**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------------------------------------**



**BÁO CÁO BÀI TẬP LỚN**

**LẬP TRÌNH GAME CỜ CARO BẰNG C# WINFORM**

**GVHD:**  Ths. Phạm Huy Thông

**Nhóm:** 01

**Thành viên:** Hà Xuân Phúc

Đinh Quang Nam

Nguyễn Tuấn Minh

Hoàng Quốc Quý

Trần Bá Hải

**Lớp:** 20232IT6040001 **Khóa:** K17

Hà Nội – 2024

# MỤC LỤC

**[CHƯƠNG I: TỔNG QUAN VỀ TRÒ CHƠI XO](#_heading=h.oyhmj0xtr0ap) 4**

[1. Giới thiệu chung](#_heading=h.106y0u4ho0qn) 4

[1.1. Mục đích](#_heading=h.tgnylcccxdmu) 5

[1.2. Phạm vi](#_heading=h.x35out1cqjj4) 5

[1.3. Tài liệu kham khảo](#_heading=h.uos7vn9etj8g) 6

[2. Mô tả chung về hệ thống](#_heading=h.h2140ef9ed2a) 7

[2.1. Danh sách các tác nhân và mô tả](#_heading=h.xjhn1jt1xvz) 7

[2.2. Danh sách các chức năng trong ứng dụng](#_heading=h.v9pi7nopo2v9) 8

**[CHƯƠNG II: THIẾT KẾ PHẦN MỀM](#_heading=h.gjdgxs) 9**

[1. Cơ sở dữ liệu](#_heading=h.izro6bfxgvyv) 9

[1.1. Biểu đồ thực thể liên kết ERD](#_heading=h.uzzzqrg1cbjl) 9

[1.2. Thiết kế bảng](#_heading=h.l8v0l5boeuzg) 9

[2. Mô tả use case](#_heading=h.izro6bfxgvyv) 10

[2.1. Các use case tổng quan](#_heading=h.s3ku3y3jbuf) 10

[2.2. Các use case front end](#_heading=h.lf7b2jmj38s8) 11

[2.3. Các use case backend](#_heading=h.jtph0uo84wb) 11

[2.4. Mô tả chi tiết use case](#_heading=h.1or7lv48yy9z) 11

[2.4.1. Mô tả use case xem lịch sử đấu](#_heading=h.w8mrfirlt59f) 11

[2.4.2. Mô tả use case bắt đầu trận đấu mới](#_heading=h.t19l4ou8qfld) 12

[2.4.3. Mô tả use case đi lại nước cờ cũ](#_heading=h.x0ggku9szan) 13

[2.4.4. Mô tả use case kết nối hai người chơi](#_heading=h.igeakb67eabv) 14

[2.4.5. Mô tả use case xóa lịch sử đấu](#_heading=h.ogahlwl4ag7t) 15

[3. Phân tích use case](#_heading=h.ot8iqfpsigb3) 16

[3.1. Phân tích use case xem lịch sử đấu](#_heading=h.qabcpyt9wrzt) 16

[3.1.1. Biểu đồ VOPC](#_heading=h.ea784lru710k) 16

[3.1.2. Biểu đồ trình tự](#_heading=h.r6fv16ka0jt8) 17

[3.2. Phân tích use case bắt đầu trận đấu mới](#_heading=h.nek80bd89f9o) 18

[3.2.1. Biểu đồ VOPC](#_heading=h.8eswc9926xm2) 18

[3.2.2. Biểu đồ trình tự](#_heading=h.fn2xwai5akdh) 19

[3.3. Phân tích use case đi lại nước cờ cũ](#_heading=h.mx947so6nibw) 20

[3.3.1. Biểu đồ VOPC](#_heading=h.j046gmwegoha) 20

[3.3.2. Biểu đồ trình tự](#_heading=h.qmbrta16jrii) 21

[3.4. Phân tích use case kết nối hai người chơi](#_heading=h.5a47zzg31lcc) 22

[3.4.1. Biểu đồ VOPC](#_heading=h.xke2ibxm9h5m) 22

[3.4.2. Biểu đồ trình tự](#_heading=h.4zjr7qs3eivt) 23

[3.5. Phân tích use case xóa lịch sử đấu](#_heading=h.x22xinu7sh03) 24

[3.5.1. Biểu đồ VOPC](#_heading=h.qo2qxs5ubim7) 24

[3.5.2. Biểu đồ trình tự](#_heading=h.fy78nakte8zp) 25

[4. Thiết kế giao diện](#_heading=h.91v8cu27homa) 26

[4.1. Giao diện use case xem lịch sử đấu](#_heading=h.dcchewtecnmw) 26

[4.1.1. Hình dung màn hình](#_heading=h.d291stxhaf5c) 26

[4.1.2. Biểu đồ lớp màn hình](#_heading=h.9c0smpaux4nu) 26

[4.1.3. Biểu đồ cộng tác của màn hình](#_heading=h.koq0xcr2kwcv) 27

[4.2. Giao diện use case bắt đầu trận đấu mới](#_heading=h.wnvdz8ij0zbf) 27

[4.2.1. Hình dung màn hình](#_heading=h.d291stxhaf5c) 27

[4.2.2. Biểu đồ lớp màn hình](#_heading=h.9c0smpaux4nu) 28

[4.2.3. Biểu đồ cộng tác của màn hình](#_heading=h.koq0xcr2kwcv) 28

[4.3. Giao diện use case đi lại nước cờ cũ](#_heading=h.dcokx3v2t6z) 29

[4.3.1. Hình dung màn hình](#_heading=h.d291stxhaf5c) 29

[4.3.2. Biểu đồ lớp màn hình](#_heading=h.9c0smpaux4nu) 29

[4.3.3. Biểu đồ cộng tác của màn hình](#_heading=h.koq0xcr2kwcv) 30

[4.4. Giao diện use case kết nối hai người chơi](#_heading=h.p79b9ppf2t5t) 30

[4.4.1. Hình dung màn hình](#_heading=h.d291stxhaf5c) 30

[4.4.2. Biểu đồ lớp màn hình](#_heading=h.9c0smpaux4nu) 31

[4.4.3. Biểu đồ cộng tác của màn hình](#_heading=h.koq0xcr2kwcv) 31

[4.5. Giao diện use case xóa lịch sử đấu](#_heading=h.q3b5vw4a9ey7) 32

[4.5.1. Hình dung màn hình](#_heading=h.d291stxhaf5c) 32

[4.5.2. Biểu đồ lớp màn hình](#_heading=h.9c0smpaux4nu) 32

[4.5.3. Biểu đồ cộng tác của màn hình](#_heading=h.koq0xcr2kwcv) 33

**[CHƯƠNG III: KẾT QUẢ THỰC NGHIỆM](#_heading=h.9k3y3nmda241) 34**

[1. Hình ảnh màn hình trang chủ](#_heading=h.oybifjvah8c3) 34

[2. Hình ảnh giao diện khi chơi game](#_heading=h.ig0kmymnhs0c) 35

[3. Hình ảnh khi đang kết nối mang LAN](#_heading=h.k2lgjlewqzh6) 36

[4. Hình ảnh các chức năng có trong menu](#_heading=h.tvnb5voi97rs) 37

**[KẾT LUẬN](#_heading=h.9w6ti2l49d1) 39**

# 

# CHƯƠNG I: TỔNG QUAN VỀ TRÒ CHƠI XO

## Giới thiệu chung

Trò chơi Tic Tac Toe (XO) lập trình bằng C# và WinForms là một ứng dụng đơn giản nhưng hấp dẫn, thường được sử dụng để học lập trình cơ bản và làm quen với giao diện người dùng Windows. Dưới đây là một tổng quan về trò chơi Tic Tac Toe lập trình bằng C# WinForms:

1. **Giao diện người dùng (UI)**:
   * Bảng chơi Tic Tac Toe được hiển thị trong một cửa sổ ứng dụng.
   * Ô vuông trên bảng được tạo thành từ các Control hoặc PictureBox để hiển thị các biểu tượng X và O.
2. **Logic của trò chơi**:
   * Logic của trò chơi quản lý việc xác định người chơi hiện tại và xử lý các bước của họ trên bảng chơi.
   * Nó kiểm tra các điều kiện chiến thắng sau mỗi nước đi để xác định người chơi nào chiến thắng hoặc nếu trò chơi kết thúc hòa.
3. **Chức năng của trò chơi**:
   * Khởi tạo bảng chơi và các biểu tượng ban đầu (ví dụ: ô trống).
   * Xử lý sự kiện khi người chơi nhấp vào một ô trên bảng.
   * Hiển thị kết quả của trò chơi sau mỗi nước đi và cuối cùng khi có người chiến thắng hoặc hòa.
   * Có thể kết nối 2 người chơi ở hai máy khác nhau
4. **Tính năng bổ sung**:
   * Có thể thêm các tính năng như xác định người chơi thắng sau mỗi trận đấu, tính điểm, hoặc thậm chí là tạo chế độ chơi đơn người với máy tính.
5. **Giao diện người dùng đẹp mắt**:
   * Trò chơi có thể được thiết kế với giao diện người dùng đẹp mắt và thân thiện với người dùng, bao gồm các biểu tượng và hiệu ứng hấp dẫn.

Việc lập trình trò chơi Tic Tac Toe trong C# WinForms là một dự án thú vị và giúp bạn hiểu rõ hơn về lập trình giao diện người dùng và xử lý logic trò chơi trong môi trường Windows.

### Mục đích

Mục đích của việc lập trình trò chơi Tic Tac Toe bằng C# WinForms có thể được mô tả như sau:

1. **Học lập trình cơ bản**: Trò chơi Tic Tac Toe là một dự án lập trình phổ biến cho người mới bắt đầu với lập trình C# và WinForms. Nó cung cấp một dự án thực tế và thú vị để áp dụng những kiến thức đã học như điều khiển giao diện người dùng, xử lý sự kiện và quản lý logic ứng dụng.
2. **Thực hành xử lý sự kiện**: Việc lập trình trò chơi Tic Tac Toe giúp người học thực hành việc xử lý các sự kiện người dùng như nhấp chuột vào các ô trên bảng chơi và xử lý hành động tương ứng.
3. **Hiểu về logic game**: Trò chơi Tic Tac Toe đơn giản nhưng cần phải hiểu rõ về logic của nó, bao gồm cách kiểm tra điều kiện chiến thắng và xử lý trạng thái của bảng chơi. Việc này giúp người học phát triển kỹ năng tư duy logic và giải quyết vấn đề.
4. **Tạo ra một ứng dụng thực tế**: Lập trình trò chơi Tic Tac Toe bằng C# WinForms giúp người học tạo ra một ứng dụng thực tế có thể chơi được. Điều này mang lại sự hứng thú và tự hào khi hoàn thành một dự án độc lập.
5. **Mở rộng và tùy chỉnh**: Sau khi hoàn thành, người học có thể mở rộng và tùy chỉnh trò chơi Tic Tac Toe theo ý muốn, bằng cách thêm tính năng mới như chế độ chơi với máy tính hoặc cải thiện giao diện người dùng.

Tóm lại, mục đích của việc lập trình trò chơi Tic Tac Toe bằng C# WinForms là học và thực hành kỹ năng lập trình cơ bản, logic game, và tạo ra một ứng dụng thực tế có thể chơi được.

### Phạm vi

Phạm vi của việc lập trình trò chơi Tic Tac Toe bằng C# WinForms có thể được xác định như sau:

1. **Giao diện người dùng (UI)**:
   * Thiết kế và triển khai giao diện người dùng thân thiện và dễ sử dụng.
   * Hiển thị bảng chơi Tic Tac Toe và các ô trống.
   * Xử lý sự kiện khi người dùng nhấp vào các ô trên bảng.
2. **Logic của trò chơi**:
   * Xác định và thực hiện các quy tắc chơi Tic Tac Toe.
   * Kiểm tra và xử lý các trường hợp chiến thắng hoặc hòa.
   * Quản lý trạng thái của bảng chơi và các nước đi của người chơi.
3. **Tính năng cơ bản**:
   * Cho phép hai người chơi tham gia trò chơi.
   * Hiển thị kết quả của trò chơi sau mỗi nước đi.
   * Cho phép người chơi bắt đầu trò chơi mới sau khi kết thúc một trận.
   * Thiết lập kết nối giữa các máy tính trong mạng LAN để người chơi có thể tìm kiếm và kết nối với nhau.
4. **Tính linh hoạt và mở rộng**:
   * Dễ dàng mở rộng và tùy chỉnh tính năng của trò chơi.
   * Thêm tính năng như chế độ chơi với máy tính hoặc chế độ chơi online.
   * Tùy chỉnh giao diện người dùng và hiệu ứng hình ảnh để làm cho trò chơi thú vị hơn.
5. **Thử nghiệm và sửa lỗi**:
   * Thực hiện kiểm thử để đảm bảo tính ổn định và đúng đắn của ứng dụng.
   * Sửa lỗi và cải thiện trải nghiệm người dùng dựa trên phản hồi và phát hiện lỗi trong quá trình thử nghiệm.

