

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**NIÊN LUẬN CƠ SỞ
NGÀNH CÔNG NGHỆ THÔNG TIN**

Đề tài

**GAME TIC TAC TOE
ỨNG DỤNG THUẬT TOÁN MINIMAX
& LẬP TRÌNH SOCKET**

**Sinh viên: Trần Quốc Toàn
Mã số: B1606944
Khóa: 42**

Cần Thơ, 11/2019

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**NIÊN LUẬN CƠ SỞ
NGÀNH CÔNG NGHỆ THÔNG TIN**

Đề tài

**TIC TAC TOE
ỨNG DỤNG THUẬT TOÁN MINIMAX
& LẬP TRÌNH SOCKET**

**Người hướng dẫn
TS.Trần Công Ân**

**Sinh viên thực hiện
Trần Quốc Toàn
Mã số: B1606944
Khóa: 42**

Cần Thơ, 11/2019

1. Lời cảm ơn

Em xin chân thành cảm ơn TS.Trần Công Án, Bộ môn Công nghệ Thông tin, Khoa Công nghệ Thông tin và Truyền thông, trường Đại học Cần Thơ đã cố vấn, giảng dạy và giúp đỡ sinh viên hoàn thành đề tài này!

2. Mục lục

1. Lời cảm ơn	1
2. Mục lục	2
3. Danh mục đồ thị, biểu bảng và màn hình	3
4. Tóm lược	4
5. Phần giới thiệu.....	5
6. Phần nội dung.....	6
6.1. Chương 1 - Đặc tả yêu cầu:.....	6
6.2. Chương 2 - Thiết kế giải pháp:	6
6.2.1. Thuật toán Minimax.....	6
6.2.2. Cây trò chơi (Game Tree):	7
6.2.3. Xử lý trạng thái trò chơi:	9
6.2.4. Xử lý thời gian thực (Real Time):	11
6.3. Chương 3 - Cài đặt giải pháp:	12
6.4. Chương 4 – Đánh giá kiểm thử:	13
7. Phần kết luận	16
8. Tài liệu tham khảo	17

3. Danh mục đồ thị, biểu bảng và hình ảnh

Hình 1: Ví dụ Minimax.....	6
Hình 2: Cây trò chơi (Game Tree)	8
Hình 3: Khởi tạo bàn cờ 3x3	9
Hình 4: Các nước đi chiến thắng.....	9
Hình 5: Mảng lưu các nước đi chiến thắng.....	10
Hình 6: Cơ chế hoạt động Socket IO.....	11
Hình 7: Node JS – Socket IO	12
Hình 8: Chế độ chơi hai người một máy(X thắng)	13
Hình 9: Chế độ chơi người một máy(O thắng)	14
Hình 10: Chế độ chơi người một máy(X và O hòa).....	14
Hình 11: Chế độ chơi giữa người và máy.....	15
Hình 12: Chế độ chơi trực tuyến	15

4. Tóm lược

Như chúng ta đã biết, công nghệ thông tin đang bao phủ, tràn ngập khắp thế giới, trở nên quen thuộc với con người và gắn bó mật thiết hỗ trợ con người rất nhiều trong tất cả mọi công việc. Trong thời đại công nghệ 4.0 hiện nay, sự phát triển vượt bậc của công nghệ đã mang đến cho thế giới những thay đổi vô cùng to lớn. Với việc ứng dụng các kỹ thuật công nghệ hiện đại đã mang đến nhiều lợi ích tuyệt vời cho cuộc sống.

Thuật toán góp một phần không hề nhỏ để mang đến hiệu năng cao của chương trình. Thuật Toán (dù là phức tạp hay không) đóng vai trò là các khối tư duy hữu ích để giải quyết các vấn đề logic chúng ta gặp trong tương lai hiệu quả hơn. Có rất nhiều thuật toán ứng dụng vào thực tế hay những trò chơi đối kháng như: heuristic, Minimax, ...

5. Phân giới thiệu

Tic-tac-toe là một trò chơi phổ biến dùng viết trên bàn cờ giấy có 5ang ô, 3x3. Hai người chơi, người dùng ký hiệu O, người kia dùng ký hiệu X, lần lượt điền ký hiệu của mình vào các ô. Người thắng là người thể tạo được đầu tiên một dãy ty ký hiệu của mình, ngang dọc hay chéo đều được.

Để tạo ra một chương trình Tic Tac Toe hoàn hảo, một thuật toán có thể tính toán tất cả nước đi và xác định nước đi có lợi là thiết yếu. Minimax là một trong số những thuật toán có thể ứng dụng vào chương trình Tic Tac Toe. Thông qua việc tìm hiểu về thuật toán Minimax sẽ giúp ta hiểu được một trò chơi đối kháng như Tic Tac Toe sẽ xử lý các nước đi như thế nào.

