cấu hình có thể phức tạp. - Đòi hỏi kiến thức chuyên sâu để tối ưu hóa và quản lý hiệu quả. - Hiệu suất có thể không cao bằng một số hệ quản trị cơ sở dữ liệu thương mại trong các ứng dụng yêu cầu cao về tốc độ và dung lượng.
MongoDB	- Là cơ sở dữ liệu NoSQL, sử dụng mô hình tài liệu để lưu trữ dữ liệu. - Phổ biến cho các ứng dụng yêu cầu tính linh hoạt cao trong việc lưu trữ và truy xuất dữ liệu không cấu trúc.	- Dễ dàng mở rộng và phù hợp với các hệ thống phân tán. Miễn phí và mã nguồn mở. - Hỗ trợ lưu trữ dữ liệu không cấu trúc và linh hoạt trong việc thay đổi cấu trúc dữ liệu.	- Thiếu tính toàn vẹn dữ liệu so với các RDBMS. - Không hỗ trợ tốt các giao dịch phức tạp. - Yêu cầu thiết kế cẩn thận để tránh vấn đề về hiệu suất và bảo mật.

		- Hiệu suất cao cho các truy vấn không chuẩn hóa.	- Không phù hợp cho các ứng dụng yêu cầu tính nhất quán mạnh và quản lý giao dịch phức tạp.
--	--	---	---

3. *Kết luận*

3.1. Sơ đồ thực thể liên kết ERD

Mục đích của phần mềm nhằm tạo ra hệ thống học từ vựng với các chức năng (function) mà người dùng/vai trò người dùng có thể sử dụng tại thời điểm chạy.

Trước khi đăng nhập, người dùng có thể xem các public collection và standard collection nhưng không thể tương tác hoặc tạo collection riêng. Bất kỳ ai cũng có thể đăng ký, đăng nhập để trải nghiệm đầy đủ tính năng. Sau khi đăng nhập vào hệ thống, người dùng được cấp một không gian lưu trữ cho collection cá nhân (study set), có thể tạo và quản lý collection chứa flashcards theo nhu cầu bản thân. Người dùng có thể học, thêm mới, xem, chỉnh sửa, hoặc xóa flashcard. Có thể thiết lập quyền riêng tư cho collection của mình là public hoặc private. Người dùng cũng có thể xem, tìm kiếm public collection của người khác và sử dụng standard collection của hệ thống, và clone collection đó vào không gian lưu trữ cá nhân.

Quản trị hệ thống đóng vai trò quan trọng trong việc quản lý thông báo và người dùng, đảm bảo an ninh, quyền riêng tư, và bảo trì hệ thống. Họ chịu trách nhiệm cập nhật và duy trì cơ sở hạ tầng kỹ thuật số, cũng như quản lý dữ liệu để bảo đảm nội dung luôn chất lượng và hiện đại.

3.2. Kỹ thuật sử dụng

- Phát triển ứng dụng: Flutter

Lý do: Ngôn ngữ cho phép phát triển ứng dụng trên cả 2 nền tảng Android và iOS.

- Phát triển Frontend: JavaScript (ReactJS)

Lý do:

- + ReactJS có cộng đồng lớn, nhiều tài liệu, ví dụ minh họa và thư viện hỗ trợ.
- + ReactJS sử dụng kiến trúc component, giúp tách biệt các phần giao diện thành các thành phần nhỏ. Điều này giúp quản lý mã nguồn dễ dàng hơn và tái sử dụng các component.

- Backend: Laravel/PHP

Lý do: Framework PHP phổ biến, có sẵn nhiều công cụ hỗ trợ phát triển nhanh. Bảo mật tốt và dễ tích hợp với nhiều dịch vụ khác nhau (database, cloud).

- Database: MySQL

Lý do:

- + MySQL có cộng đồng lớn và tài liệu phong phú (miễn phí, mã nguồn mở).
- + Hiệu suất cao: MySQL xử lý nhanh và ổn định, phù hợp với các ứng dụng web và hệ thống vừa và nhỏ.
- + Dễ sử dụng: Giao diện thân thiện, cú pháp đơn giản, dễ học và triển khai.
- + Tương thích tốt: Hoạt động trên nhiều nền tảng, tích hợp dễ dàng với PHP

- Quản lý code: Github

4. ***Repository Github***

- Github frontend: [MemoTakara-ReactJS](#)
- Github backend: [MemoTakara-BE](#)