Với phạm vi này, bạn có thể tạo ra một ứng dụng Tic Tac Toe hoàn chỉnh, có khả năng chơi và thực hiện logic của trò chơi một cách chính xác và dễ hiểu.

### Tài liệu kham khảo

Dưới đây là một số tài liệu tham khảo có thể giúp bạn trong việc lập trình trò chơi Tic Tac Toe bằng C# WinForms:

1. **Microsoft Docs - Windows Forms**: Trang web chính thức của Microsoft cung cấp hướng dẫn chi tiết về lập trình Windows Forms bằng C#. Bạn có thể tìm thấy các ví dụ và hướng dẫn về cách sử dụng các Control và xử lý sự kiện.
   * Link: [Microsoft Docs - Windows Forms](https://docs.microsoft.com/en-us/dotnet/desktop/winforms/?view=netdesktop-5.0)
2. **Tutorialspoint - C# Windows Forms**: Trang web này cung cấp các bài hướng dẫn từ cơ bản đến nâng cao về lập trình Windows Forms bằng C#. Bạn có thể tìm thấy hướng dẫn về tạo giao diện người dùng, xử lý sự kiện, và thực hiện các tính năng của ứng dụng.
   * Link: Tutorialspoint - C# Windows Forms
3. **Codeproject - Tic Tac Toe Game in C#**: Trang web này cung cấp một số dự án mẫu và mã nguồn mở về trò chơi Tic Tac Toe được lập trình bằng C# WinForms. Bạn có thể tham khảo mã nguồn này để hiểu cách thực hiện các tính năng cơ bản của trò chơi.
   * Link: Codeproject - Tic Tac Toe Game in C#
4. **Youtube Tutorials**: Youtube cung cấp nhiều video hướng dẫn về lập trình Tic Tac Toe bằng C# WinForms. Bạn có thể tìm kiếm các video hướng dẫn từ các kênh như "The Coding Train" hoặc "Programming Knowledge".
5. **GitHub Repositories**: Nhiều mã nguồn mở của trò chơi Tic Tac Toe bằng C# WinForms có sẵn trên GitHub. Bạn có thể tìm kiếm và tham khảo mã nguồn này để hiểu cách thực hiện các tính năng cụ thể của trò chơi.
   * Ví dụ: [Tic Tac Toe Game in C#](https://github.com/James231/Tic-Tac-Toe-C-Sharp)

Những tài liệu tham khảo trên sẽ cung cấp cho bạn những kiến thức cần thiết để bắt đầu và phát triển ứng dụng Tic Tac Toe bằng C# WinForms.

## Mô tả chung về hệ thống

### Danh sách các tác nhân và mô tả

|  |  |  |
| --- | --- | --- |
| Tác nhân | Mô tả tác nhân | Ghi chú |
| Player | Là người dùng cuối cùng của ứng dụng, tham gia vào trò chơi Tic Tac Toe bằng cách chọn ô trống trên bảng và đặt biểu tượng của mình (X hoặc O) vào ô đó. |  |
| Người quản trị | Người quản trị là cá nhân hoặc nhóm người chịu trách nhiệm cho việc phát triển, triển khai và duy trì ứng dụng Tic Tac Toe. Họ có thể là các nhà phát triển phần mềm, kỹ sư phần mềm hoặc người quản lý dự án. |  |
| HeThongCSDL | Hệ thống CSDL được sử dụng để lưu trữ và quản lý dữ liệu liên quan đến trò chơi Tic Tac Toe, bao gồm thông tin về người chơi, lịch sử trận đấu, điểm số và các thông tin khác có liên quan. |  |

## 

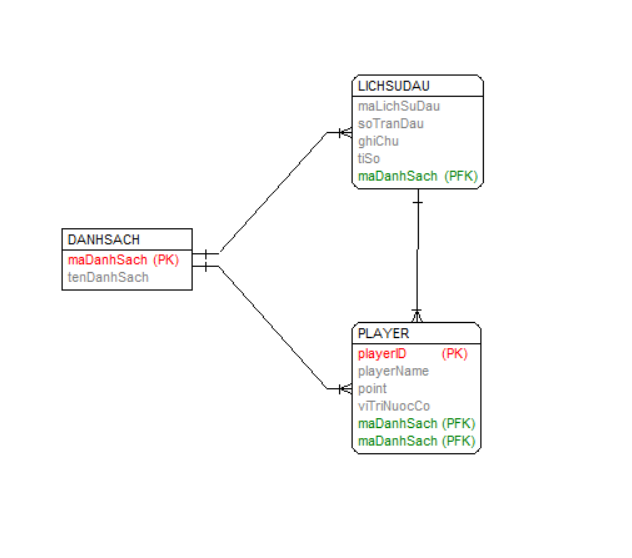
### Danh sách các chức năng trong ứng dụng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tên Use Case | Mô tả ngắn gọn Use case | Chức năng | Ghi chú |
| UC\_001 | Use case xem lịch sử đấu | Cho phép người dùng xem lại các trận đấu đã diễn ra | Hiển thị danh sách các trận đấu  Hiển thị chi tiết từng trận đấu |  |
| UC\_002 | Use case bắt đầu trận đấu mới | Cho phép người dùng bắt đầu trận đấu mới | Thiết lập trận đấu  Tìm đối thủ |  |
| UC\_003 | Use case đi lại nước cờ cũ | Cho phép người dùng đi lại các nước cờ đã chơi trước đó | Hiển thị lịch sử nước cờ  Thực hiện lại nước cờ đã chọn |  |
| UC\_004 | Use case kết nối hai người chơi | Cho phép hai người chơi kết nối và thi đấu với nhau | Tìm và kết nối với các đối thủ  Bắt đầu trận đấu |  |
| UC\_005 | Use case xóa lịch sử đấu | Cho phép người dùng xóa các lịch sử đấu đã chơi | Hiển thị danh sách các trận đấu đã chơi  Cho phép người dùng chọn và xóa trận đấu |  |

# CHƯƠNG II: THIẾT KẾ PHẦN MỀM

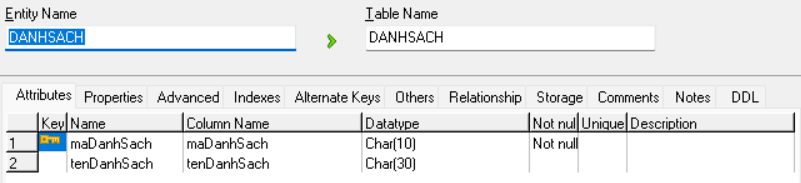
## Cơ sở dữ liệu

### Biểu đồ thực thể liên kết ERD

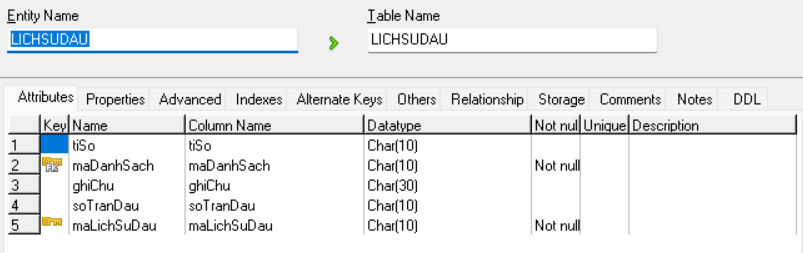
****

### Thiết kế bảng

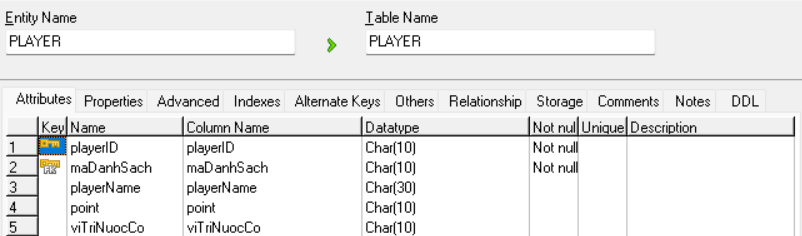
* Bảng DANHSACH



* Bảng LICHSUDAU

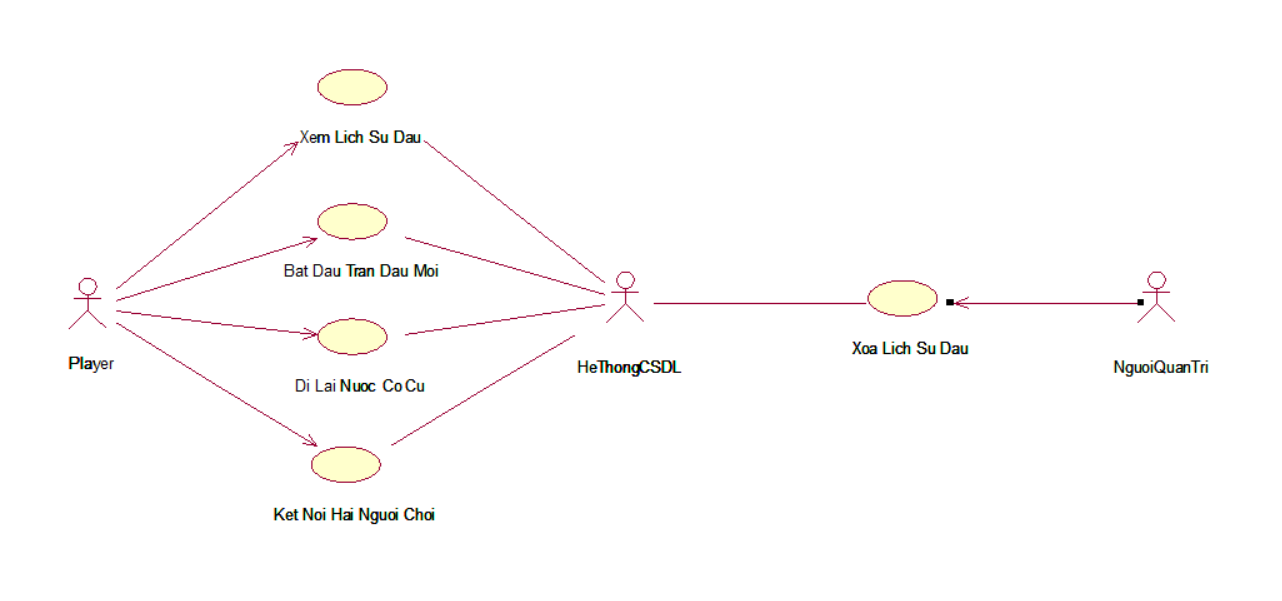


* Bảng PLAYER

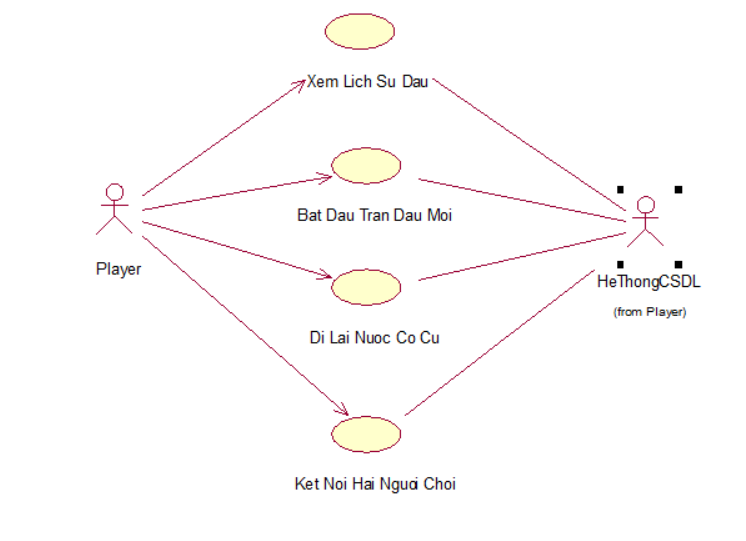
****

## Mô tả use case

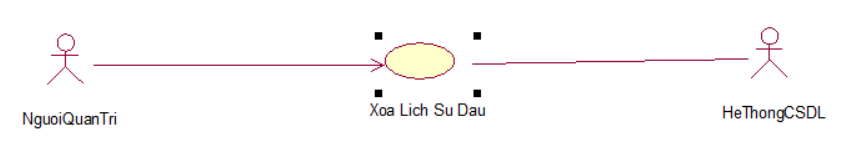
### Các use case tổng quan

****

### Các use case front end

****

### Các use case backend

****

### Mô tả chi tiết use case

#### Mô tả use case xem lịch sử đấu

* **Tên use case**
* Xem lịch sử đấu
* **Mô tả use case**
* Use case cho phép Player xem lịch sử đấu
* **Luồng sự kiện**