Mục tiêu đề tài là tìm hiểu và cài đặt trò chơi Tic Tac Toe ứng dụng giải thuật Minimax, kết hợp với lập trình socket xử lý thời gian thực. Để dựa vào đó ta có thể hiểu được cách áp dụng 1 giải thuật và thực tế là như thế nào.

Bố cục của bản báo cáo gồm 3 phần: phần giới thiệu, phần nội dung và phần kết luận. Trong đó phần nội dung gồm có 4 chương:

Chương 1 – Đặc tả yêu cầu: Giới thiệu các chức năng chính của trò chơi Tic Tac Toe.

Chương 2 – Thiết kế giải pháp: Trình bày những kiến thức liên quan đến giải thuật Minimax, cây trò chơi, socket.

Chương 3 – Cài đặt giải pháp: Mô tả cách thức cài đặt thiết kế.

Chương 4 – Đánh giá kiểm thử: Mô tả kiểm thử và kiểm thử sản phẩm.

6. Phần nội dung

6.1. Chương 1 – Đặc tả yêu cầu

Những chức năng chính của trò chơi Tic Tac Toe

I. Chế độ chơi giữa người với người:

- Chế độ chơi 2 người trên cùng một máy
- Chế độ chơi Online: Ứng dụng lập trình socket để xử lý thời gian thực giữa 2 người chơi

II. Chế độ chơi giữa người và máy tính:

Ứng dụng thuật toán Minimax vào chế độ chơi để tìm đường đi tiếp theo tối ưu nhất cho máy tính

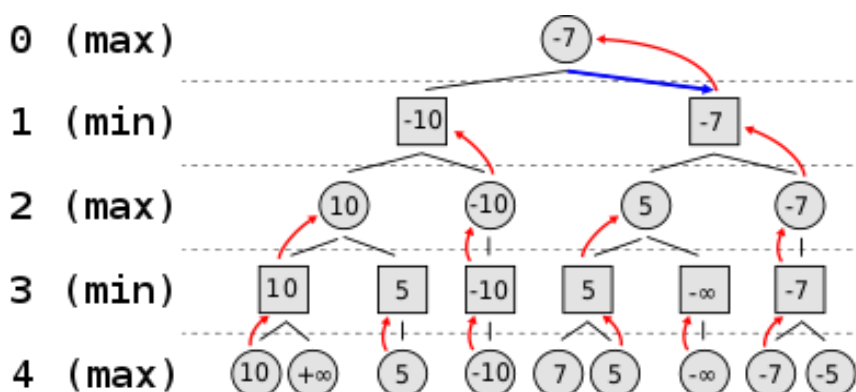
6.2. Chương 2 – Thiết kế giải pháp

6.2.1. Thuật toán Minimax

Minimax (còn gọi là minmax) là một phương pháp trong lý thuyết quyết định có mục đích là tối thiểu hóa (*minimize*) tổn thất vốn được dự tính có thể là “tối đa” (*maximize*). Có thể hiểu ngược lại là, nó nhằm tối đa hóa lợi ích vốn được dự tính là *tối thiểu* (maximin). Nó bắt nguồn từ trò chơi có tổng bằng không. Nó cũng được mở rộng cho nhiều trò chơi phức tạp hơn và giúp đưa ra các quyết định chung khi có sự hiện diện của sự không chắc chắn.[1]

Một phiên bản của giải thuật áp dụng cho các trò chơi như tic-tac-toe, khi mà mỗi người chơi có thể thắng, thua, hoặc hòa. Nếu người chơi A có thể thắng trong 1 nước đi, thì “nước đi tốt nhất” chính là nước đi để dẫn đến kết quả thắng đó. Nếu người B biết rằng có một nước đi mà dẫn đến tình huống người A có thể thắng ngay ở nước đi tiếp theo, trong khi nước đi khác thì sẽ dẫn đến tình huống mà người chơi A chỉ có thể, tốt nhất, là hòa thì nước đi tốt nhất của người B chính là nước đi sau.

Để dễ hình dung, ta tham khảo hình sau:



Hình 1: Ví dụ Minimax

Hai Đối thủ trong 1 chò chơi được gọi là MIN và MAX. MAX là đại diện cho đối thủ quyết định dành thắng lợi hay cố gắng tối ưu hóa ưu thế của mình. Ngược lại MIN là đối thủ cố gắng tối thiểu hóa điểm số của MAX. Ta giả thiết MIN cũng dùng thông tin như MAX.

