

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG



MÔN HỌC: LẬP TRÌNH MẠNG CĂN BẢN

BÁO CÁO ĐỒ ÁN CUỐI KỲ

FunDraw

Giảng viên hướng dẫn: ThS. Lê Minh Khánh Hội

Lớp: NT106.P13

Sinh viên thực hiện:

Nguyễn Lộc Tỷ - 23521757

(Nhóm trưởng: 23521757@gm.uit.edu.vn)

Nguyễn Thanh Tùng - 23521744

Đỗ Đức Minh Triết – 23521650

Tạ Quốc Tuấn – 23521728

TP. Hồ Chí Minh, ngày 25 tháng 12 năm 2024

Mục lục

I. GIỚI THIỆU ĐỀ TÀI	3
1. Lý do chọn đề tài	3
2. Mục tiêu đề tài	3
3. Giới thiệu đề tài	3
4. Các tính năng của đề tài	4
II. PHÂN TÍCH THIẾT KẾ HỆ THỐNG	4
1. Sơ đồ kiến trúc tổng quan hệ thống	4
2. Network Stack	5
3. Sơ đồ Use case	7
4. Luồng của ứng dụng	9
Luồng đăng ký:	9
Luồng đăng nhập:	9
Luồng đăng xuất:	10
Luồng quên mật khẩu:	10
Luồng hiển thị thông tin hồ sơ người chơi:	10
Luồng đổi mật khẩu:	11
Luồng tạo phòng:	11
Luồng bắt đầu phòng chơi:	12
Luồng người vẽ chọn từ:	13
Luồng người chơi chat:	14
Luồng tiến trình trò chơi:	14
III. HIỆN THỰC HỆ THỐNG	14
1. Tính năng đăng nhập, đăng ký, quên mật khẩu	14
2. Tính năng hồ sơ người chơi, đổi mật khẩu	16
3. Tính năng trò chơi	18
4. Đặc tả API	20
IV. KẾT LUẬN, HƯỚNG PHÁT TRIỂN	28
1. Kết luận	28
2. Hướng phát triển	29
Bảng phân công công việc	29
Tài liệu tham khảo	32

I. GIỚI THIỆU ĐỀ TÀI

1. Lý do chọn đề tài

Thời đại công nghệ 4.0 hiện nay, việc tương tác trực tiếp trở nên phổ biến. Mọi người, ở mọi độ tuổi, có thể kết nối với nhau thông qua các mạng xã hội, các trò chơi online, v.v... Đồ án trò chơi “vẽ hình đoán chữ” là trò chơi này không chỉ mang lại niềm vui và sự sáng tạo cho người chơi mà còn phát triển kỹ năng giao tiếp và tư duy nhanh. Trong bối cảnh xã hội ngày càng chú trọng các hoạt động nhóm và kết nối xã hội, trò chơi này giúp tạo cơ hội tương tác giữa người chơi, từ đó cải thiện kỹ năng hợp tác, sáng tạo và diễn đạt ý tưởng một cách hiệu quả.

2. Mục tiêu đề tài

Mục tiêu nắm rõ đc các thành phần tạo nên game, có một cái nhìn khái quát, sâu hơn về game cũng như quy trình cần thiết để làm các game tương tự.

3. Giới thiệu đề tài

FunDraw là một ứng dụng game desktop cho phép người dùng có thể chơi vẽ hình và đoán hình cùng nhau.

Người vẽ sẽ chọn một trong những từ được cho sẵn để có thể vẽ và những người chơi còn lại sẽ dựa vào hình ảnh do người vẽ vẽ nên để đoán được từ và ghi điểm.

Người chơi có thể đăng nhập vào rồi tạo một phòng để người chơi khác có thể vào chơi cùng hoặc tạo một phòng riêng để có thể chơi cùng bạn bè thông qua mã phòng.

Để tham gia vào trò chơi, người chơi có thể là một trong hai loại sau:

- Người tham gia: Là người trực tiếp tham gia vào trò chơi. Người tham gia có thể nhập vào mã phòng để có thể tham gia vào một phòng chơi.

- Người tạo phòng: Là người tạo ra phòng chơi để người khác có thể vào tham gia. Người tạo phòng có quyền giới hạn số người chơi, thời gian, số vòng, đặt mật khẩu phòng hoặc giới hạn các từ có thể chọn.

Người tham gia và người tạo phòng có thể là người đoán hoặc người vẽ (tối đa 1 người vẽ). Mỗi vòng sẽ có n lượt chơi tương ứng với n người chơi. Điểm sẽ được tính như sau:

- Điểm sẽ bắt đầu bằng số thời gian của lượt. Giả sử là 120 giây. Mỗi giây điểm sẽ bị trừ đi 1.
- Khi người chơi đoán được thì sẽ được cộng điểm bằng số điểm bắt đầu trừ đi số điểm đã bị trừ. Ví dụ, người chơi đoán được sau 15 giây trôi qua, thì điểm sẽ là $120 - 15 = 105$ điểm.
- Người vẽ mỗi khi người chơi đoán được thì sẽ được số điểm bằng phân nửa số điểm của người đoán. Ví dụ như trên thì người vẽ sẽ được 52.5 điểm làm tròn lên là 53 điểm.
- Càng nhiều người đoán được càng nhanh thì điểm của người vẽ càng cao, ngược lại điểm càng thấp.

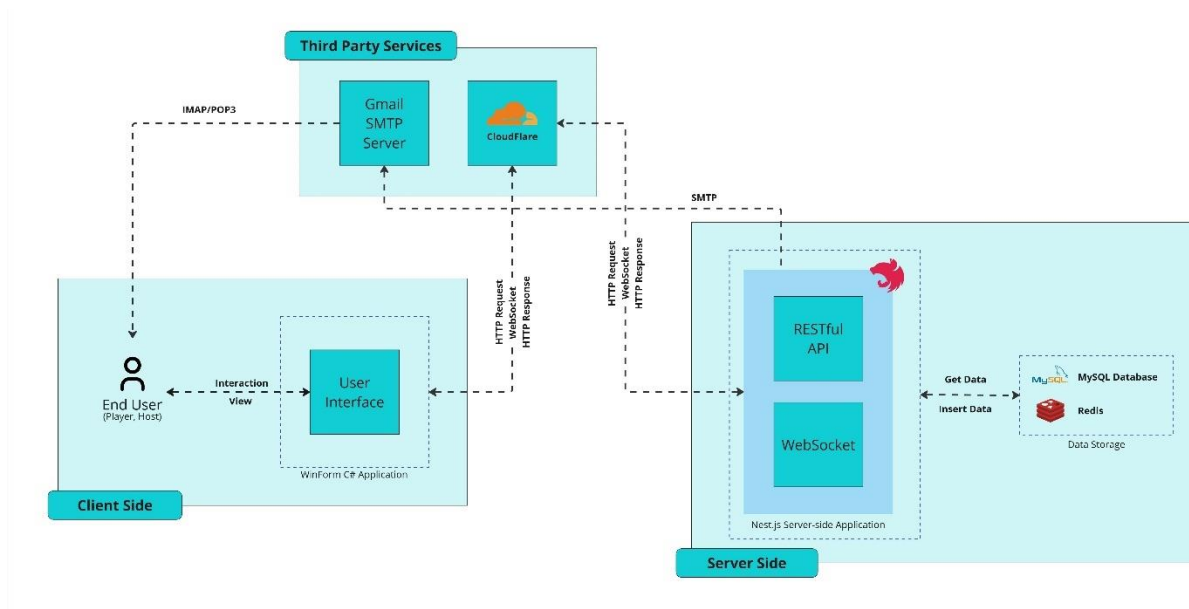
4. Các tính năng của đề tài

Các tính năng của trò chơi FunDraw bao gồm:

- Đăng ký và đăng nhập người dùng
- Đặt lại mật khẩu thông qua email
- Xem hồ sơ người chơi bao gồm tên đăng nhập, email, ngày tham gia và nút đổi mật khẩu
- Tham gia phòng chơi hoặc tạo phòng thông qua WebSocket

II. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

1. Sơ đồ kiến trúc tổng quan hệ thống

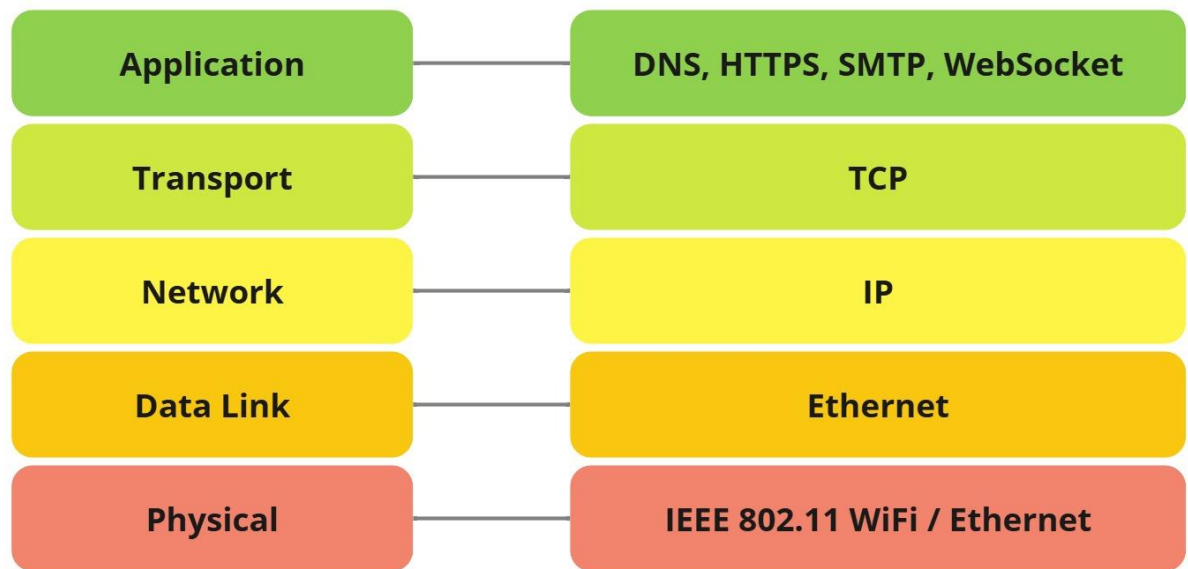


Sơ đồ kiến trúc tổng quan hệ thống

Các công nghệ sử dụng:

- Client Side: gồm ứng dụng WinForm C# kết hợp thư viện GunaUI [1], SkiaSharp [2], và Socket.IO C# [3]
- Server Side:
 - Nest.js [4]: Ứng dụng Server-side
 - MySQL: Cơ sở dữ liệu chính
 - Redis [5]: Dùng cho cache dữ liệu
 - Socket.IO [6]: Dùng cho tạo và quản lý kết nối WebSocket
- Third Party Service:
 - SMTP Server: Cho chức năng gửi mail đặt lại mật khẩu.
 - CloudFlare [7]: Là Proxy đóng vai trò lá chắn bảo vệ web server khỏi các cuộc tấn công từ chối dịch vụ phân tán (DDoS) và kiểm soát lưu lượng truy cập.

2. Network Stack



Cấu trúc phân lớp mạng

Tầng Application: Sử dụng DNS (Domain Name System) để phân giải tên miền của Web Server (fundraw.lt.id.vn), dùng HTTPS (Port 443) để gửi các HTTP Request và Response được bảo mật, SMTP (Port 465) để gửi mail từ phía Server, và WebSocket (Port 443) để client có thể kết nối liên tục đến Server.

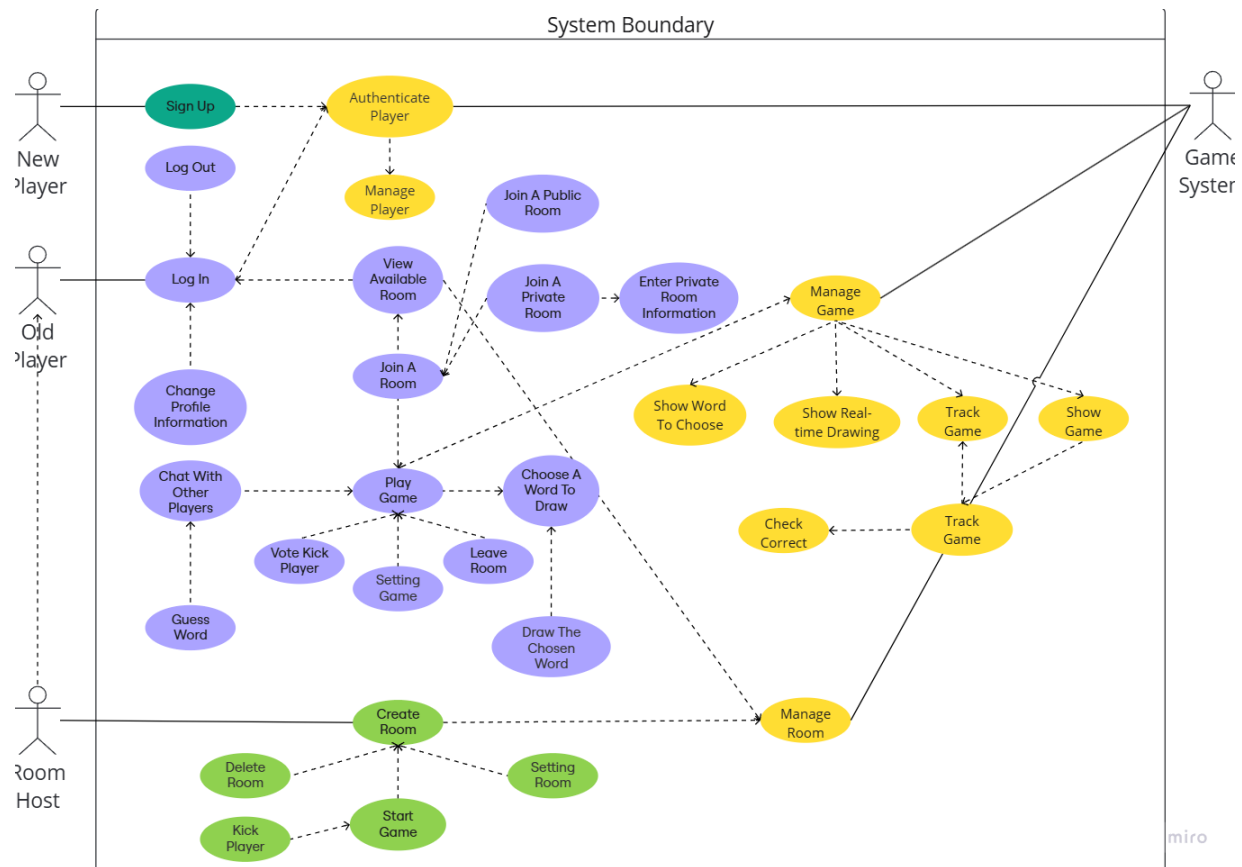
Tầng Transport: Giao thức TCP (Transmission Control Protocol) được sử dụng xuyên suốt trong các phiên truyền thông giữa ứng dụng phía Client và phía Server để đảm bảo độ tin cậy và toàn vẹn của thông tin.