1. Luồng cơ bản
2. Use case bắt đầu khi Player kich nút “Xem lịch sử đấu” trong danh mục menu. Hệ thống sẽ lấy thông tin lịch sử đấu từ bảng LICHSUDAU và hiển thị thông tin lên màn hình. Use case kết thúc.
3. Luồng rẽ nhánh
4. Tại bất kỳ thời điểm nào trong luồng cơ bản nếu không kết nối được với cơ sơ dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt**
* Không có
* **Tiền điều kiện**
* Không có
* **Hậu điều kiện**
* Không có
* **Điểm mở rộng**
* Không có

#### Mô tả use case bắt đầu trận đấu mới

* **Tên use case**
* Bắt đầu trận đấu mới.
* **Mô tả vắn tắt**
* Use case bắt đầu trận đấu mới cho phép Player bắt đầu một trận đấu mới.
* **Luồng sự kiện.**

1. **Luồng cơ bản:**
2. Use case bắt đầu khi Player kích vào nút “Menu” trong thanh menu. Hệ thống sẽ hiển thị 1 danh sách các lựa chọn lên màn hình gồm: New game, Undo, Quit game.
3. Player kick vào nút “New game”. Hệ thống sẽ xóa hết những nước cờ đã đi trước đó trong bảng PLAYER và bắt đầu lại một ván mới cho Player. Use case kết thúc.
4. **Luồng rẽ nhánh:**
5. Tại bất kì bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt**
* Không
* **Tiền điều kiện**
* Không
* **Hậu điều kiện**
* không
* **Điểm mở rộng**
* Không

#### Mô tả use case đi lại nước cờ cũ

* **Tên use case**
* Đi lại nước cờ caro cũ
* **Mô tả use case**
* Use case cho phép Player đi lại nước cờ cũ đã đi sai.
* **Luồng sự kiện**

1. **Luồng cơ bản**
2. Use case bắt đầu khi Player chọn chức năng "Đi lại nước cờ caro cũ". Hệ thống hiển thị thông báo “ Bạn có muốn đi lại nước cờ cũ ?”. Player chọn “ Xác Nhận”. Hệ thống lấy thông tin vị trí nước cờ cũ trong bảng PLAYER và hiển thị lại bàn cờ caro với các bước đi trước đó.
3. **Luồng sự kiện**
4. Tại bước 2 trong luồng cơ bản, nếu player chọn “Hủy” thì trận đấu sẽ tiếp tục mà không thực hiện đi lại nước cờ cũ. Use case kết thúc
5. Tại bất kì bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặt biệt**
* Nếu không có trận đấu nào được tìm thấy, hệ thống hiển thị thông báo "Không có trận đấu nào được tìm thấy".
* **Tiền điều kiện**
* Đã kết thúc trận đấu caro và đã lưu lại các bước đi của trận đấu đó.
* **Hậu điều kiện**
* Hiển thị lại bàn cờ caro với các bước đi đã thực hiện.
* **Điểm mở rộng**
* Không có

#### Mô tả use case kết nối hai người chơi

* **Tên use case**
* Kết nối hai người chơi
* **Mô tả use case**
* Use case cho phép kết nối hai người chơi với nhau.
* **Luồng sự kiện**

1. **Luồng cơ bản**
2. Use case sẽ bắt đầu khi Player chọn chức năng “Online gaming” trên màn hình chế độ của màn hình. Hệ thống sẽ hiển thị ra thông tin playerName, địa chỉ IP, và bàn cờ.
3. Player nhập địa chỉ IP muốn kết nối và kích nút “ CONNECT”.Hệ thống sẽ hiển thị lên trên màn hình thông báo “The server created successfully”.

**Luồng rẽ nhánh**

1. Tại bước 1b trong luồng cơ bản nếu Player kích nút “Hủy bỏ”, hệ thống sẽ bỏ qua thao và trở về màn hình chính. Use case kết thúc.
2. Nếu player thứ 2 nhập sai địa chỉ IP thì màn hình sẽ hiển thị thông báo “ Thất bại” và trở về màn hình chính. Use case kết thúc.
3. Tại bất kì bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt**
* Không có
* **Tiền điều kiện**
* Không có
* **Hậu điều kiện**
* Không có
* **Điểm mở rộng**
* Không có

#### Mô tả use case xóa lịch sử đấu

* **Tên use case**
* Xóa lịch sử đấu
* **Mô tả use case**
* Use case cho phép Player xóa lịch sử các trận đã đấu
* **Luồng sự kiện**

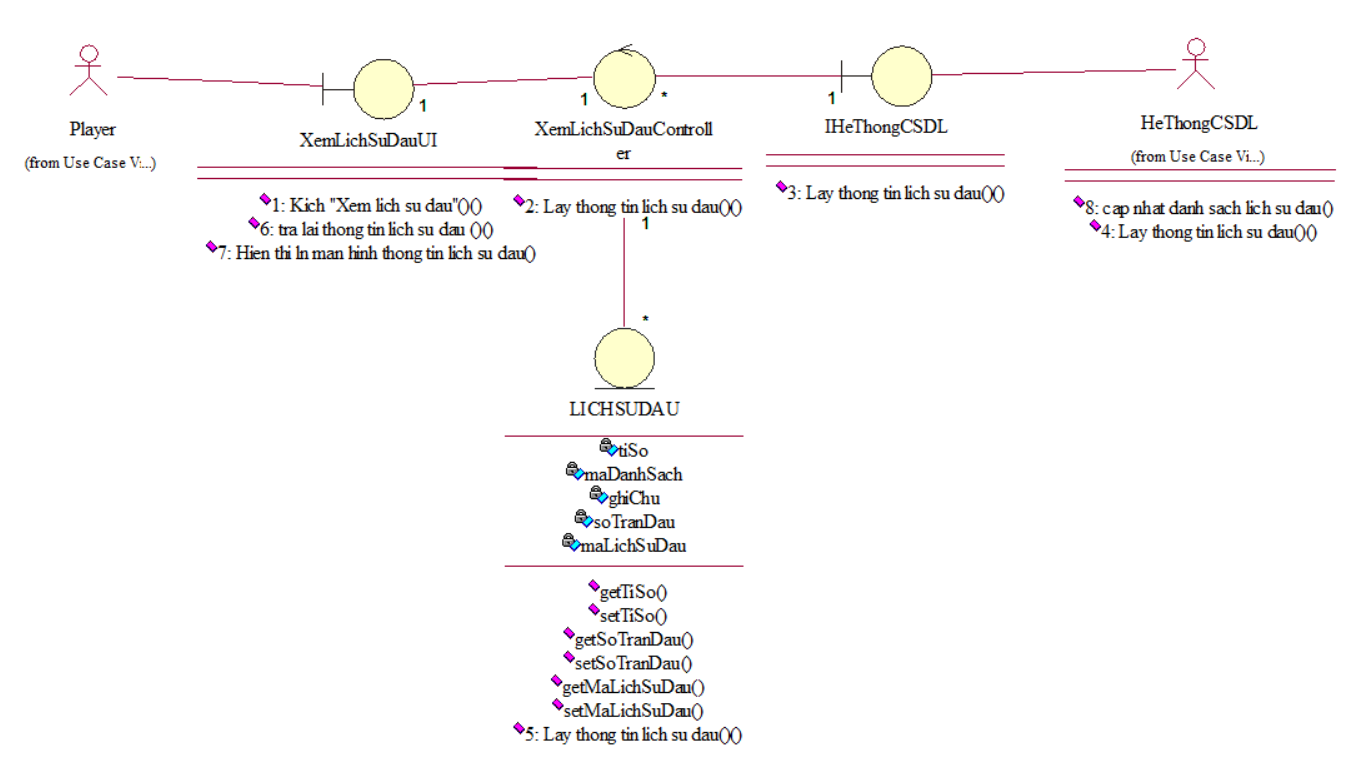
1. **Luồng cơ bản**
2. Use case bắt đầu khi người quản trị nhập “Mã lịch sử đấu” kích nút “Xóa” bên dưới bảng thông tin lịch sử đấu, hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa.
3. Người quản trị kích nút “Đồng ý” hệ thống sẽ xóa thông tin trận đấu có “Mã lịch sử đấu” trong bảng LICHSUDAU, DANHSACH và hiển thị danh sách lịch sử đấu đã cập nhật. Use case kết thúc
4. **Luồng rẽ nhánh**
5. Tại bước 1b trong luồng cơ bản nếu người quản trị kích nút “Không đồng ý”, hệ thống sẽ bỏ qua thao tác xóa và hiển thị danh sách lịch sử đấu. Use case kết thúc.
6. Tại bất kỳ bước nào trong luồng cơ bản nếu không kết nối với cơ sở dữ liệu hệ thống sẽ hiển thị thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt**
* Danh sách trận đấu không bỏ trống.
* **Tiền điều kiện**
* Đăng nhập với vai trò người quản trị
* **Hậu điều kiện**
* Không có
* **Điểm mở rộng**
* Không có