Khi áp dụng thủ tục Minimax ta đánh dấu luân phiên từng mức trong không gian tìm kiếm phù hợp với đối thủ có bước đi ở mức đó. Trong ảnh trên MIN được quyền đi trước. Từng nút lá được gán giá trị là 0 hay 1 tùy theo đó là thắng cuộc với MIN hay MAX. Minimax sẽ truyền giá trị này lên cao dần trên đồ thị qua các nút cha mẹ kế tiếp nhau theo luật sau:

- Nếu trạng thái cha mẹ là nút MAX, gán cho nó giá trị tối đa của các con cháu của nó.

- Nếu trạng thái cha mẹ là nút MIN, gán cho nó giá trị tối thiểu của các con cháu của nó.

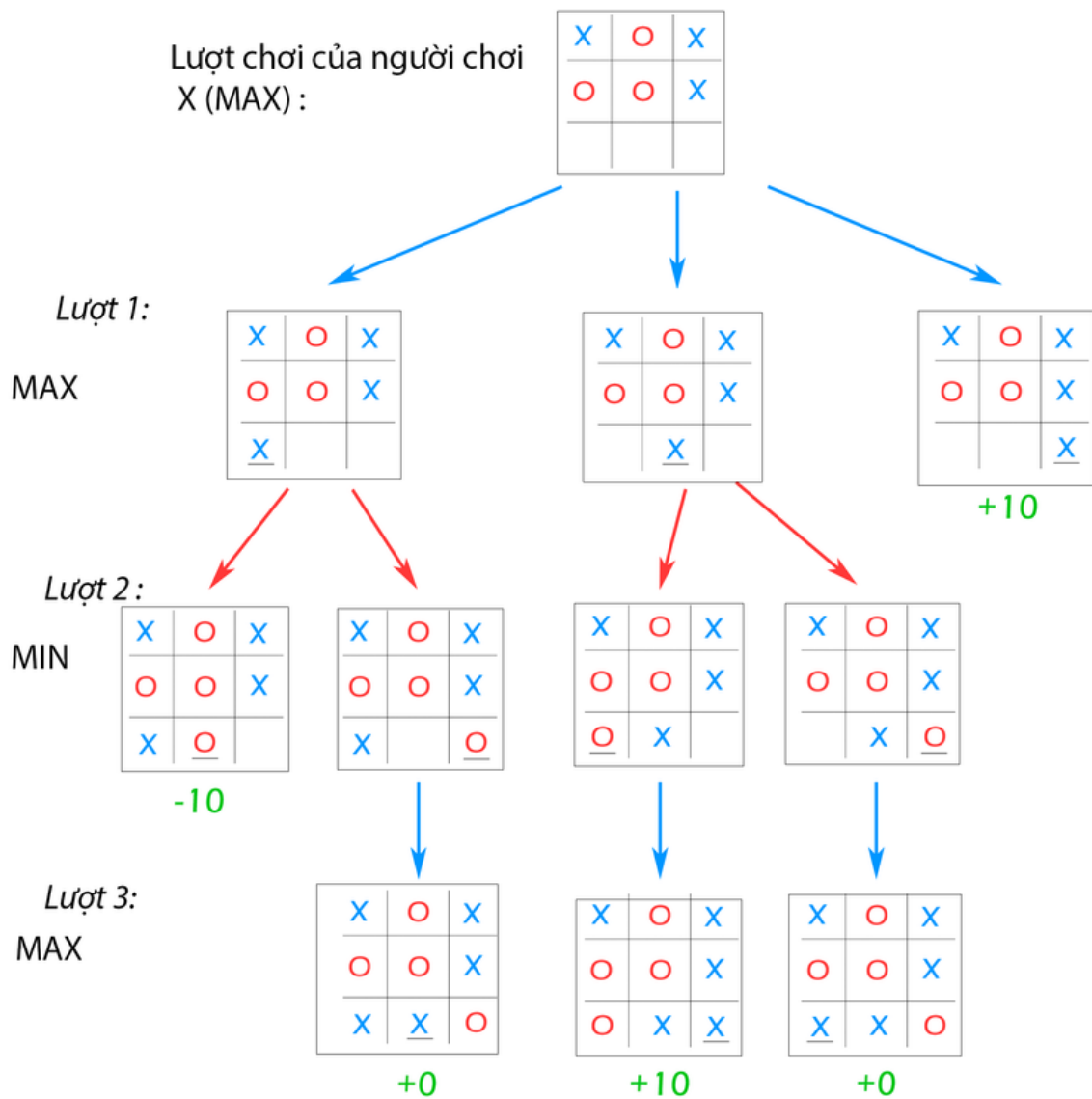
Giá trị được gán cho từng trạng thái bằng cách đó sẽ chỉ rõ trạng thái tốt nhất mà đối thủ này có thể đạt được. Các giá trị này sẽ được dùng để lựa chọn các bước đi có thể có.

