

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

Web Game Mô Phỏng Trò Chơi
“Bật Hình Nhảy Số”

Người hướng dẫn: **PGS. TS. TRẦN MẠNH HUY**
Sinh viên thực hiện: **NGUYỄN HOÀNG LINH**
Mã số sinh viên: **2051220165**
Lớp: **20CT2**

Đà Nẵng,/2024

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

Web Game Mô Phỏng Trò Chơi
“Bật Hình Nhảy Số”

Người hướng dẫn: PGS. TS. TRẦN MẠNH HUY
Sinh viên thực hiện: NGUYỄN HOÀNG LINH
Mã số sinh viên: 2051220165
Lớp: 20CT2

Đà Nẵng,/2024

{Trang trắng này dùng để dán bản Nhận xét của người hướng dẫn, hoặc thay trang này bằng Nhận xét của người hướng dẫn}

{Trang trắng này dùng để dán bản Nhận xét của người phản biện, hoặc thay trang này bằng Nhận xét của người phản biện}

TÓM TẮT

Tên đề tài: Web Game Mô Phỏng Trò Chơi “Bật Hình Nhảy Số”

Tên đề tài: Web Game Mô Phỏng Trò Chơi “Bật Hình Nhảy Số”

Sinh viên thực hiện: Nguyễn Hoàng Linh

Mã số SV: 2051220165 Lớp: 20CT2

Nội dung báo cáo gồm có:

Chương 1: Tổng quan về đề tài web game mô phỏng trò chơi “bật hình nhảy số”: Chương này trình bày về bối cảnh, lý do chọn đề tài và nội dung xây dựng và ý nghĩa thực tiễn của hệ thống web game mô phỏng trò chơi bật hình nhảy số.

Chương 2: Cơ sở lý thuyết: Giới thiệu các nền tảng lý thuyết như lập trình web, game mô phỏng, xử lý file .zip, kiểm tra đáp án, và điều khiển thời gian thực

Chương 3: Phân tích hệ thống: Chương này Mô tả chi tiết các yêu cầu chức năng, phi chức năng, thiết kế hệ thống, cơ sở dữ liệu, và giao diện người dùng, đồng thời định hình luồng tương tác và quy trình vận hành hệ thống..

Chương 4: Cài đặt, triển khai và đánh giá: Trình bày môi trường phát triển, các bước triển khai, kiểm thử, và phát hành sản phẩm. Đánh giá kết quả đạt được và đề xuất hướng phát triển trong tương lai..

LỜI NÓI ĐẦU

Trong quá trình hoàn thành đồ án tốt nghiệp với đề tài "Web game mô phỏng trò chơi bật hình nhảy số," em đã may mắn nhận được rất nhiều sự giúp đỡ và động viên từ nhiều phía. Em xin chân thành gửi lời cảm ơn đến tất cả những ai đã đồng hành và ủng hộ em trong hành trình này.

Trước hết, em xin bày tỏ lòng biết ơn sâu sắc đến PGS. TS Trần Mạnh Huy, người đã tận tình hướng dẫn và chỉ bảo em trong suốt thời gian thực hiện đồ án. Kiến thức sâu rộng, sự chỉ dẫn cẩn thận, và những nhận xét quý báu của thầy đã giúp em có được định hướng rõ ràng và tự tin hơn trong quá trình nghiên cứu và phát triển hệ thống. Em rất cảm kích sự sẵn lòng của thầy trong việc hỗ trợ và truyền cảm hứng để em có thể đạt được kết quả tốt nhất.

Em cũng xin gửi lời cảm ơn chân thành đến Khoa Công nghệ Thông tin, Trường Đại học Kiến trúc Đà Nẵng, đã tạo điều kiện học tập và nghiên cứu thuận lợi cho em. Nhờ vào môi trường học tập đầy đủ và sự hướng dẫn của các thầy cô trong khoa, em đã có thể trau dồi kiến thức và kỹ năng cần thiết cho quá trình nghiên cứu của mình.

Cuối cùng, em muốn dành những lời cảm ơn sâu sắc nhất đến gia đình và bạn bè, những người đã luôn kề vai sát cánh, động viên và cổ vũ tinh thần em trong suốt quá trình học tập. Tình yêu và sự hy sinh vô điều kiện của gia đình là nguồn động lực to lớn giúp em vượt qua mọi khó khăn và thử thách để hoàn thành tốt đồ án này.

Em hy vọng rằng hệ thống mà em đã xây dựng sẽ đóng góp phần nào vào việc phát triển các ứng dụng mô phỏng trong giáo dục và giải trí. Đồng thời, em mong rằng những kiến thức và kinh nghiệm có được từ đồ án này sẽ là nền tảng vững chắc cho bước đường tiếp theo trong sự nghiệp của em.

Em xin trân trọng cảm ơn!

CAM ĐOAN

Tôi xin cam đoan rằng đồ án tốt nghiệp với đề tài "Web game mô phỏng trò chơi bật hình nhảy số" là kết quả nghiên cứu và làm việc nghiêm túc của bản thân tôi. Toàn bộ nội dung và các kết quả được trình bày trong đồ án là do tôi thực hiện dưới sự hướng dẫn của PGS. TS Trần Mạnh Huy.

Tôi khẳng định rằng, ngoài những tài liệu tham khảo được trích dẫn trong đồ án, tôi không sao chép hoặc sử dụng nội dung của bất kỳ công trình nào khác. Nếu có bất kỳ vi phạm nào liên quan đến vấn đề bản quyền và sở hữu trí tuệ, tôi xin hoàn toàn chịu trách nhiệm trước nhà trường và pháp luật.

Tôi xin chân thành cảm ơn!

Sinh viên thực hiện

{Chữ ký, họ và tên sinh viên}

MỤC LỤC

TÓM TẮT.....	i
LỜI NÓI ĐẦU	i
CAM ĐOAN	ii
MỤC LỤC	iii
DANH SÁCH CÁC B	x
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT	xi
MỞ ĐẦU	1
Chương 1: TỔNG QUAN VỀ ĐỀ TÀI WEB GAME MÔ PHỎNG TRÒ CHƠI “BẬT HÌNH NHẢY SỐ”	3
1.1 Bối cảnh và lý do chọn đề tài	3
1.2 Nội dung đề tài	3
1.3 Lựa chọn công nghệ.....	4
Chương 2: CƠ SỞ LÝ THUYẾT	5
2.1. Giới thiệu framework NextJS	5
2.1.1. NextJS là gì?	5
2.1.2. Lợi thế của NextJS.....	5
2.1.3. Nhược điểm của NextJS	6
2.2. Hệ quản trị cơ sở dữ liệu PostgreSQL	6
2.2.1 Khái niệm	6
2.2.2 Tính năng PostgreSQL.....	7
2.2.3 Ưu điểm và nhược điểm của PostgreSQL	7
2.2.4 Vì sao nên sử dụng PostgreSQL thay vì MySQL?	8
2.3. Ngôn ngữ Python	9
2.4. Django và Django REST Framework.....	12
2.5. Mô hình Client-Server	13

2.5.1. Nguyên tắc hoạt động	14
2.5.2. Ưu điểm	15
2.5.3. Nhược điểm	15
2.6. Công cụ phát triển Visual Studio Code	16
2.6.1. Visual Studio Code là gì	16
2.6.1. Tại sao nên sử dụng Visual Studio Code?	16
Chương 3: PHÂN TÍCH, THIẾT KẾ HỆ THỐNG	18
3.1 Mô tả bài toán	18
3.1.1 Yêu cầu của người chơi	19
3.1.2 Yêu cầu admin.....	21
3.2 Phân tích hệ thống	22
3.2.1 chức năng đăng nhập của người chơi	22
3.2.2. Chức năng đăng ký của người chơi.....	23
3.2.3 Chức năng cập nhập thông tin tài khoản:	25
3.2.4 chức năng quản lý các chủ đề cá nhân.....	26
3.2.5 Chức năng tham gia phòng chơi.....	28
3.2.6 Chức năng tạo phòng chơi	29
3.2.7 Chức năng chơi ở chế độ 1vs1.....	31
3.2.8 Chức năng chơi ở chế độ có giám khảo	34
3.2.9 Chức năng bảng xếp hạng.....	37
3.2.10 Chức năng quản lý user ở CMS	38
3.2.11 Chức năng tạo chủ đề hệ thống	40
3.3. Thiết kế cơ sở dữ liệu.....	42
3.3.1. Mô hình dữ liệu quan hệ	42
3.3.2 Danh mục các bảng cơ sở dữ liệu	42
3.4. Thiết kế giao diện	45
3.4.1. Giao diện đăng nhập.....	45

3.4.2. Giao diện đăng ký	45
3.4.3. Giao diện Trang chủ	46
3.4.4. Giao diện setting user	46
3.4.5. Giao diện tạo topic	47
3.4.6. Giao diện detail topic	47
3.4.7. Giao diện tạo phòng	48
3.4.8. Giao diện phòng chờ	49
3.4.9. Giao diện player ở chế độ 1vs1	50
3.4.10. Giao diện giám khảo	50
3.4.11. Giao diện player ở chế độ có giám khảo	51
3.4.12. Giao diện kết thúc game đấu	51
3.4.13. Giao diện bảng xếp hạng	52
3.4.11 Giao diện cms-topics	52
3.4.12 Giao diện cms-user	53
Chương 4: CÀI ĐẶT, TRIỂN KHAI PHÁT HÀNH	54
4.1 Cài đặt và triển khai hệ thống	54
4.2 Áp dụng socket vào quá trình vận hành trò chơi	55
4.2.1 Connect user	55
4.2.2. Start game	55
4.2.3. Tương tác player với nhau	56
4.2.4. Quản lý câu hỏi	57
4.2.5. Quản lý thời gian	58
4.2.6. Quản lý disconnection	59
4.3 Triển khai phát hành sản phẩm	59
4.3.1. Triển khai sản phẩm lên server	59
4.3.2. Treo quảng cáo và tìm kiếm người dùng tiềm năng	60
4.2.3. Tích hợp Google Analytics	60

3.2.4. Đánh giá và tối ưu hóa.....	60
KẾT LUẬN.....	62

DANH SÁCH HÌNH ẢNH

Hình 2.1 NextJS.....	5
Hình 2.2 PostgreSQL.....	6
Hình 2.3 Mô tả cơ sở dữ liệu PostgreSQL	8
Hình 2.4 Python logo.....	10
Hình 2.5 Truy cập liên kết.....	11
Hình 2.6 Tiến hành cài đặt	11
Hình 2.7 Mở Command Prompt	11
Hình 2.8 Mô hình Client Server	14
Hình 2.9 Client Server	15
Hình 2.10 Visual Studio Code.....	16
Hình 3.1 Use Case chức năng đăng nhập	22
Hình 3.2 Sơ đồ tuần tự chức năng đăng nhập	23
Hình 3.3 Use Case chức năng đăng ký.....	23
Hình 3.4 Biểu đồ tuần tự đăng ký của người chơi.....	24
Hình 3.5 Use Case chức năng cập nhập thông tin.....	25
Hình 3.6 Biểu đồ tuần tự chức năng cập nhập thông tin tài khoản	26
Hình 3.7 Use Case chức năng quản lý chủ đề cá nhân.....	26
Hình 3.8 Biểu đồ tuần tự chức năng quản lý chủ đề cá nhân	27
Hình 3.9 Use Case chức năng tham gia phòng.....	28
Hình 3.10 Biểu đồ tuần tự chức năng tham gia phòng chơi.....	29
Hình 3.11 Use Case chức năng tạo phòng chơi.....	29
Hình 3.12 Biểu đồ tuần tự chức năng tạo phòng chơi.....	31
Hình 3.13 Use Case chức năng chơi ở chế độ 1vs1	31
Hình 3.14 Biểu đồ hoạt động chức năng chơi ở chế độ 1 vs 1.....	33
Hình 3.15 Use Case chức năng chơi ở chế độ có giám khảo	34

Hình 3.16 Biểu đồ hoạt động chức năng chơi ở chế độ có giám khảo.....	36
Hình 3.17 Use Case chức năng bảng xếp hạng	37
Hình 3.18 Biểu đồ hoạt động chức năng bảng xếp hạng.....	38
Hình 3.19 Use Case chức năng quản lý user	38
Hình 3.20 Biểu đồ hoạt động chức năng quản lý user.....	39
Hình 3.21 Use Case chức năng tạo chủ đề hệ thống	40
Hình 3.22 Biểu đồ hoạt động chức năng tạo chủ đề hệ thống.....	41
Hình 3.23 Mô hình dữ liệu quan hệ.....	42
Hình 3.24 Giao diện đăng nhập.....	45
Hình 3.25 Giao diện đăng ký.....	45
Hình 3.26 Giao diện Trang chủ	46
Hình 3.27 Giao diện setting user	46
Hình 3.28 Giao diện tạo topic.....	47
Hình 3.29 Giao diện chi tiết topic	47
Hình 3.30 Giao diện tạo phòng.....	48
Hình 3.31 Giao diện chọn chủ đề	48
Hình 3.32 Giao diện phòng chờ chế độ 1vs 1	49
Hình 3.33 Giao diện phòng chờ chế độ có giám khảo	49
Hình 3.34 Giao diện chơi ở chế độ 1vs 1	50
Hình 3.35 Giao diện của giám khảo	50
Hình 3.36 Giao diện chơi ở chế độ có giám khảo	51
Hình 3.37 Giao diện kết thúc ván đấu	51
Hình 3.38 Giao diện bảng xếp hạng	52
Hình 3.39 Giao diện CMS-topic.....	52
Hình 3.40 Giao diện CMS-user	53
Hình 4.1 Hàm connect user	55
Hình 4.2 Hàm bắt đầu game đấu	55

Hình 4.3 Hàm quản lý 2 player tương tác	56
Hình 4.4 Hàm quản lý chuyển ảnh	57
Hình 4.5 Hàm quản lý thời gian chơi	58
Hình 4.6 Hàm quản lý thời gian chơi	59

DANH SÁCH CÁC BẢNG

Bảng 2.1 Vì sao nên sử dụng PostgreSQL thay vì MySQL?	9
Bảng 2.2 Django và Django REST Framework	13
Bảng 3.1 Mô tả Use Case đăng nhập.....	23
Bảng 3.2 Mô tả Use Case đăng ký.....	24
Bảng 3.3 Mô tả Use Case cập nhật tài khoản	25
Bảng 3.4 Mô tả Use Case quản lý đề cá nhân	27
Bảng 3.5 Mô tả Use Case tham gia phòng chơi	28
Bảng 3.6 Mô tả Use Case tạo phòng chơi	30
Bảng 3.7 Mô tả Use Case chức năng chơi ở chế độ 1vs1	32
Bảng 3.8 Mô tả Use Case chức năng chơi ở chế độ có giám khảo	35
Bảng 3.9 Mô tả Use Case chức năng bảng xếp hạng	37
Bảng 3.10 Mô tả Use Case chức năng quản lý user	39
Bảng 3.11 Mô tả Use Case chức năng tạo chủ đề hệ thống	40
Bảng 3.12 Mô tả các bảng cơ sở dữ liệu	42
Bảng 3.13 Mô tả bảng User	43
Bảng 3.14 Mô tả bảng Topic	43
Bảng 3.15 Mô tả bảng question.....	43
Bảng 3.16 Mô tả bảng room.....	44
Bảng 3.17 Mô tả bảng ranking	44

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Ký hiệu	Mô tả
API	Application Programming Interface
ES6	ECMA Script 6
SEO	Search Engine Optimization
SSR	Server-side Rendering

MỞ ĐẦU

1. Mục đích thực hiện đề tài

Đề tài “Web game mô phỏng trò chơi bật hình nhảy số” nhằm xây dựng một hệ thống giải trí trực tuyến, giúp người dùng rèn luyện trí nhớ và phản xạ qua các trò chơi tương tác. Hệ thống cung cấp các tính năng như phòng chờ, nhiều chế độ chơi, và hỗ trợ chủ đề cá nhân để tăng trải nghiệm người dùng.

2. Mục tiêu đề tài

Mục tiêu tổng quát

Mục tiêu chính của đề tài là phát triển một hệ thống quản lý trò chơi dựa trên web, cho phép người dùng tham gia và trải nghiệm các trò chơi theo nhiều chế độ khác nhau như 1vs1 và có giám khảo, đồng thời cung cấp các chức năng quản lý dữ liệu người chơi và hệ thống một cách hiệu quả. Chức năng:

Mục tiêu cụ thể

- Xây dựng chức năng quản lý tài khoản người dùng: Cho phép đăng ký, đăng nhập, chỉnh sửa thông tin cá nhân, và quản lý các chủ đề cá nhân.
- Phát triển hệ thống phòng chơi: Hỗ trợ người dùng tạo phòng chơi mới hoặc tham gia vào phòng chơi sẵn có với các thiết lập linh hoạt.
- Thiết kế các chế độ chơi: Cung cấp các chế độ chơi như 1vs1 và có giám khảo để người dùng có thể trải nghiệm những thử thách khác nhau.
- Hệ thống xếp hạng người chơi: Sử dụng điểm ELO để đánh giá và xếp hạng người chơi trong hệ thống.
- Quản lý nội dung hệ thống: Tạo và quản lý các chủ đề câu hỏi từ kho dữ liệu hệ thống nhằm đa dạng hóa nội dung trò chơi.

3. Phạm vi và đối tượng nghiên cứu

Phạm vi nghiên cứu tập trung vào phát triển một hệ thống web game với các tính năng cốt lõi phục vụ mục đích giải trí và luyện tập trí nhớ. Đối tượng nghiên cứu bao gồm công nghệ web, các thuật toán xử lý hình ảnh và dữ liệu thời gian thực, cũng như các công cụ quản lý dữ liệu người dùng và nội dung chủ đề.

4. Phương pháp nghiên cứu

Phương pháp nghiên cứu trong đề tài bao gồm:

- Phương pháp nghiên cứu lý thuyết và thực nghiệm, bao gồm: khảo sát các công nghệ phát triển web và game mô phỏng, thiết kế và xây dựng mô hình hệ thống, kiểm thử và đánh giá hiệu quả.
- Phương pháp phân tích yêu cầu và thiết kế mô hình dữ liệu cũng được áp dụng để đảm bảo hệ thống hoạt động ổn định và có thể mở rộng.

5. Cấu trúc của đồ án tốt nghiệp

Đồ án tốt nghiệp được cấu thành từ các chương sau:

Chương 1: Tổng quan về đề tài web game mô phỏng trò chơi “bật hình nhảy số”:

Trình bày tổng quan về đề tài, mục tiêu, và ý nghĩa thực tiễn của hệ thống web game mô phỏng trò chơi bật hình nhảy số.

Chương 2: Cơ sở lý thuyết: Giới thiệu các nền tảng lý thuyết như lập trình web, game mô phỏng, xử lý file .zip, kiểm tra đáp án, và điều khiển thời gian thực.

Chương 3: Phân tích và thiết kế hệ thống: Mô tả chi tiết các yêu cầu chức năng, phi chức năng, thiết kế hệ thống, cơ sở dữ liệu, và giao diện người dùng, đồng thời định hình luồng tương tác và quy trình vận hành hệ thống.

Chương 4: Thiết kế hệ thống: Trình bày môi trường phát triển, các bước triển khai, kiểm thử, và phát hành sản phẩm. Đánh giá kết quả đạt được và đề xuất hướng phát triển trong tương lai.

Chương 1: TỔNG QUAN VỀ ĐỀ TÀI WEB GAME MÔ PHỎNG TRÒ CHƠI “BẬT HÌNH NHẢY SỐ”

1.1 Bối cảnh và lý do chọn đề tài

Trong thời đại số hóa, nhu cầu giải trí trực tuyến đang tăng mạnh, đặc biệt với các trò chơi web có khả năng giúp người dùng rèn luyện trí nhớ và phản xạ nhanh. Các trò chơi như nhận diện hình ảnh trong thời gian ngắn thu hút đông đảo người chơi vì vừa mang tính giải trí, vừa hỗ trợ phát triển kỹ năng tư duy.

Đề tài “Web game mô phỏng trò chơi bật hình nhảy số” được chọn với mục tiêu tạo ra một trò chơi tương tác cao, giúp người chơi rèn luyện trí nhớ trong môi trường trực tuyến. Đồng thời, đây là cơ hội để người thực hiện áp dụng kiến thức về lập trình web, thiết kế hệ thống, và học hỏi kỹ năng xử lý dữ liệu đa người dùng, phù hợp với các yêu cầu trong ngành công nghệ thông tin.

1.2 Nội dung đề tài

Xây dựng một web game mô phỏng, bao gồm các tiêu chí và chức năng

Tiêu chí:

- Tính tương tác cao: Trò chơi cần mang lại trải nghiệm mượt mà, phản hồi nhanh để tăng tính hấp dẫn.
- Giao diện thân thiện: Giao diện đơn giản, dễ hiểu và trực quan để người chơi dễ dàng làm quen và thao tác.
- Tính ổn định và hiệu suất cao: Đảm bảo hệ thống có thể xử lý nhiều người chơi đồng thời và hoạt động ổn định trong thời gian dài.
- Tính năng động và linh hoạt: Cho phép người chơi cập nhật và tùy chỉnh tài khoản, chủ đề chơi, cài đặt âm thanh và thiết bị đầu vào.
- Tính mở rộng: Hệ thống phải có khả năng mở rộng để bổ sung thêm các tính năng và nội dung trong tương lai.

Chức năng:

Chức năng người dùng

- Quản lý tài khoản: Đăng ký, đăng nhập, cập nhật thông tin cá nhân (tên, avatar, âm thanh nền).

- Quản lý chủ đề cá nhân: Tạo và tải lên chủ đề riêng bằng file .zip chứa hình ảnh và tên chủ đề đồng thời có thể cập nhập thông tin từng nội dung trong chủ đề.
- Tạo và tham gia phòng chơi: Tạo phòng với ID riêng, chọn chủ đề, chế độ chơi và thời gian.
- Chế độ chơi 1vs1 và có giám khảo
- Tính toán elo người chơi và đưa ra xếp hạng người chơi trong toàn hệ thống
- Chức năng quản trị (CMS)
- Quản lý người dùng: Thêm, xóa, cập nhật tài khoản người dùng.
- Quản lý chủ đề: Tạo và quản lý chủ đề cho toàn hệ thống.
- Thống kê: Theo dõi số liệu người chơi và hiệu quả hoạt động.

1.3 Lựa chọn công nghệ

Sử dụng PostgreSQL làm cơ sở dữ liệu cho hệ thống.

Tìm hiểu, nghiên cứu và áp dụng NextJS framework cũng như HTML, CSS trong quá trình xây dựng giao diện cho đề tài.