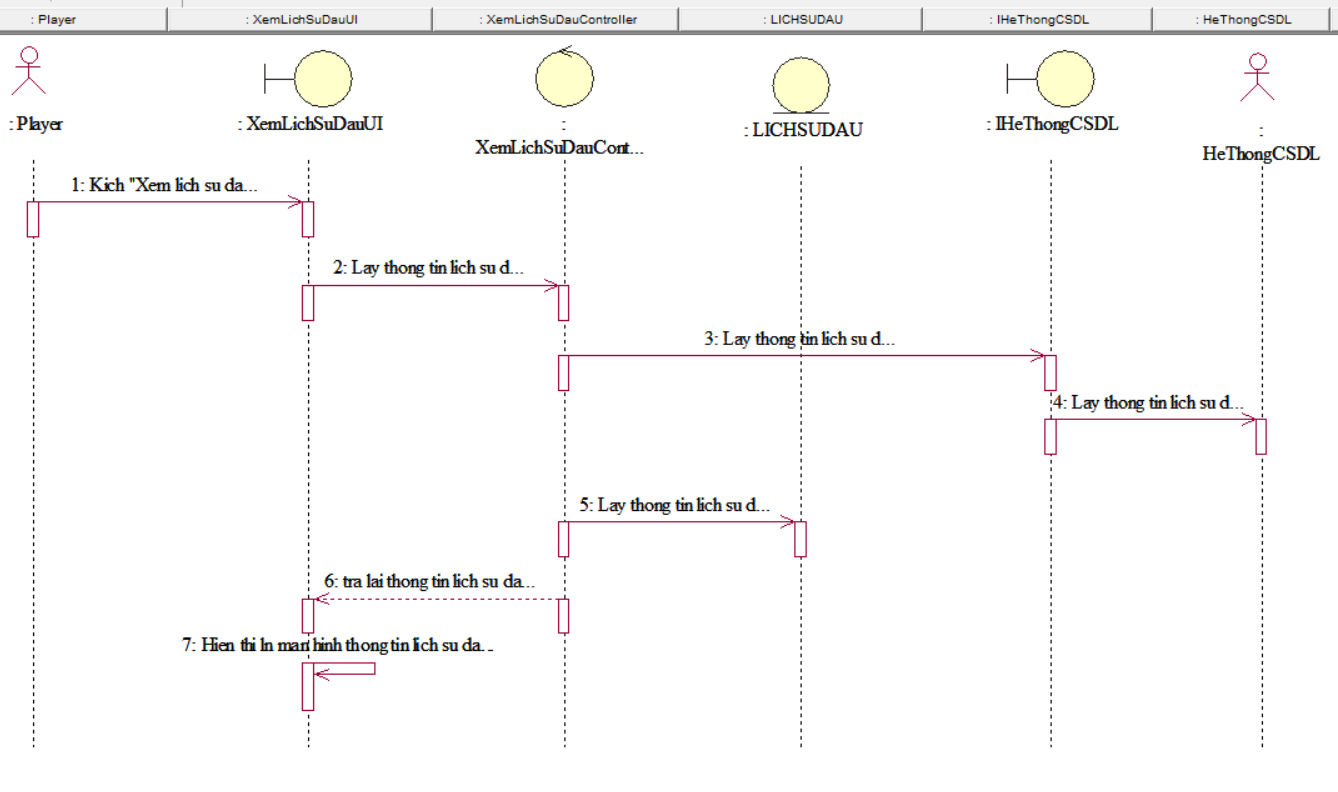
## Phân tích use case

### Phân tích use case xem lịch sử đấu

#### Biểu đồ VOPC

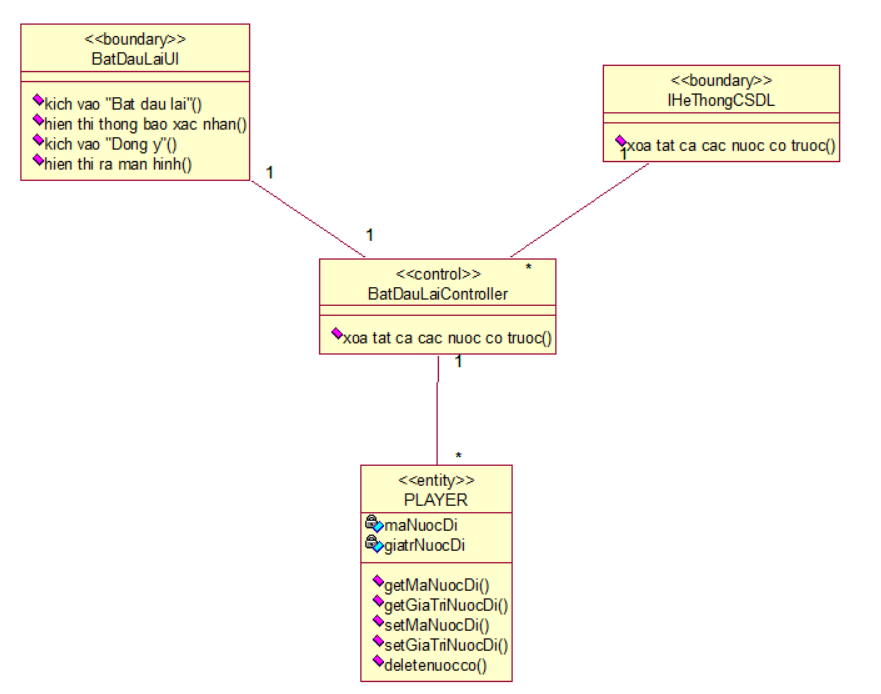


#### Biểu đồ trình tự

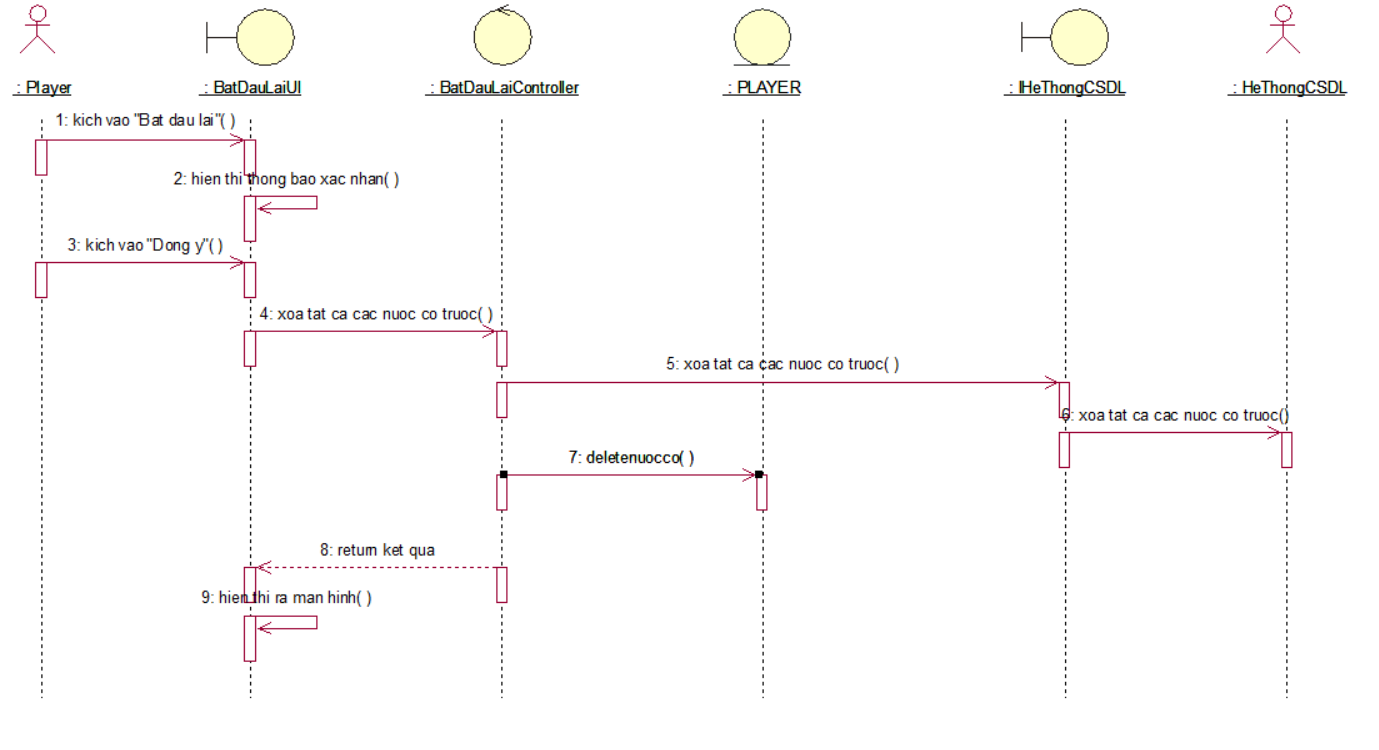


### Phân tích use case bắt đầu trận đấu mới

#### Biểu đồ VOPC

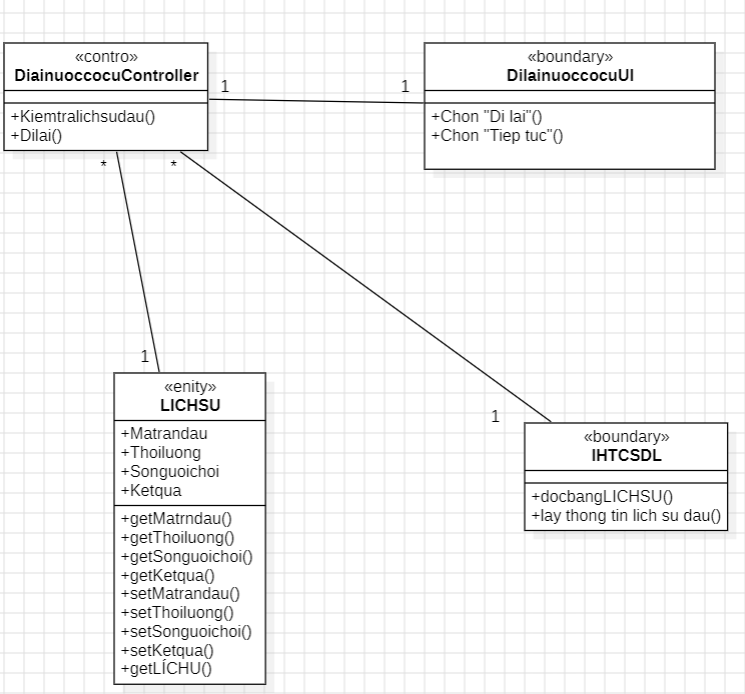


#### Biểu đồ trình tự

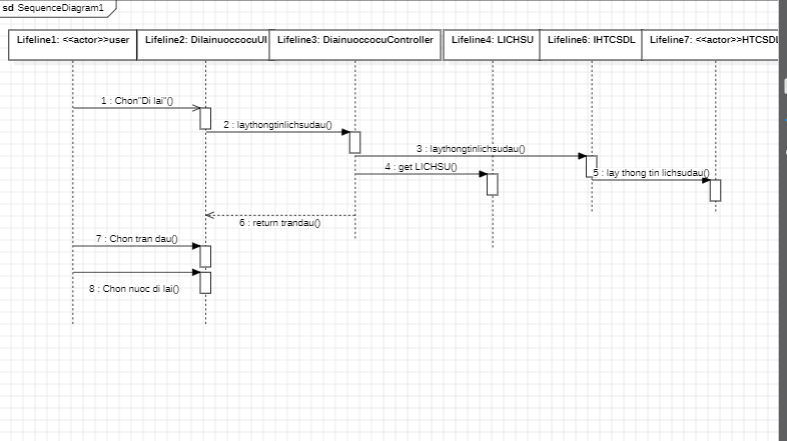


### Phân tích use case đi lại nước cờ cũ

#### Biểu đồ VOPC

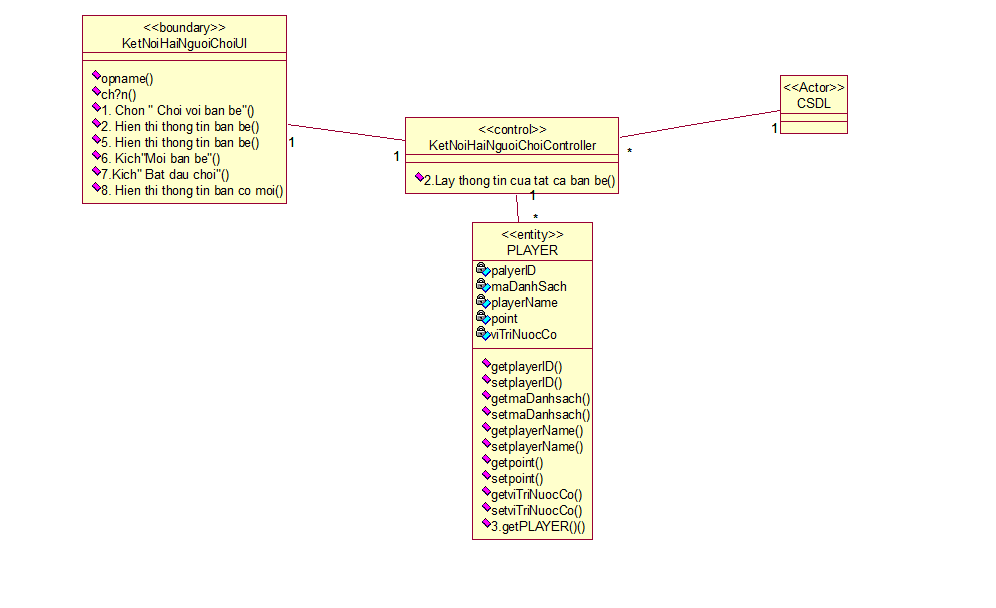


#### Biểu đồ trình tự

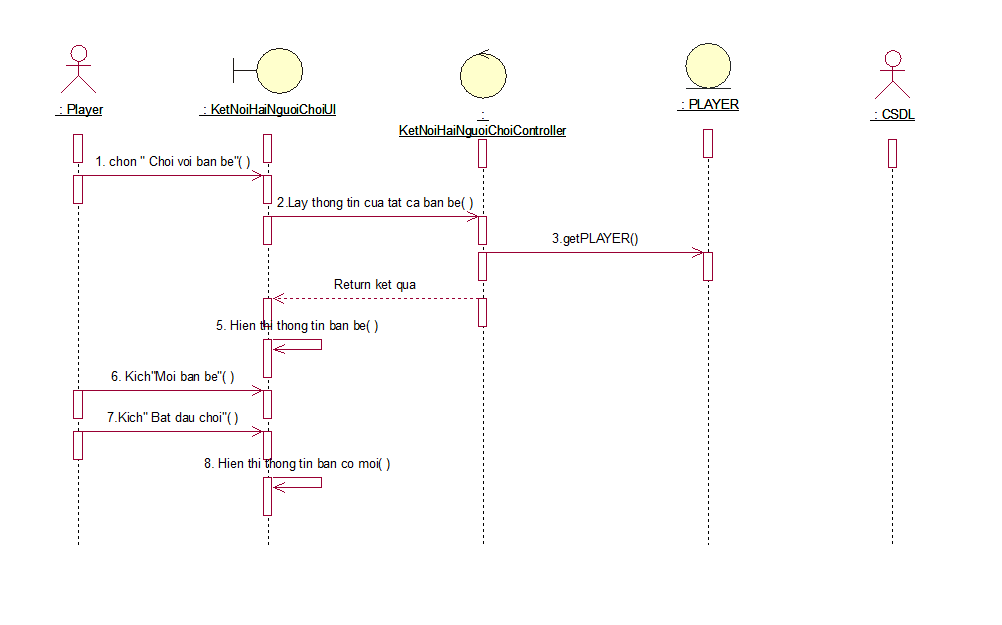


### Phân tích use case kết nối hai người chơi

#### Biểu đồ VOPC

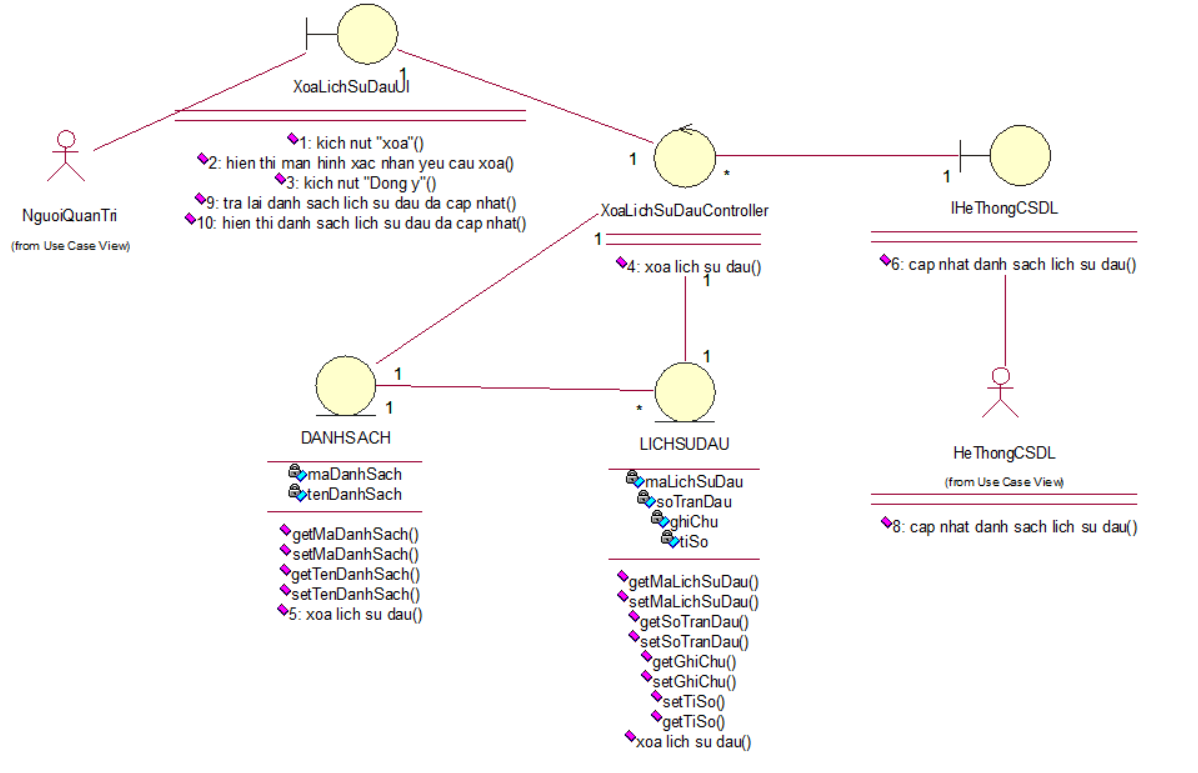


#### Biểu đồ trình tự



### Phân tích use case xóa lịch sử đấu

#### Biểu đồ VOPC



#### Biểu đồ trình tự

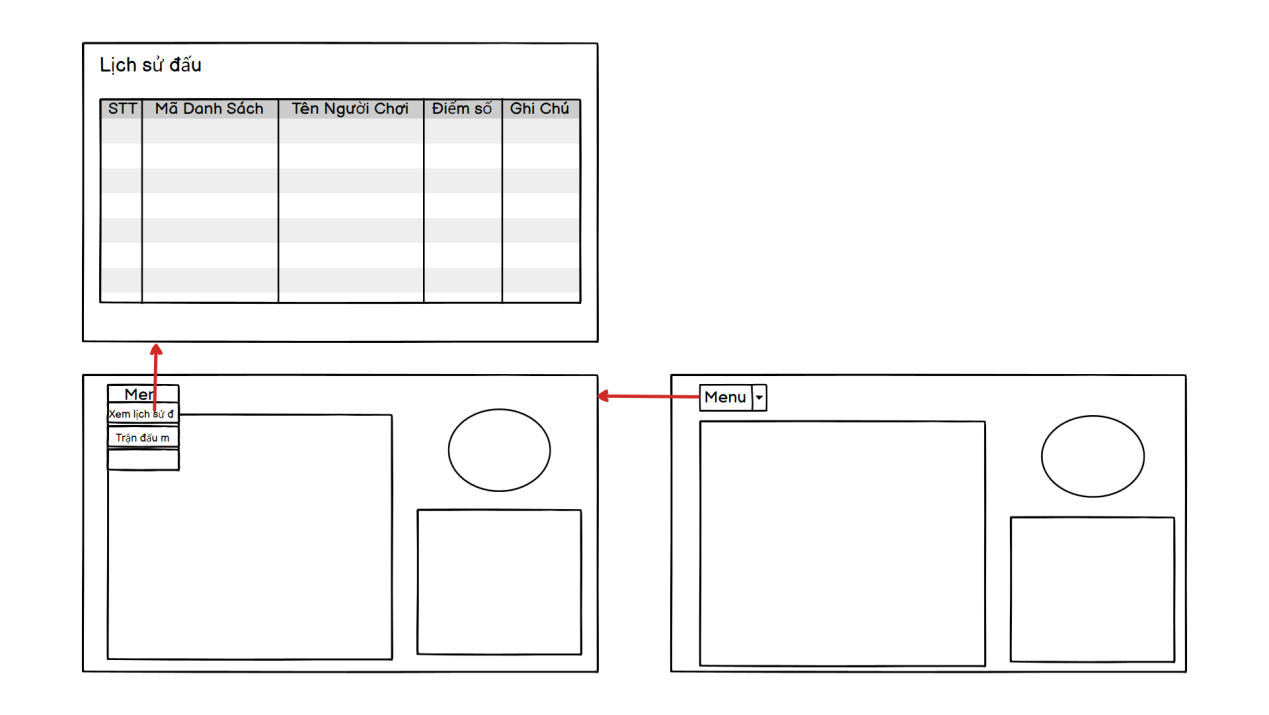


## 

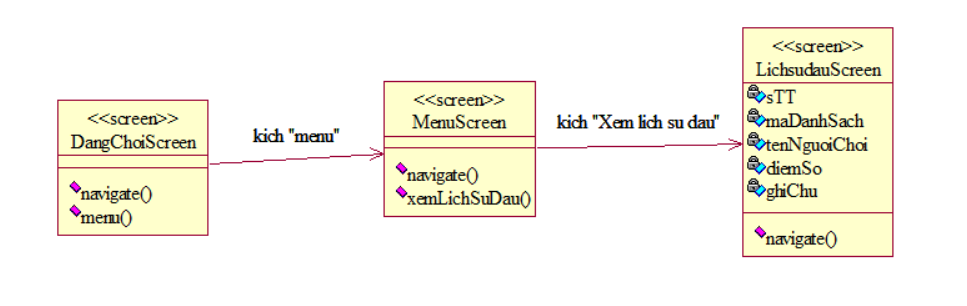
## Thiết kế giao diện

### Giao diện use case xem lịch sử đấu

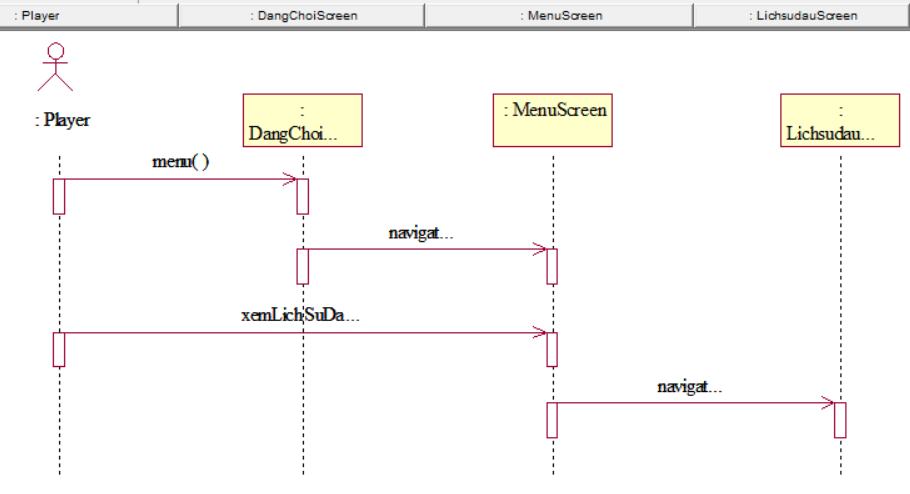