6.2.2. Cây trò chơi (Game Tree)

Trò chơi có một trạng thái bắt đầu và mỗi nước đi sẽ biến đổi trạng thái hiện hành thành một trạng thái mới. Trò chơi sẽ kết thúc theo một quy định nào đó, theo đó thì cuộc chơi sẽ dẫn đến một trạng thái phản ánh có một người thắng cuộc hoặc một trạng thái mà cả hai đấu thủ không thể phát triển được nước đi của mình, ta gọi nó là trạng thái hòa cờ. Ta tìm cách phân tích xem từ một trạng thái nào đó sẽ dẫn đến đấu thủ nào sẽ thắng với điều kiện cả hai đấu thủ đều có trình độ như nhau.

Một trò chơi như vậy có thể được biểu diễn bởi một cây, gọi là cây trò chơi. Mỗi một nút của cây biểu diễn cho một trạng thái. Nút gốc biểu diễn cho trạng thái bắt đầu của cuộc chơi. Mỗi nút lá biểu diễn cho một trạng thái kết thúc của trò chơi (trạng thái thắng thua hoặc hòa). Nếu trạng thái x được biểu diễn bởi nút n thì các con của n biểu diễn cho tất cả các trạng thái kết quả của các nước đi có thể xuất phát từ trạng thái x . [2]

Để dễ dàng hình dung hơn, ta tham khảo hình sau :



Hình 2: Cây trò chơi (Game Tree)

Ta có thể gán cho mỗi nút lá một giá trị để phản ánh trạng thái thắng thua hay hòa của các đấu thủ. Đối với bài toán này ta gán cho nút lá các giá trị như sau:

- 10 nếu tại đó người đi X thắng
- 10 nếu tại đó người đi X thua
- 0 nếu cả hai hòa nhau

6.2.3. Xử lý trạng thái trò chơi

Khởi tạo bàn cờ Tic Tac Toe 3x3, gồm 9 ô. Đánh thứ tự các ô lần lượt từ 0 đến 8 như hình sau:

0	1	2
3	4	5
6	7	8

Hình 3: Khởi tạo bàn cờ 3x3

Ví dụ người chơi 1: X, người chơi 2 (hoặc Computer): O. Quy luật chơi của game, khi có 3 vị trí liên tiếp theo 9ang dọc, ngang hoặc chéo có kí tự giống nhau thì game sẽ xét trạng thái thắng thua giữa 2 người chơi. Để dễ hình dung ta tham khảo hình sau:

0	1	2
3	4	5
6	7	8

Hình 4: Các nước đi chiến thắng

Các vị trí ô trên bàn cờ, sẽ được lưu vào một mảng chiến thắng để xét trạng thái thắng-thua khi kết thúc trò chơi. Để dễ hình dung ta tham khảo hình sau:

[
[0	1	2]
[3	4	5]
[6	7	8]
[0	3	6]
[1	4	7]
[2	5	8]
[0	4	8]
[2	4	6]
]		

Hình 5: Mảng lưu các nước đi chiến thắng

Khi xuất hiện các vị trí thuộc combo của mảng trên, thì trò chơi sẽ kết thúc. Khi đó sẽ dựa vào kí tự X hoặc O ta sẽ xét thắng thua cho 2 người chơi. Người chơi thắng sẽ được cộng 1 điểm vào bảng điểm.

Trong trường hợp khi 2 người chơi đã đánh hết 9 ô của bàn cờ, mà vẫn chưa xuất hiện các vị trí của combo của mảng trên. Thì trạng thái trò chơi sẽ là hòa giữa hai người chơi.