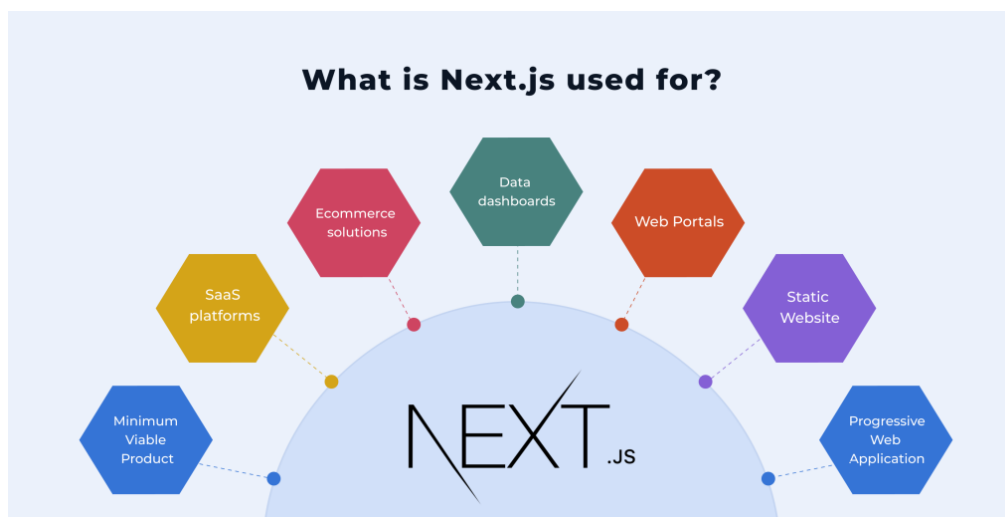
Sử dụng Django, Django REST framework, JWT (JSON Web Token) để xây dựng chức năng phía server.

Chương 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu framework NextJS

2.1.1. NextJS là gì?

Next.js là một framework front-end React được phát triển dưới dạng open-source bổ sung các khả năng tối ưu hóa như render phía máy chủ (SSR) và tạo trang web static. Next.js xây dựng dựa trên thư viện React, có nghĩa là các ứng dụng Next.js sử dụng core của React và chỉ thêm các tính năng bổ sung. Việc triển khai ứng dụng SSR cho phép máy chủ truy cập tất cả dữ liệu được yêu cầu và xử lý JavaScript cùng nhau để hiển thị trang. Sau đó, trang được gửi lại toàn bộ cho trình duyệt và ngay lập tức được hiển thị. SSR cho phép các trang web load trong thời gian nhỏ nhất và tăng trải nghiệm người dùng với khả năng phản hồi nhanh hơn.



Hình 2.1 NextJS

2.1.2. Lợi thế của NextJS

- SEO website rất tốt và giúp giảm chi phí, giúp trang web của bạn hiển thị cao hơn trên các trang kết quả của công cụ tìm kiếm. SSR làm cho các trang web xếp hạng tốt hơn cho SEO vì chúng tải nhanh hơn và nhiều nội dung trang web có thể được quét bởi các trình theo dõi SEO
- Next.js cũng cho phép chỉnh sửa thẻ <head> của một trang web, điều mà không thể thực hiện trong React. Các tag <head> là một phần cốt lõi của siêu dữ liệu trên trang web và góp phần vào đánh giá xếp hạng SEO của trang web.
- Tích hợp bộ nhớ đệm và tối ưu hóa tĩnh tự động

- Server phụ trách hiển thị các trang
- Framework của React vì thế nếu làm tốt với React thì có thể code Next.js dễ hơn
- Tinh chỉnh cấu hình webpack / babel
- Hỗ trợ tích hợp cho Route Page, CSS, JSX và TypeScript
- Nhanh chóng thêm các plugin để tùy chỉnh Next.js theo nhu cầu của trang cụ thể.

2.1.3. Nhược điểm của NextJS

Nhược điểm thực sự duy nhất của NextJS là nó là một opinionated framework, có nghĩa là nó có một phương pháp và bộ công cụ cụ thể mà nó muốn bạn sử dụng để xây dựng các ứng dụng của mình

2.2. Hệ quản trị cơ sở dữ liệu PostgreSQL

2.2.1 Khái niệm

PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ và đối tượng (object-relational database management system) miễn phí và nguồn mở (RDBMS) tiên tiến nhất hiện nay. khả năng mở rộng cao và tuân thủ các tiêu chuẩn kỹ thuật. Nó được thiết kế để xử lý một loạt các khối lượng công việc lớn, từ các máy tính cá nhân đến kho dữ liệu hoặc dịch vụ Web có nhiều người dùng đồng thời.



Hình 2. 2 PostgreSQL

- PostgreSQL được phát triển bởi PostgreSQL Global Development Group, Phát hành lần đầu: 08/07/1996
- PostgreSQL linh động có thể chạy được trên nhiều nền tảng khác nhau như Mac OS X, Solaris và Windows.
- PostgreSQL là một phần mềm mã nguồn mở miễn phí bởi vậy PostgreSQL có thể được dùng, sửa đổi và phổ biến bởi bất kỳ ai cho bất kỳ mục đích nào.
- PostgreSQL có tính ổn định cao.
- PostgreSQL là hệ thống quản lý cơ sở dữ liệu đầu tiên triển khai tính năng kiểm soát đồng thời nhiều phiên bản (MVCC).

2.2.2 Tính năng PostgreSQL

- Hỗ trợ developer xây dựng ứng dụng.
- Cho phép quản trị viên xây dựng môi trường chống lỗi bằng cách bảo vệ tính toàn vẹn của dữ liệu.
- Tương thích với nhiều nền tảng sử dụng tất cả các ngôn ngữ và middleware chính.
- Cung cấp cơ chế khóa phức tạp nhất.
- Hỗ trợ kiểm soát đồng thời nhiều phiên bản.
- Chức năng lập trình ở phía máy chủ đã trải qua quá trình phát triển.
- Tuân theo tiêu chuẩn ANSI SQL.
- Hỗ trợ đầy đủ kiến trúc mạng client-server..
- Đồng bộ dữ liệu dựa trên log và trigger với SSL.
- Máy chủ đứng và khả năng sẵn có cao.
- Hỗ trợ hướng đối tượng và tương thích với ANSI-SQL2008.
- Hỗ trợ JSON cho phép liên kết với các kho dữ liệu khác như NoSQL, hoạt động như một trung tâm liên kết cho các cơ sở dữ liệu đa ngôn ngữ.

2.2.3 Ưu điểm và nhược điểm của PostgreSQL

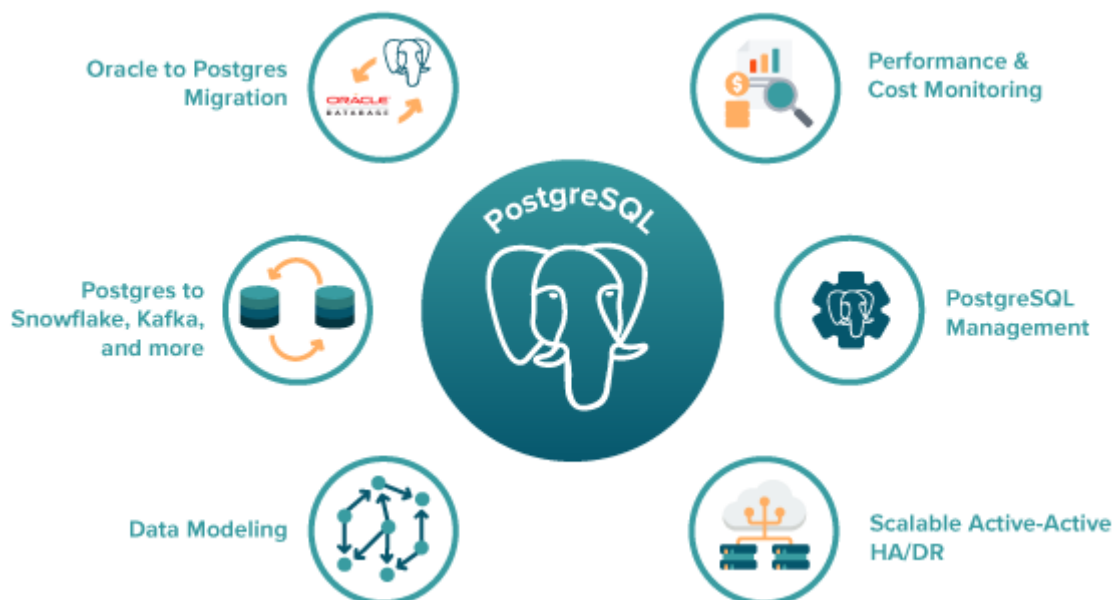
Ưu điểm:

- Chạy trên các ứng dụng web: PostgreSQL có thể được sử dụng để chạy các trang web động và ứng dụng web như một phần của bộ công cụ LAMP.
- Chức năng ghi nhật ký trước: PostgreSQL có khả năng chịu lỗi cao nhờ chức năng ghi nhật ký trước, giúp khôi phục dữ liệu hiệu quả.
- Mã nguồn mở: Là phần mềm miễn phí với mã nguồn mở, PostgreSQL cho phép người dùng tùy chỉnh theo nhu cầu kinh doanh.
- Hỗ trợ đối tượng địa lý: PostgreSQL có khả năng hỗ trợ dữ liệu địa lý, phù hợp với các dịch vụ dựa trên vị trí và hệ thống thông tin địa lý (GIS).
- Dễ sử dụng: Người dùng không cần quá nhiều đào tạo để sử dụng PostgreSQL vì nó có giao diện thân thiện.
- Bảo trì thấp: Quản lý và bảo trì PostgreSQL dễ dàng, phù hợp cho cả doanh nghiệp và các ứng dụng nhúng

Nhược điểm:

- Thiếu hỗ trợ từ tổ chức lớn: PostgreSQL không thuộc sở hữu của một tổ chức lớn, làm giảm sự uy tín trong cộng đồng, dù nó có nhiều tính năng mạnh mẽ.

- Cải thiện tốc độ tốn công: Việc tối ưu hóa tốc độ trong PostgreSQL cần nhiều công việc hơn so với MySQL, do PostgreSQL ưu tiên tính tương thích hơn.
- Hỗ trợ từ các ứng dụng nguồn mở hạn chế: Nhiều ứng dụng mã nguồn mở hỗ trợ MySQL nhưng không hỗ trợ PostgreSQL.
- Hiệu suất thấp hơn MySQL: Theo các chỉ số hiệu suất, PostgreSQL có thể chậm hơn MySQL trong một số trường hợp



Hình 2.3 Mô tả cơ sở dữ liệu PostgreSQL

2.2.4 Vì sao nên sử dụng PostgreSQL thay vì MySQL?

Danh mục	MySQL	PostgreSQL
Công nghệ cơ sở dữ liệu	MySQL đơn thuần là một hệ thống quản lý cơ sở dữ liệu quan hệ.	PostgreSQL là một hệ thống quản lý cơ sở dữ liệu quan hệ đối tượng.
Tính năng	MySQL hỗ trợ số ít các tính năng cơ sở dữ liệu như chế độ xem, điều kiện kích hoạt và quy trình.	PostgreSQL hỗ trợ hầu hết các tính năng cơ sở dữ liệu nâng cao như chế độ xem cụ thể hóa, điều kiện kích hoạt INSTEAD OF và quy trình được lưu trữ bằng nhiều ngôn ngữ.
Loại dữ liệu	MySQL hỗ trợ các loại dữ liệu số, ký tự, ngày và giờ, không gian và JSON.	PostgreSQL hỗ trợ tất cả các loại dữ liệu MySQL cùng với các loại dữ liệu hình học, liệt kê, địa

		chỉ mạng, mảng, phạm vi, XML, hstore và kết hợp.
Tuân thủ ACID	MySQL chỉ tuân thủ ACID với công cụ lưu trữ InnoDB và NDB Cluster.	PostgreSQL luôn tuân thủ ACID.
Chỉ mục	MySQL hỗ trợ chỉ mục B-tree và R-tree.	PostgreSQL hỗ trợ nhiều loại chỉ mục như chỉ mục biểu thức, chỉ mục một phần và chỉ mục băm cùng với dạng cây.
Hiệu năng	MySQL có hiệu năng cao hơn đối với các thao tác đọc thường xuyên.	PostgreSQL có hiệu năng cao hơn đối với các thao tác ghi thường xuyên
Hỗ trợ người mới bắt đầu	MySQL dễ hơn khi bắt đầu sử dụng. MySQL có một bộ công cụ đa dạng hơn cho người dùng không chuyên về kỹ thuật.	PostgreSQL phức tạp hơn khi bắt đầu sử dụng. PostgreSQL có một bộ công cụ hạn chế cho người dùng không chuyên về kỹ thuật.

Bảng 2.1 Vì sao nên sử dụng PostgreSQL thay vì MySQL?

PostgreSQL là hệ quản trị cơ sở dữ liệu mã nguồn mở, tương thích với nhiều ngôn ngữ lập trình như Python, Java, Perl, Ruby và hỗ trợ dữ liệu JSON linh hoạt. Nó có hiệu suất cao với khả năng xử lý dữ liệu lớn, truy vấn song song và phân vùng bảng. PostgreSQL cũng mở rộng tốt theo cả chiều ngang và dọc, đáp ứng các ứng dụng từ nhỏ đến phức tạp. Bảo mật mạnh mẽ với xác thực đa cấp, mã hóa và hỗ trợ SSL. Cộng đồng người dùng lớn và tích cực, giúp hỗ trợ qua các diễn đàn, mailing list.

2.3. Ngôn ngữ Python

Python là một ngôn ngữ lập trình thông dịch, dễ học và mạnh mẽ, được sử dụng rộng rãi trong nhiều lĩnh vực từ phát triển phần mềm đến khoa học dữ liệu và trí tuệ nhân tạo. Python nổi bật với cú pháp đơn giản và dễ đọc, giúp người lập trình dễ dàng tạo ra các ứng dụng và dự án phức tạp một cách nhanh chóng và hiệu quả.



Hình 2.4 Python logo

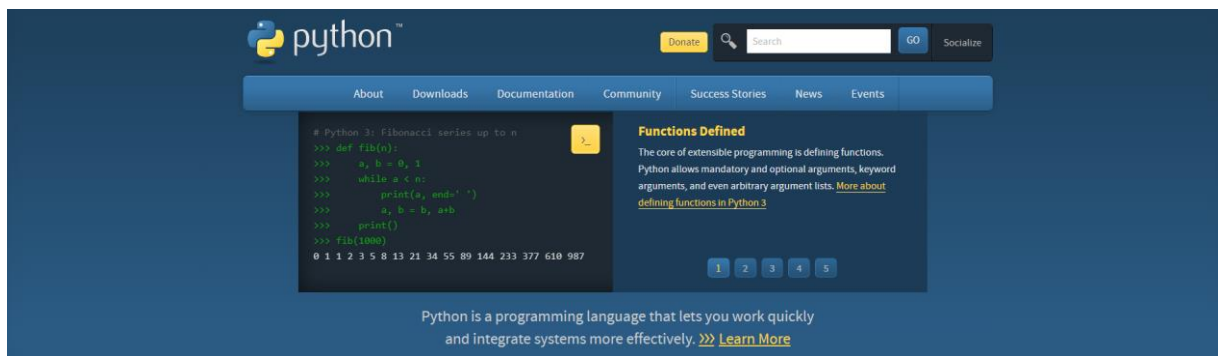
Một số tính năng nổi bật của Python:

- **Cú Pháp Đơn Giản:** Python có cú pháp dễ đọc và dễ hiểu, giúp lập trình viên tập trung vào việc giải quyết vấn đề mà không bị phân tâm bởi cú pháp phức tạp.
- **Hỗ Trợ Toàn Diện:** Python có một cộng đồng lớn và tích hợp nhiều thư viện và framework, giúp lập trình viên giảm thời gian và công sức trong việc phát triển ứng dụng.
- **Đa Năng và Mạnh Mẽ:** Python được sử dụng cho nhiều mục đích từ việc phát triển ứng dụng desktop, web, mobile đến xử lý dữ liệu và trí tuệ nhân tạo.
- **Thân Thiện với Người Dùng:** Python có cộng đồng sôi nổi và tài liệu phong phú, cùng với một số IDE như PyCharm giúp việc lập trình trở nên dễ dàng và hiệu quả.
- **Quản Lý Môi Trường Ảo:** Python hỗ trợ quản lý môi trường ảo, giúp phân biệt rõ ràng giữa các phiên bản và phụ thuộc của các gói Python khác nhau, đảm bảo tính nhất quán và giảm xung đột.
- **Tích Hợp với Công Cụ Phát Triển Khác:** Python tích hợp tốt với các công cụ phát triển phổ biến khác như Git, Mercurial và SVN để quản lý phiên bản và kiểm soát mã nguồn.

Ngoài ra, Python cũng là một công cụ mạnh mẽ cho việc xử lý dữ liệu và tính toán khoa học. Với các thư viện như NumPy, Pandas và SciPy, Python đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trong lĩnh vực khoa học dữ liệu và trí tuệ nhân tạo.

Cài đặt:

Bước 1: Truy cập liên kết <https://www.python.org/downloads/> và chọn phiên bản python mới nhất



Hình 2.5 Truy cập liên kết

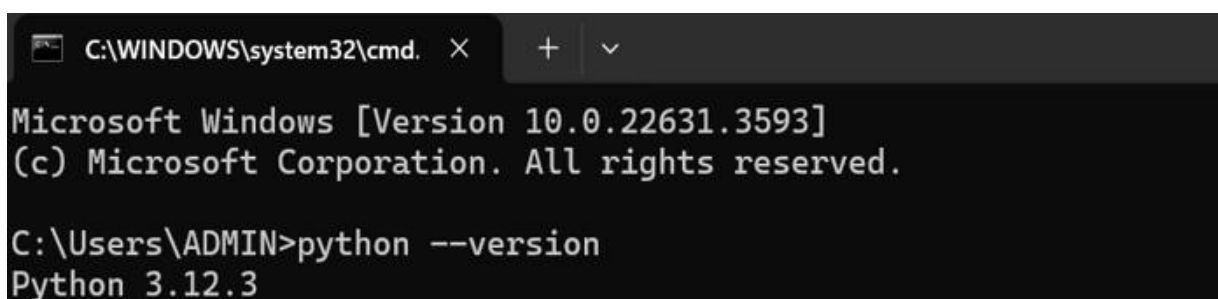
Bước 2: Tiến hành cài đặt

Sau khi tải, mở file cài đặt Python (đuôi file là .exe) và chọn "Install Now"



Hình 2.6 Tiến hành cài đặt

Bước 3: Mở Command Prompt, gõ python --version để kiểm tra phiên bản Python đã cài đặt



Hình 2.7 Mở Command Prompt

2.4. Django và Django REST Framework

Django và Django REST Framework hoạt động cùng nhau một cách liền mạch. Django cung cấp nền tảng phát triển ứng dụng web mạnh mẽ với các chức năng như ORM, routing, xác thực và phân quyền, trong khi DRF mở rộng Django để hỗ trợ xây dựng các API RESTful hiện đại. Điều này giúp các nhà phát triển có thể dễ dàng xây dựng ứng dụng web truyền thống và cung cấp API trong cùng một dự án Django.

Django và Django REST Framework đều là framework web được sử dụng để phát triển ứng dụng web, nhưng chúng có những mục đích và tính năng riêng:

Danh mục	Django	Django REST Framework
Mục đích	Django là một khuôn khổ web cấp cao để xây dựng các ứng dụng web. Nó tuân theo mô hình kiến trúc Model-View-Template (MVT).	DRF là một bộ công cụ mạnh mẽ được thiết kế riêng để xây dựng Web API bằng Django. Nó mở rộng Django để tạo điều kiện thuận lợi cho việc tạo RESTful API.
Đặc trưng	- Full-Stack Framework : Cung cấp bộ công cụ hoàn chỉnh để xây dựng các ứng dụng web, bao gồm ORM (Ánh xạ quan hệ đối tượng), xác thực và giao diện quản trị. - Kết xuất HTML : Chủ yếu phục vụ nội dung HTML, phù hợp với các ứng dụng web truyền thống. - Định tuyến URL : Quản lý URL và ánh xạ chúng tới các chế độ xem trả về phản hồi	- Phát triển API : Tập trung vào việc tạo API phục vụ phản hồi JSON hoặc XML, phù hợp với các ứng dụng phía máy khách và ứng dụng di động. - Tuần tự hóa : Cung cấp các trình tuần tự hóa để chuyển đổi các kiểu dữ liệu phức tạp (như bộ truy vấn) thành JSON và ngược lại. - Xác thực và Quyền : Cung cấp hỗ trợ tích hợp cho nhiều phương thức xác thực khác nhau (ví dụ: dựa trên mã thông báo, dựa trên phiên) và các lớp quyền có thể tùy chỉnh. - API có thể duyệt : Bao gồm giao diện web thân thiện với người

	HTML. - Xử lý biểu mẫu : Bao gồm hỗ trợ tích hợp cho biểu mẫu, xác thực và hiển thị.	dùng để kiểm tra điểm cuối API trực tiếp trong trình duyệt. - ViewSets & Routers : Đơn giản hóa việc tạo chế độ xem API và định tuyến URL thông qua các viewset và bộ định tuyến, giảm mã chuẩn.
--	---	---

Bảng 2. 2 Django và Django REST Framework

2.5. Mô hình Client-Server

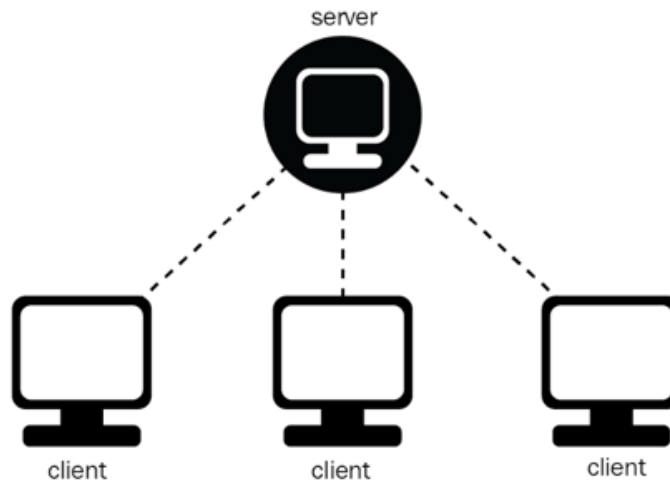
Mô hình client - server là mô hình giúp các máy tính giao tiếp truyền tải dữ liệu cho nhau. Nhắc đến Client với Server thì đây là 2 từ khóa khá phổ biến và ai cũng biết.

Client và server về bản chất thì nó là 2 máy tính giao tiếp và truyền tải dữ liệu cho nhau.

Máy tính đóng vai trò là máy khách – **Client**: Với vai trò là máy khách, chúng sẽ không cung cấp tài nguyên đến các máy tính khác mà chỉ sử dụng tài nguyên được cung cấp từ máy chủ. Một client trong mô hình này có thể là một server cho mô hình khác, tùy thuộc vào nhu cầu sử dụng của người dùng.

