

TRƯỜNG ĐẠI HỌC SÀI GÒN
KHOA CÔNG NGHỆ THÔNG TIN



PHÁT TRIỂN PHẦN MỀM MÃ NGUỒN MỞ

Xây dựng ứng dụng game 2 người chơi

Caro trên PYTHON

GVHD: Từ Lăng Phiêu

SV: Nguyễn Văn Tiến Dũng - 3120410084

Lai Quang Vinh - 3120410613

TP. HỒ CHÍ MINH, THÁNG 4/2024

MỤC LỤC

PHẦN 1: CƠ SỞ LÝ THUYẾT	4
1.1 Ngôn ngữ Python	4
1.1.1 Giới thiệu về ngôn ngữ python	4
1.1.2 Ưu điểm của Python	4
1.1.3 Nhược điểm của Python	5
1.2 Các thư viện chính đã được sử dụng	5
1.2.1 pygame	5
1.2.2 pygame-menu	6
1.2.3 tkinter	7
1.2.4 socket	8
1.2.5 queue	9
1.2.6 threading	10
PHẦN 2: THIẾT KẾ ỨNG DỤNG	11
2.1 Cấu trúc mã nguồn	11
2.2 Mô hình ứng dụng	11
2.3 Các tính năng được xây dựng	11
2.4 Flowchart	13
PHẦN 3: KẾT QUẢ THỰC HIỆN	15
3.1 Giao diện menu và trò chơi	15
3.1.1 Giao diện menu	15
3.1.2 Giao diện trò chơi	16
3.2 Kết nối mạng socket	17
3.2.1 Host Server	17
3.2.2 Join Server	18
3.3 Xử lý các nước đi	19
3.4 Xử lý lượt đi lần lượt	19
3.5 Kiểm tra chiến thắng	19
3.6 Gửi thông báo và ghi nhận điểm số	21
3.6.1 Gửi thông báo	21
3.6.2 Ghi nhận điểm số	21
PHẦN 4: HƯỚNG DẪN CÀI ĐẶT ỨNG DỤNG VÀ MÔI TRƯỜNG	22
NHIỆM VỤ VÀ VAI TRÒ CỦA TỪNG THÀNH VIÊN	24



TÀI LIỆU THAM KHẢO

25

GIỚI THIỆU

Trò chơi Caro - một trò chơi với nguồn gốc từ trò chơi gomoku cổ điển, nổi tiếng với tên gọi "Five in a Row", là một trong những trò chơi chiến thuật thú vị và sâu sắc. Với lịch sử hơn một thế kỷ, Caro không chỉ là một trò chơi giải trí, mà còn là một bài toán chiến thuật được nhiều người yêu thích và tìm hiểu.

Trò chơi Caro đòi hỏi người chơi phải có sự tư duy chiến thuật, khả năng dự đoán và phản ứng nhanh nhạy. Trên bàn cờ 15×15 , hai người chơi sẽ thay phiên nhau đánh dấu X hoặc O trên các ô trống, với mục tiêu tạo thành một dãy liên tiếp gồm 5 dấu X hoặc O theo hàng ngang, cột dọc hoặc đường chéo.

Trò chơi Caro đã trở thành một biểu tượng văn hóa và là một phần không thể thiếu trong cuộc sống hàng ngày của nhiều người. Không chỉ là một trò chơi giải trí, Caro còn là một nền tảng để thể hiện sự sáng tạo, suy luận và chiến thuật của người chơi.

Trong phiên bản Python của trò chơi Caro, người chơi sẽ trải nghiệm cảm giác hồi hộp khi đối đầu với một người chơi khác qua mạng. Bằng cách sử dụng thư viện đồ họa hoặc giao diện dòng lệnh, chúng ta có thể tạo ra giao diện đồ họa hoặc dòng lệnh thân thiện để người chơi có thể tham gia vào trò chơi một cách dễ dàng.

Caro không chỉ là một trò chơi, mà còn là một cách để kết nối và tạo ra những kỷ niệm đáng nhớ giữa các người chơi. Hãy chuẩn bị tinh thần và tham gia vào thế giới thú vị của trò chơi Caro ngay hôm nay!

PHẦN 1: CƠ SỞ LÝ THUYẾT

1.1 Ngôn ngữ Python

1.1.1 Giới thiệu về ngôn ngữ python

Python là một ngôn ngữ lập trình bậc cao cho các mục đích lập trình đa năng, do Guido van Rossum tạo ra và lần đầu ra mắt vào năm 1991. Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình và là ngôn ngữ lập trình dễ học; được dùng rộng rãi trong phát triển trí tuệ nhân tạo. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu.

1.1.2 Ưu điểm của Python

Là một ngôn ngữ mạnh mẽ và phổ biến hiện nay, Python mang lại nhiều ưu điểm với người dùng như:

- **Dễ học và dễ đọc:** Python có cú pháp đơn giản và rõ ràng, giúp việc học và hiểu ngôn ngữ này trở nên dễ dàng cho người mới học lập trình. Cú pháp gọn nhẹ cũng giúp code Python dễ đọc và dễ bảo trì.
- **Hỗ trợ thư viện phong phú:** Python có một cộng đồng lớn và nhiều thư viện phát triển, cung cấp các công cụ và chức năng đa dạng để giải quyết các vấn đề khác nhau.
- **Đa nền tảng:** Python là ngôn ngữ đa nền tảng, có thể chạy trên nhiều hệ điều hành như Windows, macOS và Linux. Điều này giúp xây dựng ứng dụng di động và máy tính để bàn trở nên linh hoạt và dễ dàng.
- **Hỗ trợ cho nhiều lĩnh vực:** Python không chỉ giới hạn trong một lĩnh vực cụ thể. Nó có thể được sử dụng trong phát triển web, phân tích dữ liệu, trí tuệ nhân tạo, trò chơi, mạng, và nhiều lĩnh vực khác.
- **Tích hợp và mở rộng:** Python có khả năng tích hợp tốt với các ngôn ngữ khác như C, C++, và Java bằng cách đó chúng ta có thể tích hợp các module mở rộng để tăng hiệu năng hoặc sử dụng các thư viện bên thứ ba đã có sẵn.
- **Giảm chi phí bảo trì:** Do tính đơn giản của nó, Python giúp bảo trì ứng dụng dễ dàng hơn và do đó, giảm chi phí liên quan, đây là một lợi thế lớn.

1.1.3 Nhược điểm của Python

Bên cạnh những ưu điểm kể trên thì Python vẫn có những hạn chế riêng như:

- **Chậm hơn so với ngôn ngữ biên dịch:** Python là ngôn ngữ thông dịch, do đó thường chạy chậm hơn so với ngôn ngữ biên dịch như C++ hoặc Java.
- **Tiêu thụ bộ nhớ lớn:** Các cấu trúc của Python đòi hỏi nhiều không gian bộ nhớ hơn. Ngôn ngữ này không thích hợp để sử dụng cho sự phát triển trong điều kiện bộ nhớ hạn chế.
- **Phát hiện lỗi trong mã:** Vì Python được thực thi thông qua trình thông dịch thay vì trình biên dịch, nên không thể phát hiện lỗi trong quá trình biên dịch.
- **Hạn chế thiết kế:** Python không cung cấp một giao diện người dùng đồ họa (GUI) tiêu chuẩn vì vậy có thể gây ra một số khó khăn trong việc phát triển giao diện người dùng phức tạp.
- **Khó kiểm tra:** Vì nó là một ngôn ngữ dựa trên trình thông dịch, rất khó để chạy các bài kiểm tra trên mã được viết bằng Python. Tất cả các lỗi chỉ xuất hiện trong thời gian chạy, điều này khiến việc kiểm tra các đoạn mã được viết bằng Python rất khó khăn.

