Tài Liệu Thiết Kế

của

Trò Chơi Tic Tac Toe

Phiên bản v1.0 được phê chuẩn

Được chuẩn bị bởi:

Lê Nguyên Thức\_B1400731

22/10/2017

**Mục Lục**

[Theo dõi phiên bản tài liệu 1](#_Toc495654838)

[1. Giới thiệu 2](#_Toc495654839)

[1.1 Mục đích 2](#_Toc495654840)

[1.2 Phạm vi 2](#_Toc495654841)

[1.3 Qui ước tài liệu 3](#_Toc495654842)

[1.4 Bảng chú giải thuật ngữ 3](#_Toc495654843)

[1.5 Tài liệu tham khảo 3](#_Toc495654844)

[1.6 Tổng quan về tài liệu 4](#_Toc495654845)

[2. Tổng quan hệ thống 4](#_Toc495654846)

[3. Kiến trúc hệ thống 5](#_Toc495654847)

[3.1 Thiết kế kiến trúc 5](#_Toc495654848)

[4. Thiết kế theo chức năng 6](#_Toc495654849)

[4.1 Các chức năng của hệ thống 6](#_Toc495654850)

[4.2 Chức năng giao diện trò chơi (UC\_001). 7](#_Toc495654852)

[4.3 Chức năng thoát trò chơi và chơi mới (UC\_002) 9](#_Toc495654853)

[4.4 Chức năng chọn người hay máy đi trước (UC\_003) 10](#_Toc495654854)

[5. Thuật toán AI trong trò chơi 11](#_Toc495654855)

# Theo dõi phiên bản tài liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên** | **Ngày** | **Lý do thay đổi** | **Phiên bản** |
| Tài liệu thiết kế trò chơi | 02/03/2017 |  | V1.0 |

# Giới thiệu

## Mục đích

* Tài liệu thiết kế này mô tả các mục tiêu thiết kế và các cân nhắc, cung cấp tổng quan cấp cao về kiến ​​trúc hệ thống, và mô tả thiết kế dữ liệu gắn với hệ thống, cũng như giao diện người chơi và các kịch bản hoạt động.Thiết kế hệ thống tiếp tục được phân tách ra thành ở mức độ thấp chi tiết về thiết kế các thành phần con, bao gồm cả phần cứng, phần mềm, điều khiển toàn vẹn của hệ thống và giao diện bên ngoài.
* Tài liệu này cung cấp cho các đối tượng:
* Nhóm phát triển phần mềm: đọc và hiều về các thiết kế giao diện, thiết kếcơ sở dữ liệu nhằm thực hiện đúng như thiết kế.
* Người thiết kế: tham khảo về thiết kế kiến trúc và giao diện phần mềm.
* Người chơi: đáp ứng tính giải trí,giảm strees trong những lúc căng thẳng học tập và công việc trong cuộc sống.

## Phạm vi

* Lĩnh vực sản phẩm:
* Ứng dụng công nghệ thông tin trong quản lý.
* Sản phẩm ứng dụng nghiêm cứu trong lĩnh vực giải trí sử dụng trí tệu nhân tạo trong công nghệ thông tin.
* Phạm vi: Được triển khai trên phạm vi toàn thể con người có nhu cầu giải trí, cần tính tiện lợi và nhanh chóng nhưng vẫn đảm bảo về mặt giải trí.
* Đối tượng phục vụ của phần mềm: tất cả mọi người từ trẻ em tới người lớn.
* Nội dung cơ bản: Nghiên cứu ứng dụng game trò chơi sử dụng chế độ AI, thuật toán cắt tỉa alpha vào game (Tic Tac Toe).

## Qui ước tài liệu

Cấu trúc của tài liệu: Các tiêu đề được đánh dấu theo danh sách đánh số (ví dụ, 1, 2, 3, v.v ...), các tiêu đề nhỏ hơn là đánh dấu theo tiểu mục được đánh số (ví dụ: 1.1, 1.2, 1.3, v.v ...). Các mục dạng liệt kê được thể hiện bằng dấu chấm đầu dòng (•).