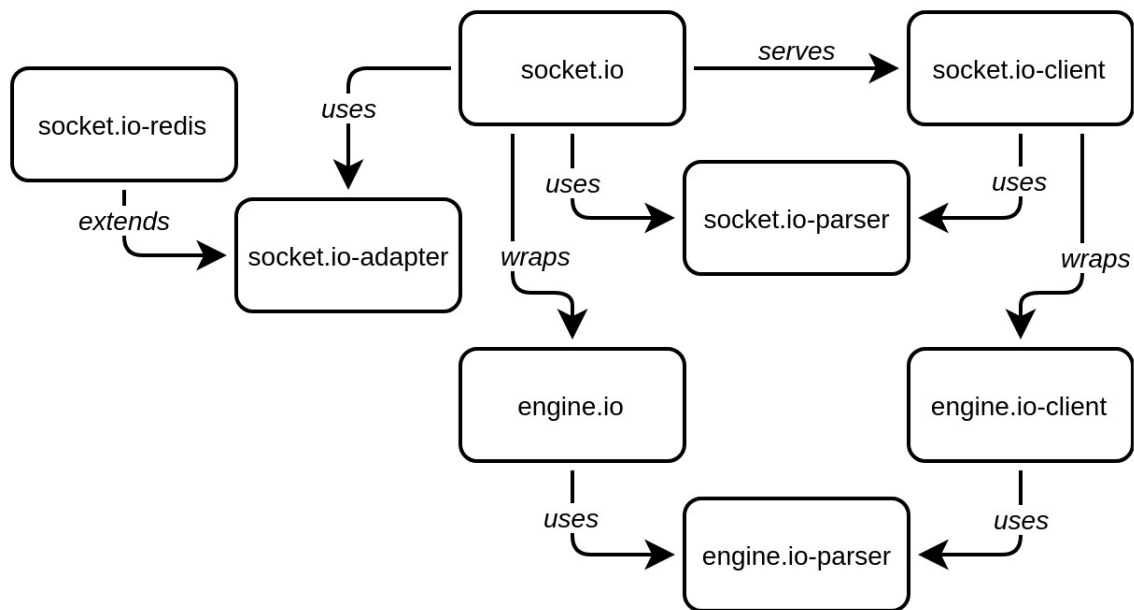
Cả 2 người chơi khi vào trạng thái hòa sẽ không được cộng điểm. Và bắt đầu lại ván cờ từ đầu.

6.2.4. Xử lý thời gian thực (Real Time)

Trong chế độ chơi Online. Vấn đề đặt ra là chúng ta phải xử lý được thời gian thực giữa 2 người chơi trong 1 ván đấu bên phía Client. Ta sẽ sử dụng thư viện Socket.IO.[3]

Cấu trúc realtime sử dụng socket bao gồm 2 phần: phía server, phía client.

Cơ chế hoạt động của một ứng dụng realtime đó là thông qua server để lắng nghe (listen) data và truyền data về các máy client. Vì vậy cần cài khai báo sử dụng socketio ở cả phía server và client. Để dễ dàng hình dung hơn, ta tham khảo hình sau:



Hình 6: Cơ chế hoạt động Socket.IO

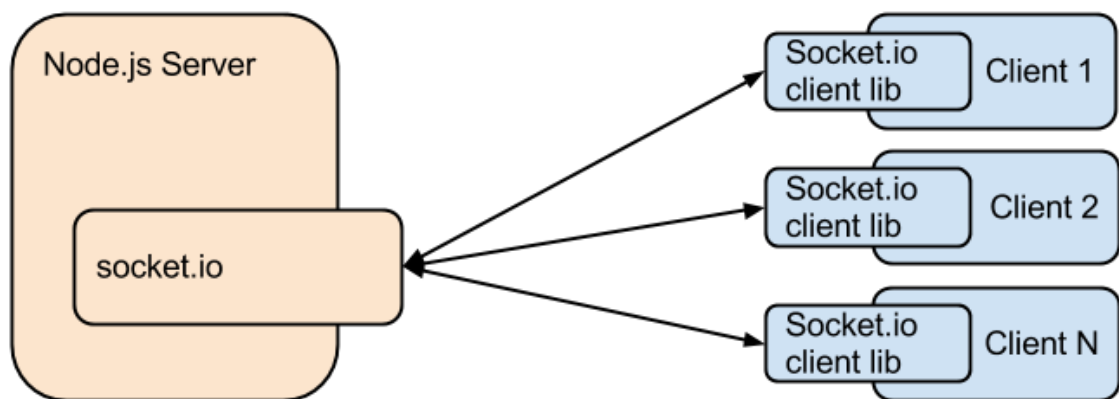
Áp dụng vào trò chơi Tic Tac Toe, khi người chơi đánh một nước cờ, thì client sẽ tạo sự kiện và server sẽ lắng nghe sự kiện đó. Từ đó server sẽ gửi sự kiện đó cho bên người chơi khác. Server sẽ xử lý các sự kiện của client và xử lý các trạng thái trò chơi đến khi có kết quả, thì server sẽ tạo sự kiện thông báo cho các client trạng thái hiện tại của trò chơi.

6.3. Chương 3 – Cài đặt giải pháp

Giao diện bên phía Client được viết bằng templates Handlebars (.hbs). Kết hợp JavaScript để tạo các hiệu ứng.