Máy tính đóng vai trò là máy chủ – **Server**: Là máy tính có khả năng cung cấp tài nguyên và các dịch vụ đến các máy khách khác trong hệ thống mạng. Server đóng vai trò hỗ trợ cho các hoạt động trên máy khách client diễn ra hiệu quả hơn

Mô hình Client Server là mô hình mạng máy tính trong đó các máy tính con được đóng vai trò như một máy khách, chúng làm nhiệm vụ gửi yêu cầu đến các máy chủ. Để máy chủ xử lý yêu cầu và trả kết quả về cho máy khách đó



Hình 2.8 Mô hình Client Server

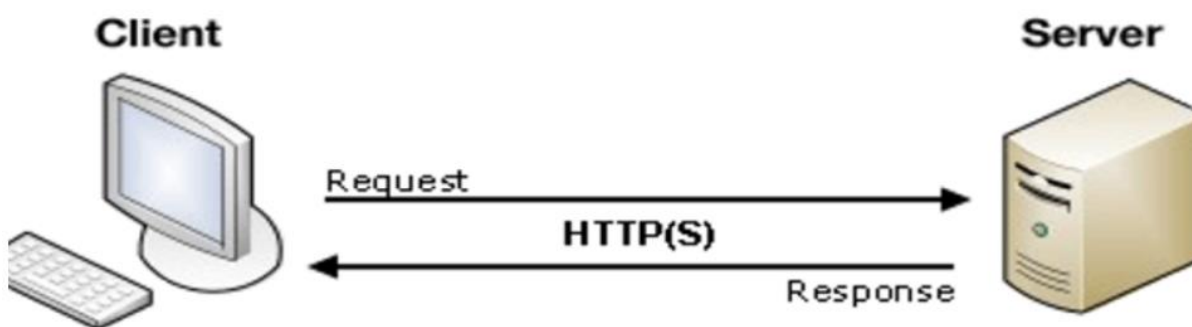
2.5.1. Nguyên tắc hoạt động

Trong mô hình Client Server, server chấp nhận tất cả các yêu cầu hợp lệ từ mọi nơi khác nhau trên Internet, sau đó trả kết quả về máy tính đã gửi yêu cầu đó

Máy tính được coi là máy khách khi chúng làm nhiệm vụ gửi yêu cầu đến các máy chủ và đợi câu trả lời được gửi về.

Để máy khách và máy chủ có thể giao tiếp được với nhau thì giữa chúng phải có một chuẩn nhất định, và chuẩn đó được gọi là giao thức. Một số giao thức được sử dụng phổ biến hiện nay như: HTTPS, TCP/IP, FTP,...

Nếu máy khách muốn lấy được thông tin từ máy chủ, chúng phải tuân theo một giao thức mà máy chủ đó đưa ra. Nếu yêu cầu đó được chấp nhận thì máy chủ sẽ thu thập thông tin và trả về kết quả cho máy khách yêu cầu. Bởi vì Server - máy chủ luôn luôn trong trạng thái sẵn sàng để nhận request từ client nên chỉ cần client gửi yêu cầu tín hiệu và chấp nhận yêu cầu đó thì server sẽ trả kết quả về phía client trong thời gian ngắn nhất.



Hình 2.9 Client Server

2.5.2. Ưu điểm

Giúp chúng ta có thể làm việc trên bất kì một máy tính nào có hỗ trợ giao thức truyền thông. Giao thức chuẩn này cũng giúp các nhà sản xuất tích hợp lên nhiều sản phẩm khác nhau mà không gặp phải khó khăn gì.

Có thể có nhiều server cùng làm một dịch vụ, chúng có thể nằm trên nhiều máy tính hoặc một máy tính.

Chỉ mang đặc điểm của phần mềm mà không hề liên quan đến phần cứng, ngoài yêu cầu duy nhất là server phải có cấu hình cao hơn các client.

Hỗ trợ người dùng nhiều dịch vụ đa dạng và sự tiện dụng bởi khả năng truy cập từ xa.

Cung cấp một nền tảng lý tưởng, cho phép cung cấp tích hợp các kỹ thuật hiện đại như mô hình thiết kế hướng đối tượng, hệ chuyên gia, hệ thông tin địa lý (GIS)

2.5.3. Nhược điểm

Vấn đề bảo mật dữ liệu thông tin đôi khi còn chưa được an toàn lắm. Vì do phải trao đổi dữ liệu giữa 2 máy tính khác nhau ở 2 khu vực địa lý cách xa nhau. Và đây cũng nhược điểm duy nhất của mô hình này.

Tuy nhiên vấn đề này thì có một số giao thức đã hỗ trợ bảo mật dữ liệu khi truyền tải. Giao thức được sử dụng phổ biến như HTTPS.

2.6. Công cụ phát triển Visual Studio Code

2.6.1. Visual Studio Code là gì

Visual Studio Code chính là ứng dụng cho phép biên tập, soạn thảo các đoạn code để hỗ trợ trong quá trình thực hiện xây dựng, thiết kế website một cách nhanh chóng. Visual Studio Code hay còn được viết tắt là VS Code. Trình soạn thảo này vận hành mượt mà trên các nền tảng như Windows, macOS, Linux. Hơn thế nữa, VS Code còn cho khả năng tương thích với những thiết bị máy tính có cấu hình tầm trung vẫn có thể sử dụng dễ dàng.



Hình 2.10 Visual Studio Code

Visual Studio Code hỗ trợ đa dạng các chức năng Debug, đi kèm với Git, có Syntax Highlighting. Đặc biệt là tự hoàn thành mã thông minh, Snippets, và khả năng cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép các lập trình viên thay đổi Theme, phím tắt, và đa dạng các tùy chọn khác. Mặc dù trình soạn thảo Code này tương đối nhẹ, nhưng lại bao gồm các tính năng mạnh mẽ

2.6.1. Tại sao nên sử dụng Visual Studio Code?

- Hỗ trợ đa nền tảng: Linux, Mac, Windows,...
- Hỗ trợ đa ngôn ngữ: C/C++, C#, JavaScript, JSON, Visual Basic, HTML, CSS,...
- Ít dung lượng
- Tính năng mạnh mẽ
- Intellisense chuyên nghiệp

- Giao diện thân thiện
- Kiến trúc mạnh mẽ và người dùng có thể khai thác mở rộng
- Số lượng người sử dụng lớn tạo nên ộng đồng hỗ trợ rộng rãi

Chương 3: PHÂN TÍCH, THIẾT KẾ HỆ THỐNG

3.1 Mô tả bài toán

- Tìm hiểu cách vận hành một hệ thống quản lý trò chơi và dữ liệu người dùng
- Xác định giải pháp, hướng phát triển xây dựng hệ thống trò chơi
- Xây dựng hệ thống web site có các chức năng cơ bản như sau:

Hệ thống CMS quản lý user trong hệ thống và tạo các topic của hệ thống mà người chơi có thể sử dụng

Đăng ký và đăng nhập: cho phép người dùng tạo tài khoản, đăng nhập để trải nghiệm trò chơi

Quản lý tài khoản cá nhân: người dùng quản lý thông tin của chính mình đồng thời tạo ra các topic cá nhân

Tham gia phòng chơi hoặc tạo phòng chơi: khi ở menu trang chủ, người chơi có thể tham gia các phòng có sẵn ở hệ thống, hoặc có thể tạo và tùy chỉnh phòng chơi của mình, và bạn bè của người chơi có thể tìm kiếm thông qua id của phòng

Trải nghiệm trò chơi với luật chơi như sau:

Hệ thống sẽ hiện ngẫu nhiên ảnh trong chủ đề, 2 người chơi theo lượt, người chơi đến lượt phải trả lời đúng nội dung ảnh để cập, chỉ khi trả lời đúng, bộ đếm thời gian (giảm dần) sẽ dừng lại và lượt chơi được chuyển cho người chơi khác, người chơi không trả lời được có thể trả lời lại hoặc chuyển đổi bức ảnh khác để trả lời, người chơi nào có bộ đếm thời gian về 0 trước sẽ thua trận đấu. Người thắng sẽ được cộng điểm elo và người thua sẽ bị trừ điểm elo.

Có 2 chế độ chơi:

- Solo 1vs1: 2 người chơi trong phòng chơi phải nhập đáp án vào ô đáp án và hệ thống sẽ xác định đúng sai
- Có giám khảo: sẽ có 1 người thứ 3 tham gia vào phòng chơi với vai trò giám khảo, giám khảo có thể thấy đáp án, 2 người chơi giao tiếp trực tiếp bằng lời nói và giám khảo xác định đúng sai và nhấn nút báo hiệu cho người chơi biết

Bảng xếp hạng người chơi: người chơi có điểm elo cá nhân và hệ thống dùng điểm đó để xếp hạng người chơi trong toàn hệ thống

3.1.1 Yêu cầu của người chơi

Chức năng đăng nhập của người chơi

Khi đăng nhập vào website, người dùng nhập email và mật khẩu đã đăng ký để sử dụng website. Nếu hệ thống kiểm tra nhập đúng sẽ hướng người dùng tới màn hình trang chủ, nếu sai sẽ đưa ra thông báo để người dùng nhập lại. Khi đăng nhập thành công người dùng sẽ được cấp một cặp token (access token và refresh token) đã được mã hóa để sử dụng. Trường hợp người dùng đăng nhập lại sau khi thoát trang web, nếu thời gian cấp token còn có hiệu lực thì người dùng sẽ không cần phải đăng nhập lại.

Nếu thời hạn của token hết thì người dùng buộc phải đăng nhập lại từ đầu

Chức năng đăng kí của người chơi:

Khi người dùng muốn tạo tài khoản mới để sử dụng website, họ truy cập vào trang đăng ký. Trên trang đăng ký, người dùng điền thông tin cá nhân cần thiết như tên, địa chỉ email và mật khẩu. Họ cũng cần nhập lại mật khẩu để xác nhận. Sau khi điền đầy đủ thông tin và ấn nút "Đăng ký", hệ thống sẽ kiểm tra tính hợp lệ của thông tin.

Nếu thông tin không đúng định dạng hoặc email đã được đăng ký trước đó, hệ thống sẽ hiển thị thông báo lỗi yêu cầu người dùng nhập lại thông tin. Nếu thông tin hợp lệ, hệ thống sẽ mã hóa mật khẩu và lưu thông tin người dùng vào cơ sở dữ liệu. Tiếp theo, hệ thống sẽ gửi một email xác nhận đăng ký thành công đến địa chỉ email mà người dùng đã cung cấp. Email này chứa một liên kết (link) xác nhận đăng ký. Người dùng sẽ cần bấm vào liên kết xác nhận trong email để hoàn tất quá trình đăng ký. Sau khi xác nhận, họ sẽ nhận thông báo đăng ký thành công và được chuyển hướng về trang chủ.

Chức năng cập nhật thông tin tài khoản:

Người dùng đã đăng nhập vào hệ thống và muốn cập nhật thông tin tài khoản để đảm bảo rằng thông tin của họ được cài đặt chính xác. Chức năng cập nhật thông tin cho tài khoản cho phép người dùng chỉnh sửa, bổ sung và cập nhật cài đặt một cách linh hoạt.

Người dùng nhấn biểu tượng hồ sơ của bạn, nơi họ thấy các trường thông tin cần thiết như ảnh đại diện, username, language,... Họ cập nhật các thông tin một cách chi tiết và chính xác.

Sau khi điền đầy đủ thông tin cần thiết, họ ấn nút "Cập nhật" và hệ thống sẽ kiểm tra tính hợp lệ của dữ liệu. Nếu thông tin hợp lệ, thông tin tài khoản sẽ được cập nhật trong cơ sở dữ liệu

Chức năng quản lý các chủ đề cá nhân

Người chơi sau khi vào trang cài đặt thông tin cá nhân, có nhu cầu tạo ra những chủ đề, những bộ câu hỏi của riêng mình khi chơi với người khác.

Người chơi tạo bộ câu hỏi mới bằng 1 file zip chứa ảnh và tên của ảnh chính là đáp án của ảnh đó. Sau khi đặt tên cho chủ đề, hệ thống sẽ tự động extract file zip và tạo chủ đề mới cho người chơi và cập nhật vào cơ sở dữ liệu

Người chơi sau đó có thể cập nhật chỉnh sửa chủ đề hoặc xóa chủ đề ngay trong màn hình cài đặt thông tin cá nhân

Chức năng tham gia phòng chơi

Người chơi có thể tham gia phòng chơi bằng cách nhấn vào danh sách phòng có sẵn trên trang chủ hoặc tìm kiếm phòng bằng ID.

Khi chọn từ danh sách, người chơi sẽ thấy các thông tin như tên phòng, số lượng người chơi và trạng thái. Nếu phòng còn trống, hệ thống sẽ đưa người chơi vào phòng. Với tìm kiếm bằng ID, người chơi nhập mã ID của phòng vào ô tìm kiếm, hệ thống kiểm tra tính hợp lệ và đưa người chơi vào nếu phòng tồn tại và còn chỗ. Trường hợp phòng đầy hoặc không tồn tại, hệ thống sẽ hiển thị thông báo lỗi. Sau khi tham gia, người chơi được chuyển đến giao diện phòng chờ để bắt đầu trải nghiệm.

Chức năng tạo phòng chơi

Người chơi có thể tạo phòng chơi mới từ giao diện trang chủ. Khi tạo phòng, người chơi cần thiết lập các thông tin như: chọn chủ đề (có thể chọn nhiều chủ đề từ kho cá nhân hoặc hệ thống), chế độ chơi (Solo 1vs1 hoặc Có giám khảo), và thời gian chơi. Sau khi điền đầy đủ thông tin, người chơi nhấn nút "Tạo phòng". Hệ thống sẽ kiểm tra tính hợp lệ của thông tin, tạo phòng chơi mới và hiển thị mã ID phòng để chia sẻ với bạn bè. Người chơi sau khi tạo phòng sẽ được chuyển trực tiếp đến giao diện phòng chờ để chờ người chơi khác tham gia hoặc bắt đầu ngay khi đủ điều kiện.

Chức năng chơi ở chế độ 1vs1

Trong chế độ 1vs1, phòng chơi có hai người chơi đối đầu trực tiếp. Hệ thống hiển thị lần lượt các hình ảnh ngẫu nhiên thuộc chủ đề đã chọn. Mỗi người chơi sẽ trả lời nội dung của ảnh trong lượt của mình bằng cách nhập đáp án vào ô trả lời. Nếu trả

lời đúng, bộ đếm thời gian giảm dần của người chơi sẽ dừng lại và lượt chơi chuyển sang đối thủ. Nếu trả lời sai, người chơi có thể thử lại hoặc chuyển sang bức ảnh khác, nhưng thời gian vẫn tiếp tục giảm. Người chơi nào hết thời gian trước sẽ thua trận đấu. Sau trận, hệ thống tính toán điểm Elo để cập nhật xếp hạng cho cả hai người chơi.

Chức năng chơi ở chế độ có giám khảo

Trong chế độ có giám khảo, phòng chơi gồm 3 người: 2 người chơi và 1 giám khảo. Hệ thống hiển thị lần lượt các hình ảnh ngẫu nhiên thuộc chủ đề đã chọn. Người chơi đến lượt trả lời bằng cách giao tiếp trực tiếp với giám khảo, không cần nhập đáp án. Giám khảo có quyền xem đáp án đúng của hình ảnh và quyết định xem câu trả lời của người chơi có chính xác hay không bằng cách nhấn nút "Đúng" hoặc "Sai". Nếu đáp án đúng, bộ đếm thời gian của người chơi dừng lại và lượt chơi được chuyển cho đối thủ. Nếu đáp án sai, người chơi có thể tiếp tục trả lời hoặc chuyển sang ảnh khác, nhưng thời gian vẫn giảm. Người chơi nào hết thời gian trước sẽ thua trận. Sau trận đấu, hệ thống cập nhật điểm Elo cho người thắng và thua

Chức năng bảng xếp hạng

Người chơi có thể truy cập bảng xếp hạng từ trang chủ bằng cách nhấn vào mục "Bảng xếp hạng". Hệ thống hiển thị danh sách các người chơi được sắp xếp theo điểm Elo từ cao đến thấp. Mỗi mục trong bảng xếp hạng hiển thị các thông tin như: tên người chơi, điểm Elo. Dữ liệu được cập nhật theo thời gian thực để phản ánh kết quả mới nhất từ các trận đấu.

3.1.2 Yêu cầu admin

Chức năng quản lý user ở CMS

Quản trị viên có thể quản lý người dùng trong hệ thống thông qua giao diện CMS. Các chức năng bao gồm: cập nhật email và tên người dùng, đổi mật khẩu, và xóa tài khoản. Quản trị viên có thể chỉnh sửa email hoặc tên người dùng trong danh sách, sau đó lưu thay đổi. Hệ thống sẽ kiểm tra tính hợp lệ và cập nhật thông tin vào cơ sở dữ liệu.

Đối với đổi mật khẩu, quản trị viên nhập mật khẩu mới cho người dùng, hệ thống sẽ mã hóa và lưu vào cơ sở dữ liệu. Khi xóa người dùng, quản trị viên xác nhận thao tác, và hệ thống sẽ xóa toàn bộ thông tin liên quan đến tài khoản đó khỏi cơ sở dữ liệu.

-Chức năng tạo topic hệ thống

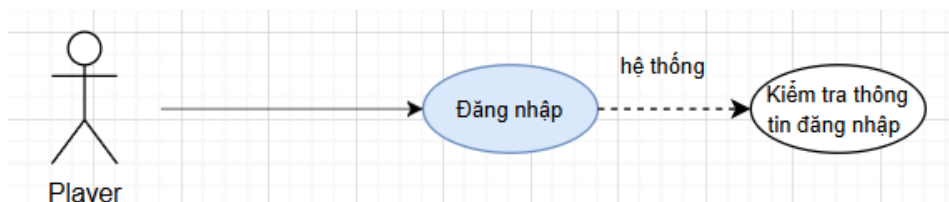
Quản trị viên có thể tạo các topic hệ thống để người chơi sử dụng trong trò chơi. Tại giao diện CMS, quản trị viên nhập tên topic và tải lên một file zip chứa các

hình ảnh liên quan. Tên của mỗi file ảnh trong zip sẽ được hệ thống tự động sử dụng làm đáp án tương ứng. Sau khi tải lên, hệ thống kiểm tra tính hợp lệ của file zip, giải nén và lưu các thông tin về topic, hình ảnh và đáp án vào cơ sở dữ liệu. Quản trị viên cũng có thể chỉnh sửa hoặc xóa các topic hệ thống đã tạo ngay trong giao diện CMS

3.2 Phân tích hệ thống

3.2.1 chức năng đăng nhập của người chơi

-Biểu đồ Use case đăng nhập của người tìm việc



Hình 3.1 Use Case chức năng đăng nhập

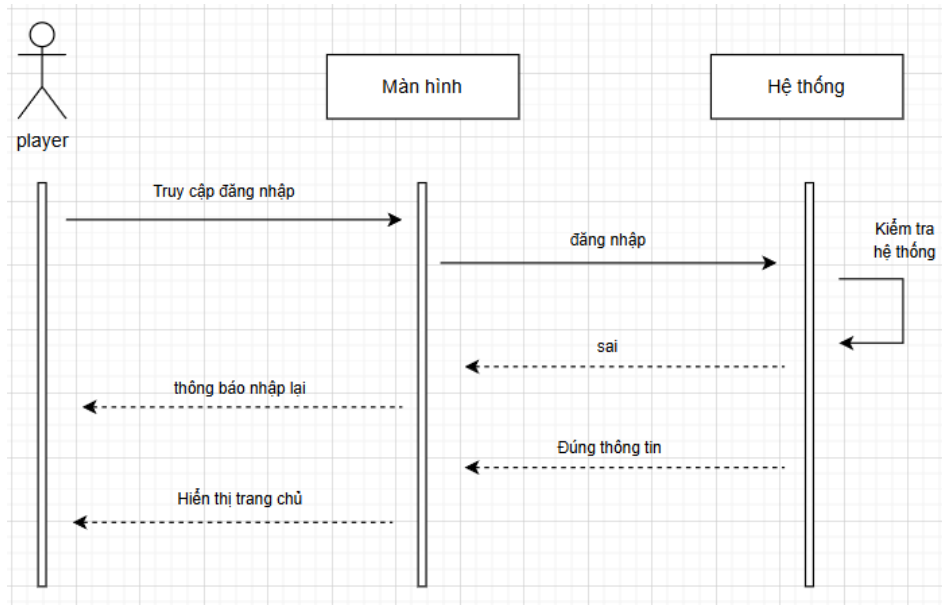
-Mô tả Use Case đăng nhập của người chơi

Use Case	Đăng nhập
Mục tiêu	Người chơi muốn đăng nhập vào tài khoản của mình
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã có tài khoản và truy cập trang đăng nhập
Kết quả đầu ra	Người chơi đăng nhập thành công và có quyền truy cập vào tài khoản
Luồng sự kiện chính	1. Người tìm việc khởi động Use Case này bằng cách truy cập trang đăng nhập 2. Hệ thống yêu cầu nhập thông tin bao gồm email và mật khẩu 3. Người chơi nhập thông tin đăng nhập 4. Hệ thống kiểm tra tính hợp lệ của thông tin đăng nhập và xác định tài khoản người dùng 5. Nếu thông tin đúng, hệ thống xác thực đăng nhập và tạo cặp token (access token và refresh token) 6. Người chơi được chuyển hướng vào trang chủ của trò chơi, đã đăng nhập và sẵn sàng sử dụng các chức năng

Luồng sự kiện phụ	Nếu thông tin đăng nhập không hợp lệ, hệ thống thông báo cho người chơi và yêu cầu nhập lại
-------------------	---

Bảng 3.1 Mô tả Use Case đăng nhập

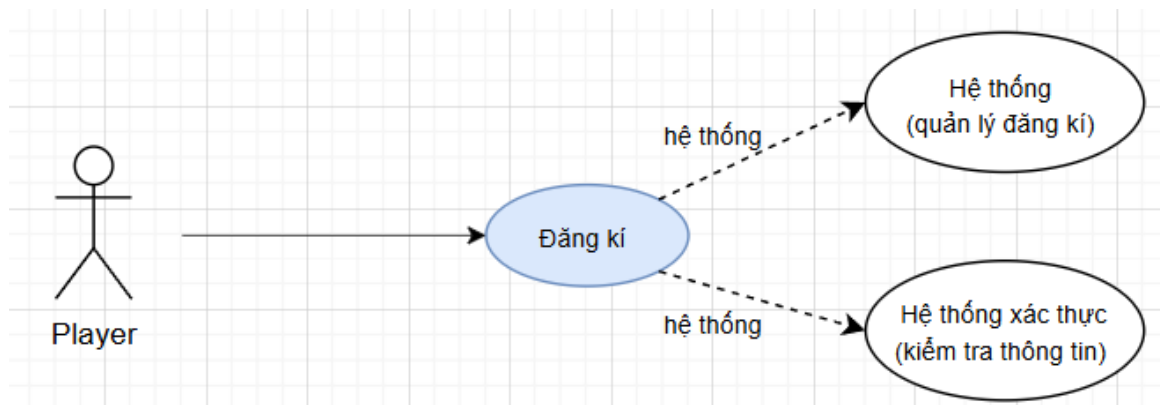
-Biểu đồ tuần tự đăng nhập của người chơi