1.2 Các thư viện chính đã được sử dụng

1.2.1 pygame

Pygame là một thư viện của ngôn ngữ lập trình Python và là một tập hợp các mô-đun Python được thiết kế riêng để lập trình trò chơi. Pygame được viết bởi Pete Shinnars và phát hành chính thức từ năm 2000. Pygame có thể chạy trên nhiều nền tảng và hệ điều hành khác nhau. Với thư viện pygame trong Python, các nhà phát triển có thể sử dụng công cụ và chức năng mở rộng để tạo ra các trò chơi ấn tượng. Bởi vậy, Pygame đang ngày càng phổ biến với nhà phát triển vì tính đơn giản, linh hoạt, dễ sử dụng.

Ưu điểm:

- **Dễ học và sử dụng:** Pygame được thiết kế để dễ dàng tiếp cận cho người mới học và người mới bắt đầu phát triển game. Cú pháp đơn giản và tài liệu phong phú giúp người dùng nhanh chóng làm quen với thư viện.
- **Đa nền tảng:** Pygame có sẵn trên nhiều nền tảng khác nhau, bao gồm Windows, macOS và Linux. Điều này cho phép bạn phát triển game và ứng dụng đa nền tảng một cách dễ dàng.

- **Hỗ trợ đồ họa và âm thanh:** Pygame cung cấp các chức năng mạnh mẽ để vẽ đồ họa, xử lý hình ảnh, và phát nhạc, âm thanh trong game. Bạn có thể tạo ra các hiệu ứng hình ảnh và âm thanh đa dạng để làm cho game của bạn sống động hơn.
- **Cộng đồng hỗ trợ lớn:** Pygame có một cộng đồng đông đảo và năng động. Bạn có thể tìm thấy nhiều tài liệu, hướng dẫn và nguồn tài nguyên sẵn, ngoài ra người dùng cũng có thể chia sẻ và trao đổi ý tưởng, mã nguồn, và góp ý với nhau từ cộng đồng này.

Nhược điểm:

- **Hiệu suất hạn chế:** Pygame không phải là một thư viện đồ họa chuyên nghiệp và không được tối ưu hóa để đạt hiệu suất tốt nhất. Điều này có thể gây ra hạn chế về hiệu suất khi xử lý các trò chơi có yêu cầu đồ họa và xử lý phức tạp.
- **Thiếu tính năng cao cấp:** So với một số thư viện phát triển game khác, pygame có thể thiếu một số tính năng cao cấp như hỗ trợ 3D, công cụ phân tích và chạm phức tạp, và hỗ trợ đa luồng mạnh mẽ.
- **Giao diện người dùng hạn chế:** Pygame không cung cấp sẵn các thành phần giao diện người dùng đồ họa (GUI) phức tạp. Bạn cần phải xây dựng và tùy chỉnh giao diện người dùng của mình bằng cách sử dụng các phương pháp và công cụ khác.

1.2.2 pygame-menu

Pygame-menu là một thư viện mã nguồn mở được phát triển cho Python và cung cấp các công cụ để tạo ra các menu và giao diện người dùng cho trò chơi sử dụng thư viện pygame. Với pygame-menu, việc tạo menu đa cấp, các nút tùy chỉnh, và các hiệu ứng chuyển động trở nên dễ dàng và linh hoạt hơn bao giờ hết. Pygame-menu được thiết kế để tích hợp dễ dàng với pygame, cho phép bạn sử dụng các thành phần của pygame để tạo ra các trò chơi đa dạng và phong phú về giao diện người dùng. Nó cũng được hỗ trợ bởi một cộng đồng phát triển động, với nhiều đóng góp và hỗ trợ từ các nhà phát triển khác nhau trên toàn thế giới. Tuy nhiên, tài liệu hướng dẫn không đầy đủ và một số hạn chế về hiệu suất trong một số trường hợp. Nhưng với sự linh hoạt và tính tiện lợi của nó, pygame-menu vẫn là một lựa chọn phổ biến cho việc phát triển giao diện người dùng trong trò chơi Python.

Ưu điểm:

- **Dễ Sử Dụng:** pygame-menu cung cấp một cách tiếp cận dễ dàng để tạo ra các menu trong trò chơi pygame một cách nhanh chóng và hiệu quả.
- **Tính Năng Đa Dạng:** Thư viện này cung cấp nhiều tính năng đa dạng như tạo menu đa cấp, tùy chỉnh các nút, tiêu đề, hình ảnh nền và nhiều hơn nữa.
- **Hỗ Trợ Hình Ảnh:** pygame-menu cho phép bạn sử dụng các hình ảnh làm nền cho các menu, giúp tạo ra giao diện người dùng phong phú và thú vị.
- **Tích Hợp Dễ Dàng:** Thư viện này tích hợp tốt với pygame, cho phép bạn tạo ra các menu và giao diện người dùng một cách dễ dàng trong trò chơi của mình.
- **Cộng Đồng Phát Triển Chủ Động:** pygame-menu là một dự án mã nguồn mở trên GitHub, nên luôn có sự đóng góp từ cộng đồng, giúp nó ngày càng hoàn thiện và linh hoạt hơn.

Nhược điểm:

- **Tính Linh Hoạt Hạn Chế:** Mặc dù pygame-menu cung cấp nhiều tính năng, nhưng đôi khi nó có thể hạn chế đối với các yêu cầu tùy chỉnh phức tạp hoặc không phù hợp với một số dự án cụ thể.
- **Tài Liệu Hướng Dẫn Không Đầy Đủ:** Mặc dù có sự hỗ trợ từ cộng đồng, tài liệu hướng dẫn về pygame-menu vẫn còn thiếu sót ở một số khía cạnh, đặc biệt là đối với người mới bắt đầu.
- **Hiệu Suất:** Trong một số trường hợp, việc sử dụng pygame-menu có thể ảnh hưởng đến hiệu suất của trò chơi, đặc biệt là khi có nhiều menu hoặc các hiệu ứng phức tạp được sử dụng.
- **Phụ Thuộc vào pygame:** pygame-menu là một phần của pygame, do đó nó phụ thuộc hoàn toàn vào pygame và không thể sử dụng độc lập với các thư viện khác.

1.2.3 tkinter

Tkinter là một thư viện trong ngôn ngữ lập trình Python được sử dụng để tạo giao diện đồ họa người dùng (GUI). "Tkinter" là viết tắt của "Tk interface", một toolkit đồ họa cung cấp các công cụ để phát triển giao diện người dùng. Điều này giúp cho Tkinter trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng với giao diện đồ họa đơn giản trong Python.

Ưu điểm:

- **Dễ học và sử dụng:** Tkinter được thiết kế để dễ dàng tiếp cận và sử dụng cho người mới học. Cú pháp đơn giản và tài liệu phong phú giúp người dùng nhanh chóng xây dựng các giao diện người dùng đơn giản.
- **Đa nền tảng:** Tkinter hỗ trợ đa nền tảng và có sẵn trên nhiều hệ điều hành khác nhau, bao gồm Windows, macOS và Linux. Điều này cho phép bạn phát triển ứng dụng GUI đa nền tảng một cách dễ dàng.
- **Có tính tương thích ngược:** Tkinter hỗ trợ tương thích ngược với phiên bản cũ hơn của thư viện Tk. Điều này có nghĩa là mã nguồn Tkinter viết cho phiên bản mới nhất cũng có thể chạy trên các phiên bản cũ hơn của Tk.
- **Hỗ trợ rộng:** Tkinter có một cộng đồng lớn và năng động, với nhiều tài liệu, ví dụ và hỗ trợ trực tuyến có sẵn. Bạn có thể tìm thấy nhiều tài nguyên và giải đáp thắc mắc từ cộng đồng này.

