**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HCM**

**KHOA [CÔNG NGHỆ THÔNG TIN]**

****

**ĐỒ ÁN HỌC PHẦN**

**Xây dựng game đoán số**

**Sinh viên thực hiện: Nguyễn Huỳnh Duy Tân**

**Mã số sinh viên: 21110852**

**Khóa: K21**

**Ngành: Công nghệ thông tin**

**Giảng viên hướng dẫn: Từ Tuyết Hồng**

TP.HCM, ngày 02 Tháng 12 năm 2023

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA [CÔNG NGHỆ THÔNG TIN]**

**A logo of hands holding a book

Description automatically generated**

**ĐỀ TÀI ĐỒ ÁN HỌC PHẦN**

**Xây dựng game đoán số**

Họ tên sinh viên**: Nguyễn Huỳnh Duy Tân**

Mã số sinh viên: **21110852**

Khóa: **K21**

Giảng viên hướng dẫn: **Từ Tuyết Hồng**

**NHẬN XÉT CỦA GIẢNG VIÊN**

**………………………………………………………..**

**………………………………………………………..**

**TP.HCM, ngày 02 tháng 12 năm 2023**

**Giảng viên: Ký tên**

**MỤC LỤC**

[DANH MỤC CÁC HÌNH 4](#_Toc152436618)

[DANH MỤC CÁC BẢNG 4](#_Toc152436619)

[Chương 1: GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN 5](#_Toc152436620)

[1.1. Tổng quan về Đồ án học phần 5](#_Toc152436621)

[1.2. Bố cục của báo cáo 5](#_Toc152436622)

[Chương 2: PHÂN CÔNG CÔNG VI 6](#_Toc152436623)

[Chương 3: THIẾT KẾ 7](#_Toc152436624)

[**3.1.** Thuật toán 7](#_Toc152436625)

[**3.2.** Thiết kế lớp 7](#_Toc152436626)

[[1]. A. Danh mục các lớp 7](#_Toc152436627)

[[2]. B. Mô tả các phương thức trong một lớp 9](#_Toc152436628)

[**3.3.** Thiết kế cơ sở dữ liệu 15](#_Toc152436629)

[**3.4.** Bảng mô tả các field trong table Games 16](#_Toc152436630)

[**3.5.** **.**Bảng mô tả các field trong table Players 16](#_Toc152436631)

[**3.6.** Thiết kế giao diện 17](#_Toc152436632)

[Chương 4: CÀI ĐẶT VÀ KIỂM THỬ 19](#_Toc152436633)

[Chương 5: KẾT LUẬN 21](#_Toc152436634)

# DANH MỤC CÁC HÌNH

|  |  |  |
| --- | --- | --- |
| TT | Kí hiệu | Trang |
| 1 | Hình 3.3.1 | 15 |
| 2 | Hình 3.6.1 | 17 |
| 3 | Hình 3.6.2 | 18 |
| 4 | Hình 3.6.3 | 18 |
| 5 | Hình 3.6.4 | 19 |
| 6 | Hình 3.6.5 | 19 |

# DANH MỤC CÁC BẢNG

|  |  |  |
| --- | --- | --- |
| TT | Kí hiệu | Trang |
| 1 | Bảng 2.1 | 6 |
| 2 | Bảng 3.2.1 | 7 |
| 3 | Bảng 3.2.2 | 9 |
| 4 | Bảng 3.3.1 | 15 |
| 5 | Bảng 3.4.1 | 16 |
| 6 | Bảng 3.5.1 | 16 |
| 7 | Bảng 3.6.1 | 17 |
| 8 | Bảng 4.1 | 20 |

# GIỚI THIỆU VỀ ĐỒ ÁN HỌC PHẦN

## Tổng quan về Đồ án học phần

Đồ án Học phần CNTT vận dụng kiến thức và kỹ năng về lập trình hướng đối tượng để xây dựng một phần mềm với giao diện đồ hoạ (GUI) bằng winform C# để người dùng tương tác. Phần mềm này được thiết kế để chơi một trò chơi đơn giản nhưng thú vị: đoán một số từ 1 đến 100. Người chơi sẽ cố gắng đoán số đúng trong ít lượt nhất có thể.

Với dữ liệu đầu vào duy nhất cho trò chơi này là số mà người chơi cần phải đoán. Số này phải nằm trong khoảng từ 1 đến 100 và sẽ phát sinh ngẫu nhiên sau mỗi ván chơi. Sau đó, cho phép người chơi nhập số vào, sau đó kiểm tra xem số của người chơi nhập có đúng với số ngẫu nhiên mà ván chơi đã khởi tạo chưa, nếu đúng chỉ trong 3 lần sẽ in ra thông báo "You are a lucky person today!", còn nếu đoán đúng trong vòng 4 lượt trở lên, dòng “Great guess!” sẽ được in ra. Nếu không may đoán trượt quá 10 lượt, dòng thông báo “You lost!” báo hiệu việc người chơi đã thua sẽ hiện lên và trò chơi kết thúc. Trong quá trình đoán, sẽ có những thông báo để người chơi biết mình đang ở gần hoặc xa số cần đoán, ví dụ, nếu số đoán nhỏ hơn số cần đoán trong vòng 30 số, dòng thông báo "Just a little bit more! higher!" sẽ hiện lên, để nhắc người chơi đoán số lớn hơn nữa, tương tự, trường hợp lớn hơn số cần đoán thì sẽ là "Just a little bit more! Lower!”, còn nếu nằm ở khoảng 10 đơn vị, "You are so close! A little bit…!” + “higher” hoặc lower.