#### Hình dung màn hình



#### Biểu đồ lớp màn hình

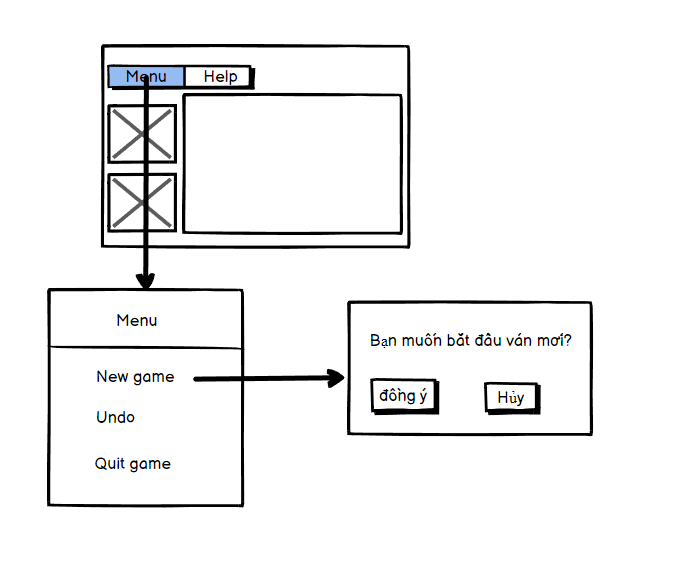


#### Biểu đồ cộng tác của màn hình



### Giao diện use case bắt đầu trận đấu mới

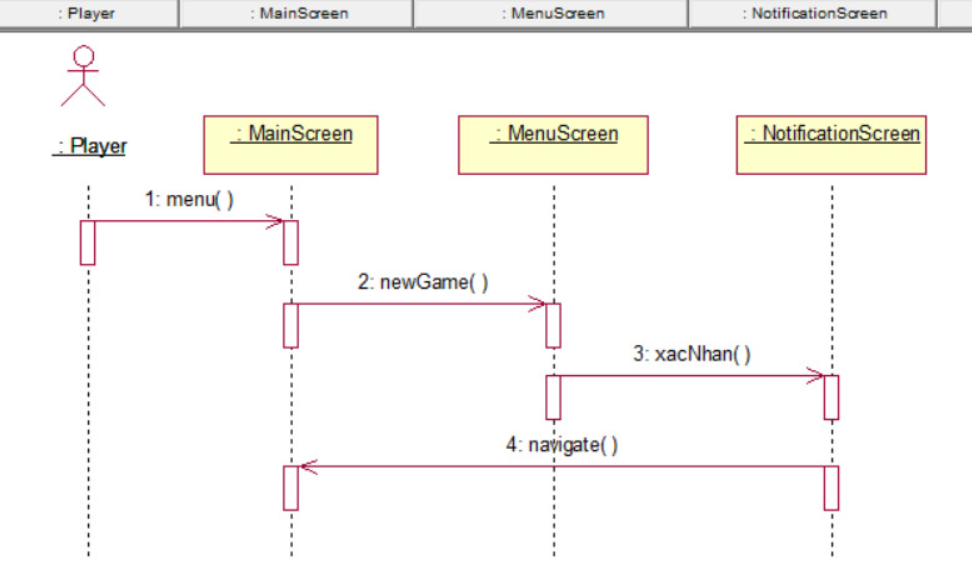
#### Hình dung màn hình



#### Biểu đồ lớp màn hình

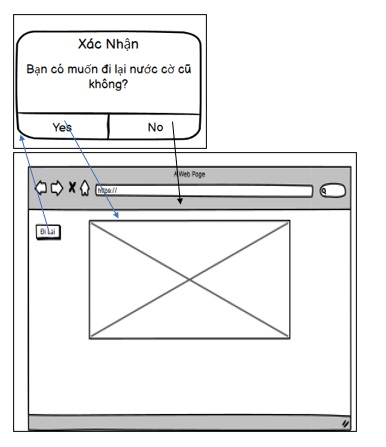
#### 

#### Biểu đồ cộng tác của màn hình

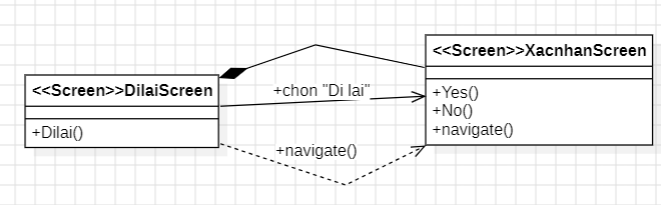


### Giao diện use case đi lại nước cờ cũ

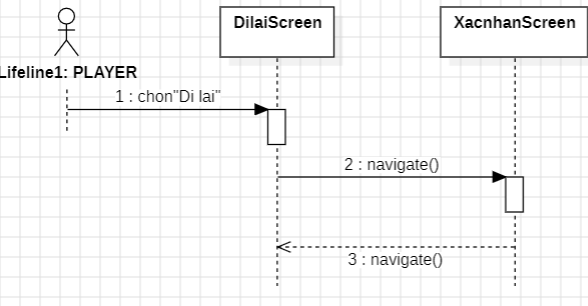
#### Hình dung màn hình



#### Biểu đồ lớp màn hình

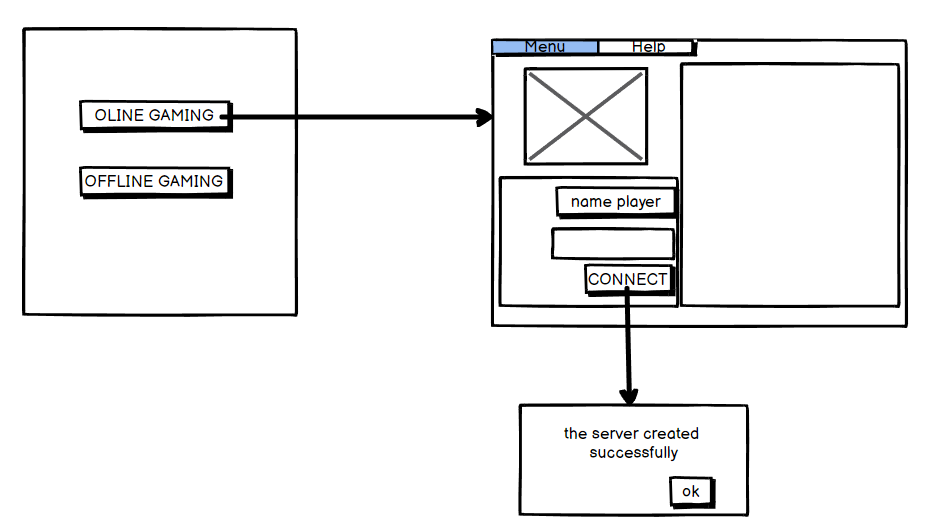


#### Biểu đồ cộng tác của màn hình

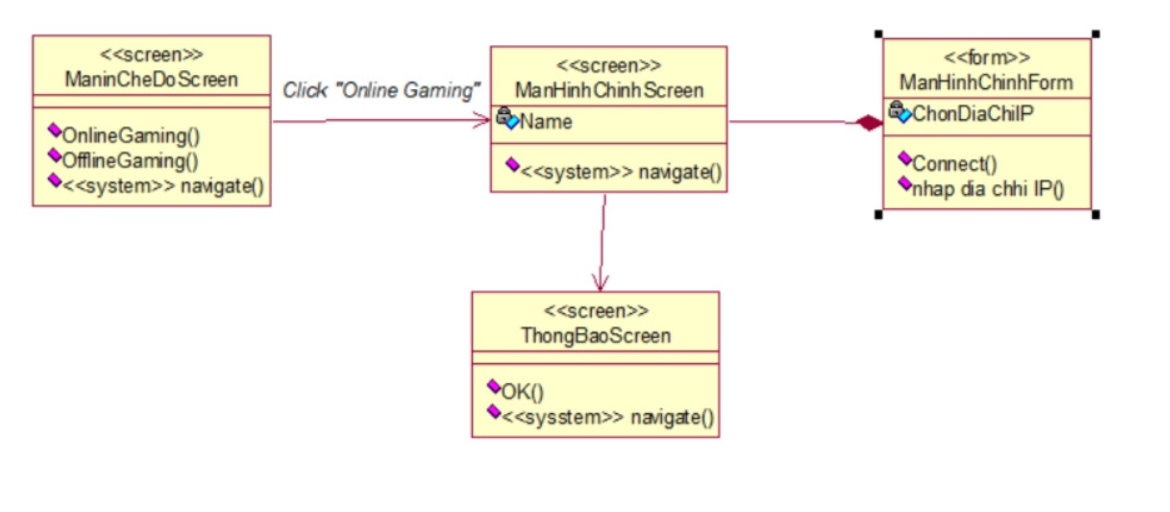


### Giao diện use case kết nối hai người chơi

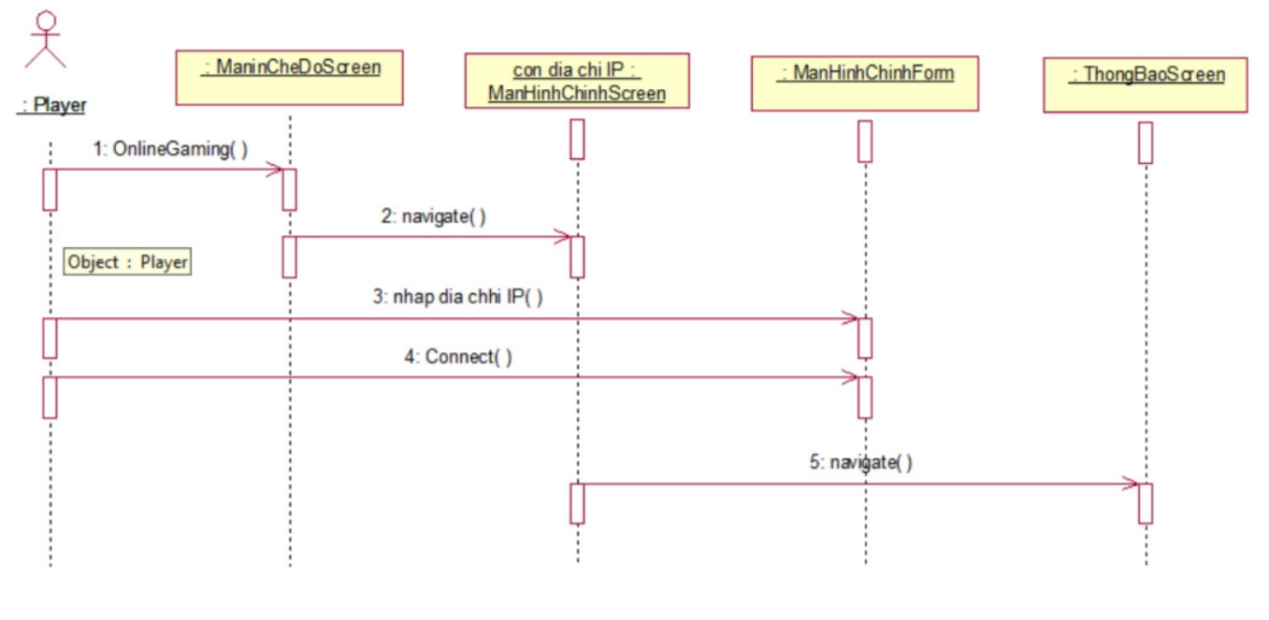
#### Hình dung màn hình



#### Biểu đồ lớp màn hình



#### Biểu đồ cộng tác của màn hình

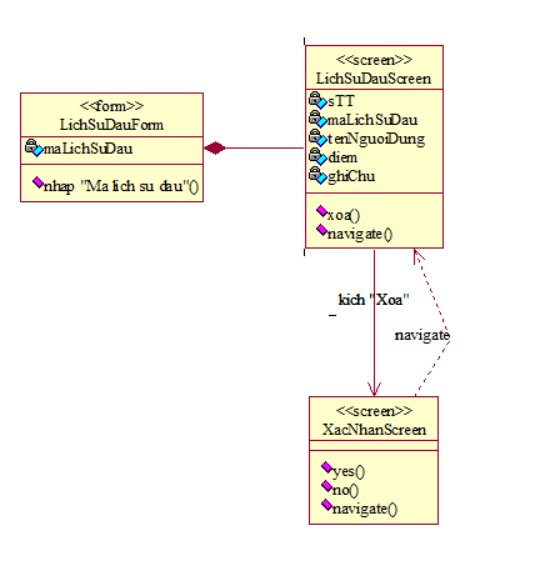


### Giao diện use case xóa lịch sử đấu

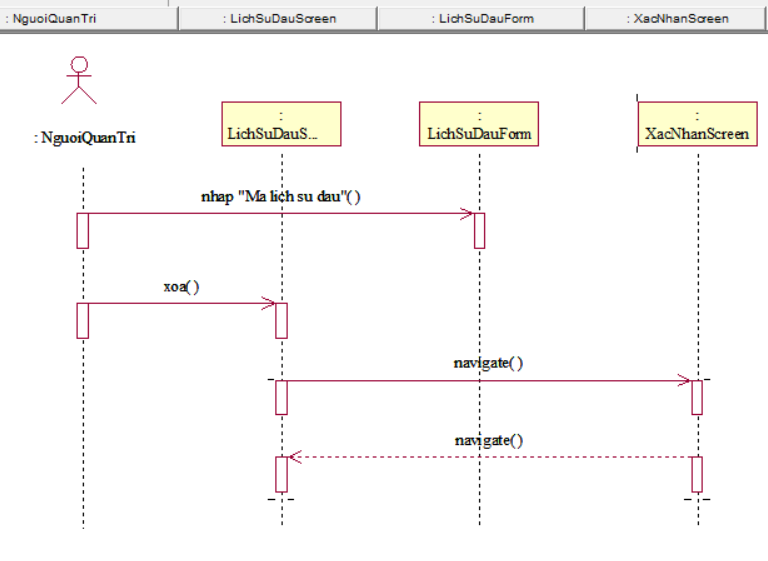
#### Hình dung màn hình



#### Biểu đồ lớp màn hình

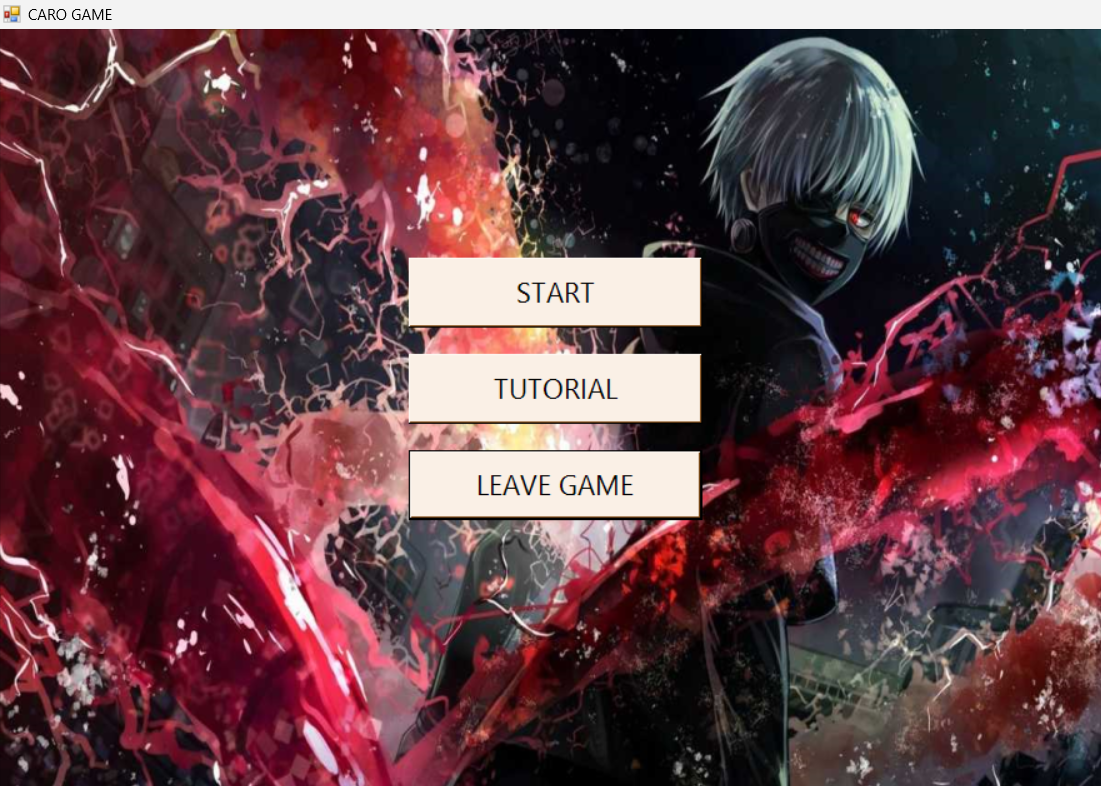