Hình 3.2 Sơ đồ tuần tự chức năng đăng nhập

3.2.2. Chức năng đăng ký của người chơi

-Biểu đồ Use Case đăng ký của người chơi



Hình 3.3 Use Case chức năng đăng ký

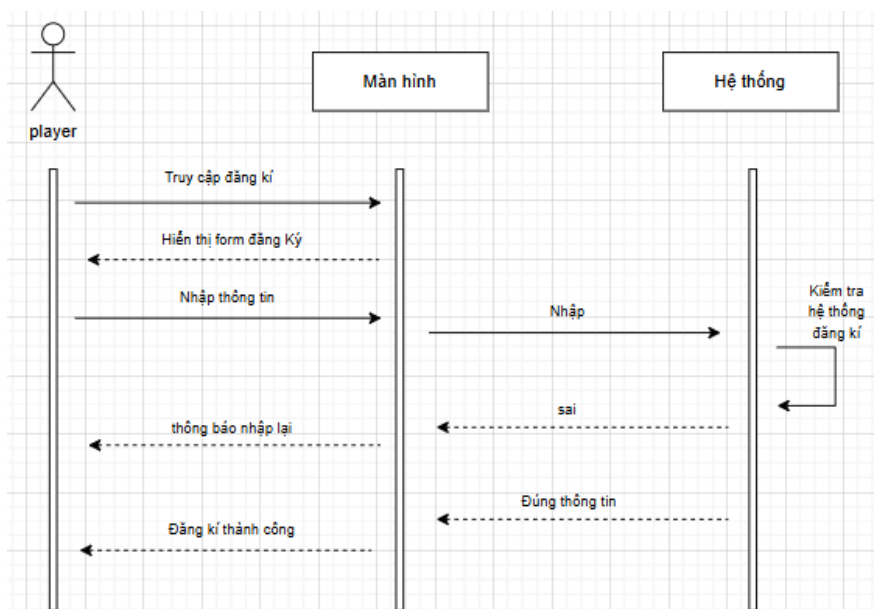
- Mô tả Use Case đăng ký của người chơi

Use Case	Đăng ký người chơi
Mục tiêu	Người chơi muốn tạo tài khoản mới để trải nghiệm trò chơi

Tác Nhân	Người chơi
Tiền điều kiện	Khả năng truy cập trang đăng ký của trang web
Kết quả đầu ra	Người chơi đăng Ký thành công và có tài khoản để sử dụng
Luồng sự kiện chính	1. Người tìm việc khởi động Use Case bằng cách truy cập trang đăng ký. 2. Hệ thống yêu cầu người chơi nhập thông cá nhân cần thiết để tạo tài khoản 3. Người tìm việc nhập thông tin đăng nhập. 4. Hệ thống kiểm tra tính hợp lệ của thông tin đăng nhập và xác nhận tính hợp lệ 5. Nếu thông tin hợp lệ, hệ thống tạo tài khoản và thông báo thành công
Luồng sự kiện phụ	Nếu thông tin không hợp lệ, hệ thống thông báo cho người chơi và yêu cầu nhập lại thông tin chính xác

3.2 Mô tả Use Case đăng ký Bảng

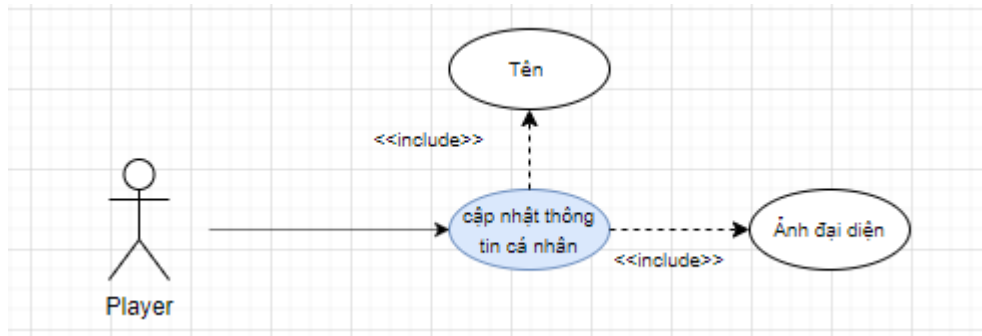
- Biểu đồ tuần tự đăng ký của người chơi



Hình 3.4 Biểu đồ tuần tự đăng ký của người chơi

3.2.3 Chức năng cập nhập thông tin tài khoản:

- Biểu đồ Use Case cập nhập thông tin tài khoản



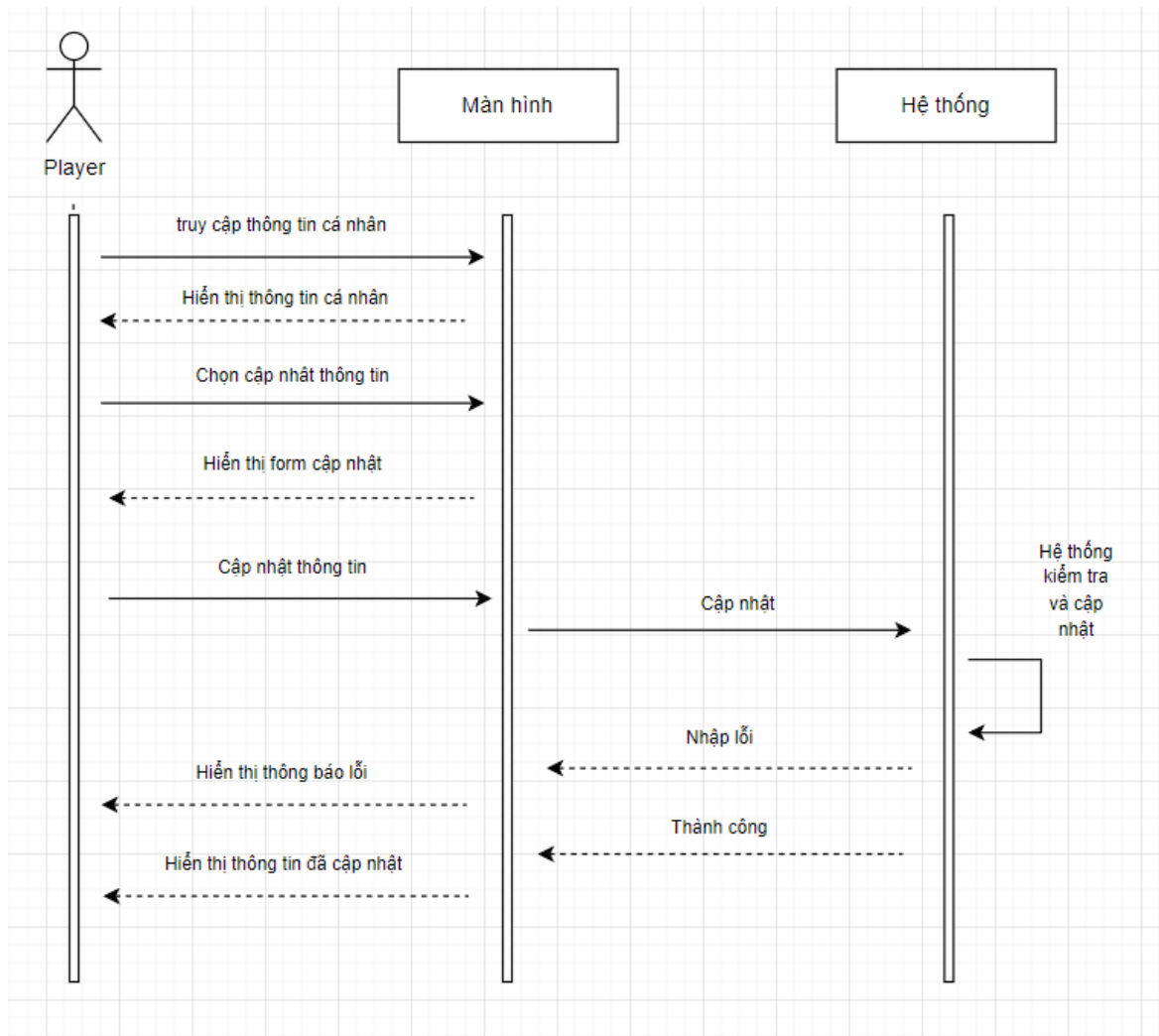
Hình 3.5 Use Case chức năng cập nhập thông tin

- Mô tả Use Case cập nhập thông tin tài khoản

Use Case	Cập nhật thông tin tài khoản
Mục tiêu	Người chơi muốn chỉnh sửa thông tin tài khoản cá nhân
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã đăng nhập và có quyền truy cập vào trang quản lý tài khoản
Kết quả đầu ra	Thông tin tài khoản của người chơi được cập nhật thành công trong cơ sở dữ liệu
Luồng sự kiện chính	1. Người chơi truy cập trang quản lý tài khoản cá nhân. 2. Hệ thống hiển thị các trường thông tin có thể chỉnh sửa. 3. Người chơi chỉnh sửa thông tin và nhấn "Cập nhật". 4. Hệ thống kiểm tra tính hợp lệ của thông tin mới. 5. Nếu thông tin hợp lệ, hệ thống lưu thay đổi vào cơ sở dữ liệu và hiển thị thông báo thành công.
Luồng sự kiện phụ	Nếu thông tin không hợp lệ (ví dụ: email sai định dạng), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

Bảng 3. 3 Mô tả Use Case cập nhập tài khoản

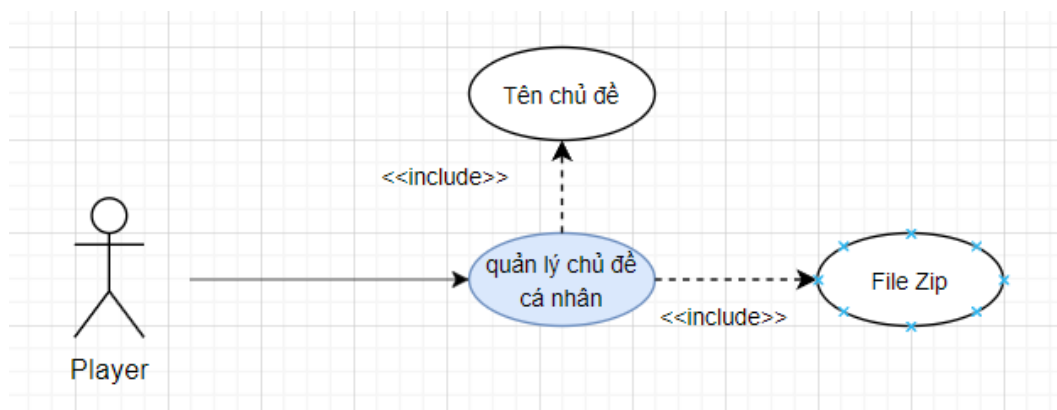
- Biểu đồ tuần tự cập nhập thông tin tài khoản



Hình 3.6 Biểu đồ tuần tự chức năng cập nhập thông tin tài khoản

3.2.4 chức năng quản lý các chủ đề cá nhân

- Biểu đồ Use Case



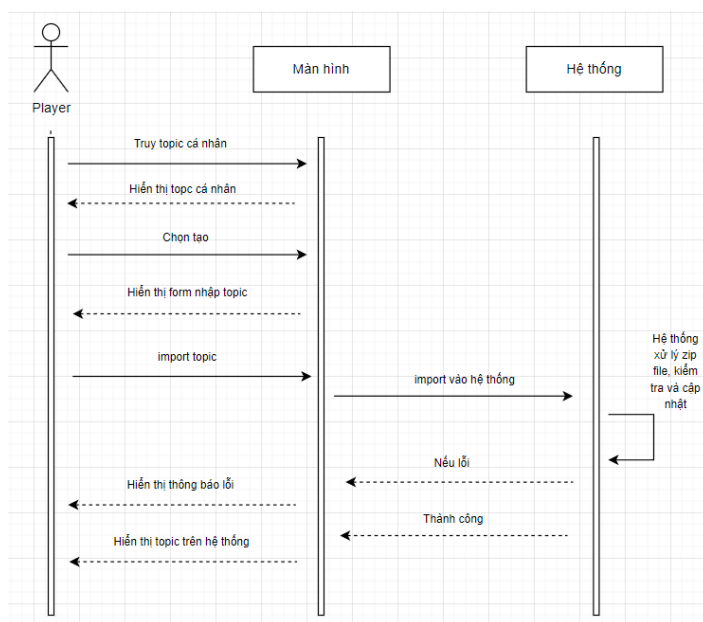
Hình 3.7 Use Case chức năng quản lý chủ đề cá nhân

- Mô tả Use Case

Use Case	Quản lý các chủ đề cá nhân
Mục tiêu	Người chơi tạo, chỉnh sửa hoặc xóa các chủ đề câu hỏi cá nhân
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã đăng nhập và có quyền truy cập vào trang thông tin tài khoản có mục chủ đề cá nhân
Kết quả đầu ra	Các chủ đề của người chơi được cập nhật trong cơ sở dữ liệu
Luồng sự kiện chính	1. Người chơi truy cập trang thông tin tài khoản 2. Người chơi chọn tạo mới, chỉnh sửa hoặc xóa một chủ đề. 3. Nếu tạo mới, hệ thống yêu cầu tải lên file zip chứa ảnh và tên chủ đề. 4. Hệ thống giải nén, kiểm tra tính hợp lệ của file và lưu thông tin vào cơ sở dữ liệu. 5. Hệ thống hiển thị danh sách chủ đề đã được cập nhật.
Luồng sự kiện phụ	Nếu file zip không hợp lệ (thiếu ảnh hoặc tên file sai định dạng), hệ thống hiển thị thông báo lỗi và yêu cầu thử lại.

Bảng 3.4 Mô tả Use Case quản lý đề cá nhân

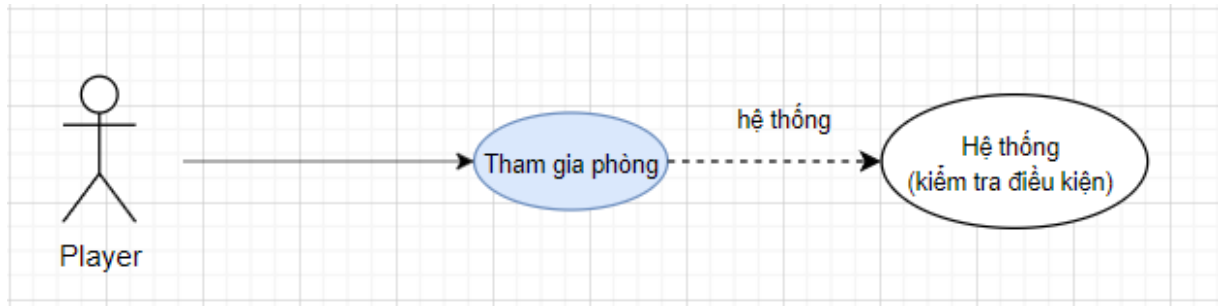
- Biểu đồ tuần tự



Hình 3.8 Biểu đồ tuần tự chức năng quản lý chủ đề cá nhân

3.2.5 Chức năng tham gia phòng chơi

- Biểu đồ Use Case



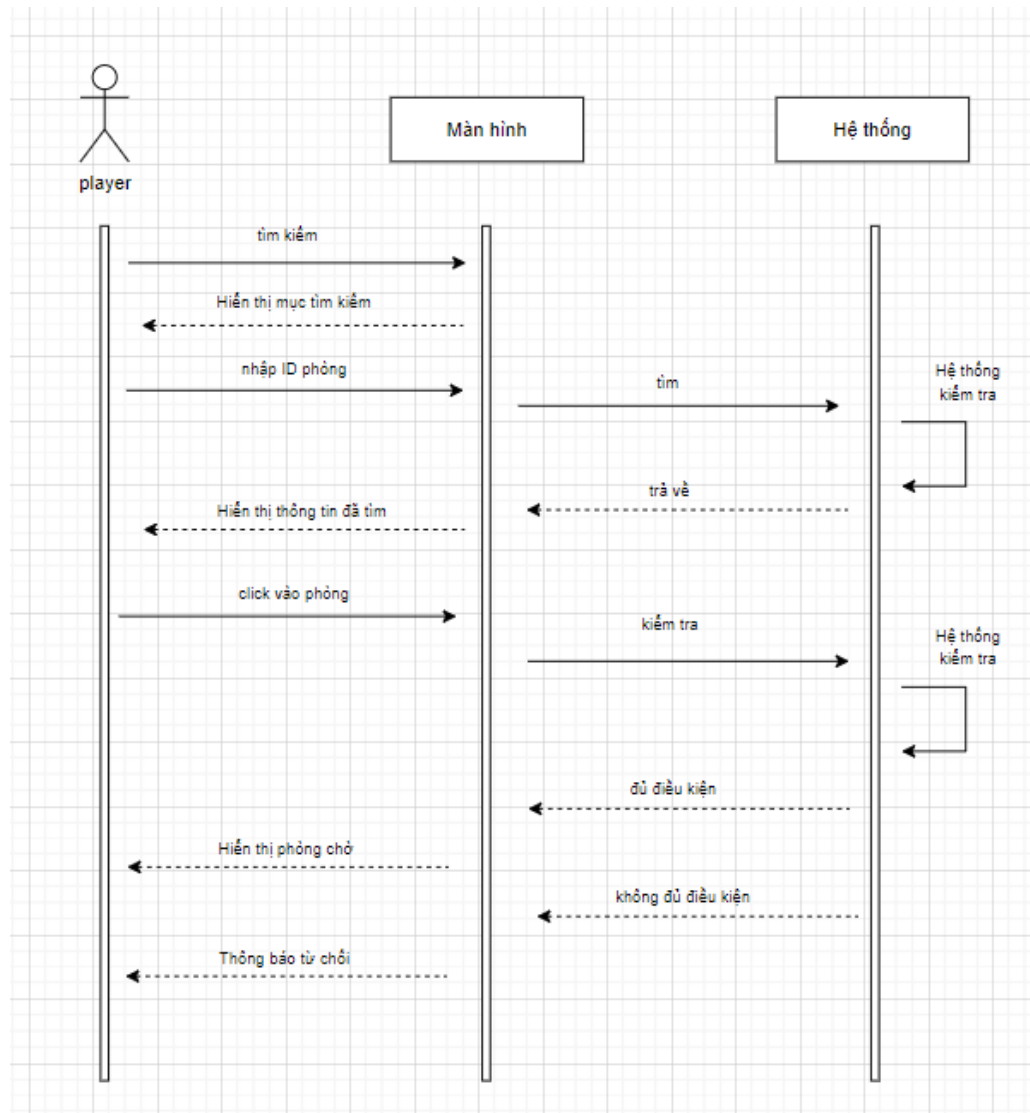
Hình 3.9 Use Case chức năng tham gia phòng

- Mô tả Use Case

Use Case	Tham gia phòng chơi
Mục tiêu	Người chơi tham gia phòng chơi đã có sẵn bằng cách nhấn vào biểu tượng phòng hoặc nhập mã ID phòng
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã đăng nhập và có quyền truy cập vào danh sách phòng chơi hoặc ID phòng
Kết quả đầu ra	Người chơi tham gia thành công và được chuyển đến giao diện phòng chờ
Luồng sự kiện chính	1. Người chơi truy cập danh sách phòng chơi hoặc nhập mã ID phòng. 2. Hệ thống kiểm tra trạng thái phòng chơi (đầy, trống, tồn tại). 3. Nếu hợp lệ, người chơi được thêm vào phòng và chuyển đến giao diện phòng.
Luồng sự kiện phụ	Nếu phòng đầy hoặc không tồn tại, hệ thống hiển thị thông báo lỗi.

Bảng 3. 5 Mô tả Use Case tham gia phòng chơi

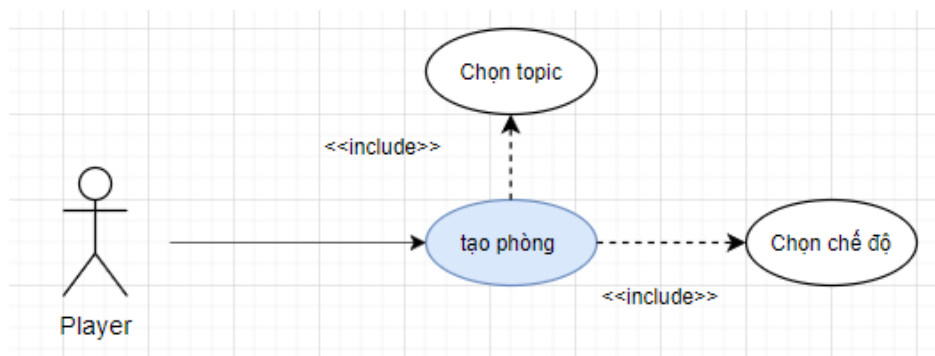
- Biểu đồ tuần tự



Hình 3.10 Biểu đồ tuần tự chức năng tham gia phòng chơi

3.2.6 Chức năng tạo phòng chơi

- Biểu đồ Use Case



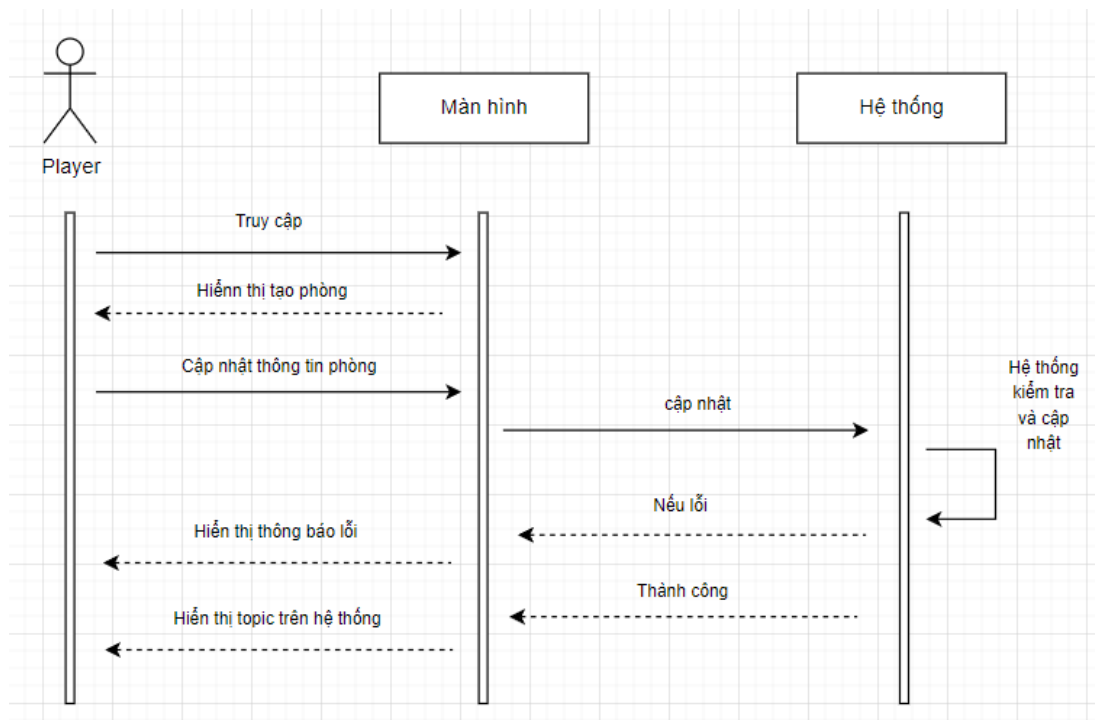
Hình 3.11 Use Case chức năng tạo phòng chơi

- Mô tả Use Case

Use Case	Tạo phòng chơi
Mục tiêu	Người chơi muốn tạo một phòng chơi mới để mời bạn bè hoặc tham gia tự do
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã đăng nhập và có quyền truy cập vào giao diện tạo phòng
Kết quả đầu ra	Phòng chơi mới được tạo thành công và lưu thông tin vào cơ sở dữ liệu và hiển thị ở trang chủ hệ thống
Luồng sự kiện chính	1. Người chơi truy cập giao diện tạo phòng. 2. Người chơi chọn các thông tin: chủ đề (có thể nhiều chủ đề), chế độ chơi và thời gian chơi. 3. Người chơi nhấn nút "Tạo phòng". 4. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu phòng mới vào cơ sở dữ liệu. 5. Hệ thống hiển thị mã ID phòng và chuyển người chơi đến giao diện phòng chờ.
Luồng sự kiện phụ	Nếu thông tin không hợp lệ (ví dụ: không chọn chủ đề), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu nhập lại.