Quy ước văn bản:

* Font: Calibri.
* Font size: 13.
* Font color text: đen.
* Font color title: đen.
* Margin-left: 3.5 cm.
* Margin-right: 2 cm.
* Margin-top: 3 cm.
* Margin-bottom: 3 cm.
* Line spacing: 1.2 pt.
* Tiêu đề được in đậm và lớn hơn nội dung 2pt. Mỗi tiêu đề sẽ tăng 2pt nếu tiêu đề đó lớn hơn tiêu đề trước.

## Bảng chú giải thuật ngữ

|  |  |  |
| --- | --- | --- |
| STT | Thuật ngữ / Từ viết tắt | Định nghĩa / Giải thích |
| 1 | SV | Sinh viên |
| 2 | CB | Cán bộ |
| 3 | HT | Hệ thống |
| 4 | MSSV | Mã số sinh viên |
| 5 | MSCB | Mã số cán bộ |
| 6 | SK | Sự kiện |
| 7 | Server | Máy chủ |
| 8 | Client | Máy trạm |
| 9 | Import | Đưa vào |

## Tài liệu tham khảo

[1] Ths Phan Phương Lan, Silde Bài giảng Nhập môn công nghệ phần mềm, mẫu hướng dẫn các tài liệu kế hoạch, đặc tả yêu cầu, tài liệu thiết kế, Khoa CNTT & TT-Đại học Cần Thơ, năm 2014. <Url: [http://elcit.ctu.edu.vn/course/view.php?id=1653](http://elcit.ctu.edu.vn/course/view.php?id=1653%20) >

[2] TS Huỳnh Xuân Hiệp và Ths Phan Phương Lan, Giáo trình nhập môn Công nghệ phần mềm, Đại học Cần Thơ 2011.

[3] <http://tailieu.vn/tag/thiet-ke-mau-co-ban.html>.

[4]<http://doc.edu.vn/tai-lieu/do-an-phan-tich-va-thiet-ke-game-don-gian-6240/>.

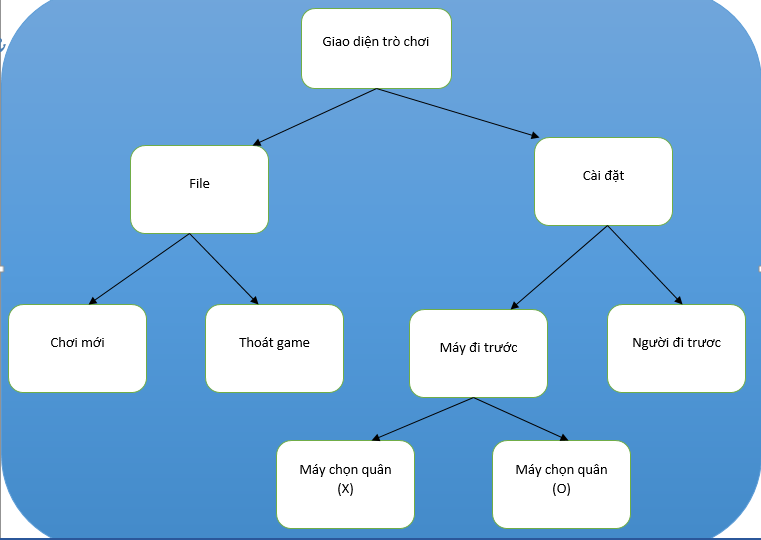
## Tổng quan về tài liệu

* Kiến trúc hệ thống:
  + Mô tả phân rả cấu trúc
  + Đưa ra các lý do lựa chọn thiết kế kiến trúc.
  + Mô hình dòng dữ liệu mức cao
* Thiết kế dữ liệu:
  + Mô tả dữ liệu
  + Các ràng buộc về dữ liệu.
* Thiết kế theo chức năng:
  + Mục đích
  + Giao diện
  + Các thành phần trong giao diện
  + Dữ liệu được sử dụng
  + Cách xử lý.
  + Hàm/ sự kiện.
  + Các ràng buộc.