Nhược điểm:

- **Giao diện người dùng đơn giản:** Tkinter cung cấp các thành phần giao diện người dùng cơ bản và không có nhiều tính năng cao cấp sẵn có. Điều này có nghĩa là bạn cần phải tự xây dựng và tùy chỉnh các thành phần giao diện phức tạp hơn.
- **Giao diện thiết kế hạn chế:** Tkinter không cung cấp các công cụ thiết kế giao diện người dùng đồ họa (GUI) phức tạp như các công cụ khác. Bạn cần phải viết mã nguồn để xây dựng và bố trí các thành phần giao diện theo yêu cầu của bạn.
- **Hạn chế đồ họa:** Tkinter không phải là một thư viện đồ họa mạnh mẽ và không hỗ trợ các hiệu ứng đồ họa phức tạp. Điều này có thể gây hạn chế về khả năng tạo ra giao diện người dùng đẹp mắt và sống động.

1.2.4 socket

Thư viện socket trong Python là một phần của thư viện chuẩn (socket) và cung cấp các công cụ để thực hiện giao tiếp mạng giữa các máy tính thông qua các giao thức mạng như TCP và UDP. Socket cho phép bạn tạo và quản lý các kết nối mạng, gửi và nhận dữ liệu qua mạng, và xử lý các yêu cầu và phản hồi từ các máy tính khác trên cùng một mạng hoặc trên Internet.

Ưu điểm:

- **Linh Hoạt và Mạnh Mẽ:** Thư viện socket trong Python cung cấp một cách mạnh mẽ và linh hoạt để thực hiện giao tiếp mạng. Bạn có thể tạo, kết nối, gửi và nhận dữ liệu qua các socket một cách dễ dàng và hiệu quả.

- **Đa Dạng Loại Giao Thức:** Socket hỗ trợ cả giao thức TCP và UDP, cho phép bạn lựa chọn loại giao thức phù hợp với nhu cầu của ứng dụng của bạn.
- **Tích Hợp Tốt với Python:** Thư viện socket là một phần của thư viện chuẩn của Python, nên việc sử dụng nó không đòi hỏi cài đặt bổ sung và dễ dàng tích hợp vào các dự án Python hiện có.

Nhược điểm:

- **Phức Tạp Trong Một Số Trường Hợp:** Việc sử dụng socket có thể trở nên phức tạp trong một số trường hợp, đặc biệt là khi bạn cần xử lý các tình huống đa luồng hoặc đồng thời.
- **Cần Kiến Thức Về Mạng:** Để sử dụng socket hiệu quả, bạn cần có kiến thức về mạng để hiểu rõ về các khái niệm như địa chỉ IP, cổng, giao thức TCP/UDP và các vấn đề liên quan đến bảo mật mạng.
- **Khả Năng Xảy Ra Lỗi:** Việc lập trình với socket có thể dễ dàng gây ra các lỗi không mong muốn, như xung đột kết nối, timeout hoặc mất kết nối.

1.2.5 queue

Thư viện Queue trong Python cung cấp các cấu trúc dữ liệu hàng đợi (queue) cho việc lưu trữ và quản lý dữ liệu theo cơ chế "first-in, first-out" (FIFO). Hàng đợi là một cấu trúc dữ liệu mà phần tử được thêm vào trước sẽ được lấy ra đầu tiên.

Ưu điểm:

- **Tính Linh Hoạt:** Thư viện Queue cung cấp một loạt các phương thức để thêm, lấy ra và xử lý dữ liệu từ hàng đợi một cách dễ dàng và linh hoạt.
- **Hiệu Suất Cao:** Với cơ chế FIFO, việc lấy ra phần tử đầu tiên từ hàng đợi có độ phức tạp thấp, giúp tăng hiệu suất cho các thuật toán liên quan đến hàng đợi.
- **Tích Hợp Sẵn trong Python:** Thư viện Queue là một phần của thư viện chuẩn của Python, nên việc sử dụng nó không đòi hỏi cài đặt bổ sung và dễ dàng tích hợp vào các dự án Python hiện có.

Nhược điểm:

- **Giới Hạn Về Tính Linh Hoạt:** Hàng đợi chỉ hỗ trợ các thao tác thêm và lấy ra phần tử theo cơ chế FIFO, không cho phép các thao tác ngẫu nhiên trên dữ liệu như các cấu trúc dữ liệu khác như danh sách liên kết.

- **Khó Khăn Trong Việc Xử Lý Phần Tử Ở Vị Trí Bất Kỳ:** Trong hàng đợi, việc xử lý phần tử ở vị trí bất kỳ trong hàng đợi có độ phức tạp cao, đòi hỏi việc duyệt qua các phần tử từ đầu đến vị trí cần xử lý.
- **Khả Năng Xảy Ra Lỗi Trong Quá Trình Xử Lý:** Việc sử dụng hàng đợi có thể dễ dàng gây ra các lỗi như tràn hàng đợi (queue overflow) hoặc truy cập vào phần tử không tồn tại (out-of-bound access) nếu không được quản lý cẩn thận.

1.2.6 threading

Thư viện threading trong Python cung cấp các công cụ để tạo và quản lý các luồng (threads), cho phép thực thi đồng thời các tác vụ khác nhau trong cùng một chương trình. Sử dụng luồng giúp tăng hiệu suất của ứng dụng bằng cách cho phép các tác vụ chạy song song.

Ưu điểm:

- **Tính Linh Hoạt:** Thư viện threading cung cấp một cách linh hoạt để tạo và quản lý các luồng, cho phép bạn kiểm soát quy trình thực thi của các tác vụ một cách dễ dàng.
- **Tăng Hiệu Suất:** Sử dụng luồng giúp tăng hiệu suất bằng cách cho phép các tác vụ chạy đồng thời, giảm thời gian chờ đợi và tận dụng các nguồn lực máy tính hiệu quả.
- **Giảm Thời Gian Xử Lý:** Bằng cách phân chia công việc thành nhiều luồng, thư viện threading giúp giảm thời gian xử lý cho các tác vụ phức tạp.

Nhược điểm:

- **Khả Năng Xảy Ra Lỗi:** Việc sử dụng luồng có thể dễ dàng gây ra các lỗi như độn độ (race condition), cần phải cẩn thận khi thiết kế và quản lý luồng để tránh các vấn đề không mong muốn.
- **Khó Khăn Trong Debug:** Debugging các vấn đề liên quan đến luồng có thể phức tạp và khó khăn hơn so với debugging các ứng dụng tuần tự.
- **Tiêu Thụ Nguồn Lực:** Sử dụng quá nhiều luồng có thể dẫn đến tiêu thụ nhiều nguồn lực của hệ thống, làm giảm hiệu suất hoặc gây ra các vấn đề về bộ nhớ.