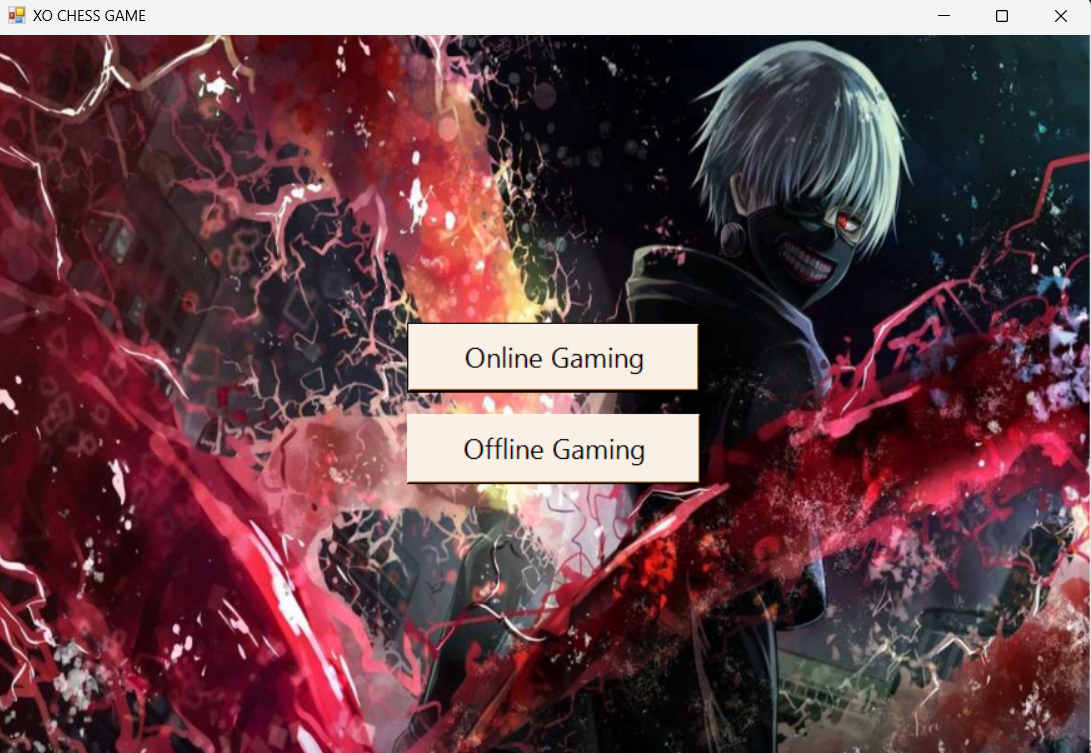
#### Biểu đồ cộng tác của màn hình



# CHƯƠNG III: KẾT QUẢ THỰC NGHIỆM

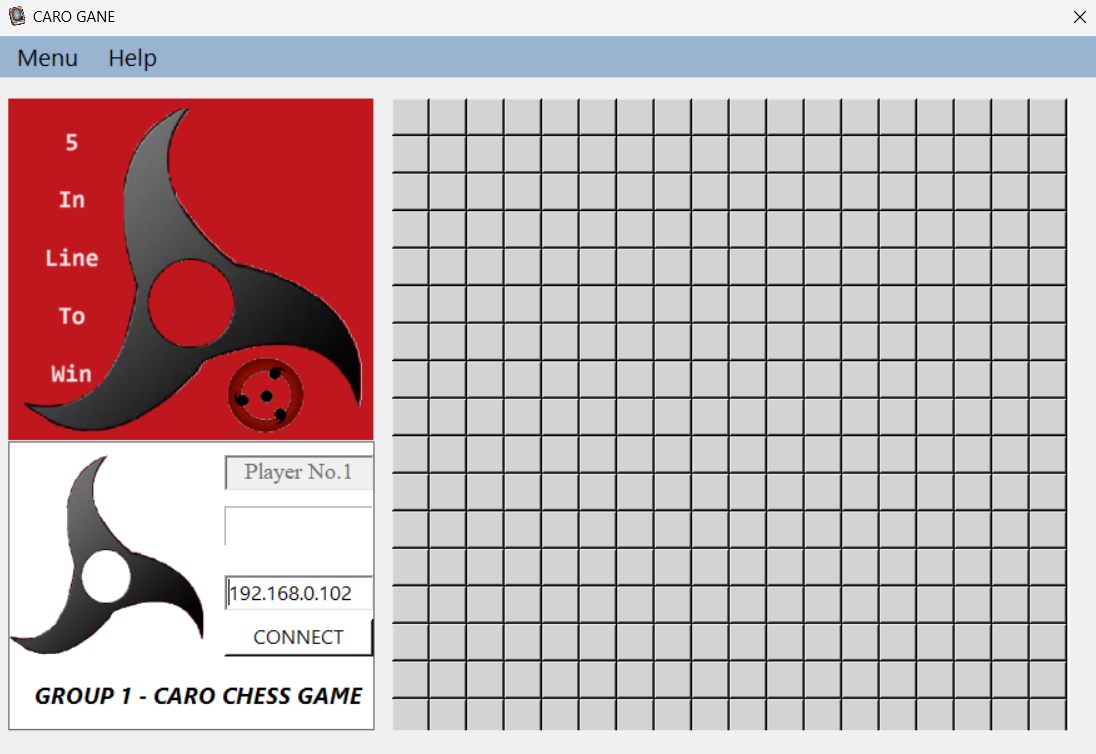
## Hình ảnh màn hình trang chủ

Hình 1: giao diện bắt đầu

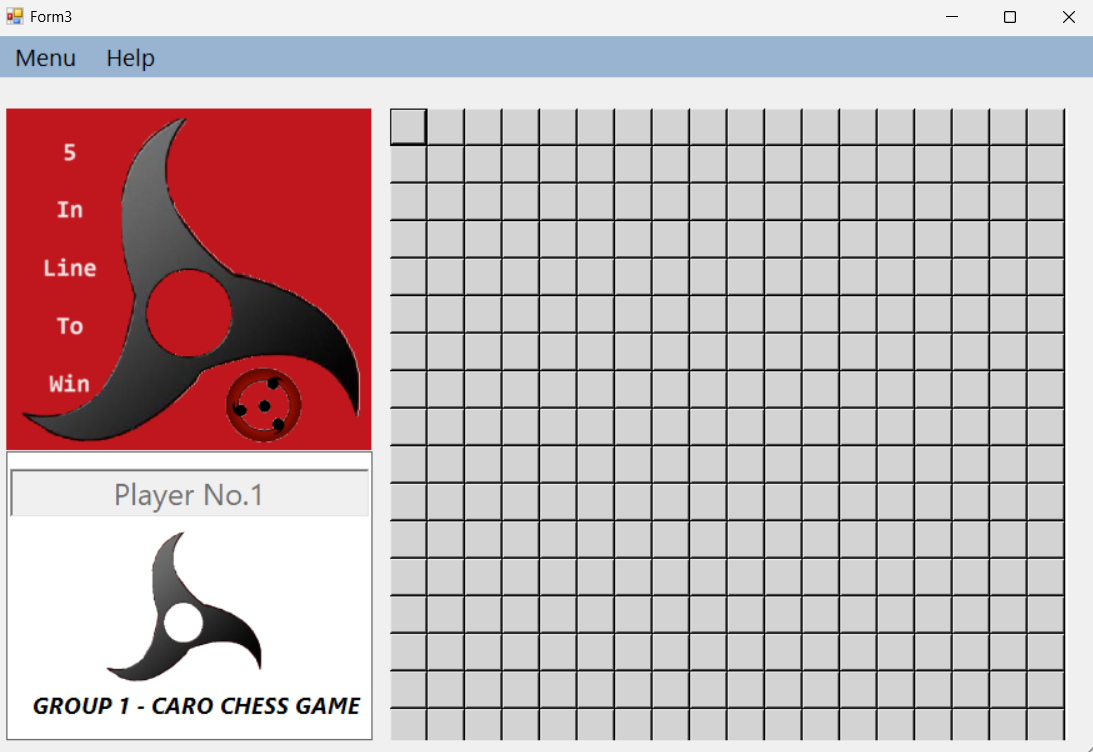
**

Hình 2: giao diện chọn chế độ

## Hình ảnh giao diện khi chơi game

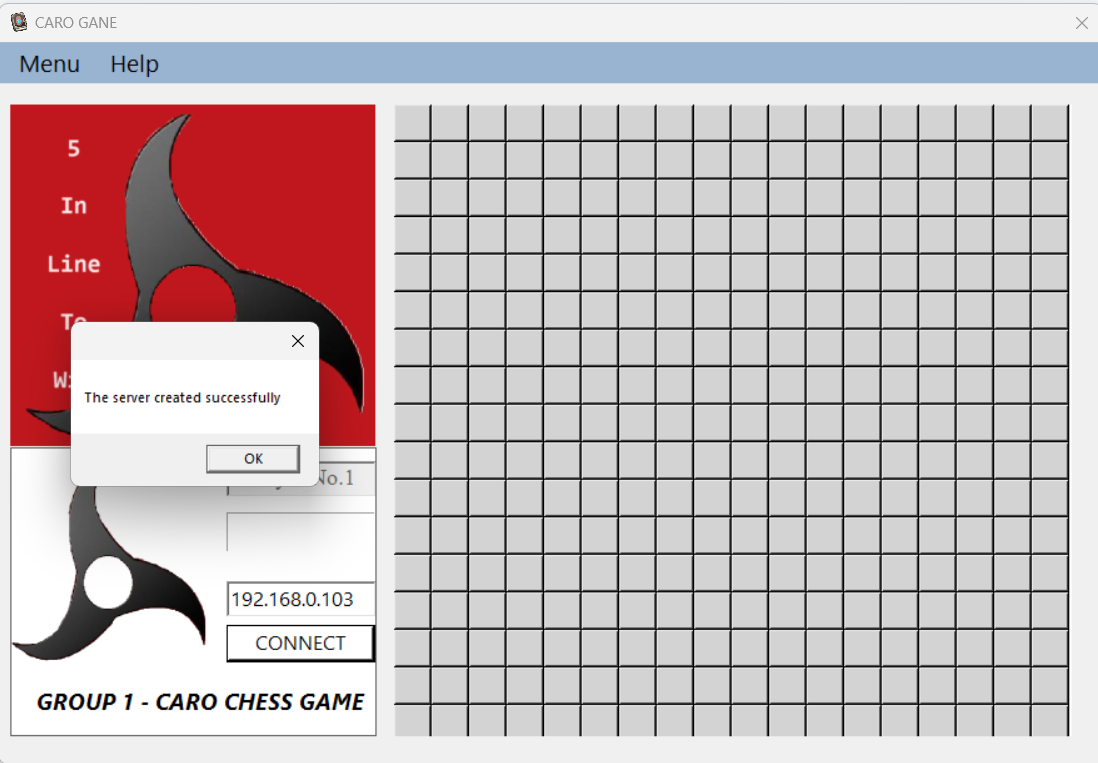
**

Hình 3: giao diện chế độ online

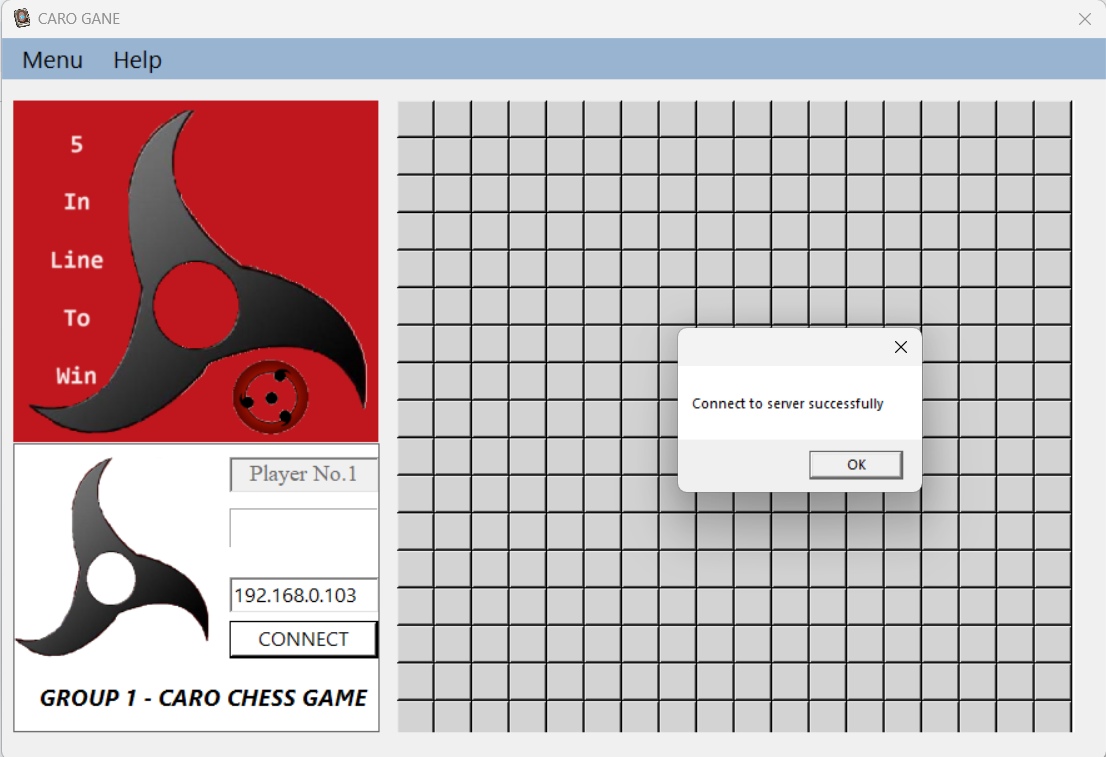
**

Hình 4: giao diện chế độ offline

## Hình ảnh khi đang kết nối mang LAN

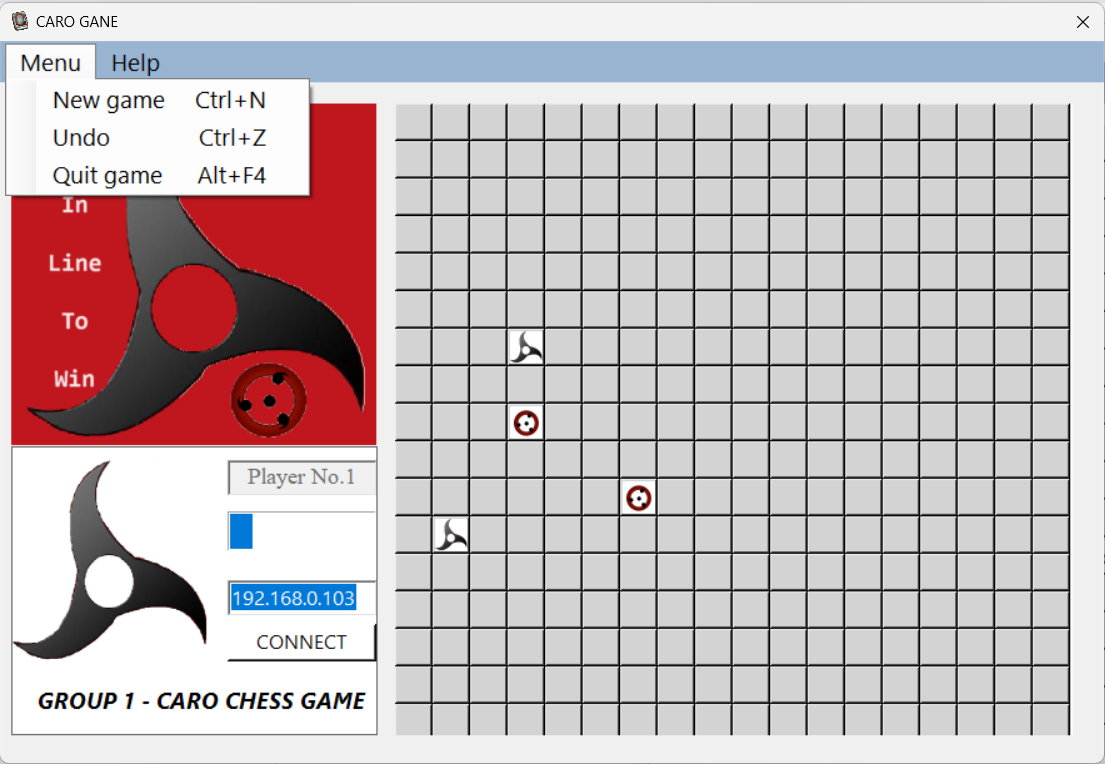


Hình 5: Giao diện khi tạo server thành công

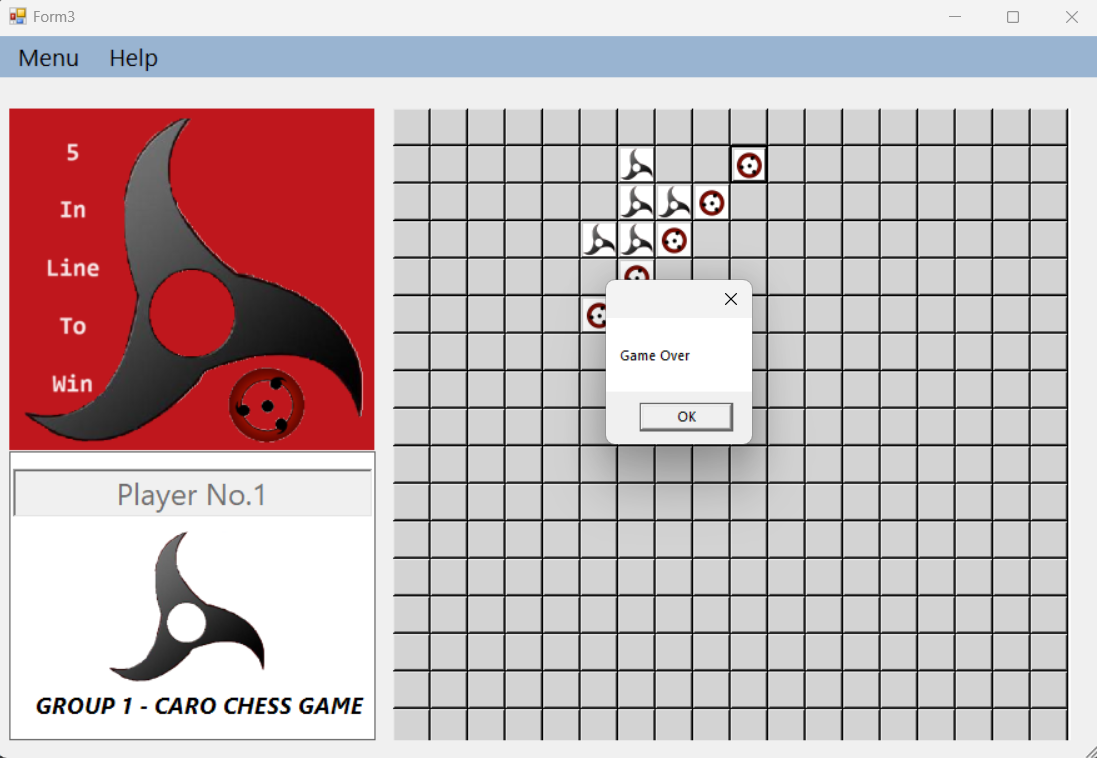


Hình 6: Giao Diện khi kết nối server thành công

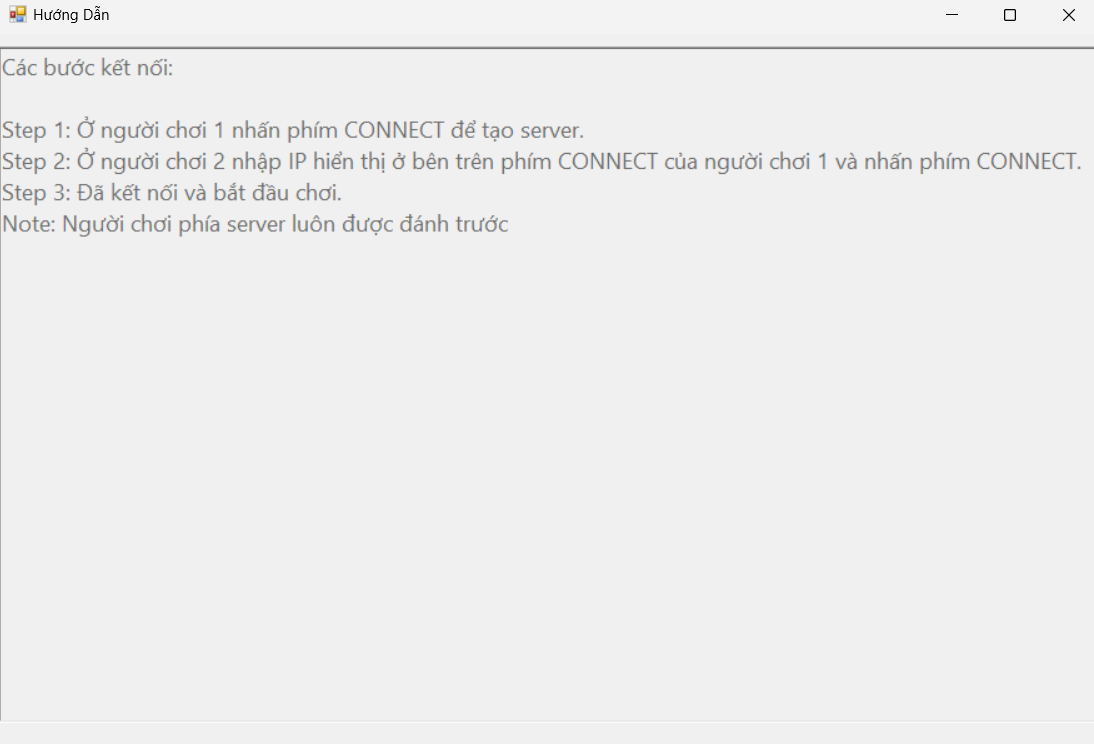
## Hình ảnh các chức năng có trong menu



Hình 7: Giao diện khi dùng menu

**

Hình 5: giao diện kết thúc

**

Hình 6: giao diện hướng dẫn

# KẾT LUẬN

Dự án phát triển trò chơi XO có kết nối mạng LAN đã đạt được các mục tiêu đề ra ban đầu và mang lại nhiều giá trị thực tiễn. Qua quá trình phát triển, chúng