Bảng 3. 6 Mô tả Use Case tạo phòng chơi

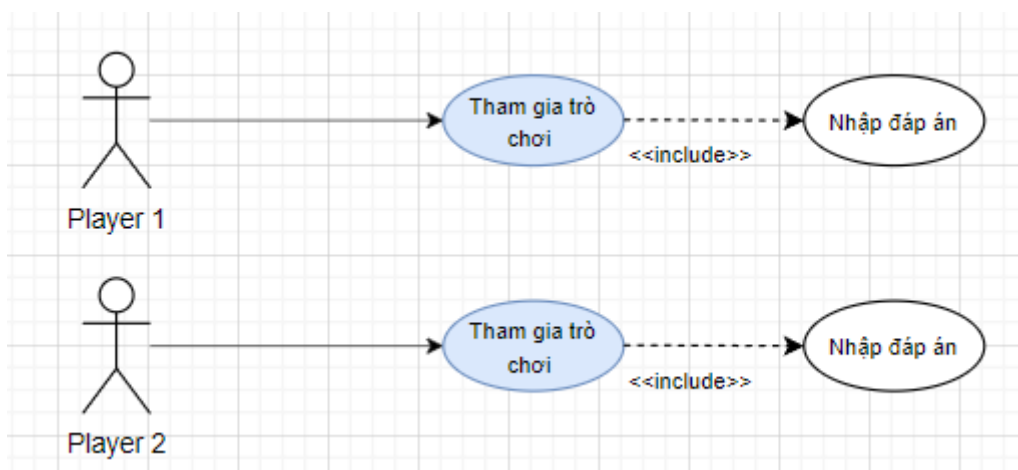
- Biểu đồ tuần tự



Hình 3.12 Biểu đồ tuần tự chức năng tạo phòng chơi

3.2.7 Chức năng chơi ở chế độ 1vs1

- Biểu đồ Use Case



Hình 3.13 Use Case chức năng chơi ở chế độ 1vs1

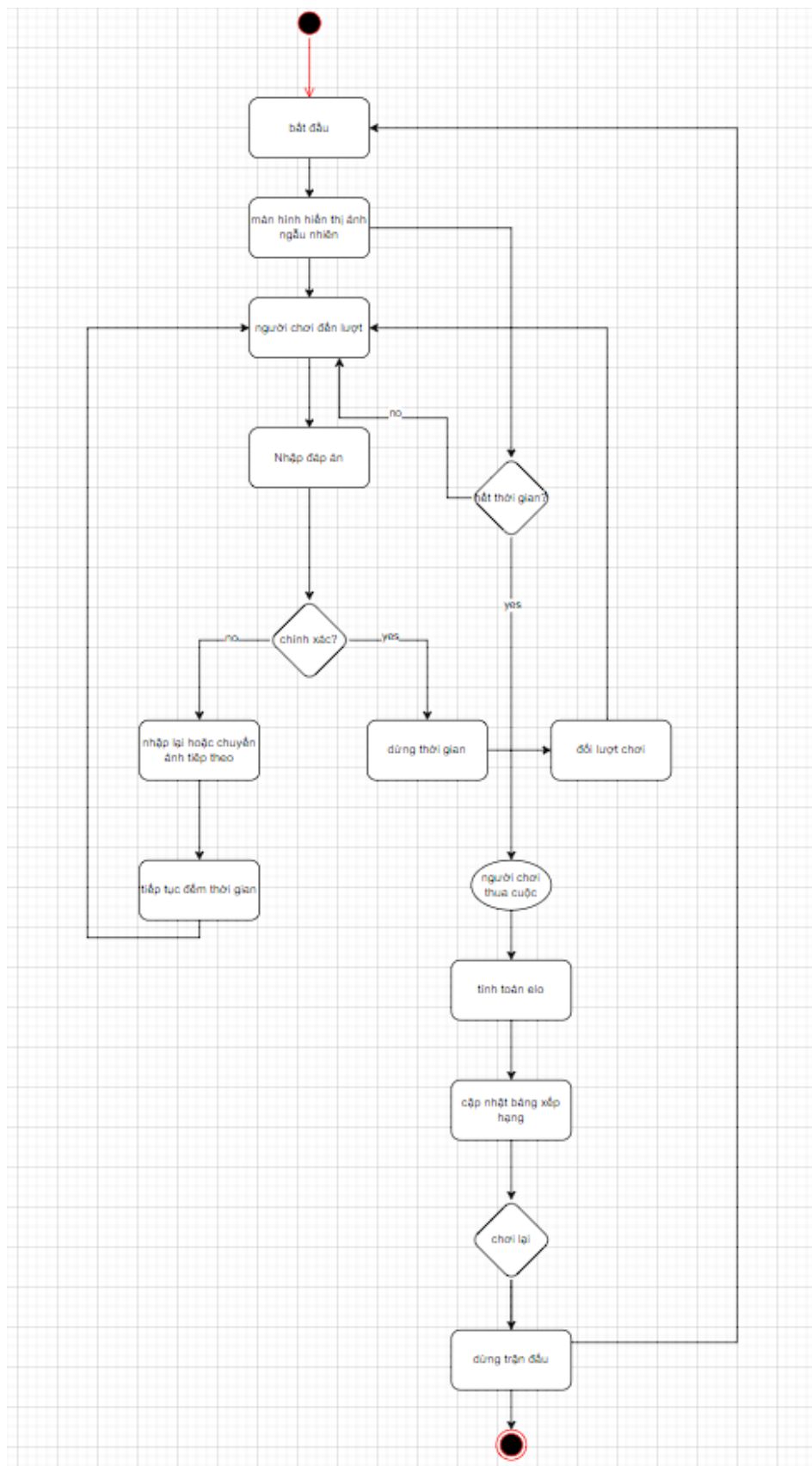
- Mô tả Use Case

Use Case	Chơi ở chế độ 1vs1
Mục tiêu	Hai người chơi đối đầu trực tiếp theo luật chơi của hệ thống
Tác Nhân	Người chơi 1, Người chơi 2

Tiền điều kiện	Hai người chơi đã tham gia phòng chơi và chọn chế độ chơi 1vs1
Kết quả đầu ra	Trận đấu kết thúc, xác định người thắng thua và cập nhật điểm Elo
Luồng sự kiện chính	1. Hệ thống hiển thị hình ảnh ngẫu nhiên từ chủ đề đã chọn. 2. Người chơi đến lượt nhập đáp án vào ô trả lời. 3. Nếu trả lời đúng, thời gian dừng lại và lượt chuyển sang đối thủ. 4. Nếu trả lời sai, người chơi có thể trả lời lại hoặc chuyển ảnh khác (thời gian tiếp tục giảm). 5. Trận đấu kết thúc khi một người chơi hết thời gian. Hệ thống tính điểm Elo và hiển thị kết quả.
Luồng sự kiện phụ	Nếu kết nối người chơi bị mất, hệ thống xác định người mất kết nối thua và tính toán kết quả

Bảng 3.7 Mô tả Use Case chức năng chơi ở chế độ 1vs1

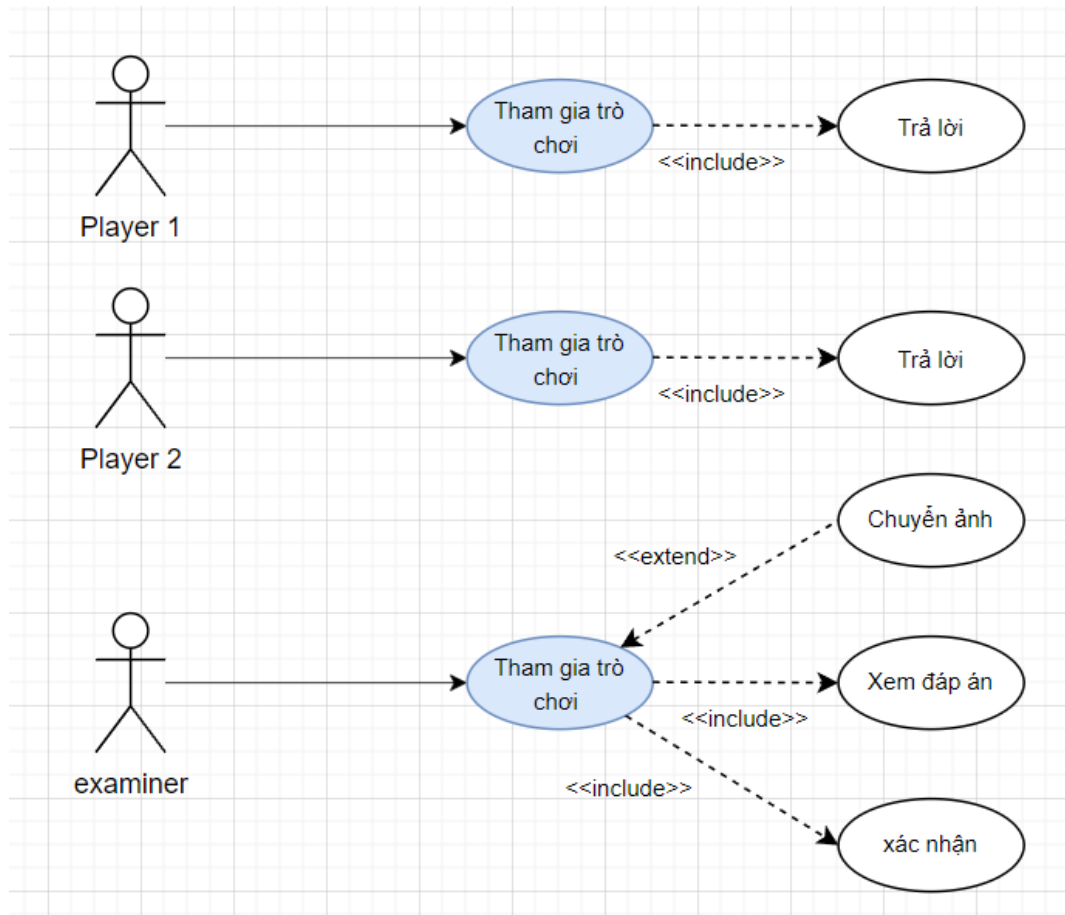
- Biểu đồ activity



Hình 3.14 Biểu đồ hoạt động chức năng chơi ở chế độ 1 vs 1

3.2.8 Chức năng chơi ở chế độ có giám khảo

- Biểu đồ Use Case



Hình 3.15 Use Case chức năng chơi ở chế độ có giám khảo

- Mô tả Use Case

Use Case	Chơi ở chế độ có giám khảo
Mục tiêu	Hai người chơi thi đấu với sự xác định đáp án đúng/sai từ giám khảo
Tác Nhân	Người chơi 1, Người chơi 2, Giám khảo
Tiền điều kiện	Hai người chơi và giám khảo đã tham gia phòng chơi, chọn chế độ có giám khảo
Kết quả đầu ra	Trận đấu kết thúc, xác định người thắng thua và cập nhật điểm Elo
Luồng sự kiện chính	1. Hệ thống hiển thị hình ảnh ngẫu nhiên từ chủ đề đã chọn. 2. Người chơi đến lượt trả lời bằng lời nói với giám khảo. 3. Giám khảo xem đáp án đúng và nhấn nút "Đúng" hoặc "Sai".

	4. Nếu đúng, thời gian dừng lại và lượt chuyển sang đối thủ. Nếu sai, người chơi có thể thử lại hoặc chuyển ảnh khác. 5. Trận đấu kết thúc khi một người chơi hết thời gian. Hệ thống tính điểm Elo và hiển thị kết quả.
Luồng sự kiện phụ	Nếu giám khảo rời phòng, hệ thống tự động kết thúc trận đấu và ghi lại kết quả chưa hoàn chỉnh.

Bảng 3.8 Mô tả Use Case chức năng chơi ở chế độ có giám khảo

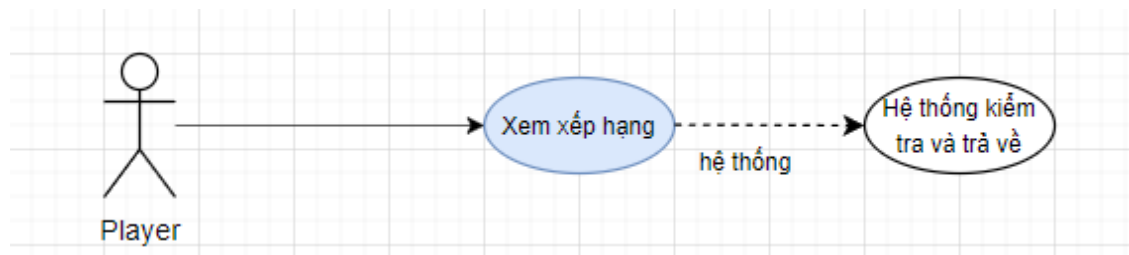
- Biểu đồ activity



Hình 3.16 Biểu đồ hoạt động chức năng chơi ở chế độ có giám khảo

3.2.9 Chức năng bảng xếp hạng

- Biểu đồ Use Case



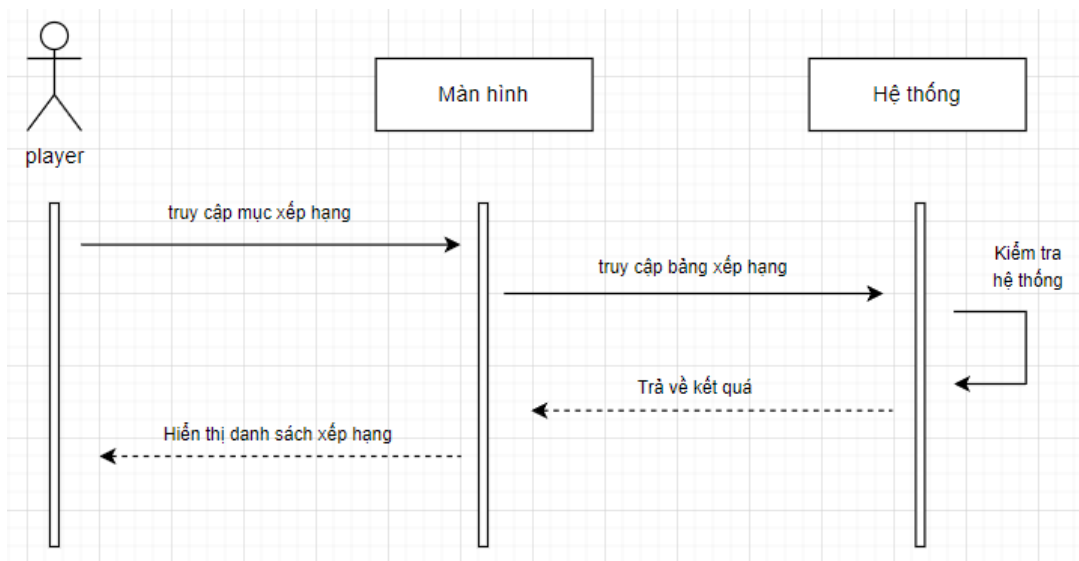
Hình 3.17 Use Case chức năng bảng xếp hạng

- Mô tả Use Case

Use Case	Bảng xếp hạng
Mục tiêu	Người chơi xem thứ hạng của mình và các người chơi khác trong hệ thống
Tác Nhân	Người chơi
Tiền điều kiện	Người chơi đã đăng nhập và có quyền truy cập vào bảng xếp hạng
Kết quả đầu ra	Bảng xếp hạng hiển thị danh sách người chơi theo thứ tự điểm Elo giảm dần
Luồng sự kiện chính	1. Người chơi truy cập mục "Bảng xếp hạng" trên giao diện trang chủ. 2. Hệ thống lấy dữ liệu điểm Elo từ cơ sở dữ liệu. 3. Hệ thống sắp xếp người chơi theo Elo và hiển thị danh sách xếp hạng.
Luồng sự kiện phụ	Nếu cơ sở dữ liệu lỗi, hệ thống hiển thị thông báo và yêu cầu thử lại sau.

Bảng 3. 9 Mô tả Use Case chức năng bảng xếp hạng

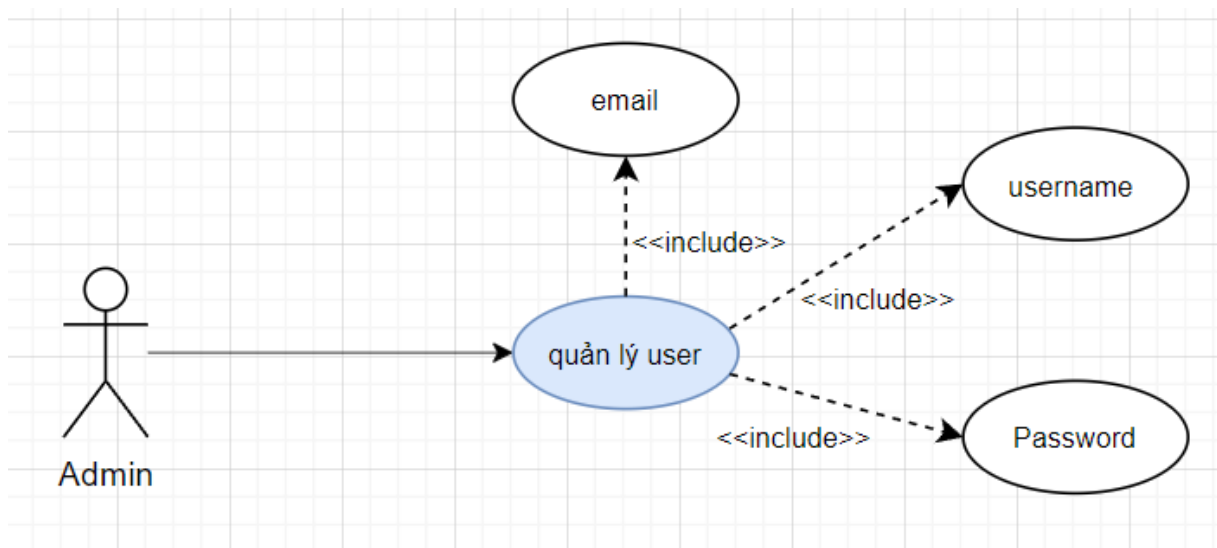
- Biểu đồ tuần tự



Hình 3.18 Biểu đồ hoạt động chức năng bảng xếp hạng

3.2.10 Chức năng quản lý user ở CMS

- Biểu đồ Use Case



Hình 3. 19 Use Case chức năng quản lý user

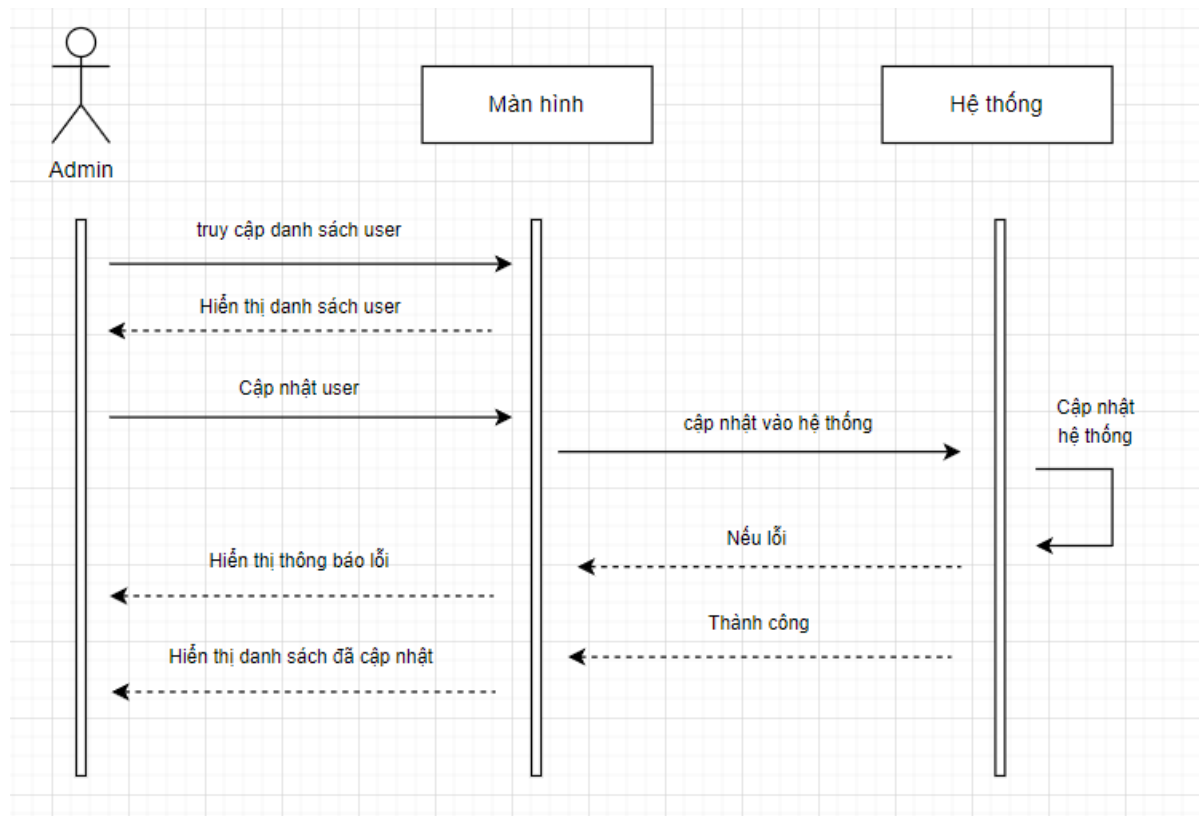
- Mô tả Use Case

Use Case	Quản lý user ở CMS
Mục tiêu	Quản trị viên quản lý danh sách người dùng trong hệ thống
Tác Nhân	Quản trị viên đã đăng nhập vào CMS và có quyền quản trị
Tiền điều kiện	Danh sách người dùng được cập nhật vào cơ sở dữ liệu

Kết quả đầu ra	Danh sách người dùng được cập nhật vào cơ sở dữ liệu
Luồng sự kiện chính	1. Quản trị viên truy cập mục quản lý người dùng. 2. Quản trị viên chọn chỉnh sửa email, username, đổi mật khẩu, hoặc xóa tài khoản. 3. Hệ thống kiểm tra và lưu thay đổi vào cơ sở dữ liệu.
Luồng sự kiện phụ	Nếu thông tin chỉnh sửa không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu nhập lại.