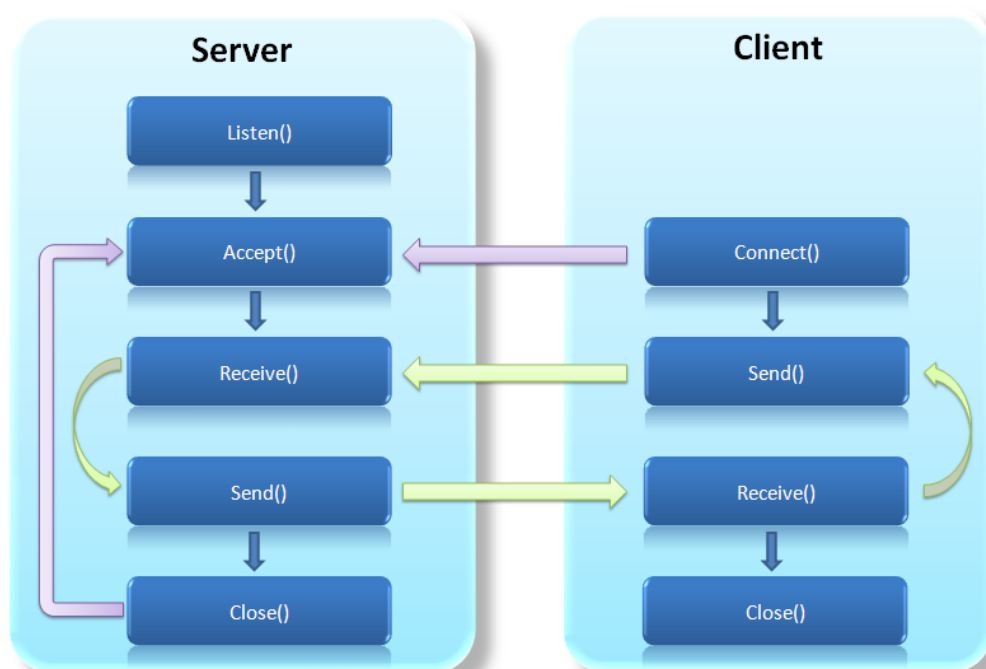
PHẦN 2: THIẾT KẾ ỨNG DỤNG

2.1 Cấu trúc mã nguồn

- File chính để chạy chương trình: `run.py`
- File cấu hình chương trình: `settings.py`, `setting_menu.py`.
- File xử lý logic game: `game.py`
- File tạo giao diện và xử lý giao diện: `main.py`, `menu.py`
- File xử lý socket: `server.py`, `main.py`

2.2 Mô hình ứng dụng

Game Caro 2 người chơi kết nối bằng Socket sử dụng mô hình Client-Server. Nhìn chung mô hình được mô tả chi tiết bằng hình bên dưới.



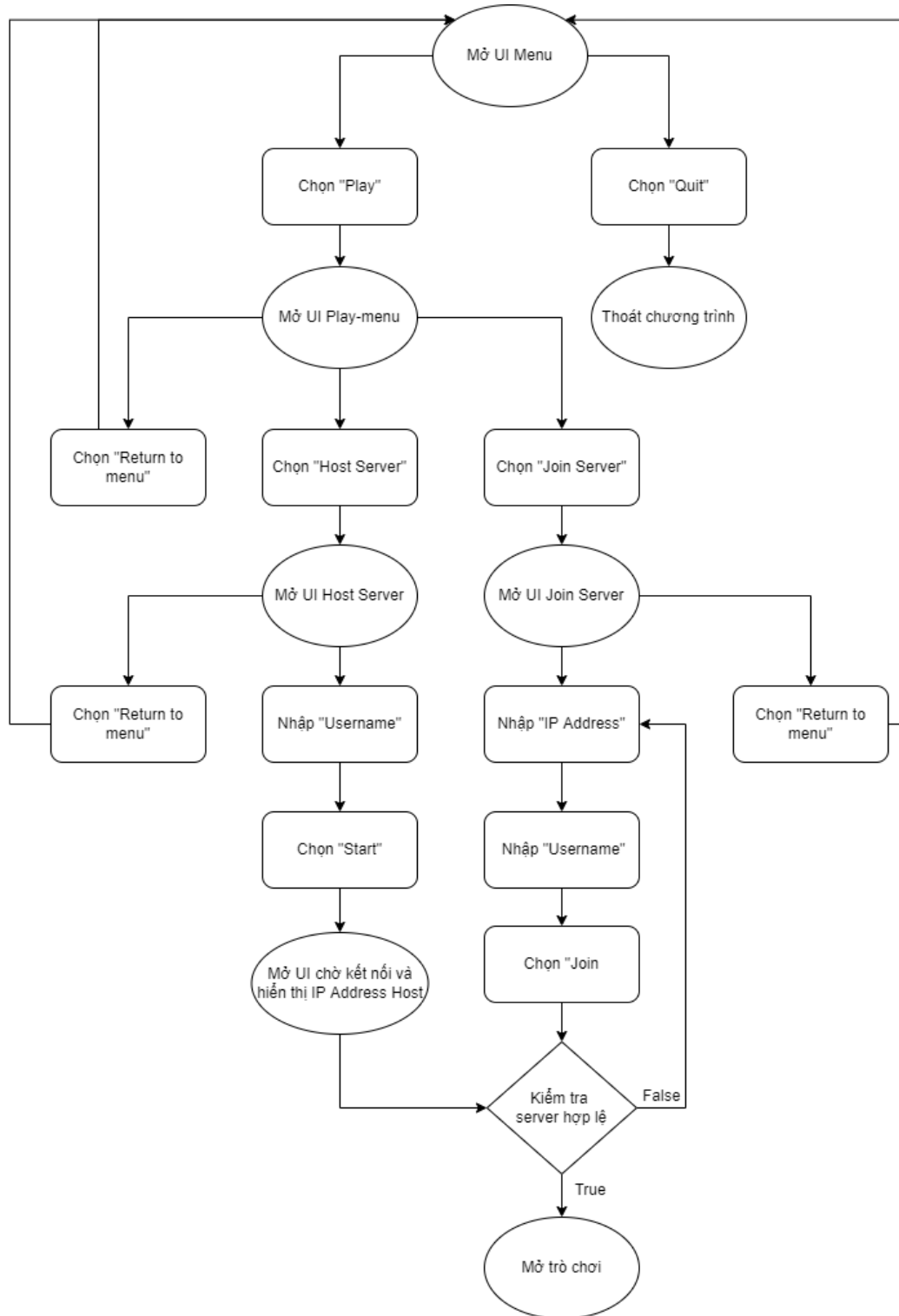
2.3 Các tính năng được xây dựng

- **Giao diện menu và trò chơi:** Tạo giao diện menu để kết nối socket và khởi động trò chơi. Tạo giao diện trò chơi bao gồm bàn cờ, các ô cờ, con trỏ chuột,