ta đã khám phá và ứng dụng các kiến thức về lập trình mạng, phát triển giao diện người dùng và xử lý các tình huống thực tế trong việc kết nối và giao tiếp giữa các máy tính. Các điểm nổi bật trong kết luận của đề tài bao gồm:

1. **Nâng cao kỹ năng lập trình và hiểu biết về mạng:**

* Dự án đã giúp người phát triển củng cố và nâng cao kỹ năng lập trình, đặc biệt là lập trình mạng, qua việc thiết lập kết nối giữa các máy tính trong mạng LAN và xử lý các giao tiếp qua mạng.
* Khả năng ứng dụng các kiến thức lý thuyết vào thực tiễn được cải thiện, giúp người phát triển tự tin hơn trong việc giải quyết các vấn đề kỹ thuật.

1. **Tạo ra một nền tảng chơi XO tiện lợi và thú vị:**

* Trò chơi XO có kết nối mạng LAN mang đến cho người chơi trải nghiệm mới mẻ, cho phép họ dễ dàng kết nối và thi đấu với nhau mà không cần gặp mặt trực tiếp.
* Giao diện người dùng được thiết kế đơn giản và thân thiện, giúp người chơi dễ dàng thao tác và tận hưởng trò chơi.

1. **Khám phá và ứng dụng các kỹ thuật lập trình mạng:**