Tầng Network: IP (Internet Protocol) cung cấp danh tính của các thiết bị được kết nối mạng, giúp các thiết bị trên mạng internet phân biệt và nhận ra nhau, từ đó có thể giao tiếp với nhau. Thiết bị đầu cuối có thể sử dụng IPv4 hoặc IPv6.

Tầng Data Link: Hầu hết các máy tính hiện đại ngày nay khi kết nối đến mạng lân cận đều sử dụng giao thức Ethernet để nhận và gửi đi các frame ở tầng này.

Tầng Physical: Nếu thiết bị đầu cuối được kết nối đến WiFi thì sẽ sử dụng giao thức thuộc họ giao thức IEEE 802.11 WiFi hoặc dùng giao thức Ethernet nếu thiết bị sử dụng mạng dây.

3. Sơ đồ Use case



Tóm tắt: Sơ đồ use case được phân loại theo từng nhóm người dùng và vai trò của họ trong hệ thống. Người chơi có thể tham gia trò chơi thông qua đăng nhập hoặc đăng ký tài khoản, tương tác với các tính năng trò chơi như chat, đoán từ, và vẽ. Chủ phòng có quyền tạo, cài đặt, và quản lý phòng chơi. Hệ thống đảm bảo các chức năng hỗ trợ như xác thực, quản lý người chơi, và xử lý trò chơi.

Người tham gia:

- Người chơi mới
- Người chơi cũ

- Chủ phòng chơi

Danh sách sự kiện:

➤ *Sự kiện chung:*

- Đăng nhập
- Đăng ký
- Đăng xuất

➤ *Sự kiện dành riêng cho người chơi cũ:*

- Thay đổi thông tin cá nhân
- Tham gia phòng chơi

➤ *Sự kiện trong phòng chơi:*

- Chat với người chơi khác
- Đoán từ
- Rời phòng

➤ *Sự kiện trong trò chơi:*

- Bắt đầu chơi
- Chọn từ để vẽ
- Vẽ từ đã chọn
- Kiểm tra từ đoán đúng
- Hiện thị thời gian thực vẽ
- Theo dõi thời gian chơi
- Theo dõi điểm số trò chơi
- Hiện thị kết quả trò chơi

➤ *Sự kiện dành riêng cho chủ phòng:*

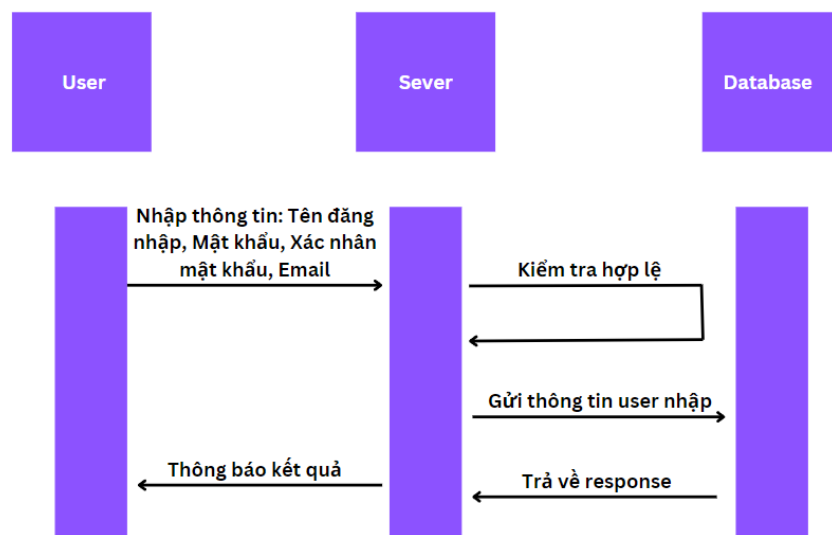
- Tạo phòng
- Cài đặt phòng
- Bắt đầu trò chơi
- Xóa phòng

➤ **Sự kiện hệ thống hỗ trợ:**

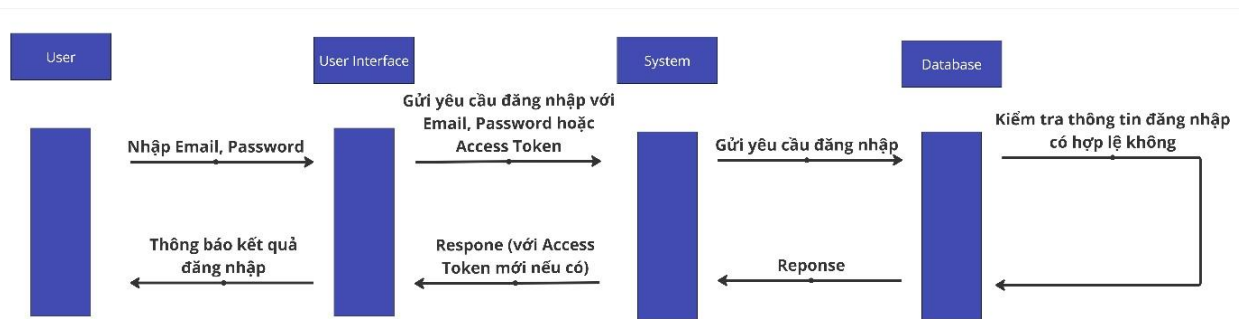
- Xác thực người chơi
- Quản lý thông tin người chơi
- Quản lý trò chơi
- Hiển thị các từ để chọn

4. Luồng của ứng dụng

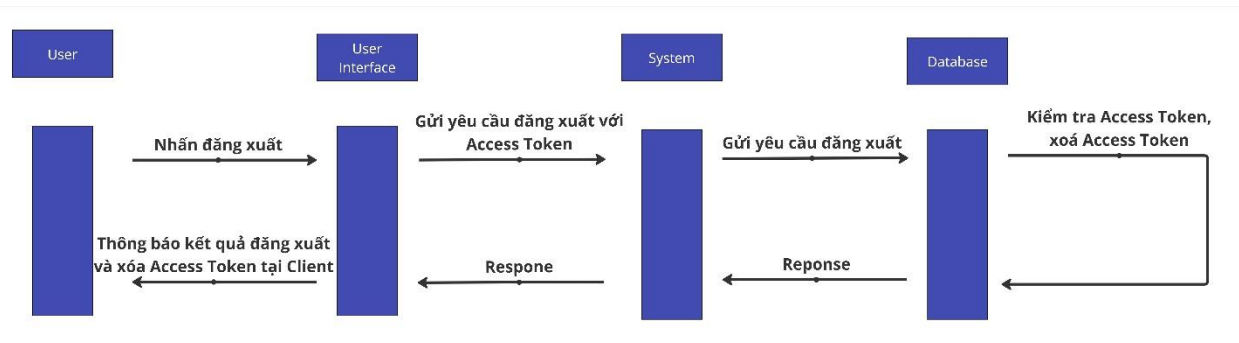
Luồng đăng ký:



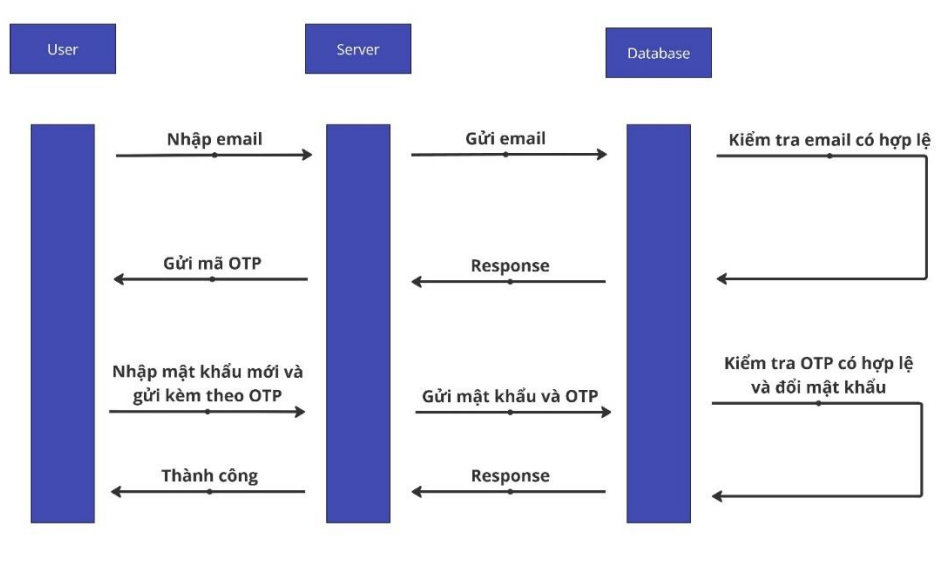
Luồng đăng nhập:



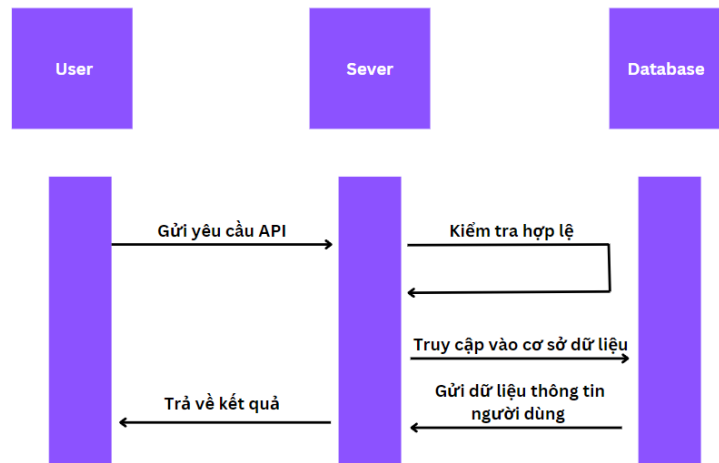
Luồng đăng xuất:



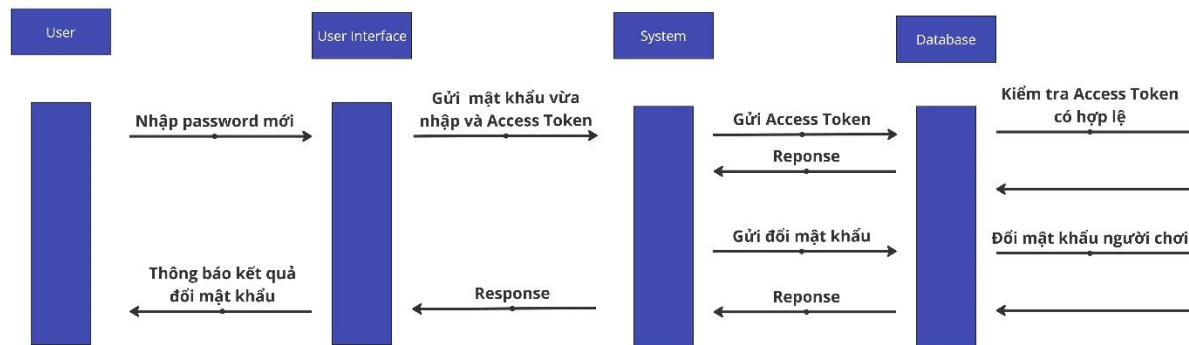
Luồng quên mật khẩu:



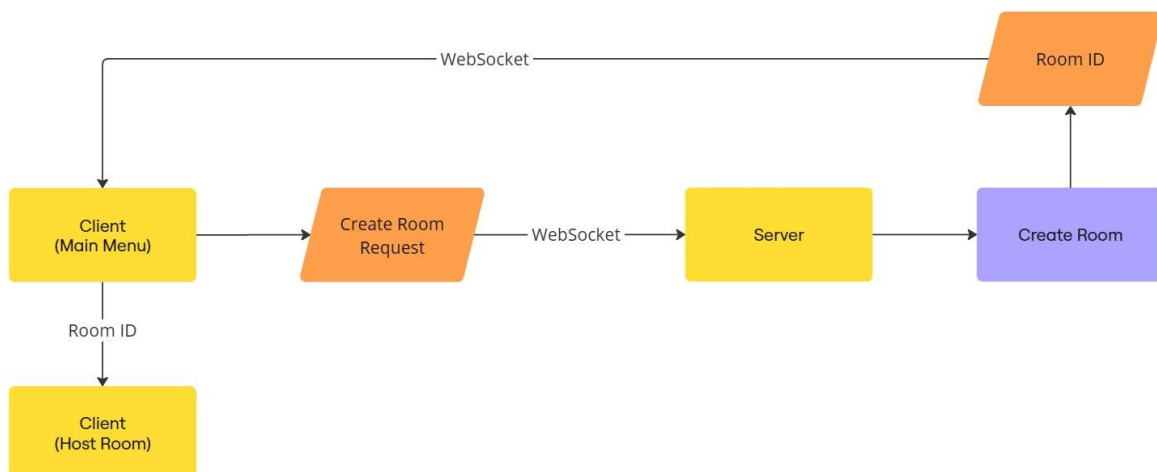
Luồng hiển thị thông tin hồ sơ người chơi:



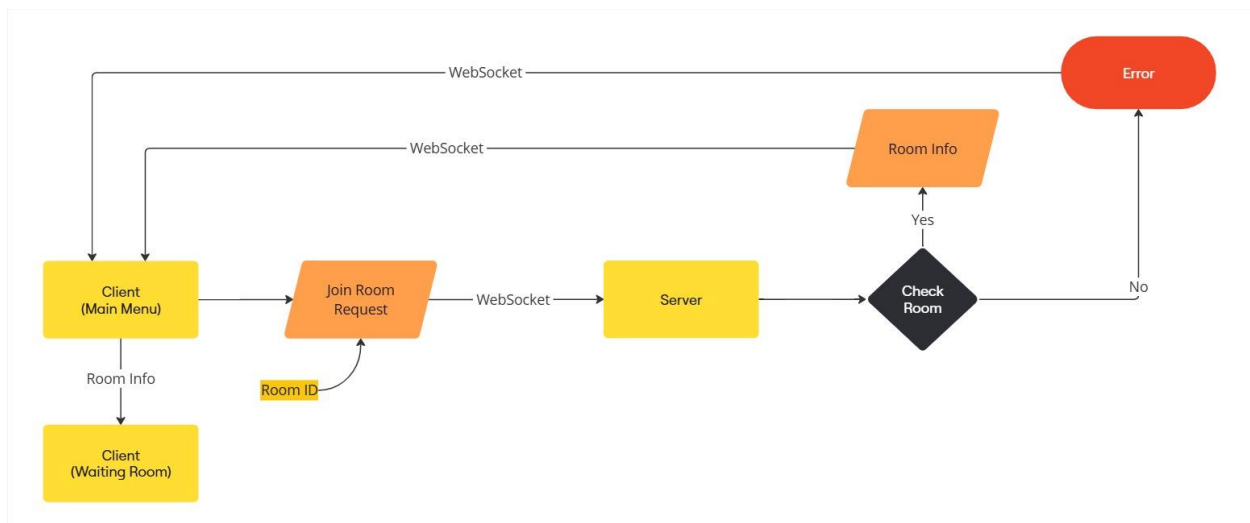
Luồng đổi mật khẩu:



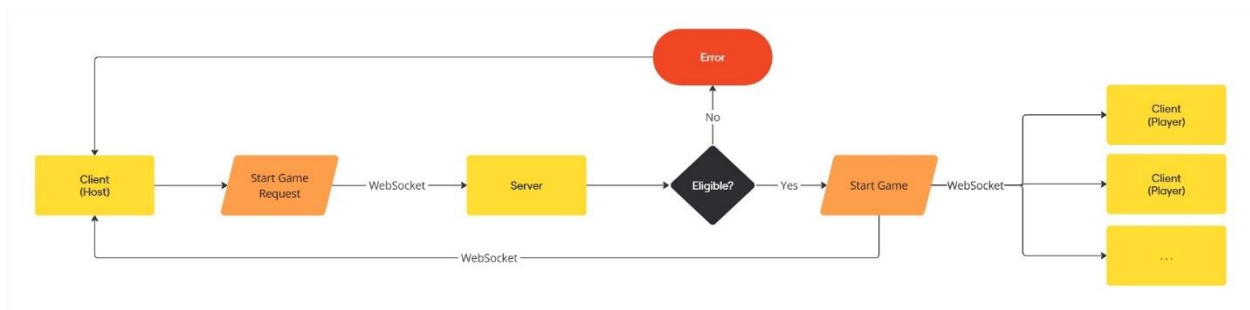
Luồng tạo phòng:



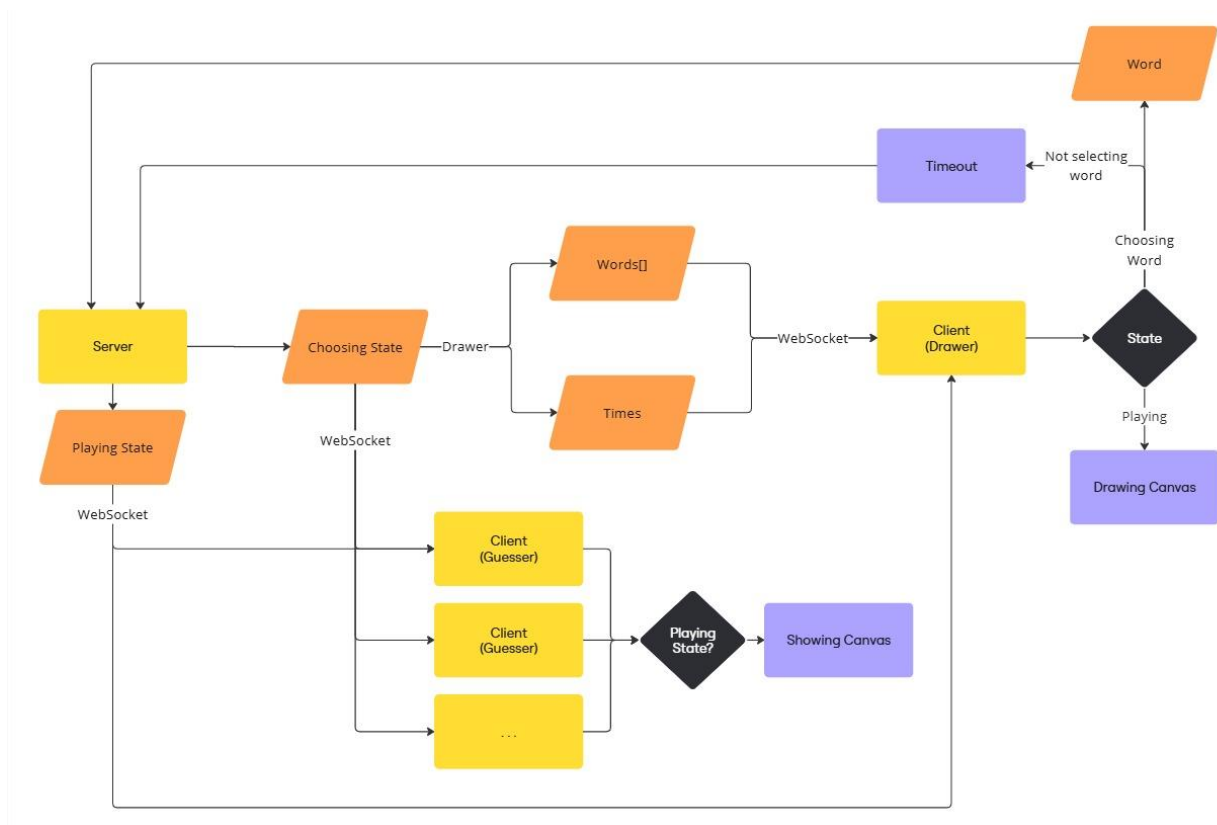
Luồng tham gia phòng:



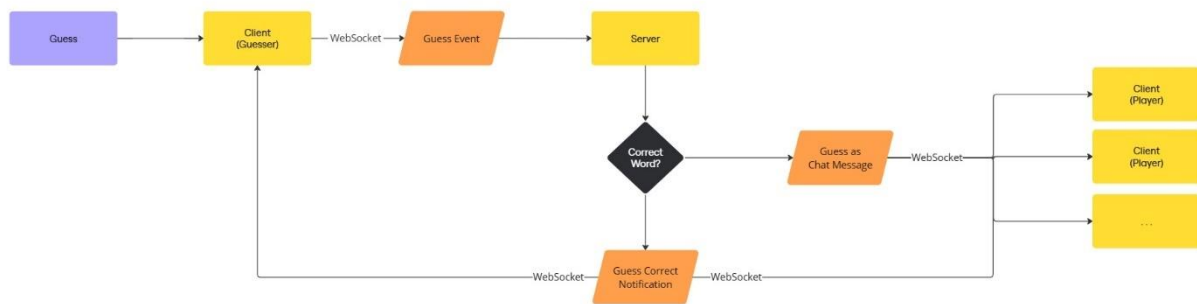
Luồng bắt đầu phòng chơi:



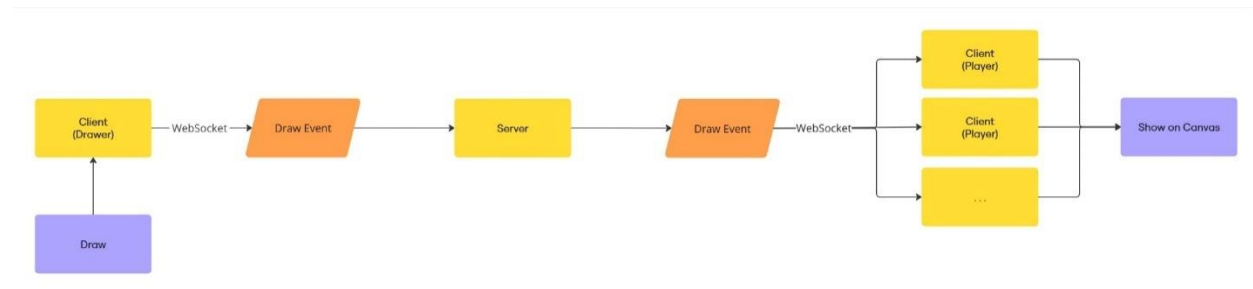
Luồng người vẽ chọn từ:



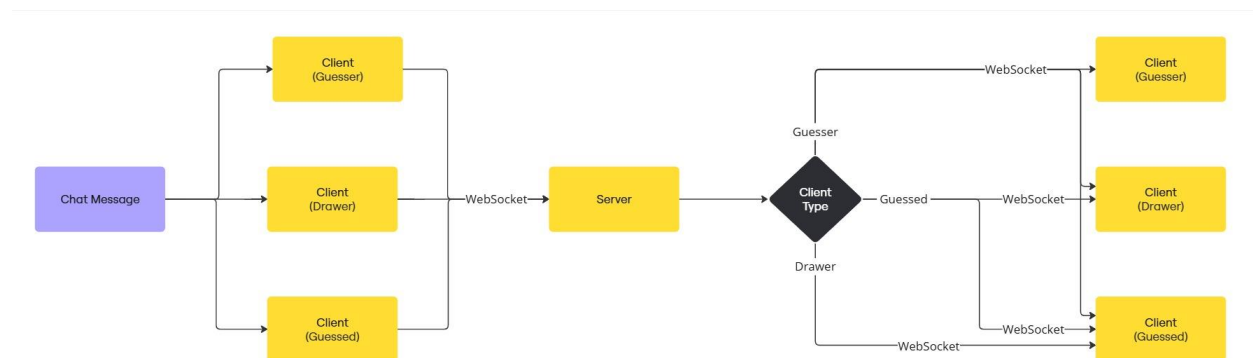
Luồng người đoán đoán từ:



Luồng người chơi chat:



Luồng tiến trình trò chơi:



III. HIỆN THỰC HỆ THỐNG

1. Tính năng đăng nhập, đăng ký, quên mật khẩu

1.1. Tính năng đăng nhập

Quy trình:

- Người dùng nhập tên đăng nhập và mật khẩu vào các ô tương ứng.
- Nhấn nút "Đăng nhập".

Xử lý:

- Ứng dụng client gửi yêu cầu POST tới server với tên đăng nhập và mật khẩu (dưới dạng không mã hóa).
- Server kiểm tra thông tin người dùng với dữ liệu lưu trữ.
- Nếu thông tin hợp lệ, server trả về accessToken và refreshToken.

- Các token này được lưu trữ cục bộ trên client.
- Ứng dụng gửi yêu cầu GET để lấy thông tin hồ sơ người dùng.

Phản hồi:

- Nếu yêu cầu GET thành công, hiển thị thông báo "Đăng nhập thành công" và chuyển hướng đến giao diện chính.
- Nếu xảy ra lỗi, hiển thị "Đăng nhập thất bại".

1.2. Tính năng đăng ký

Quy trình:

- Người dùng nhập tên đăng nhập, email, mật khẩu và xác nhận mật khẩu vào các ô tương ứng.
- Nhấn nút "Đăng ký".

Kiểm tra đầu vào:

- Kiểm tra tính hợp lệ của email bằng biểu thức chính quy (regex).

=> Nếu không hợp lệ, hiển thị thông báo yêu cầu nhập lại.

- So sánh mật khẩu và xác nhận mật khẩu.

=> Nếu không trùng khớp, cảnh báo người dùng và yêu cầu nhập lại.

Xử lý:

- Nếu thông tin hợp lệ, ứng dụng gửi yêu cầu POST tới server với các thông tin đã nhập.
- Server kiểm tra tính hợp lệ và sự tồn tại của tài khoản.

Phản hồi:

- Nếu đăng ký thành công, hiển thị thông báo "Đăng ký thành công".
- Nếu xảy ra lỗi, hiển thị "Đăng ký thất bại".

1.3. Tính năng quên mật khẩu

Quy trình:

- Người dùng nhập địa chỉ email vào ô tương ứng.
- Nhấn nút "Gửi".

Kiểm tra đầu vào:

- Kiểm tra tính hợp lệ của email bằng biểu thức chính quy (regex).

=> Nếu không hợp lệ hoặc để trống, hiển thị thông báo yêu cầu nhập lại.

Xử lý:

- Ứng dụng gửi yêu cầu POST tới server với email đã nhập.
- Server kiểm tra xem email có tồn tại trong cơ sở dữ liệu không.

Phản hồi:

- Nếu email tồn tại, server gửi email chứa thông tin đặt lại mật khẩu.
- Hiển thị thông báo "Mật khẩu đã được gửi tới email của bạn. Vui lòng kiểm tra hộp thư".
- Nếu email không tồn tại, hiển thị "Email không tồn tại. Vui lòng kiểm tra lại địa chỉ email và thử lại".
- Nếu xảy ra lỗi, hiển thị "Đã xảy ra lỗi" kèm thông tin chi tiết.

2. Tính năng hồ sơ người chơi, đổi mật khẩu

2.1. Tính năng hồ sơ người chơi

Quy trình:

- Người dùng nhấn nút "Hồ sơ người chơi".
- Ứng dụng gửi yêu cầu API tới server.

Xử lý:

- Server kiểm tra tính hợp lệ của thông tin trong yêu cầu.
- Nếu yêu cầu hợp lệ:
 - Server truy vấn dữ liệu từ cơ sở dữ liệu.
 - Cơ sở dữ liệu xử lý truy vấn và gửi kết quả về server.
- Server tổng hợp và trả dữ liệu hồ sơ người chơi về phía ứng dụng.