Bảng 3. 10 Mô tả Use Case chức năng quản lý user

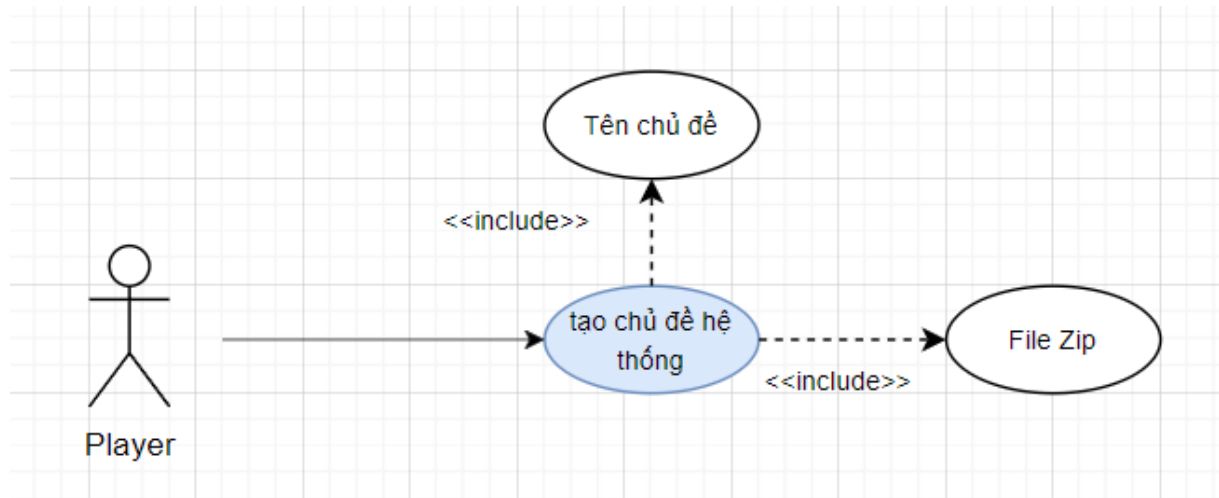
- Biểu đồ tuần tự



Hình 3.20 Biểu đồ hoạt động chức năng quản lý user

3.2.11 Chức năng tạo chủ đề hệ thống

- Biểu đồ Use Case



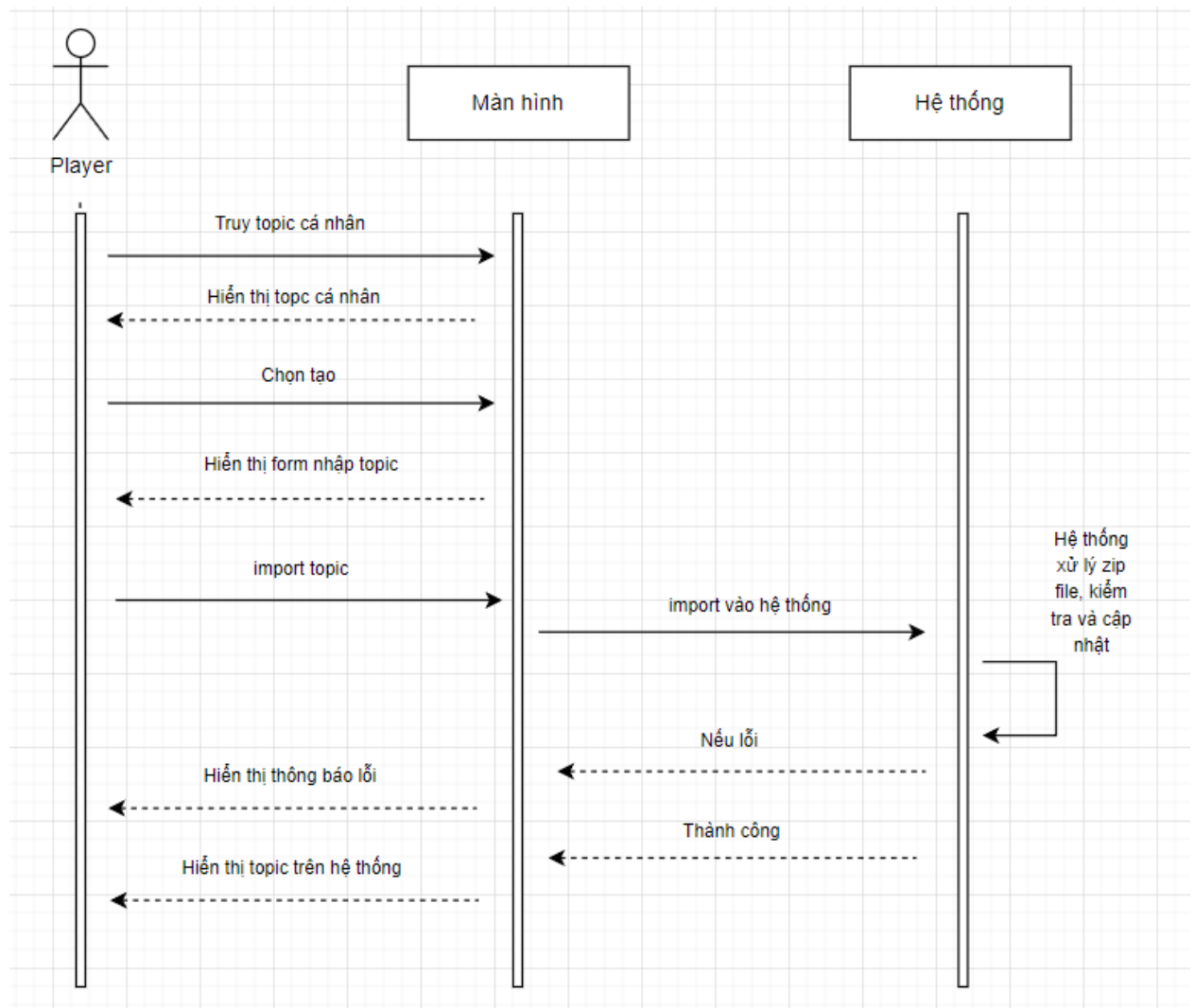
Hình 3.21 Use Case chức năng tạo chủ đề hệ thống

- Mô tả Use Case

Use Case	Tạo chủ đề hệ thống
Mục tiêu	Quản trị viên tạo các topic chung để người chơi sử dụng
Tác Nhân	Quản trị viên
Tiền điều kiện	Quản trị viên đã đăng nhập vào CMS và có quyền quản trị
Kết quả đầu ra	Topic mới được tạo và lưu vào cơ sở dữ liệu
Luồng sự kiện chính	1. Quản trị viên truy cập mục quản lý topic. 2. Nhập tên topic, tải lên file zip chứa hình ảnh và đáp án. 3. Hệ thống kiểm tra tính hợp lệ, giải nén và lưu thông tin vào cơ sở dữ liệu. 4. Hệ thống hiển thị danh sách topic đã được cập nhật.
Luồng sự kiện phụ	Nếu file zip không hợp lệ, hệ thống hiển thị thông báo lỗi và yêu cầu thử lại.

Bảng 3. 11 Mô tả Use Case chức năng tạo chủ đề hệ thống

- Biểu đồ tuần tự

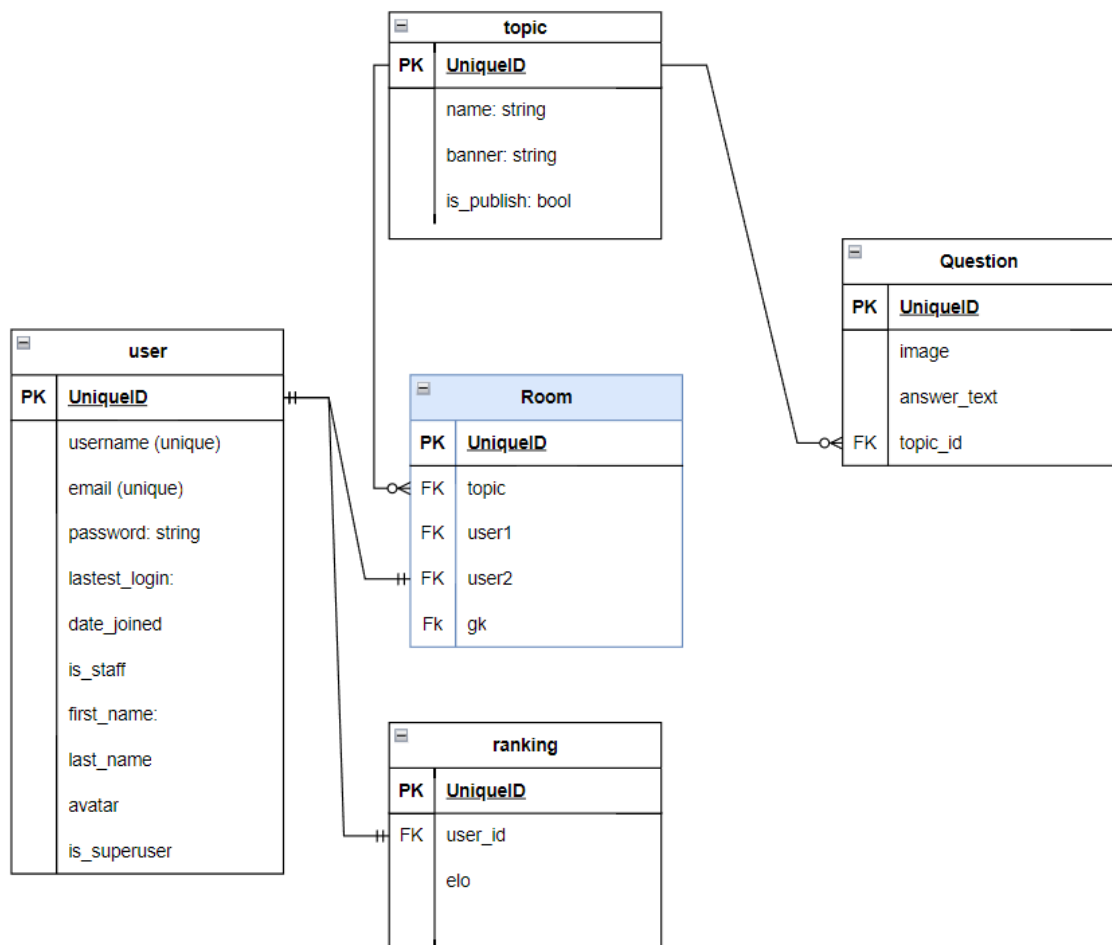


Hình 3.22 Biểu đồ hoạt động chức năng tạo chủ đề hệ thống

Hình 3.22

3.3. Thiết kế cơ sở dữ liệu

3.3.1. Mô hình dữ liệu quan hệ



Hình 3.23 Mô hình dữ liệu quan hệ

3.3.2 Danh mục các bảng cơ sở dữ liệu

STT	Tên bảng	Mô tả
1	User	Lưu trữ thông tin người dùng
2	Topic	Lưu trữ thông tin về các chủ đề
3	Question	Lưu trữ thông tin câu hỏi liên quan đến chủ đề
4	Room	Lưu trữ thông tin về phòng chơi
5	Ranking	Lưu trữ thông tin xếp hạng elo hệ thống

Bảng 3. 12 Mô tả các bảng cơ sở dữ liệu

Tên trường	Kiểu dữ liệu	Key	Ghi chú
UniqueID	String	Primary Key	ID duy nhất của người dùng
username	String	Unique	Tên đăng nhập
email	String	Unique	Email của người dùng
password	String		Mật khẩu
lastest_login	DateTime		Lần đăng nhập gần nhất
date_joined	DateTime		Ngày tham gia
is_staff	Boolean		Xác định quyền quản trị viên
is_superuser	Boolean		Quyền cao nhất
avatar	String		URL ảnh đại diện

Bảng 3. 13 Mô tả bảng Use

Tên trường	Kiểu dữ liệu	Key	Ghi chú
UniqueID	String	Primary Key	ID chủ đề
name	String		Tên chủ đề
banner	String		Hình ảnh đại diện chủ đề
is_publish	Boolean		Trạng thái công khai

Bảng 3. 14 Mô tả bảng Topic

Tên trường	Kiểu dữ liệu	Key	Ghi chú
UniqueID	String	Primary Key	ID câu hỏi
image	String		Tên Hình ảnh minh họa
answer_text	String		Nội dung câu hỏi
topic_id	String	Foreign Key	Liên kết với bảng Topic

Bảng 3. 15 Mô tả bảng question

Tên trường	Kiểu dữ liệu	Key	Ghi chú
UniqueID	String	Primary Key	ID phòng chơi
topic	String	Foreign Key	Chủ đề của phòng

User1	String	Foreign Key	Người chơi 1
User2	String	Foreign Key	Người chơi 2
gk	String	Foreign Key	Giám khảo (nếu có)

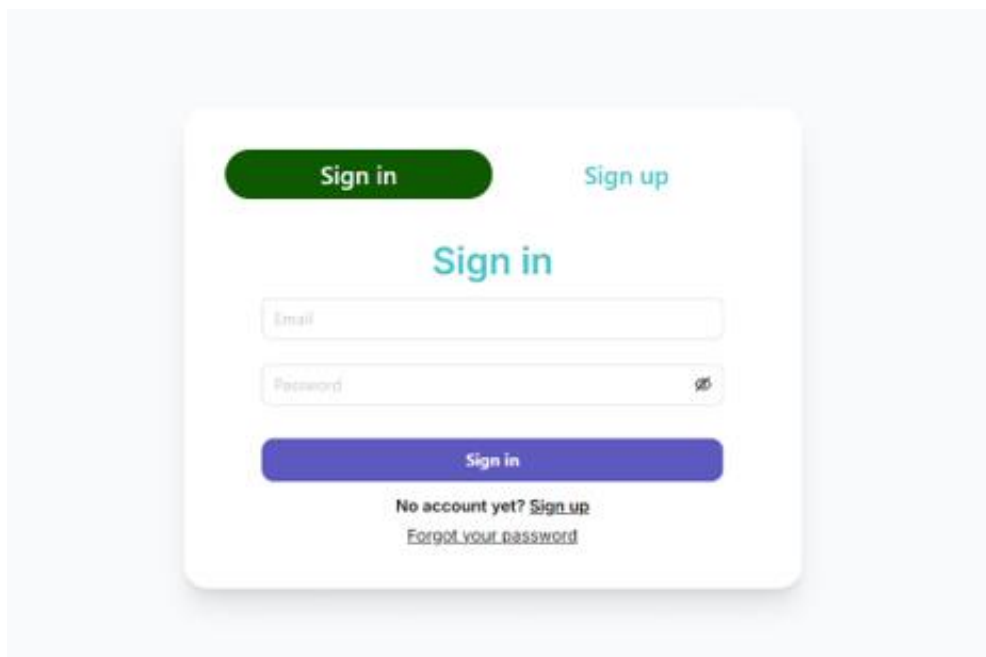
Bảng 3.16 Mô tả bảng room

Tên trường	Kiểu dữ liệu	Key	Ghi chú
UniqueID	String	Primary Key	ID bảng xếp hạng
User_id	String	Foreign Key	ID người chơi
elo	String		Điểm xếp hạng

Bảng 3.17 Mô tả bảng ranking

3.4. Thiết kế giao diện

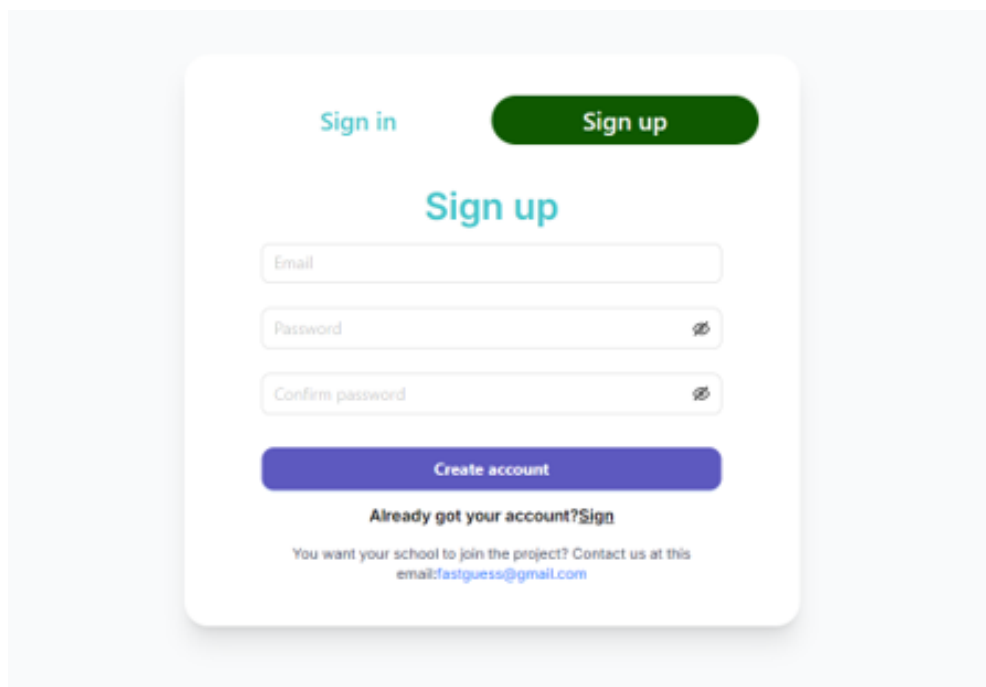
3.4.1. Giao diện đăng nhập



The image shows a sign-in form with a white background and rounded corners. At the top, there are two buttons: a green 'Sign in' button and a blue 'Sign up' button. Below these, the text 'Sign in' is displayed in blue. The form contains two input fields: 'Email' and 'Password'. The 'Password' field has a small eye icon to its right. Below the input fields is a large blue 'Sign in' button. At the bottom, there is a link 'No account yet? Sign up' and a link 'Forgot your password'.

Hình 3.24 Giao diện đăng nhập

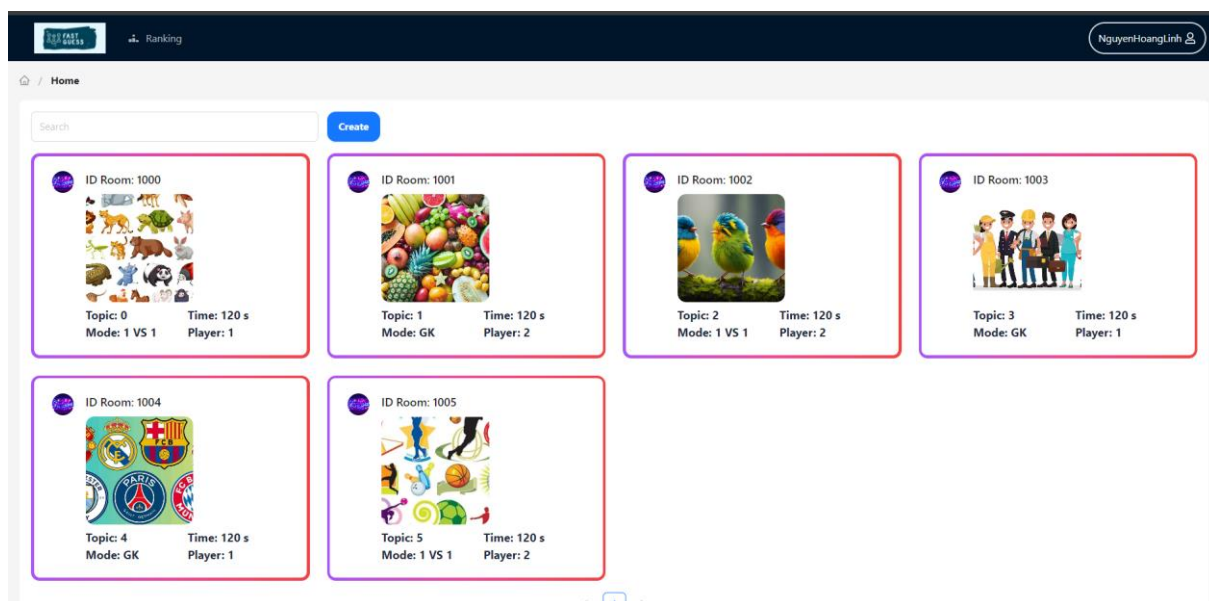
3.4.2. Giao diện đăng ký



The image shows a sign-up form with a white background and rounded corners. At the top, there are two buttons: a blue 'Sign in' button and a green 'Sign up' button. Below these, the text 'Sign up' is displayed in blue. The form contains three input fields: 'Email', 'Password', and 'Confirm password'. Each input field has a small eye icon to its right. Below the input fields is a large blue 'Create account' button. At the bottom, there is a link 'Already got your account? Sign in' and a link 'You want your school to join the project? Contact us at this email: fastguess@gmail.com'.