* Việc xây dựng trò chơi đã giúp chúng ta hiểu rõ hơn về cách thức hoạt động của mạng LAN, các giao thức truyền thông, và cách thức thiết lập kết nối giữa các máy tính.
* Các kỹ thuật xử lý lỗi và bảo mật trong giao tiếp mạng đã được áp dụng, đảm bảo trò chơi hoạt động ổn định và an toàn.

1. **Đề xuất hướng phát triển tương lai:**

* Mở rộng tính năng của trò chơi, chẳng hạn như hỗ trợ chơi qua Internet, thêm các chế độ chơi và tính năng mới để tăng tính hấp dẫn.
* Nâng cao giao diện người dùng, áp dụng các kỹ thuật đồ họa tiên tiến để cải thiện trải nghiệm người chơi.
* Tiếp tục nghiên cứu và áp dụng các công nghệ mới trong lĩnh vực lập trình mạng và phát triển trò chơi để duy trì sự hiện đại và cạnh tranh của sản phẩm.

Dự án trò chơi XO có kết nối mạng LAN không chỉ là một bước tiến trong việc áp dụng kiến thức vào thực tiễn mà còn mở ra nhiều cơ hội phát triển và sáng tạo trong tương lai. Những kinh nghiệm và bài học thu được từ dự án này sẽ là nền tảng vững chắc cho các dự án tiếp theo trong lĩnh vực công nghệ và lập trình.