Phản hồi:

- Hiển thị thông tin hồ sơ người chơi trong giao diện ứng dụng.

2.2. Tính năng đổi mật khẩu

Quy trình:

- Người dùng nhập:
 - Mật khẩu cũ.
 - Mật khẩu mới.
 - Xác nhận mật khẩu mới.
- Nhấn nút "Gửi".

Kiểm tra đầu vào:

- Kiểm tra xem người dùng đã nhập đầy đủ thông tin vào tất cả các trường:
 - **Nếu có trường bị bỏ trống:** Hiển thị thông báo yêu cầu nhập đầy đủ thông tin.
- Kiểm tra tính hợp lệ của mật khẩu:
 - **Nếu mật khẩu mới không trùng với xác nhận mật khẩu:** Hiển thị thông báo lỗi.
 - **Nếu mật khẩu mới giống mật khẩu cũ:** Hiển thị thông báo lỗi.

Xử lý:

- Nếu thông tin hợp lệ, ứng dụng gửi yêu cầu API tới server để đổi mật khẩu.

- Server kiểm tra và xử lý yêu cầu.

Phản hồi:

- Nếu đổi mật khẩu thành công: Hiển thị thông báo xác nhận.
- Nếu xảy ra lỗi: Hiển thị thông báo lỗi và yêu cầu thử lại.

3. Tính năng trò chơi

3.1. Công nghệ và thư viện sử dụng

SkiaSharp:

- Một thư viện đồ họa 2D mã nguồn mở, cung cấp khả năng vẽ hiệu năng cao và linh hoạt.
- Trong mã, **SKSurface** và **SKCanvas** được sử dụng để tạo bề mặt vẽ và thực hiện các thao tác vẽ (ví dụ: `skcanvas.DrawLine`).
- Hãy xem **SKSurface** như "tờ giấy", còn **SKCanvas** là "cây bút" mà bạn sử dụng để vẽ trên tờ giấy đó.
- **SKPaint** định nghĩa các thuộc tính của bút vẽ như màu sắc, độ rộng nét vẽ, và kiểu vẽ.

WinForms (Windows Forms):

- Framework UI cơ bản để tạo các form và xử lý sự kiện giao diện người dùng như chuột và bàn phím.
- Các thành phần chính như **canvas**, **pencil**, **eraser**, và các sự kiện liên quan (`MouseDown`, `MouseMove`, `MouseUp`) được sử dụng để tương tác với người dùng.

3.2. Tính năng vẽ

Vẽ bằng bút chì:

- Người dùng có thể chọn công cụ **bút chì (pencil)** để vẽ các đường nét trên canvas.
- Sự kiện chuột (**MouseDown, MouseMove**) được sử dụng để bắt đầu và vẽ các đường liền nét.

Tẩy xóa (Eraser):

- Công cụ tẩy xóa được triển khai bằng cách đổi màu của bút vẽ thành **SKColors.White** và tăng độ rộng nét vẽ.

Bảng màu (Color Palette):

- Người dùng có thể chọn màu sắc tùy chỉnh thông qua **hộp thoại màu** (ColorDialog) và chuyển đổi màu sang định dạng **SkiaSharp** (GetSKColor).

Xóa toàn bộ canvas:

- Chức năng xóa toàn bộ canvas bằng cách gọi **skcanvas.Clear(SKColors.White)** và làm mới giao diện bằng **canvas.Invalidate()**.

3.3. Truyền tải dữ liệu vẽ (Real-time Drawing)

Gateway:

- Sử dụng **Gateway.Instance.Emit** để gửi dữ liệu vẽ thời gian thực, bao gồm **lệnh vẽ** (drawEvent) và các thông số như tọa độ điểm đầu, điểm cuối và màu sắc.
- Các sự kiện (On handlers) **lắng nghe phản hồi** từ server để đồng bộ trạng thái canvas giữa các người chơi.

3.4. Xử lý giao diện người dùng (UI)

Invalidate và Redraw:

- **canvas.Invalidade()** được gọi để làm mới giao diện mỗi khi có thay đổi trên canvas.

Tương tác qua chuột và bàn phím:

- Các sự kiện như **MouseDown**, **MouseMove**, và **MouseUp** giúp ghi nhận thao tác vẽ của người dùng.
- **KeyDown** được sử dụng để gửi tin nhắn chat khi nhấn phím Enter.

3.5. Quản lý bộ nhớ và tài nguyên

Bitmap và LockBits:

- **Bitmap** được sử dụng để lưu trạng thái của canvas.
- **LockBits** và **UnlockBits** tối ưu hóa việc truy cập trực tiếp vào bộ nhớ bitmap để cập nhật nhanh hơn.

3.6. Tích hợp thời gian thực (Real-time Integration)

Hệ thống sự kiện Socket:

- Tích hợp các sự kiện thông qua **Gateway.Instance** để quản lý các tính năng thời gian thực như **danh sách người chơi** (playerList), **tiến độ game** (gameProgress), và **tin nhắn chat** (chatMessage).

4. Đặc tả API

4.1 Đăng nhập

Method	URL
Post	https://fundraw.lt.id.vn/auth/login

- Content-Type: application/json

Dữ liệu truyền vào API (x-www-form-urlencoded)

Tham số	Kiểu dữ liệu	Ý nghĩa
username	string	Tên tài khoản
password	string	Mật khẩu đăng nhập

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
data	object	Dữ liệu chi tiết khi đăng nhập thành công (khi mã trạng thái 200)
data.user	object	Thông tin người dùng
data.user.id	string	Mã định danh người dùng
data.user.username	string	Tên người dùng
data.user.email	string	Email người dùng
data.user.createdAt	string	Ngày tạo tài khoản
data.user.avatar	string	URL ảnh đại diện người dùng
data.accessToken	string	Token để xác thực

error	string	Mô tả lỗi (khi mã trạng thái 400)
-------	--------	-----------------------------------

4.2 Đăng ký

Method	URL
Post	https://fundraw.lt.id.vn/auth/register

- Content-Type: application/json

Dữ liệu truyền vào API (x-www-form-urlencoded)

Tham số	Kiểu dữ liệu	Ý nghĩa
username	string	Tên tài khoản đăng ký
password	string	Mật khẩu đăng ký
confirm_password	string	Xác nhận mật khẩu
email	string	Email đăng ký

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu

data	object	Dữ liệu chi tiết khi đăng ký thành công (khi mã trạng thái 201)
data.user	object	Thông tin người dùng
data.user.id	string	Mã định danh người dùng
data.user.username	string	Tên người dùng
data.user.email	string	Email người dùng
data.user.createdAt	string	Ngày tạo tài khoản
data.user.avatar	string	URL ảnh đại diện người dùng
data.accessToken	string	Token để xác thực
error	string	Mô tả lỗi (khi mã trạng thái 400)

4.3 Đăng xuất

Method	URL
Post	https://fundraw.lt.id.vn/auth/logout

- Content-Type: application/json

Dữ liệu truyền vào API (header)

Tham số	Kiểu dữ liệu	Ý nghĩa
---------	--------------	---------

Authorization	string	Token để xác thực theo chuẩn Bearer
---------------	--------	-------------------------------------

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
error	string	Mô tả lỗi (khi mã trạng thái 400)