Bên phía Server, chúng ta sẽ sử dụng Node JS để dựng server và kết hợp với thư viện Socket IO để xử lý thời gian thực. [4]

Server sẽ lắng nghe các sự kiện nước đi của Client và sẽ xử lý nước đi và trả về kết quả cho các Client. Để dễ hình dung ta tham khảo hình sau :



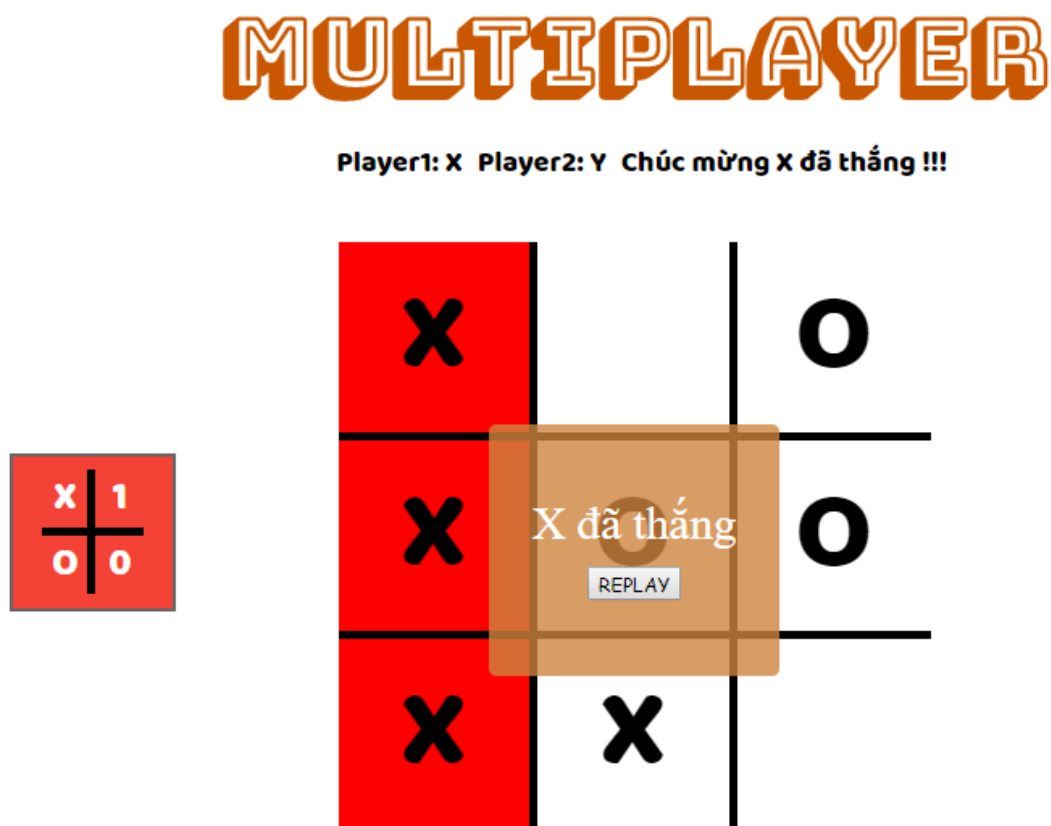
Hình 7: Node JS – Socket IO

6.4. Chương 4 – Đánh giá kiểm thử

Tiến hành chơi thử các chế độ chơi và kiểm tra hoạt động của trạng thái trò chơi. Ta thấy chương trình hoạt động tốt những chức năng đã cài đặt, và thu được kết quả như mong muốn.

Nhưng vẫn còn một số hạn chế trong việc cài đặt giải thuật. Giao diện chờ chơi còn đơn giản. Nhưng nhìn chung chương trình đã đạt được những mục tiêu cơ bản đã đề ra trước đó.

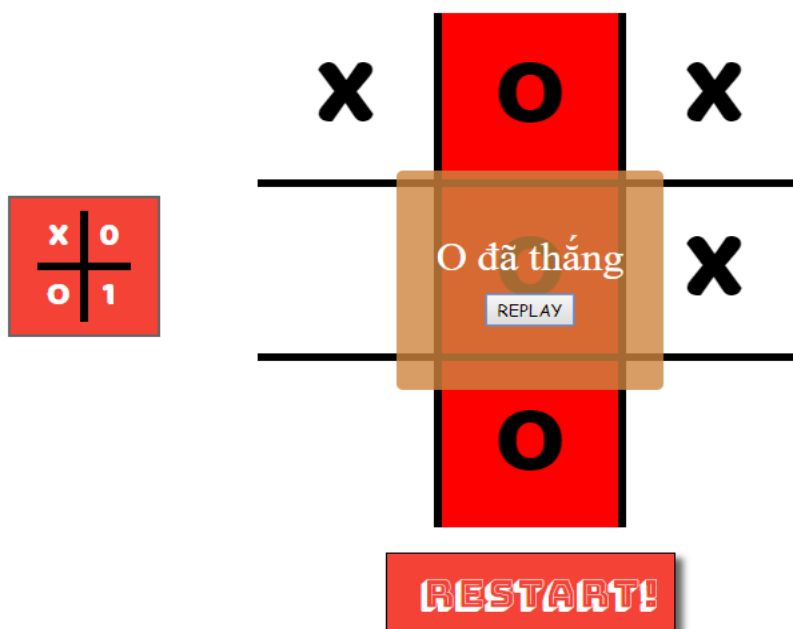
Sau đây là hình ảnh khi tiến hành chơi thực tế và kết quả thu được:



Hình 8: Chế độ chơi 2 người một máy(X thắng)

MULTIPLAYER

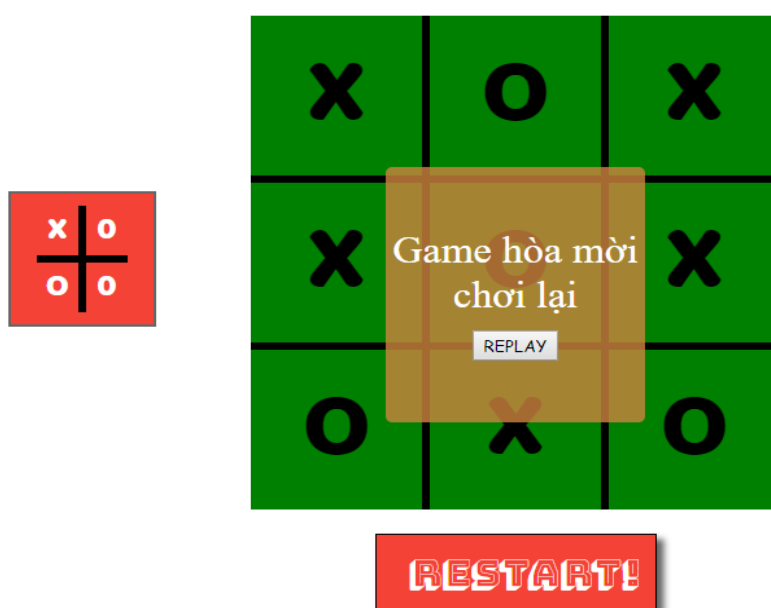
Player1: X Player2: Y Chúc mừng O đã thắng !!!



Hình 9: Chế độ chơi 2 người một máy(O thắng)