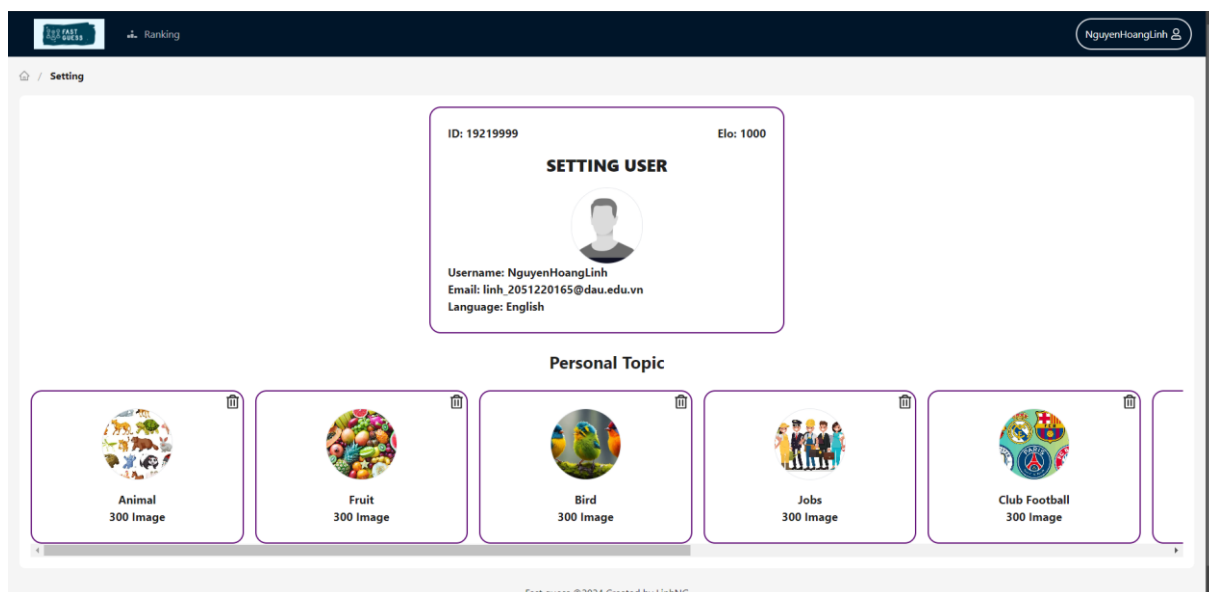
Hình 3.25 Giao diện đăng ký

3.4.3. Giao diện Trang chủ



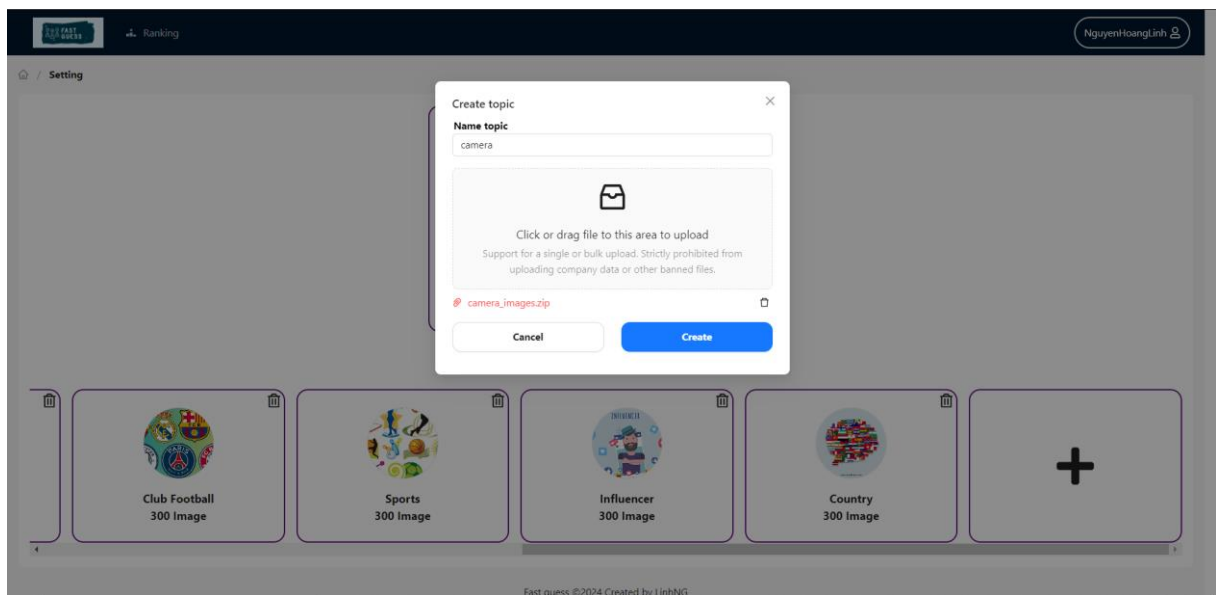
Hình 3.26 Giao diện Trang chủ

3.4.4. Giao diện setting user



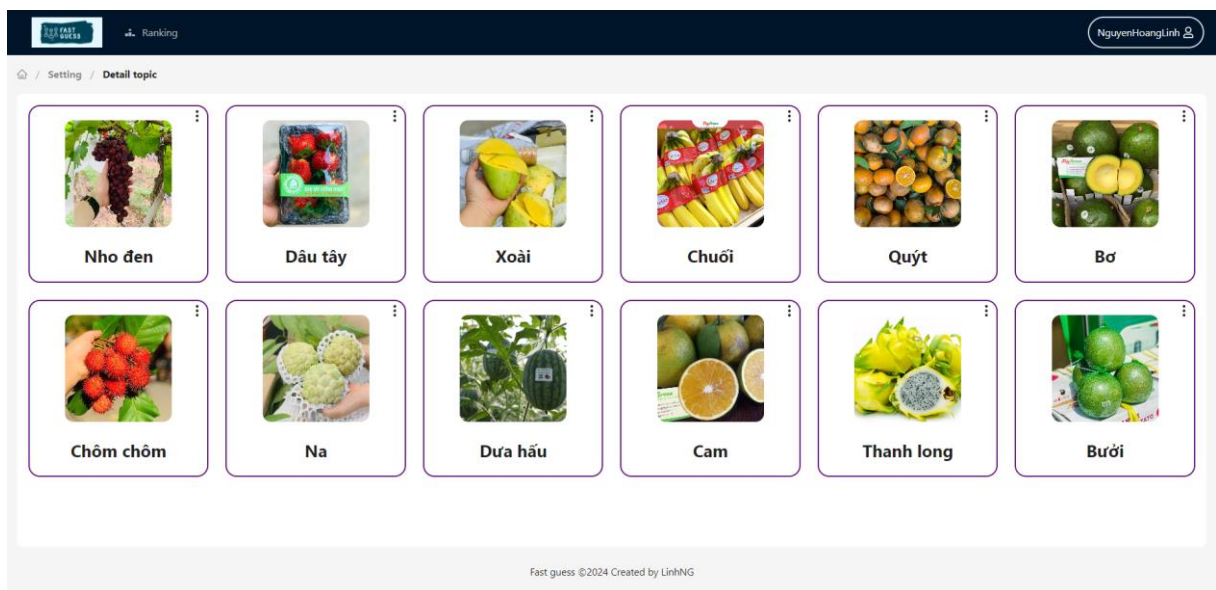
Hình 3.27 Giao diện setting user

3.4.5. Giao diện tạo topic



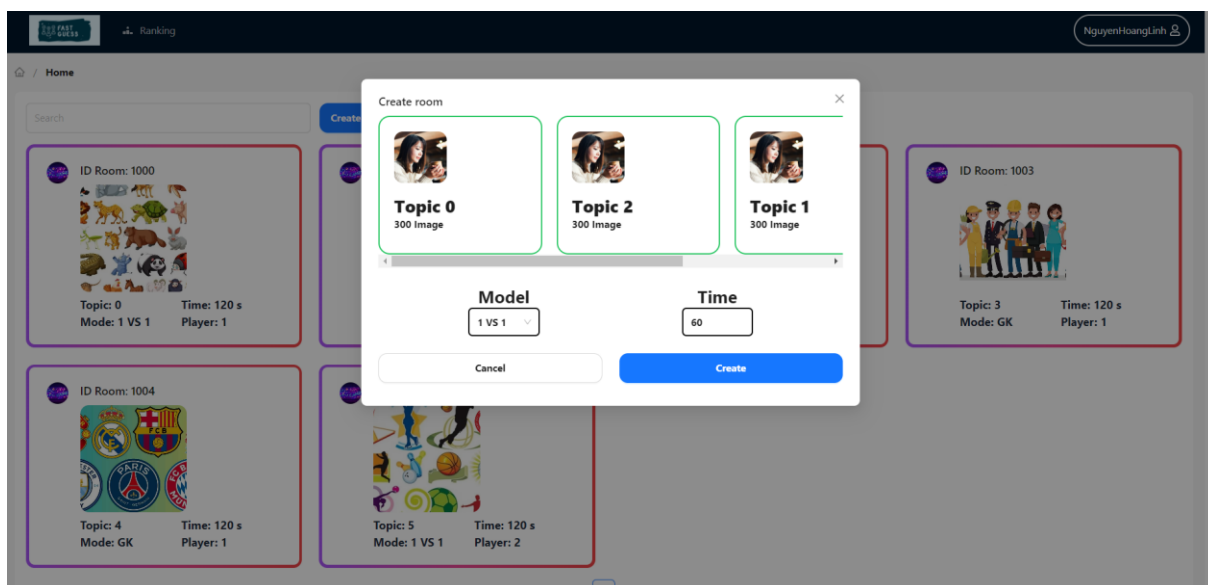
Hình 3.28 Giao diện tạo topic

3.4.6. Giao diện detail topic

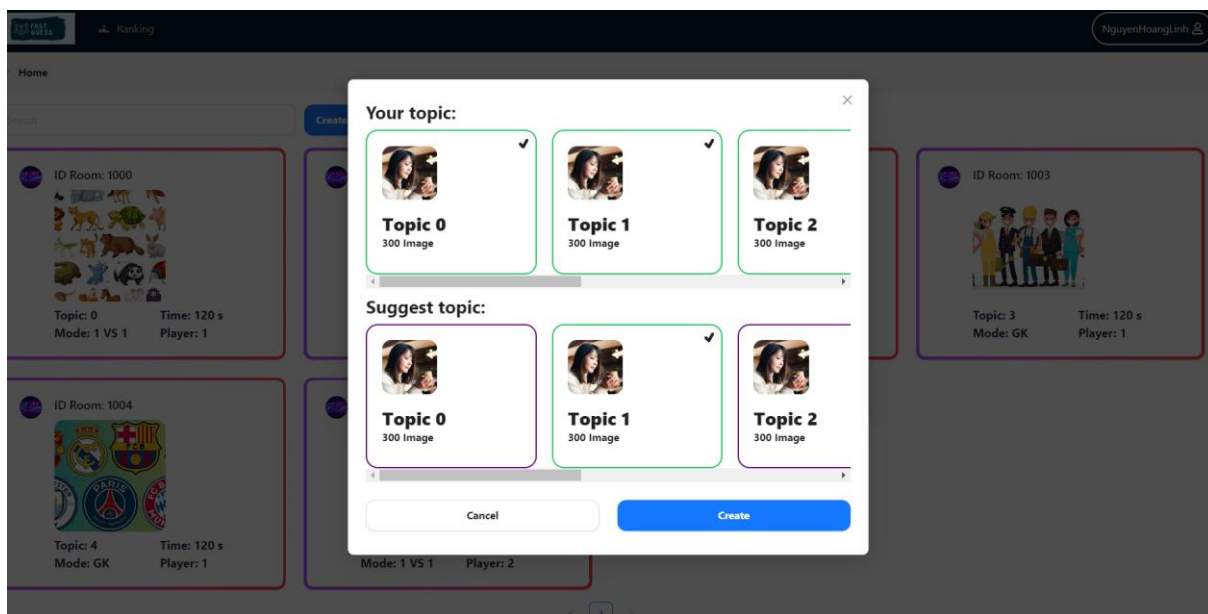


Hình 3.29 Giao diện chi tiết topic

3.4.7. Giao diện tạo phòng

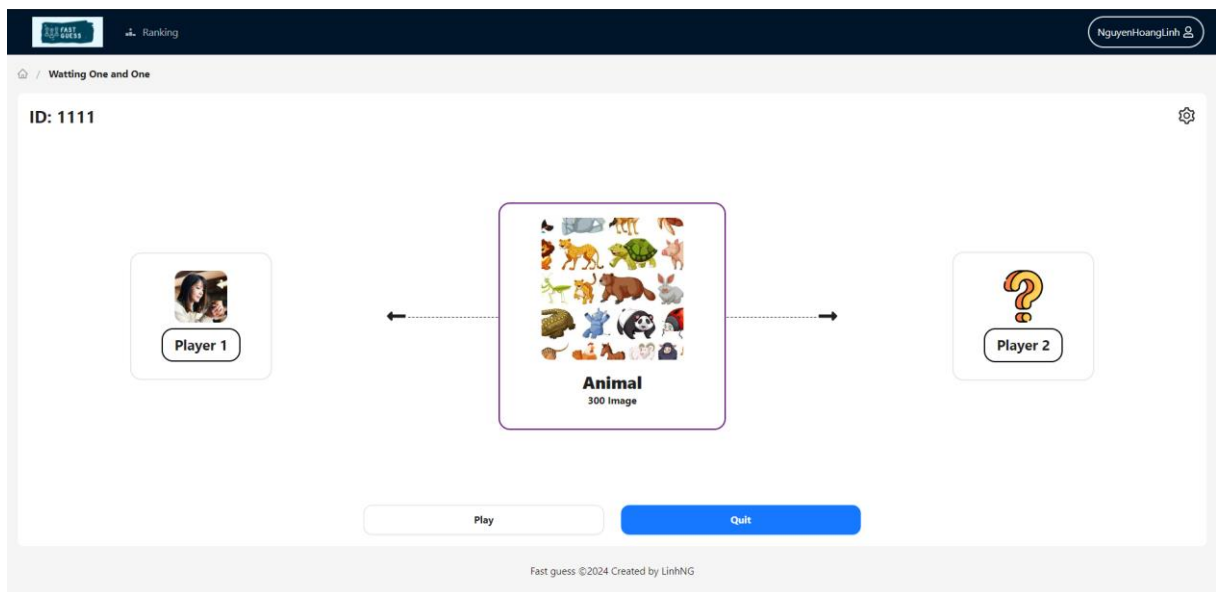


Hình 3.30 Giao diện tạo phòng

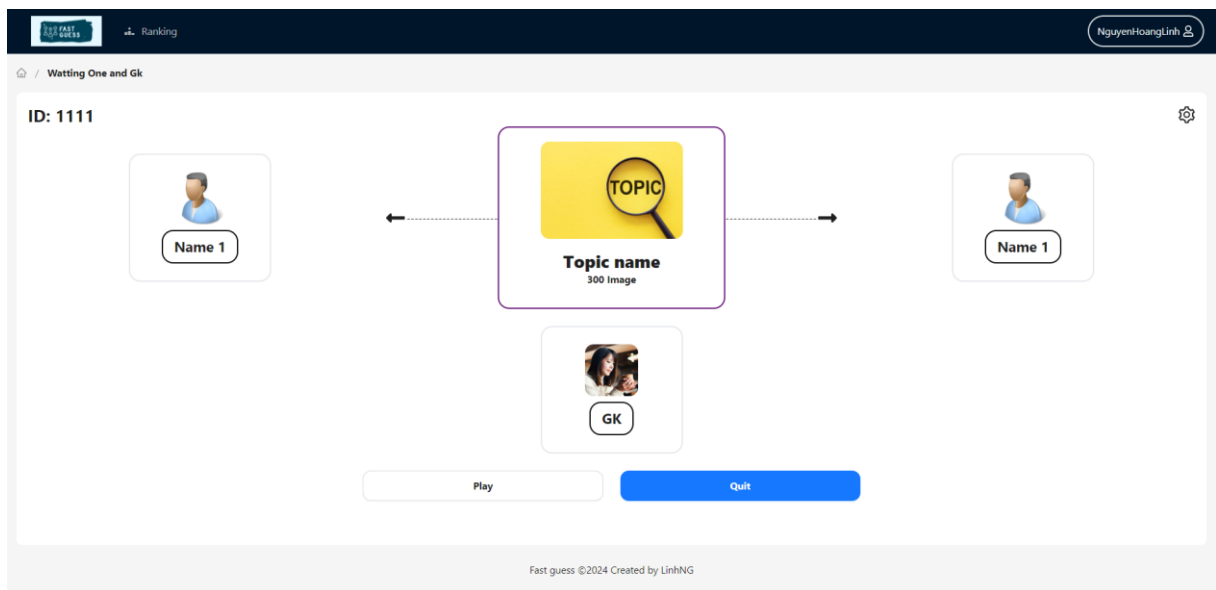


Hình 3.31 Giao diện chọn chủ đề

3.4.8. Giao diện phòng chờ

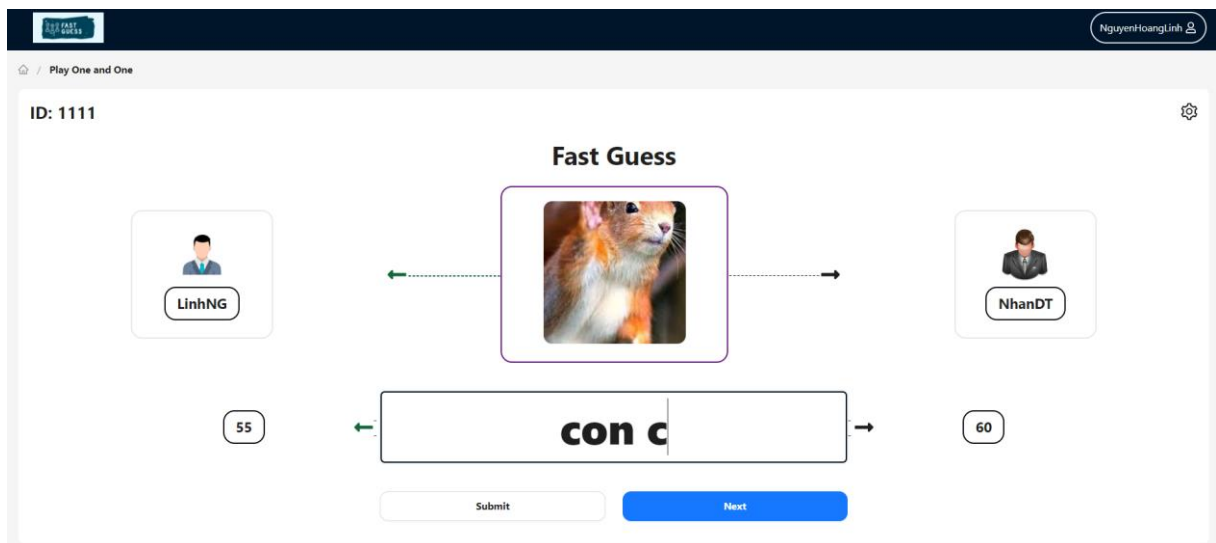


Hình 3.32 Giao diện phòng chờ chế độ 1vs 1



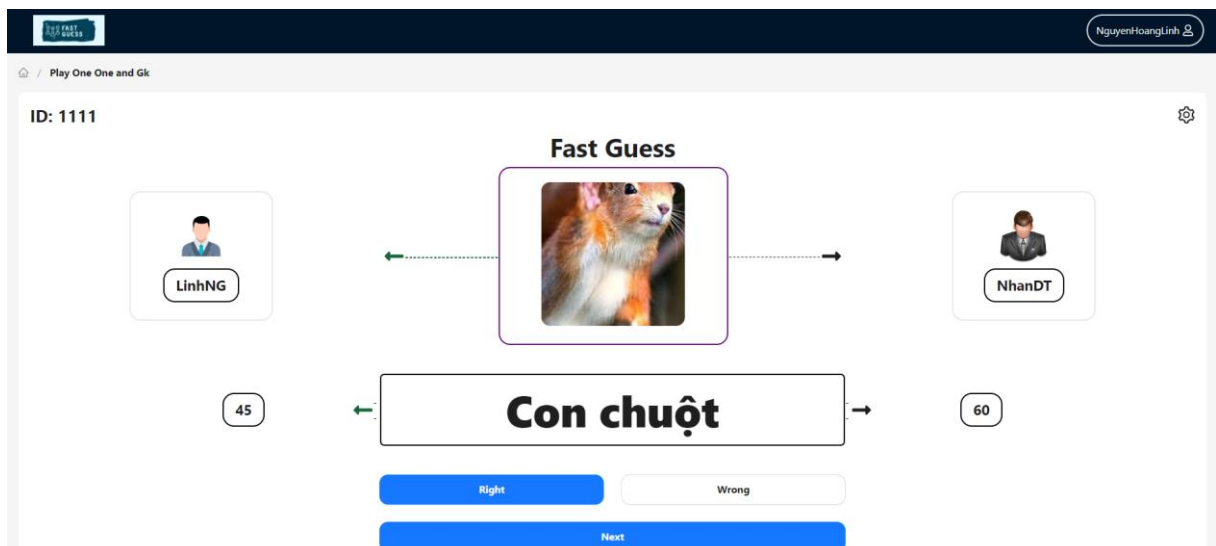
Hình 3.33 Giao diện phòng chờ chế độ có giám khảo