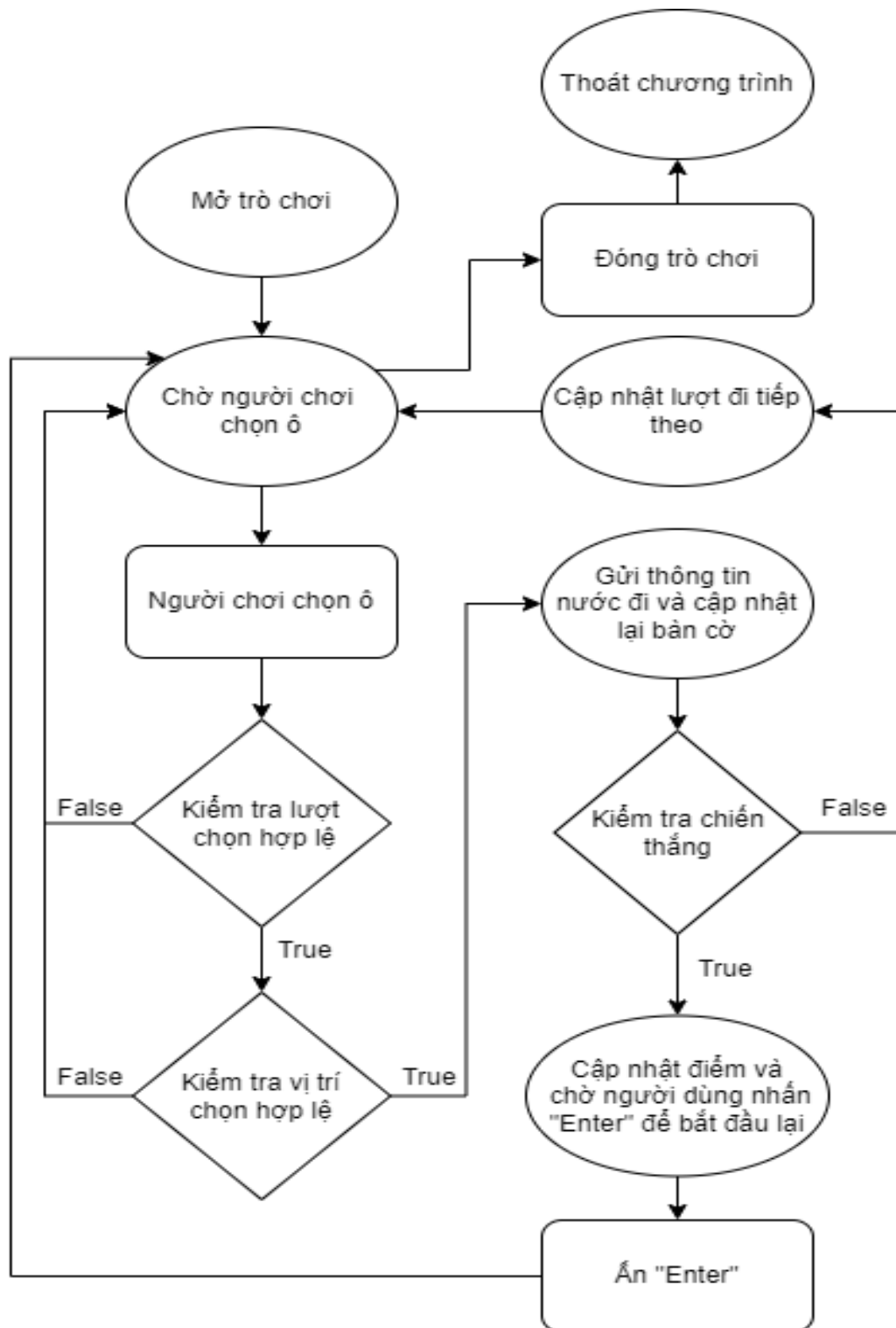
kí hiệu đại diện là "X" hoặc "O", hiển thị lượt chơi, tên người chơi và bảng điểm

- **Kết nối mạng socket:** Mỗi người chơi sẽ chạy một ứng dụng trên máy tính và một máy host sẽ gọi là Server và máy còn lại kết nối với host phải cùng địa chỉ mạng thông qua socket sẽ là Client để gửi và nhận dữ liệu qua lại.
- **Xử lý các nước đi:** Khi người chơi thực hiện một nước đi, Client sẽ gửi thông tin về nước đi đến Server, sau đó Server kiểm tra tính hợp lệ của nước đi và cập nhật trạng thái bàn cờ và điều tương tự được thực hiện khi gửi thông tin về nước đi từ Server đến Client.
- **Kiểm tra chiến thắng:** Mỗi lượt nước đi của người chơi chương trình sẽ kiểm tra xem có tạo thành dãy đủ hàng ngang, hàng dọc hoặc đường chéo để xác định người chiến thắng và tạm dừng cuộc chơi ở cả 2 phía.
- **Xử lý lượt đi lần lượt:** Mỗi lượt nước đi của người chơi sau khi kiểm tra chiến thắng và không đủ điều kiện thắng thì sẽ thay đổi lượt và thông báo cho cả 2 người chơi biết lượt đi tiếp theo là của ai.
- **Gửi thông báo và ghi nhận điểm số:** Khi một người chơi chiến thắng, chương trình sẽ gửi thông báo tới cả hai người chơi để thông báo kết quả và chấm dứt trò chơi. Sau khi bắt đầu lại trò chơi, chương trình sẽ ghi nhận lại điểm số cho người chơi chiến thắng tương ứng.