MULTIPLAYER

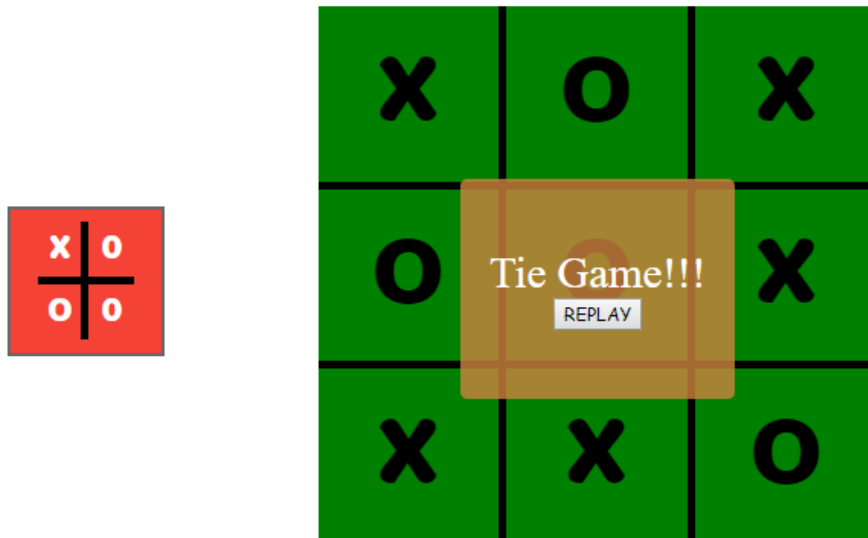
Player1: X Player2: Y Đến lượt X



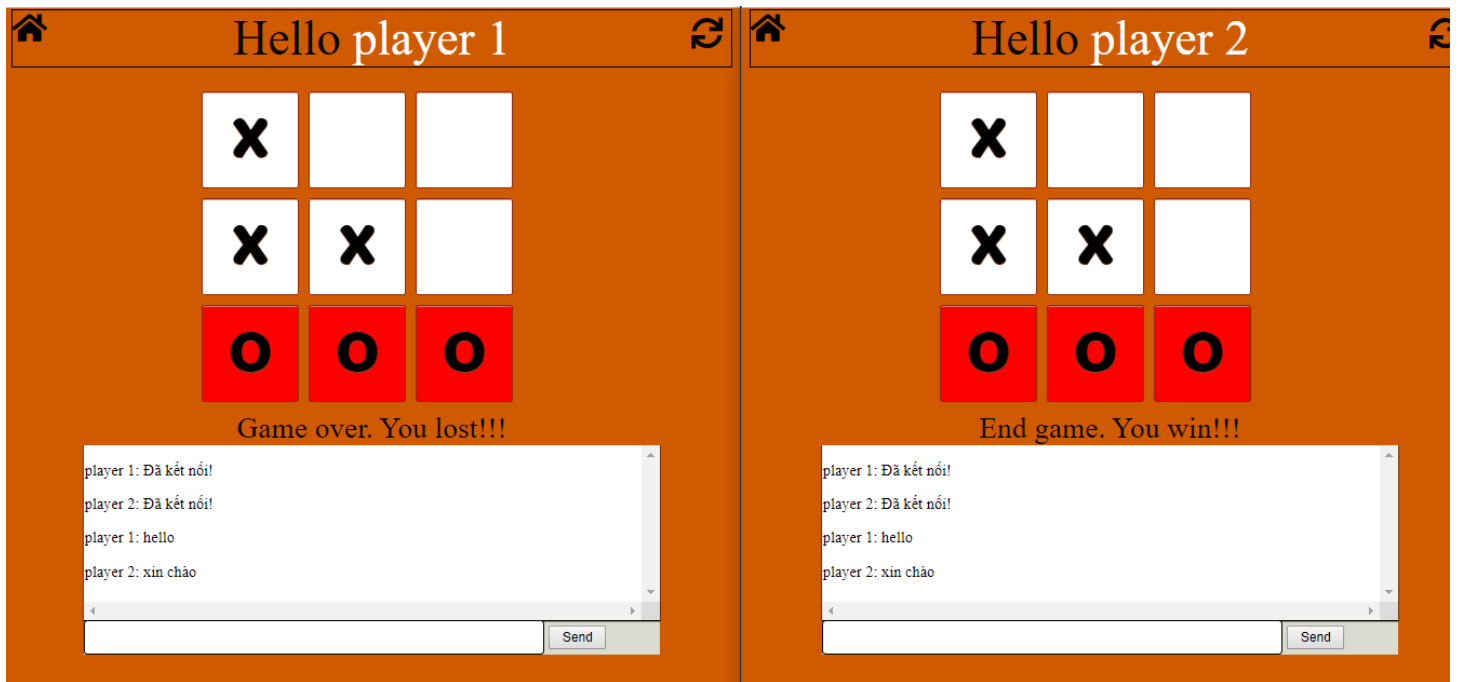
Hình 10: Chế độ chơi 2 người một máy(X và O hòa)

PLAY VS COMPUTER

Player: X Computer: Y Đến lượt X



Hình 11: Chế độ chơi giữa người và máy tính



Hình 12: Chế độ chơi trực tuyến

7. Phần kết luận

Việc ứng dụng thuật toán Minimax và lập trình xử lý thời gian thực vào trò chơi Tic Tac Toe đã giúp chúng ta hiểu được thuật toán hoạt động như thế nào và ứng dụng thuật toán vào thực tế. Xử lý thời gian thực giúp trải nghiệm người dùng thêm phong phú và chương trình thêm đa dạng.

Dù chương trình còn nhiều hạn chế, nhưng cũng đã đạt được những mục tiêu cơ bản ban đầu ta đã đề ra.

Hướng phát triển tiếp tục của sản phẩm là ta có thể tối ưu giải thuật, kết hợp Minimax với giải thuật cắt tỉa Alpha-Beta để tăng hiệu năng của chương trình. Hay nâng cấp một biến thể khác của game Tic Tac Toe là game Gomoku để nâng cấp độ khó của trò chơi. Kết hợp thêm tính năng chơi thêm Room để tăng tính đa dạng cho trò chơi.

8. Tài liệu tham khảo

1. MINIMAX - Russell, Stuart J.; Norvig, Peter (2003), Artificial Intelligence: A Modern Approach (2nd ed.), Upper Saddle River, New Jersey: Prentice Hall, pp. 163–171
2. GAME TREE - Daniel Roche (2013). SI486D: Randomness in Computing, Game Trees Unit. United States Naval Academy, Computer Science Department
3. Paul Krill (2 June 2014). "Socket.IO JavaScript framework ready for real-time apps"
4. "WebSockets - MDN". developer.mozilla.org. Mozilla Foundation. 2011-09-30. Retrieved 2011-12-10