3.4.9. Giao diện player ở chế độ 1vs1



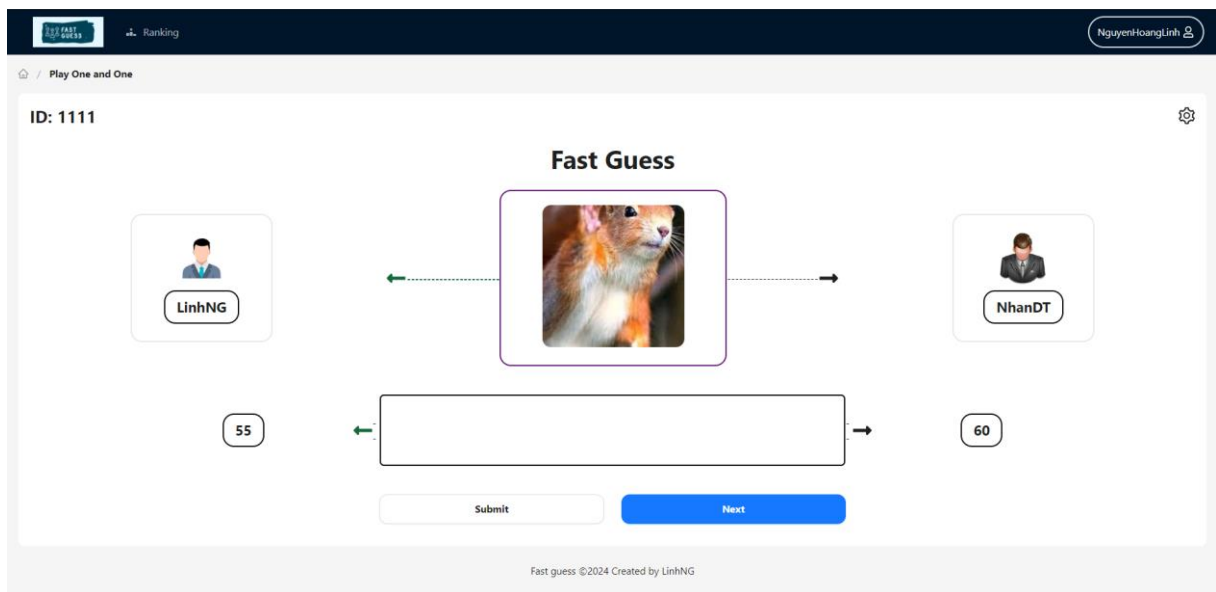
Hình 3.34 Giao diện chơi ở chế độ 1vs 1

3.4.10. Giao diện giám khảo



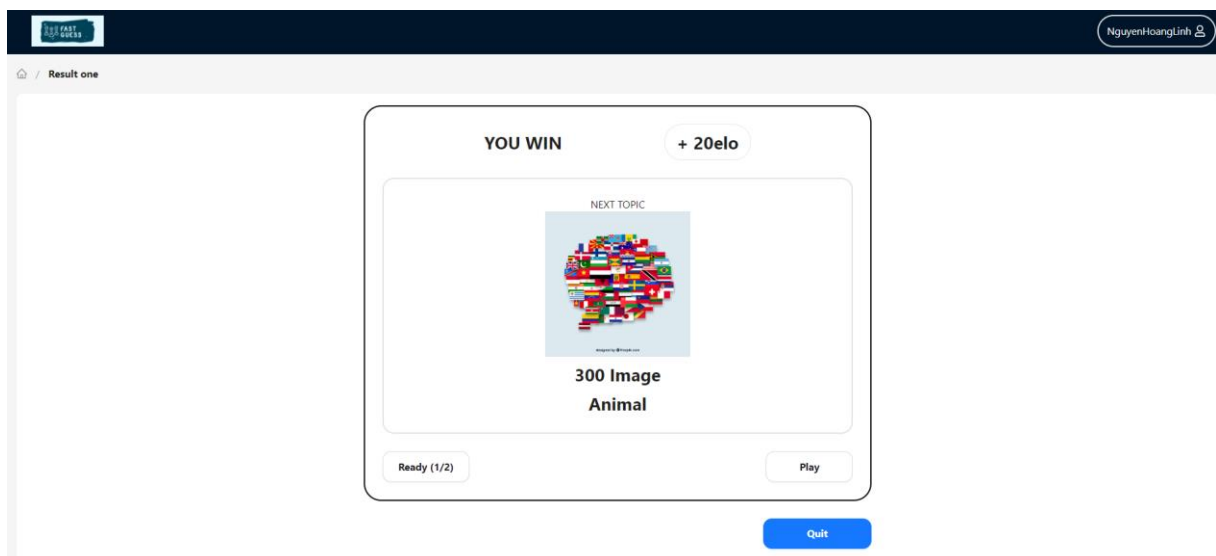
Hình 3.35 Giao diện của giám khảo

3.4.11. Giao diện player ở chế độ có giám khảo



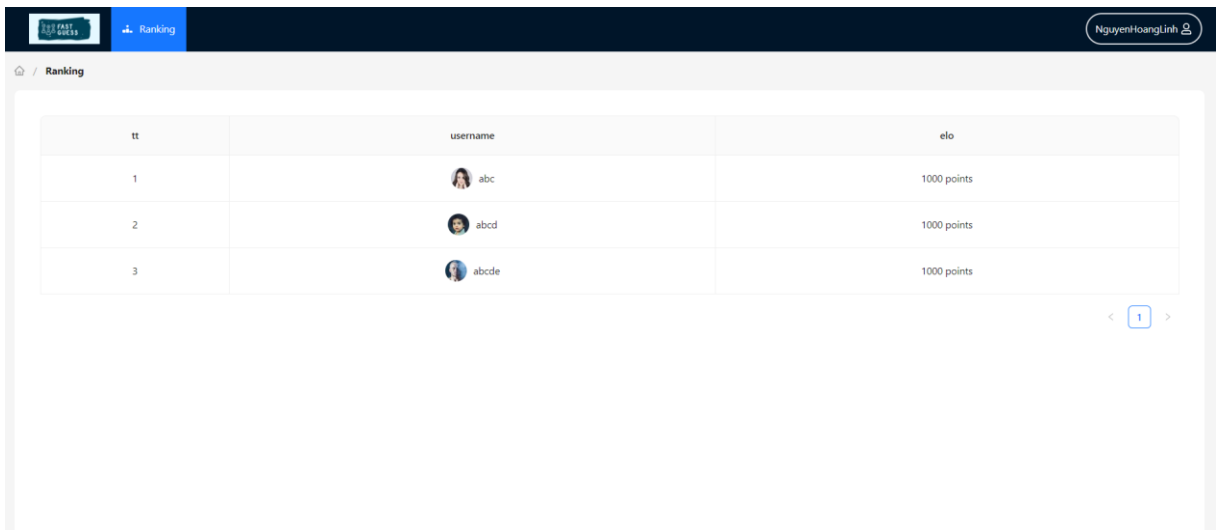
Hình 3.36 Giao diện chơi ở chế độ có giám khảo

3.4.12. Giao diện kết thúc game đấu



Hình 3.37 Giao diện kết thúc ván đấu

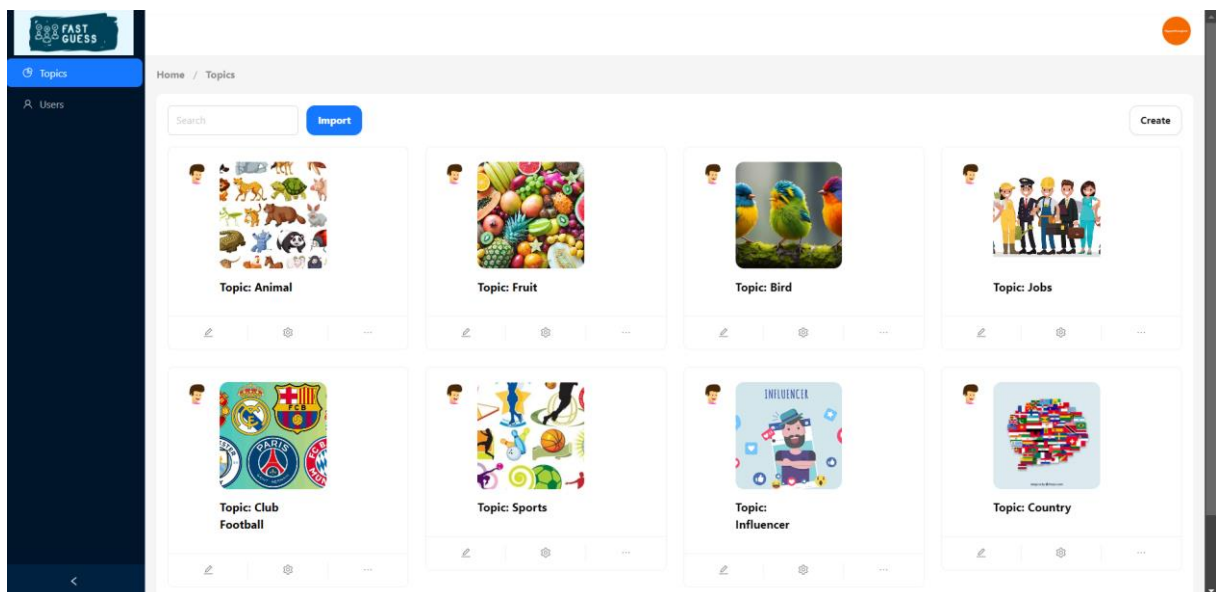
3.4.13. giao diện bảng xếp hạng



tt	username	elo
1	abc	1000 points
2	abcd	1000 points
3	abcde	1000 points

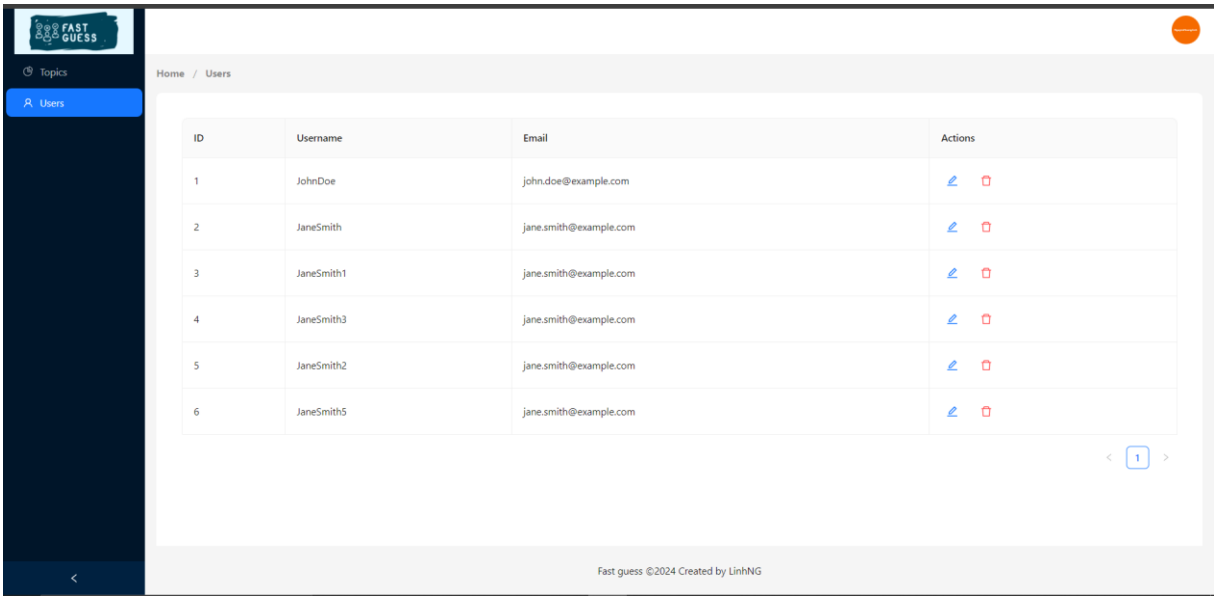
Hình 3.38 Giao diện bảng xếp hạng

3.4.11 Giao diện cms-topics



Hình 3.39 Giao diện CMS-topic

3.4.12 Giao diện cms-user



Hình 3.40 Giao diện CMS-user

Chương 4: CÀI ĐẶT, TRIỂN KHAI PHÁT HÀNH

4.1 Cài đặt và triển khai hệ thống

Cài đặt python

- `sudo apt update`
- `sudo apt install python3 python3-pip python3-venv`

Tạo và kích hoạt môi trường ảo

- `python3 -m venv venv`
- `source venv/bin/activate`

Cài đặt các thư viện cần thiết

- `pip install -r requirements.txt`

Chạy migration

- `python manage.py makemigrations`
- `python manage.py migrate`

Tạo tài khoản quản trị viên

- `python manage.py createsuperuser`

Chạy thử ứng dụng

- `python manage.py runserver`

4.2 Áp dụng socket vào quá trình vận hành trò chơi

4.2.1 Connect user

```

10 class PvPGameConsumer(AsyncWebsocketConsumer): 1 usage
11     async def connect(self):
12         self.room_name = self.scope['url_route']['kwargs']['room_name']
13         self.room_group_name = f"game_{self.room_name}"
14         self.user = self.scope["user"]
15
16         if not self.user.is_authenticated:
17             await self.close()
18             return
19
20         self.redis = await aioredis.from_url(settings.REDIS_URL, decode_responses=True)
21         await self.channel_layer.group_add(self.room_group_name, self.channel_name)
22         await self.accept()
23
24         await self.redis.sadd(f"room:{self.room_name}:players", self.user.username)
25
26         # Bắt đầu game khi đủ 2 người chơi
27         players = await self.redis.smembers(f"room:{self.room_name}:players")
28         if len(players) == 2:
29             await self.start_game(players)
30

```

Hình 4.1 Hàm connect user

4.2.2. Start game

```

59
60     async def start_game(self, players): 1 usage
61         players = list(players)
62         random.shuffle(players)
63         first_player = players[0]
64
65         await self.redis.rpush(f"room:{self.room_name}:questions", *questions)
66         await self.redis.rpush(f"room:{self.room_name}:answers", *answers)
67         # Lưu chỉ số câu hỏi hiện tại
68         await self.redis.set(f"room:{self.room_name}:current_question_index", 0)
69
70         # Khởi tạo thời gian còn lại (60 giây mỗi người)
71         for player in players:
72             await self.redis.hset(f"room:{self.room_name}:time_left", player, 60)
73
74         await self.redis.set(f"room:{self.room_name}:turn", first_player)
75
76         await self.channel_layer.group_send(
77             self.room_group_name,
78             {
79                 "type": "game_start",
80                 "players": players,
81                 "message": f"Game started! {first_player} plays first.",
82                 "current_turn": first_player
83             }
84         )
85
86         await self.send_question(first_player)
87
88         # cho chạy ngầm việc đếm ngược thời gian
89         asyncio.create_task(self.start_timer(first_player))
90

```

Hình 4.2 Hàm bắt đầu game đấu

4.2.3. Tương tác player với nhau

```

async def handle_submission(self, data): 1 usage
    print(self.user.username, data['answer'])
    players = await self.redis.smembers(f"room:{self.room_name}:players")
    current_turn = await self.redis.get(f"room:{self.room_name}:turn")
    question_index = int(await self.redis.get(f"room:{self.room_name}:current_question_index"))
    correct_answer = await self.redis.lindex(f"room:{self.room_name}:answers", question_index)

    if self.user.username != current_turn:
        await self.send(text_data=json.dumps({
            "type": "error",
            "message": "Not your turn!"
        }))
        return

    user_answer = data.get("answer")
    print(correct_answer)
    if user_answer.lower().strip() != correct_answer.lower().strip():
        await self.send(text_data=json.dumps({
            "type": "error",
            "message": "Wrong answer! Try again."
        }))
        return

    # Lưu thời gian còn lại trước khi chuyển lượt
    time_left = await self.redis.hget(f"room:{self.room_name}:time_left", current_turn)
    await self.redis.hset(f"room:{self.room_name}:time_left", current_turn, time_left)

    opponent = [player for player in players if player != current_turn][0]

    await self.channel_layer.group_send(
        self.room_group_name,
        {
            "type": "update_turn",
            "message": f"{self.user.username} answered correctly!",
            "correct_user": self.user.username,
            "next_turn": opponent
        }
    )

```

Hình 4.3 Hàm quản lý 2 player tương tác

4.2.4. Quản lý câu hỏi

```

async def handle_submission(self, data): 1 usage
    )
    await self.next_question()

async def next_question(self): 1 usage
    """Chuyển sang câu hỏi tiếp theo."""
    question_index = int(await self.redis.get(f"room:{self.room_name}:current_question_index"))
    total_questions = await self.redis.llen(f"room:{self.room_name}:questions")

    # Nếu còn câu hỏi, chuyển sang câu hỏi tiếp theo
    if question_index + 1 < total_questions:
        await self.redis.incr(f"room:{self.room_name}:current_question_index")
        players = await self.redis.smembers(f"room:{self.room_name}:players")
        current_turn = await self.redis.get(f"room:{self.room_name}:turn")
        next_player = [p for p in players if p != current_turn][0]

        # Gửi câu hỏi tiếp theo
        await self.send_question(next_player)

        # Chuyển lượt
        await self.redis.set(f"room:{self.room_name}:turn", next_player)
        asyncio.create_task(self.start_timer(next_player))
    else:
        # Hết câu hỏi => kết thúc game
        await self.channel_layer.group_send(
            self.room_group_name,
            {
                "type": "game_over",
                "winner": "Draw", # Hoặc logic khác để xác định người thắng
                "reason": "No more questions"
            }
        )
        await self.end_game()

async def start_timer(self, player): 2 usages
    """Tiếp tục đếm ngược thời gian còn lại cho người chơi."""

```

Hình 4.4 Hàm quản lý chuyển ảnh

4.2.5. Quản lý thời gian

```

190
191     async def start_timer(self, player): 2 usages
192         """Tiếp tục đếm ngược thời gian còn lại cho người chơi."""
193         time_left = int(await self.redis.hget(f"room:{self.room_name}:time_left", player))
194
195         for i in range(time_left):
196             await asyncio.sleep(1)
197             current_turn = await self.redis.get(f"room:{self.room_name}:turn")
198             if current_turn != player:
199                 # Nếu lượt chơi thay đổi, dừng đếm ngược
200                 return
201
202             time_left -= 1
203             await self.redis.hset(f"room:{self.room_name}:time_left", player, time_left)
204
205         if time_left == 0:
206             # Hết thời gian
207             await self.timeout_player(player)
208             return
209
210     async def timeout_player(self, player): 1 usage
211         players = await self.redis.smembers(f"room:{self.room_name}:players")
212         opponent = [p for p in players if p != player][0]
213
214         await self.channel_layer.group_send(
215             self.room_group_name,
216             {
217                 "type": "game_over",
218                 "winner": opponent,
219                 "loser": player,
220                 "reason": "timeout"
221             }
222         )
223
224         await self.end_game()
225

```

Hình 4.5 Hàm quản lý thời gian chơi

4.2.6. Quản lý disconnection

```

32
33 @
34     async def disconnect(self, close_code):
35         try:
36             await self.channel_layer.group_discard(self.room_group_name, self.channel_name)
37             await self.redis.srem(f"room:{self.room_name}:players", self.user.username)
38
39             players = await self.redis.smembers(f"room:{self.room_name}:players")
40             if len(players) == 1:
41                 remaining_player = list(players)[0]
42                 await self.channel_layer.group_send(
43                     self.room_group_name,
44                     {
45                         "type": "game_over",
46                         "winner": remaining_player,
47                         "loser": self.user.username,
48                         "reason": "opponent_left"
49                     }
50                 )
51                 await self.end_game()
52
53             elif len(players) == 0:
54                 await self.redis.delete(f"room:{self.room_name}:players")
55                 await self.redis.delete(f"room:{self.room_name}:state")
56                 await self.redis.delete(f"room:{self.room_name}:turn")
57                 await self.redis.delete(f"room:{self.room_name}:time_left")
58             finally:
59                 await self.redis.close()

```

Hình 4.6 Hàm quản lý thời gian chơi

4.3 Triển khai phát hành sản phẩm

4.3.1. Triển khai sản phẩm lên server

Để đảm bảo sản phẩm có thể tiếp cận được người dùng mọi lúc, tôi đã triển khai hệ thống lên một server đám mây. Cụ thể:

Lựa chọn nền tảng: Sử dụng dịch vụ cloud server từ VPS

Cấu hình server:

- Cài đặt môi trường chạy sản phẩm : django frame work
- Cấu hình cơ sở dữ liệu và kết nối với sản phẩm.
- Thiết lập domain và bảo mật (HTTPS).

Triển khai:

- Sử dụng công cụ CI/CD GitHub Actions để tự động hoá quá trình triển khai.
- Đảm bảo bản phát hành luôn đồng bộ với mã nguồn mới nhất

4.3.2. Treo quảng cáo và tìm kiếm người dùng tiềm năng

Để thu hút người dùng và gia tăng mức độ nhận diện sản phẩm, tôi thực hiện các hoạt động marketing như sau:

Treo quảng cáo trực tuyến:

- Sử dụng nền tảng Google Ads, Facebook Ads để chạy quảng cáo có mục tiêu.
- Hướng đến đối tượng người dùng tiềm năng dựa trên hành vi và sở thích.

Hợp tác với các streamer:

- Donate hoặc tài trợ các streamer trên các nền tảng như YouTube, Twitch để họ giới thiệu sản phẩm.
- Xây dựng mối quan hệ để tiếp cận cộng đồng người hâm mộ của họ, từ đó lan tỏa sản phẩm.

4.2.3. Tích hợp Google Analytics

Để theo dõi và phân tích hành vi người dùng, tôi đã tích hợp Google Analytics vào sản phẩm. Quá trình thực hiện như sau:

Thiết lập tài khoản Google Analytics:

- Tạo tài khoản, thiết lập thuộc tính và cấu hình mã theo dõi (tracking code).

Tích hợp vào sản phẩm:

- Thêm mã theo dõi vào ứng dụng hoặc trang web.
- Theo dõi các chỉ số quan trọng như số lượng người dùng, thời gian sử dụng, tỷ lệ thoát (bounce rate), v.v.

Phân tích dữ liệu:

- Sử dụng báo cáo của Google Analytics để đưa ra các quyết định cải tiến sản phẩm.
- Nhận diện các kênh quảng cáo hiệu quả, tối ưu hóa chi phí.

3.2.4. Đánh giá và tối ưu hóa

Sau khi triển khai, tôi liên tục theo dõi phản hồi từ người dùng, điều chỉnh sản phẩm và chiến lược quảng cáo:

- Thu thập phản hồi người dùng thông qua các kênh như email, mạng xã hội.
- Tối ưu hiệu suất server dựa trên lưu lượng truy cập.

- Đánh giá hiệu quả chiến dịch marketing dựa trên các chỉ số thu thập được từ Google Analytics

KẾT LUẬN

Sau thời gian nghiên cứu và phát triển, đồ án Web game mô phỏng trò chơi bật hình nhảy số đã hoàn thành với các kết quả đáng ghi nhận. Những nội dung chính có thể được tổng kết như sau:

I. Kết quả đạt được:

Hệ thống đã hoàn thiện các chức năng cốt lõi như tạo phòng chơi, tham gia phòng chơi, quản lý chủ đề, chế độ chơi 1vs1 và chế độ có giám khảo.

Hệ thống quản trị (CMS) cũng được triển khai để quản lý người dùng và chủ đề hiệu quả.

Sử dụng thành công các công nghệ như Django, NextJS và PostgreSQL để xây dựng một hệ thống hoạt động ổn định và có thể mở rộng.

Đồ án giúp tích lũy nhiều kiến thức về lập trình web, xử lý thời gian thực và quản lý cơ sở dữ liệu.

II. Hạn chế:

Giao diện người dùng tuy đã được tối ưu nhưng vẫn còn tiềm năng cải thiện về trải nghiệm tổng thể.

Chưa có tính năng đa ngôn ngữ để đáp ứng người chơi quốc tế.

Chưa tích hợp đầy đủ các báo cáo thống kê chi tiết cho quản trị viên.

III. Hướng phát triển:

Bổ sung các tính năng mới như: bảng xếp hạng toàn cầu, chế độ chơi nhóm, và các mini-game trong phòng chờ.

Cải thiện giao diện và trải nghiệm người dùng dựa trên phản hồi từ thực tế.

Triển khai thêm các công cụ phân tích dữ liệu người dùng để tối ưu hóa sản phẩm.

Phát triển tính năng đa ngôn ngữ để mở rộng tầm ảnh hưởng của sản phẩm.

Đồ án không chỉ giúp em phát triển kiến thức chuyên môn mà còn là bước đệm quan trọng trong việc áp dụng lý thuyết vào thực tiễn. Những kinh nghiệm này sẽ là hành trang quý báu cho sự nghiệp sau này. Em xin trân trọng mọi ý kiến đóng góp từ thầy cô và các bạn để cải thiện và hoàn thiện hệ thống trong tương lai.