# Tổng quan hệ thống

* Trò chơi được viết bằng ngôn ngữ C# chạy được trên các hệ điều hành của máy tính thông qua phần mềm hỗ trợ Visual Studio.
* Trò chơi cung cấp các chức năng cơ bản: chơi game,chơi mới,thoát game,tùy chọn(chơi với máy,máy đi trước,người đi trước,chọn quân cờ cho máy hay người chơi).Đây là game đối kháng chỉ người chơi với máy.

# Kiến trúc hệ thống



## Thiết kế kiến trúc

# Thiết kế theo chức năng

## Các chức năng của hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Mã | Tên chức năng | Mô tả |
| 1 | UC\_001 | -Giao diện trò chơi | -Giao diện khi bạn vào trò chơi |
| 3 | UC\_002 | -Thoát trò chơi  -Chơi mới | -Chức năng chính là thoát khỏi trò chơi khi bạn muốn kết thúc.  -Bạn có thể chơi mới lại khi kết thúc game hay đang chơi bạn cũng có thể chơi mới lại. |
| 6 | UC\_003 | -Người đi trước  -Máy đi trước  -Máy chơi X  -Máy chơi O | -Chọn người chơi sẽ đi trước hay máy đi trước.  -Máy chọn quân X hoặc quân O. |

# 

# 



## Chức năng giao diện trò chơi (UC\_001).

* **Mục đích**: Cho phép người chơi vào game thao tác các chức năng của trò chơi.
* **Giao diện**: Có tên phần mềm, có ô để nhập tài khoản, có ô nhập mật khẩu, một nút đăng nhập, một nút hủy.

Hình 1: hình ảnh giao diện trò chơi.

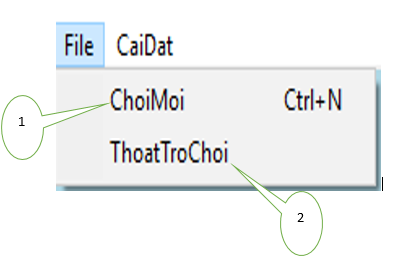
* **Các thành phần trong giao diện**:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Loại điều khiển | Giá trị mặc định | Mô tả |
| 1 | Button file | Trống | Khi nhấp chuột vào có 2 mục chức năng là chơi mới và thoát trò chơi |
| 2 | Button cài đặt | Trống | Khi nhấp chuột vào có 4 mục chức năng là người chơi trước hay máy chơi trước,máy chọn quân X hay quân O. |
| 3 | table | Trống | Người dùng click chuột vào để đánh cờ |

## 

## Chức năng thoát trò chơi và chơi mới (UC\_002)

* **Mục đích**: giúp bạn có thể chơi lại hay thoát khỏi trò chơi.
* Hình 2: hình ảnh giao diện chơi mới và thoát trò chơi



* **Các thành phần trong giao diện**:

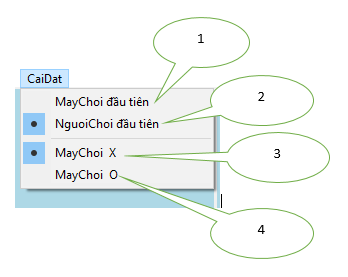
|  |  |  |  |
| --- | --- | --- | --- |
| STT | Loại điều khiển | Giá trị mặc định | Mô tả |
| 1 | Button chơi mới |  | Làm mới lại trò chơi |
| 2 | Button thoát trò chơi |  | Thoát khỏi giao diện trò chơi. |

* **Cách xử lý:** Người dùng chọn mục **file** ,sẽ có 2 chức năng sẽ tự động hiển thị trong danh mục file hiện có, người dùng có thể tùy chọn 1 trong tổng số 2 chức năng của mục file.Nếu người dùng chọn chơi mới thì giao diện trò chơi sẽ được hiện ra như lúc đầu,nếu chọn thoát trò chơi thì trò chơi ngay lập tức được đóng lại.

## Chức năng chọn người hay máy đi trước (UC\_003)

* **Mục đích**: cho phép người chơi chọn xem máy đi trước hay người chơi đi trước.

Hình 3: hình ảnh giao diện người đi trước hay máy đi trước.