2.4 Flowchart



Sơ đồ luồng dữ liệu menu

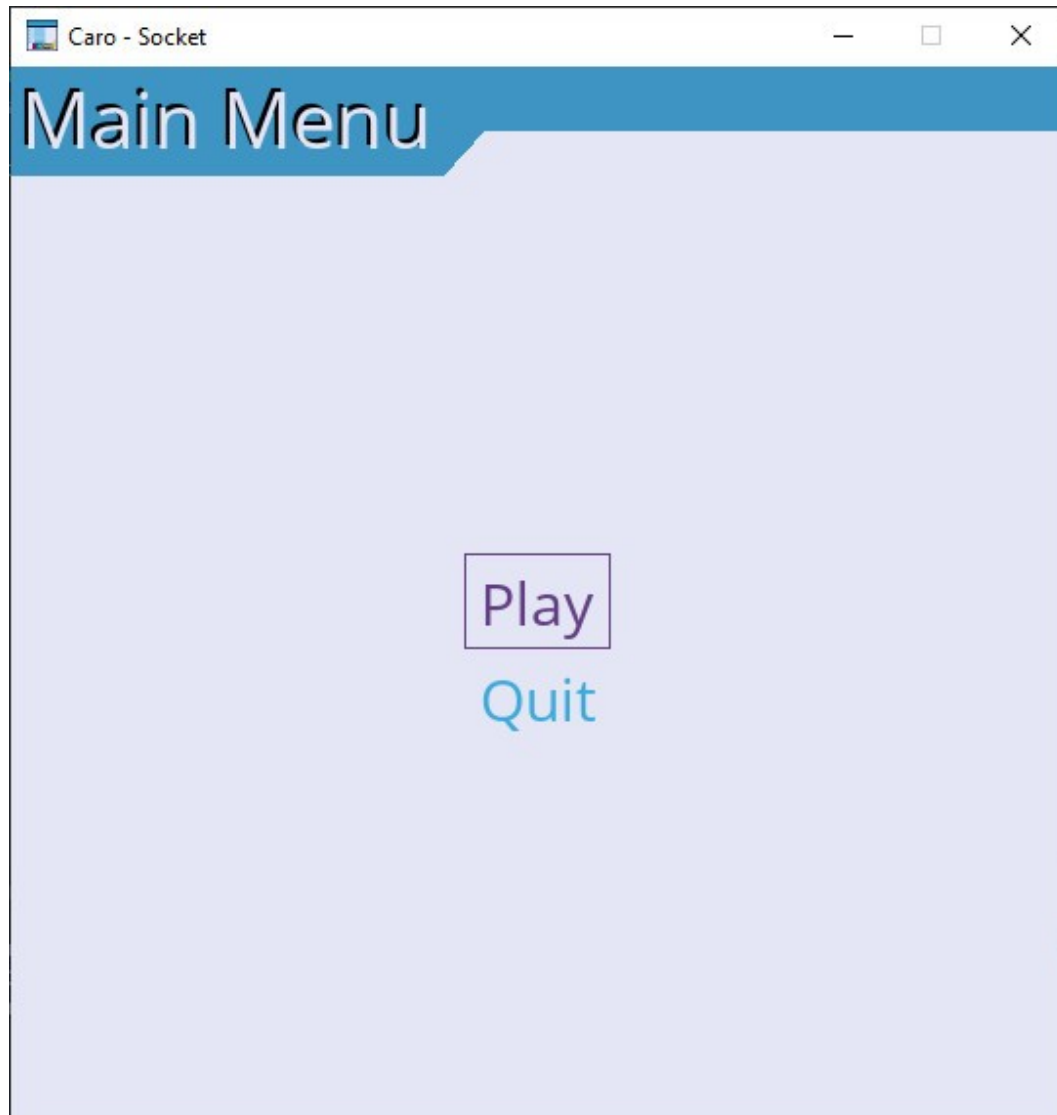


Sơ đồ luồng dữ liệu trò chơi

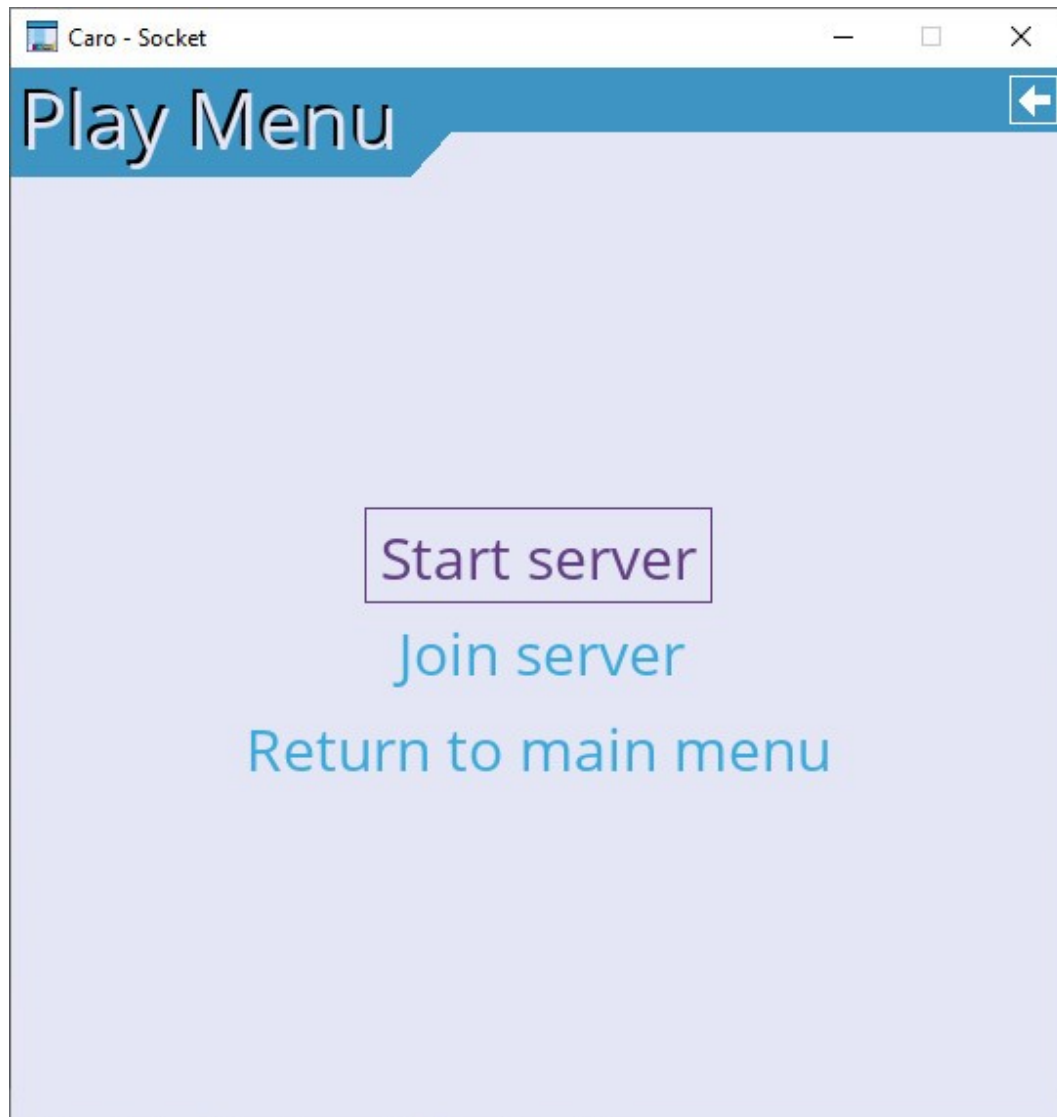
PHẦN 3: KẾT QUẢ THỰC HIỆN

3.1 Giao diện menu và trò chơi

3.1.1 Giao diện menu

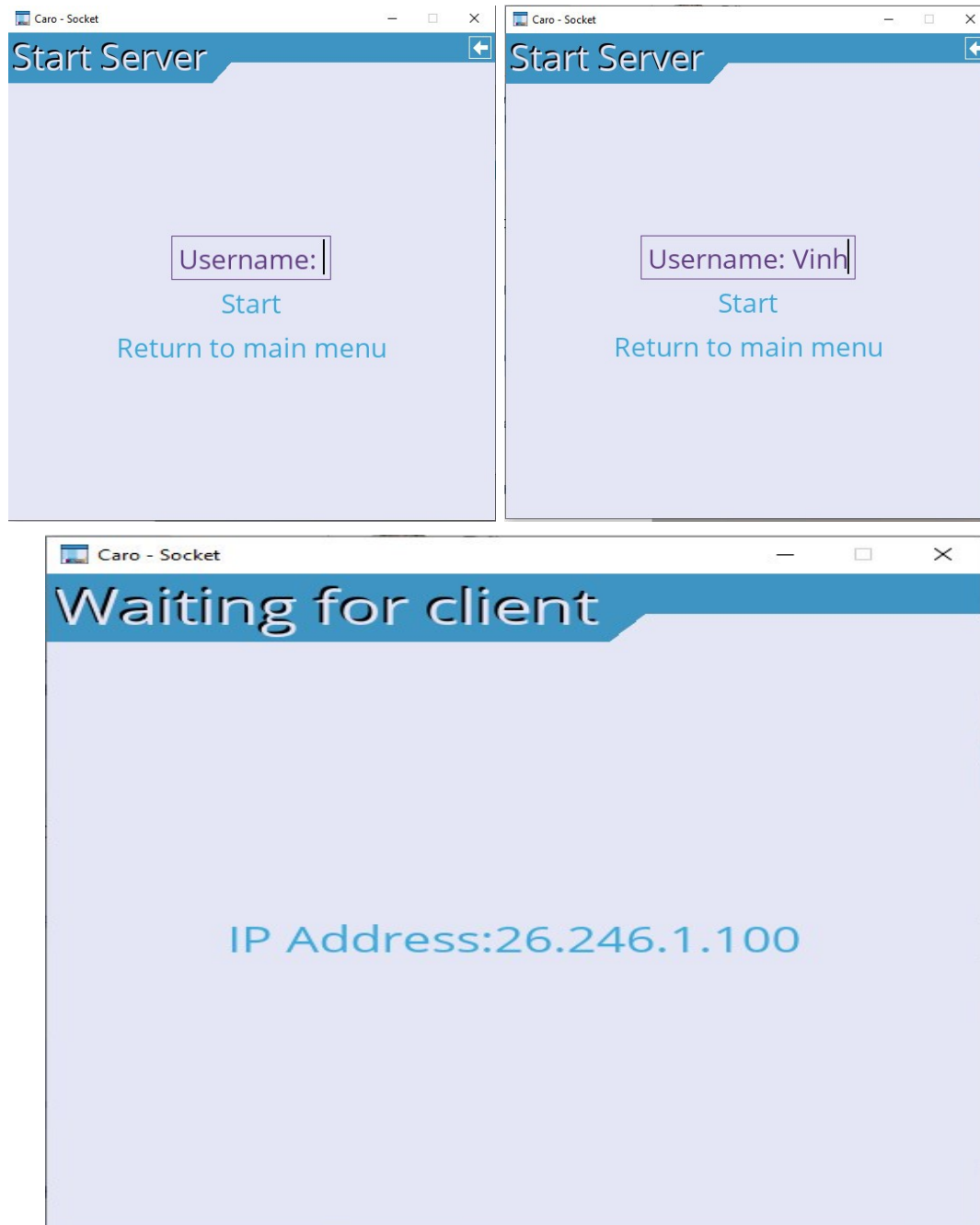


3.1.2 Giao diện trò chơi



3.2 Kết nối mạng socket

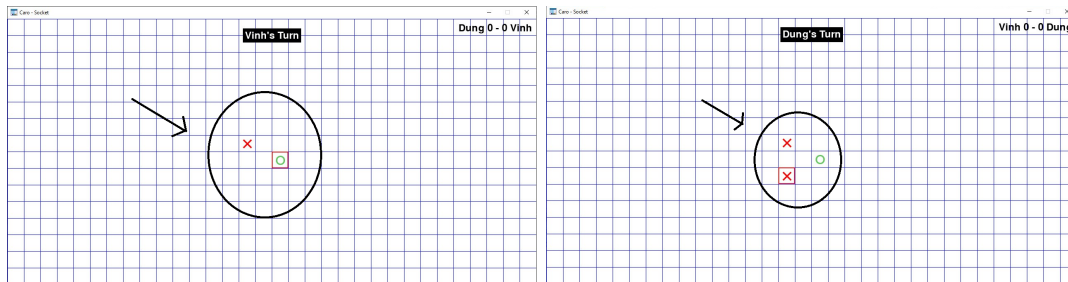
3.2.1 Host Server



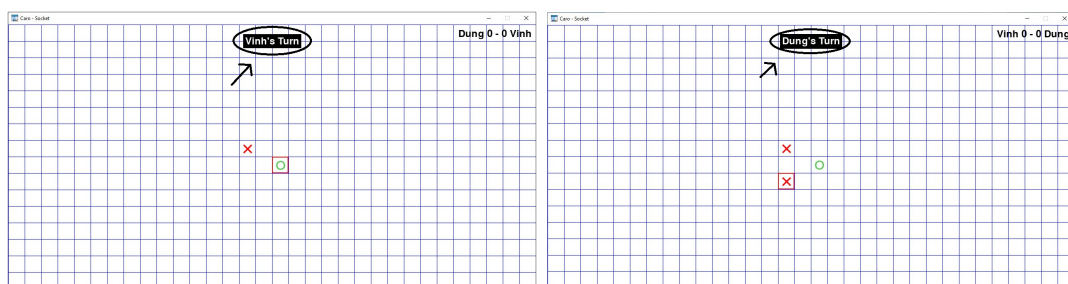
3.2.2 Join Server



3.3 Xử lý các nước đi

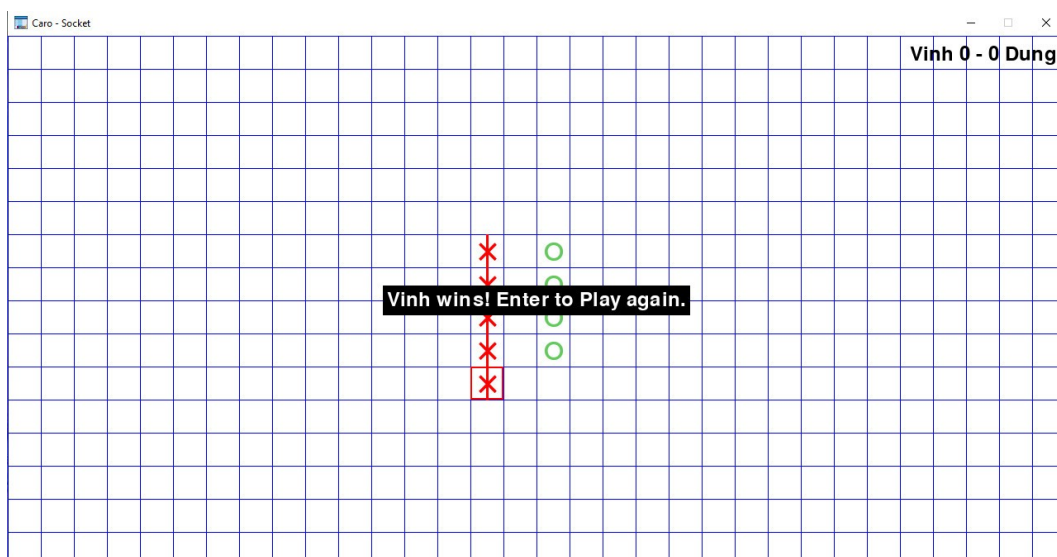


3.4 Xử lý lượt đi lần lượt

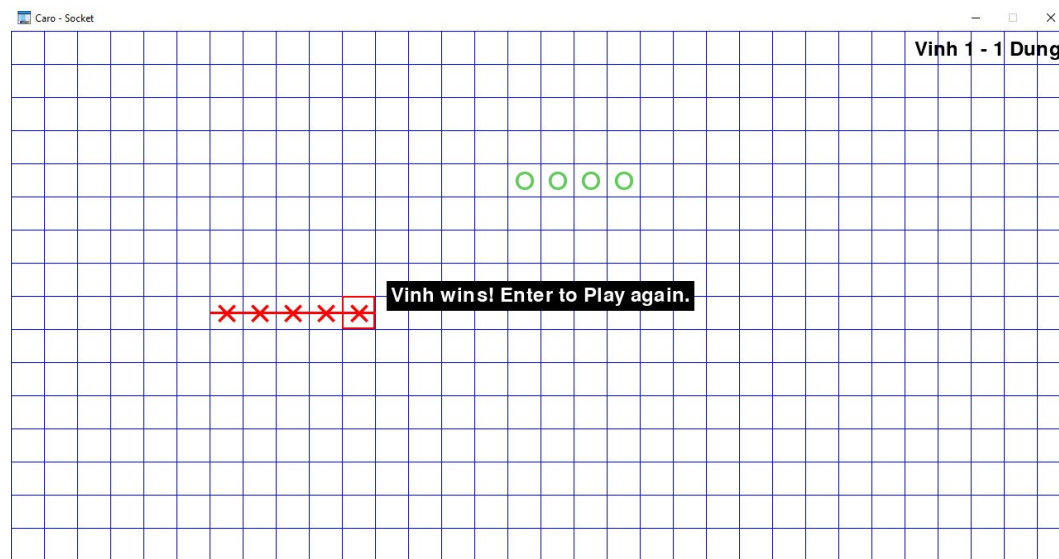


3.5 Kiểm tra chiến thắng

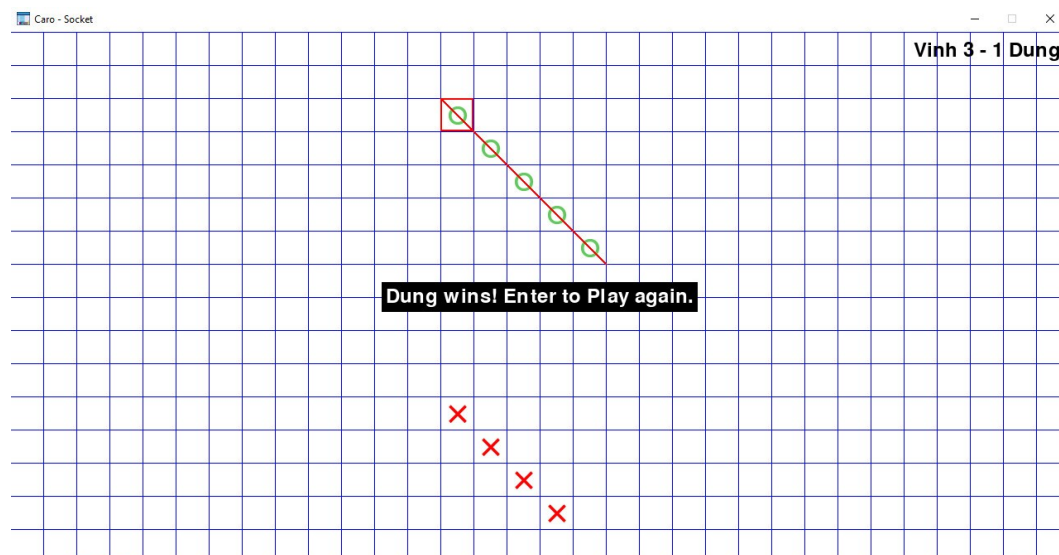
Check khi đánh thẳng bằng hàng dọc



Check khi đánh thẳng bằng hàng ngang

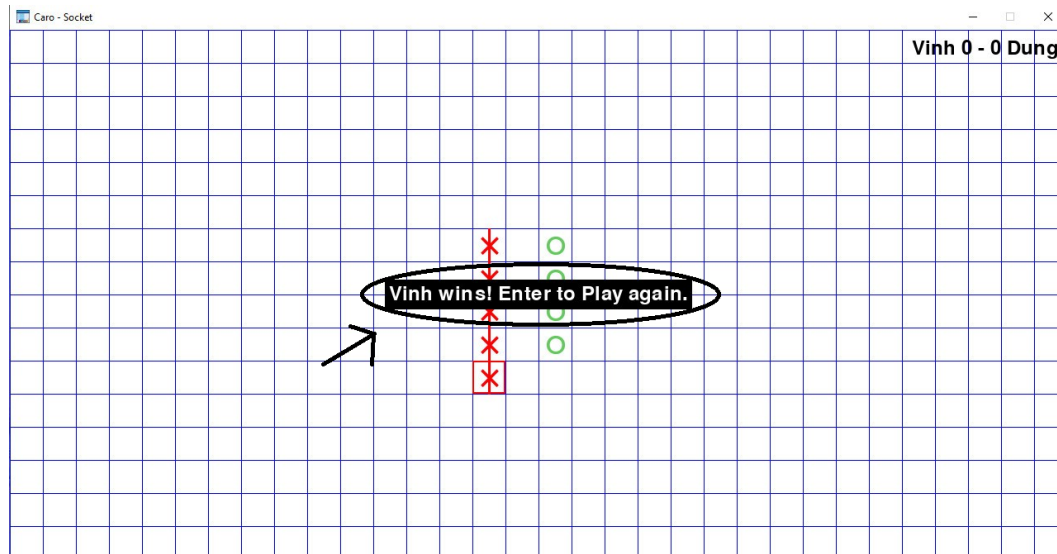


Check khi đánh thẳng bằng hàng chéo

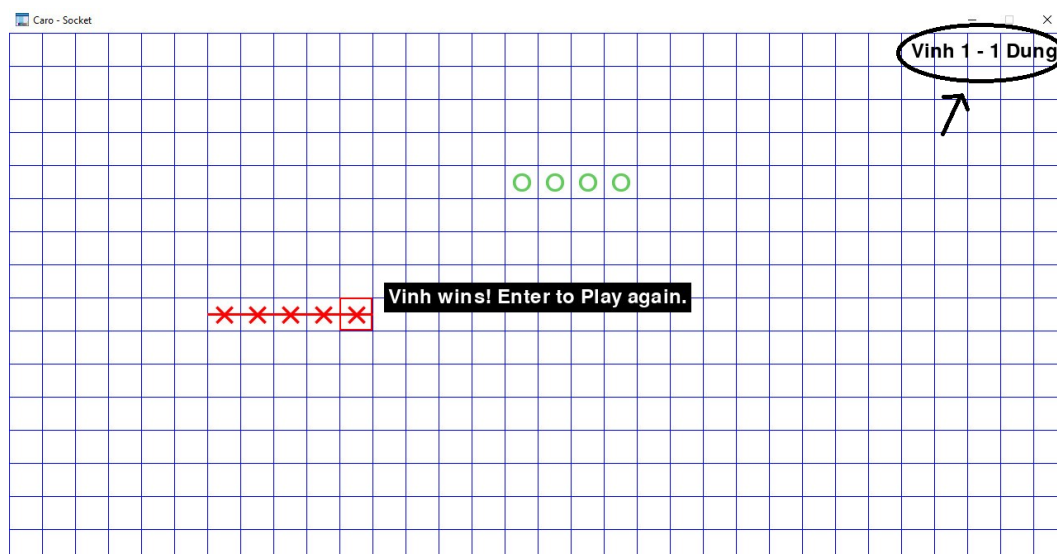


3.6 Gửi thông báo và ghi nhận điểm số

3.6.1 Gửi thông báo



3.6.2 Ghi nhận điểm số



PHẦN 4: HƯỚNG DẪN CÀI ĐẶT ỨNG DỤNG VÀ MÔI TRƯỜNG