4.4 Lấy hồ sơ người dùng

Method	URL
Get	https://fundraw.lt.id.vn/users/profile

- Content-Type: application/json

Dữ liệu truyền vào API (header)

Tham số	Kiểu dữ liệu	Ý nghĩa
Authorization	string	Token để xác thực theo chuẩn Bearer

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
data	object	Dữ liệu chi tiết khi lấy hồ sơ thành công (khi mã trạng thái 200)
data.id	string	Mã định danh người dùng
data.username	string	Tên người dùng
data.email	string	Email người dùng
data.createdAt	string	Ngày tạo tài khoản
data.user.avatar	string	URL ảnh đại diện người dùng
error	string	Mô tả lỗi (khi mã trạng thái 400)

4.5 Đổi mật khẩu

Method	URL
Post	https://fundraw.lt.id.vn/users/change-password

- Content-Type: application/json

Dữ liệu truyền vào API (header, x-www-form-urlencoded)

Tham số	Kiểu dữ liệu	Ý nghĩa
Authorization	string	Token để xác thực theo chuẩn Bearer
password	string	Mật khẩu mới
confirm_password	string	Xác nhận mật khẩu

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
error	string	Mô tả lỗi (khi mã trạng thái 400)

4.5. Đặt lại mật khẩu:

Method	URL
Post	https://fundraw.lt.id.vn/users/reset-otp

- Content-Type: application/json

Dữ liệu truyền vào API (x-www-form-urlencoded)

Tham số	Kiểu dữ liệu	Ý nghĩa
---------	--------------	---------

email	string	Email người dùng
-------	--------	------------------

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
error	string	Mô tả lỗi (khi mã trạng thái 400)

4.6 Reset OTP

Method	URL
Post	https://fundraw.lt.id.vn/users/reset-otp

- Content-Type: application/json

Dữ liệu truyền vào API (x-www-form-urlencoded)

Tham số	Kiểu dữ liệu	Ý nghĩa
email	string	Email người dùng

Dữ liệu API trả về

Tham số	Kiểu dữ liệu	Ý nghĩa
statusCode	number	Mã trạng thái HTTP phản hồi
message	string	Thông điệp mô tả kết quả của yêu cầu
error	string	Mô tả lỗi (khi mã trạng thái 400)

IV. KẾT LUẬN, HƯỚNG PHÁT TRIỂN

1. Kết luận

Nhóm đã xây dựng được một ứng dụng trò chơi theo kiến trúc client – server có xác thực, phân quyền, kết nối được Internet và tương tác với nhau thông qua nhiều client kết nối cùng một phòng.

- Repository chứa mã nguồn cho ứng dụng phía client (FunDraw):

<https://github.com/Group-10-NT106-P13/FunDraw>

- Repository chứa mã nguồn cho ứng dụng phía server (FunDraw-API):

<https://github.com/Group-10-NT106-P13/FunDraw-API>

Về tính năng, ứng dụng cho phép người chơi có thể tự tạo phòng rồi có thể mời bạn bè thông qua Mã phòng hoặc có thể tham gia vào phòng của người khác.

Đồ án đã áp dụng các công nghệ hiện đại như HTTP RESTful API, thông qua các lời gọi API để xác thực và thay đổi thông tin người dùng; WebSocket, người chơi có thể tương tác với nhau với độ trễ thấp và trực tiếp. Đặc biệt là đồ án đã được triển khai trên Internet nên bất cứ ai có thể tải ứng dụng và chơi ngay.

Tuy nhiên, vẫn còn một số hạn chế như:

- Vẫn chưa hoàn toàn có thể kiểm soát các sự kiện lỗi mà có thể xảy ra với client và server.

- Bất cập trong việc triển khai ứng dụng ra ngoài thực tế và triển khai API trên server.
- Mã nguồn ứng dụng vẫn chưa tối ưu, khó đọc và khó bảo trì.

2. Hướng phát triển

Qua những hạn chế đó, nhóm dự định tiếp tục phát triển thêm trong tương lai như:

- Kiểm soát các sự kiện cho cả Client và Server
- Kiểm soát sự kiện cho các phòng chơi bằng cách lưu vào bộ nhớ tạm (Redis [5]) và sau cùng lưu vào cơ sở dữ liệu (MySQL).
- Tích hợp tính năng kết nối lại khi Client bị ngắt kết nối giữa lúc chơi.
- Sửa đổi giao diện cho đẹp hơn, bắt mắt và thân thiện với người chơi hơn.
- Tích hợp thêm tính năng ảnh đại diện cho người chơi.

Bảng phân công công việc

STT	MSSV	Tên	Nội dung công việc phụ trách	Tự đánh giá (theo thang điểm 10)
1	23521757	Nguyễn Lộc Tỷ	<ul style="list-style-type: none"> - Phác thảo giao diện người chơi - Thiết kế kiến trúc hệ thống - Thiết kế luồng ứng dụng - Triển khai CI/CD cho cả hai repository 	9/10

			<ul style="list-style-type: none"> - Triển khai máy chủ bao gồm MySQL, Redis và Docker - Code API cho Server - Code WebSocket cho cả Server và tích hợp vào Client - Làm slide báo cáo - Viết và soạn báo cáo đồ án giữa kỳ và cuối kỳ 	
2	23521744	Nguyễn Thanh Tùng	<ul style="list-style-type: none"> - Thiết kế Userflow, luồng quên mật khẩu. - Thiết kế giao diện trò chơi, hồ sơ người chơi. - Thiết kế chức năng đăng nhập, đăng ký, đăng xuất. - Thiết kế chức năng trò chơi. - Làm slide báo cáo. - Viết và soạn báo cáo đồ án giữa kỳ và cuối kỳ. 	8.5/10
3	23521650	Đỗ Đức Minh Triết	<ul style="list-style-type: none"> - Vẽ sơ đồ Use case - Vẽ luồng đăng nhập và đăng xuất người chơi 	8.5/10

			<ul style="list-style-type: none"> - Thiết kế tổng quan và chi tiết giao diện màn hình chính, tạo phòng - Sửa và cải thiện giao diện - Tích hợp chức năng quên mật khẩu và đổi mật khẩu người chơi - Làm slide báo cáo - Viết và soạn báo cáo đồ án cuối kỳ 	
4	23521728	Tạ Quốc Tuấn	<ul style="list-style-type: none"> - Vẽ luồng đăng ký và hồ sơ người chơi - Vẽ Network Stack - Thiết kế tổng quan và chi tiết giao diện đăng nhập, đăng ký, quên mật khẩu - Sửa và cải thiện giao diện - Tích hợp chức năng gọi API cho hồ sơ người chơi - Làm slide báo cáo - Viết và soạn báo cáo đồ án cuối kỳ 	8.5/10

Tài liệu tham khảo

- [1] GunaUI: <https://gunaui.com/>.
- [2] SkiaSharp: <https://www.nuget.org/packages/SkiaSharp/>.
- [3] Socket.io Client C#: <https://www.nuget.org/packages/SocketIOClient>.
- [4] Nest.js: <https://nestjs.com/>.
- [5] Redis: <https://redis.io/>.
- [6] Socket.IO: <https://socket.io/>.
- [7] CloudFlare: <https://www.cloudflare.com/>.