* **Các thành phần trong giao diện**:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Loại điều khiển | Giá trị mặc định | Mô tả |
| 1 | Button máy chơi đầu tiên |  | Chọn máy chơi trước |
| 2 | Button người chơi đầu tiên |  | Chọn người chơi trước |
| 3 | Button máy chơi X |  | Chọn máy chơi quân X |
| 4 | Button máy chơi O |  | Chọn máy chơi quân O |

* **Cách xử lý:** Người dùng chọn mục **cài đặt**,1 loạt chức năng sẽ tự động hiển thị trong danh mục cài đặt hiện có, người dùng có thể tùy chọn 2 trong tổng số 4 chức năng của mục cài đặt.Sau khi thực hiện hai trong những hoạt động trên trò chơi sẽ hiện lại mục giao diện chính để chơi game.

# Thuật toán AI trong trò chơi

//Khởi tạo alpha,beta

int alpha = -2;

int beta = 2;

DiChuyen move = NhanGiaTriDiChuyenToiUu(this.BanCo, this.LoaiNguoiChoi, alpha, beta);

Object[] args = { move };

MakeComputerMoveDelegate MakeMoveDelegate = new MakeComputerMoveDelegate(this.TuDongDiChuyen);

this.BeginInvoke(MakeMoveDelegate, args);

}

//Thuật toán tiềm kiếm minimax chuẩn với cắt tỉa alpha-beta.Tính trong luồng sự kiện:

private DiChuyen NhanGiaTriDiChuyenToiUu(OCo banco, int NguoiChoiHienTai, int alpha, int beta)

{

DiChuyen DiChuyenToiUu = null;

int iPossibleMoves = banco.iEmptySquares;

//trò chơi bắt đầu bởi 1 ô cờ ngẫu nhiên

//1 trong số 2 người chơi sẽ được chọn ngẫu nhiên để đánh

Random rand = new Random();

int i = rand.Next(banco.iBoardSize);

int j = rand.Next(banco.iBoardSize);

while (iPossibleMoves > 0)

{

//tìm tiếp nước đi tiếp theo

do

{

if (i < KichThuocBang - 1)

{

i++;

}

else if (j < KichThuocBang - 1)

{

i = 0; j++;

}

else

{

i = 0; j = 0;

}

} while (banco.aiBoard[i, j] != OCo.Empty) ;

DiChuyen NewMove = new DiChuyen(i, j);

iPossibleMoves--;

// DiChuyen

OCo NewBoard = new OCo(banco);

NewBoard.MakeMove(NguoiChoiHienTai, NewMove);

NewBoard.KiemTraBanCo();

if (NewBoard.BoardState == GameState.InProgress)

{

DiChuyen tempMove = NhanGiaTriDiChuyenToiUu(NewBoard, -NguoiChoiHienTai, alpha, beta);

NewMove.iRank = tempMove.iRank;

}

else

{

if(NewBoard.BoardState == GameState.Hoa)

NewMove.iRank = 0;

else

{

if (NewBoard.BoardState == GameState.MayThang)

NewMove.iRank = -1;

else

{

if (NewBoard.BoardState == GameState.NguoiChoiThang)

NewMove.iRank = 1;

}

}

}

//kiểm tra di chuyển mới có phải tối ưu không:

if (DiChuyenToiUu == null ||

(NguoiChoiHienTai == TicTacToe.MayChoi && NewMove.iRank < DiChuyenToiUu.iRank) ||

(NguoiChoiHienTai == TicTacToe.NguoiChoi && NewMove.iRank > DiChuyenToiUu.iRank))

{

DiChuyenToiUu = NewMove;

}

//cập nhật alpha,beta nếu cần:

if (NguoiChoiHienTai == TicTacToe.MayChoi && DiChuyenToiUu.iRank < beta)

beta = DiChuyenToiUu.iRank;

if (NguoiChoiHienTai == TicTacToe.NguoiChoi && DiChuyenToiUu.iRank > alpha)

alpha = DiChuyenToiUu.iRank;

//kiểm tra điều kiện cắt tỉa:

if (alpha > beta)

{

//tỉa nhánh

iPossibleMoves = 0;

}

}

return DiChuyenToiUu;

}