Bước 1: Tải xuống source-code từ GitHub

Tải xuống source-code từ GitHub thông qua đường dẫn sau:

<https://github.com/DungIT2k2/Caro-Socket>

Giải nén và thu được thư mục game Caro.

Bước 2: Cài đặt python với hệ điều hành tương thích

Đối với Windows:

- Truy cập đường dẫn sau: <https://www.python.org/downloads/>
- Tải file .exe và tiến hành cài đặt trên máy tính.

Đối với Ubuntu:

- Mở Terminal và nhập lệnh:

```
sudo apt install python3
```

- Tiếp theo tiến hành cài đặt pip:

```
sudo apt install -y python3-pip
```

- Nâng cấp pip và setuptools mới nhất bằng lệnh:

```
pip3 install --upgrade pip
```

```
pip3 install --upgrade setuptools
```

Bước 3: Cài đặt các thư viện cần thiết

Mở Terminal trong thư mục Caro, và tiến hành cài đặt thư viện

Đối với Windows:

- Sử dụng lệnh:

```
pip install -r requirements.txt
```

Đối với Ubuntu:

- Sử dụng lệnh:

```
pip3 install -r requirements.txt
```

- Cài đặt thêm thư viện "tk":

```
sudo apt install python3-tk
```

Bước 4: Chạy ứng dụng

Tiến hành chạy ứng dụng trên hệ điều hành tương thích **Đối với Windows:**

- Sử dụng lệnh:

```
python run.py
```

Đối với Ubuntu:

- Sử dụng lệnh:

```
python3 run.py
```




NHIỆM VỤ VÀ VAI TRÒ CỦA TỪNG THÀNH VIÊN

Thành viên 1: Nguyễn Văn Tiến Dũng - 3120410084

- Thiết kế giao diện trò chơi
- Xây dựng socket Client-Server
- Xử lý logic nước đi, lượt đi và kết nối đến socket
- Viết báo cáo bằng Latex.

Thành viên 2: Lai Quang Vinh - 3120410613

- Thiết kế giao diện menu.
- Xử lý kiểm tra chiến thắng, ghi điểm và kết nối đến socket
- Thiết kế web tĩnh về thông tin chương trình
- Viết báo cáo bằng Latex.



TÀI LIỆU THAM KHẢO

- [1] Dr. John Hunt.(2019). Advanced Guide to Python 3 Programming, Chapter 12. Nhà xuất bản: Springer
- [2] Hướng dẫn xây dựng game Caro cơ bản bằng Python ([Video](#))
- [3] Tài liệu giới thiệu về pygame-menu ([Link](#))
- [4] (Guide) Socket Programming in Python ([Link](#))
- [5] Python GUI Programming With Tkinter ([Link](#))
- [6] Hướng dẫn kết nối Socket (UDP) cho ứng dụng lần lượt ([Video](#))