Vì là một trò chơi, mục đích chính của phần mềm này là giải trí, với giao diện đơn giản, dễ sử dụng, khởi động game sẽ là màn hình để người chơi nhập tên, sau đó sẽ là màn hình bắt đầu game để người chơi nhập và kiểm tra dự đoán của mình. Người sử dụng có thể tận dụng để giết thời gian hoặc tự thử thách bản thân bằng cách đoán đúng nhiều số nhất với ít lượt nhất,…

## Bố cục của báo cáo

Báo cáo gồm những nội dung như sau:

Chương 1: Giới thiệu đồ án của Học phần

Chương 2: Phân công công việc

Chương 3: Thiết kế

Chương 4: Cài đặt và kiểm thử

Chương 5: Kết luận

# PHÂN CÔNG CÔNG VI

Bảng 2.1

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên SV | Mô tả khái quát mảng công việc SV thực hiện trong đồ án. | Ước tính phần trăm đóng góp |
| 1 | Nguyễn Huỳnh Duy Tân | Xây dựng game đoán số sử dụng winform C# | 100% |

1. **THIẾT KẾ**

## Thuật toán

Sử dụng lớp Random trong thư viện System.Random có sẵn, sau đó gọi đến hàm Random.Next(n1, n2) để tạo ra một số nguyên ngẫu nhiên trong khoảng từ n1 (bao gồm n1) đến n2 (không bao gồm).

## Thiết kế lớp

### A. Danh mục các lớp

Sinh viên phụ trách: Nguyễn Huỳnh Duy Tân (Nguồn tham khảo: <https://youtu.be/S0GPI1mGT_A?si=p4c-DHnQozIwfjUV>)

Bảng 3.2.1

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp Thừa kế hoặc dẫn xuất từ (ghi rõ tên lớp cơ sở) – nếu có | Mục đích |
| 1 | Games.cs | Đại diện cho 1 ván chơi, lưu trữ thông tin và cập nhật thông tin của 1 ván chơi gồm ID, ID người chơi, số lượt đoán trong 1 ván, số cần đoán, kết quả. Ngoài ra còn các phương thức để khởi tạo ván game và khởi tạo số ngẫu nhiên |
| 2 | GameDAO.cs | Sử dụng các câu truy vấn SQL để tương tác với bảng cơ sở dữ liệu Games như thêm, cập nhật thông tin trong một ván chơi. |
| 3 | Player.cs | Đại diện thông tin của một người chơi, dùng để tạo mới, lưu trữ và cập nhật thông tin. |
| 4 | PlayerDAO.cs | Sử dụng các câu truy vấn SQL để tương tác với bảng cơ sở dữ liệu Players như thêm, cập nhật thông tin của người chơi trong và sau khi hoàn thành một ván chơi. |
| 5 | NumberGuessingGame.cs  Thừa kế class Form tự khởi tạo. | Giao diện và xử lí chính của ứng dụng. Lấy thông tin từ các class Player và Games để khởi tạo, xử lí và sau đó gọi đến các class DAO để tiến hành xử lí thông tin khi người dùng kích hoạt một event nào đó (như khi click vào một nút trên giao diện) trong một ván game. |
| 6 | DBConnection | Tạo và quản lý kết nối với cơ sở dữ liệu và thực hiện các thao tác trên cơ sở dữ liệu. |

### B. Mô tả các phương thức trong một lớp

Sinh viên phụ trách: Nguyễn Huỳnh Duy Tân (Nguồn tham khảo: https://youtu.be/S0GPI1mGT\_A?si=p4c-DHnQozIwfjUV)

Bảng 3.2.2

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng khai báo |
| 1 | Games(int guessCount, int number, string winLose, int gameID)  Input: guessCount, number, winLose, gameID  Output: không có | Khởi tạo đối tượng Games dùng để cập nhật thông tin cho ván game hiện tại. | Games.cs (19) |
| 2 | Games(int playerID, int guessCount, int number, string winLose)  Input: playerID, guessCount, number, winLose  Output: không có. | Dùng để khởi tạo ván game mới với các thông tin mặc định được truyền vào | Games.cs (27) |
| 3 | getRandomNum(int n1, int n2)  Input: n1, n2  Output: một số nguyên ngẫu nhiên. | Tạo ra một số ngẫu nhiên trong khoảng từ n1 đến n2. Nếu n1 lớn hơn n2, phương thức sẽ hoán đổi giá trị của n1 và n2 trước khi tạo số ngẫu nhiên. | Game.cs (41) |
| 4 | Add(Games g)  Input: g  Output: gameID | Thêm mới một ván game vào cơ sở dữ liệu sau đó trả về ID của ván game vừa được thêm bằng cách gọi phương thức GetID ở lớp . | GameDAO.cs (15) |
| 5 | Update(Games g)  Input: g  Output: không có | Cập nhật thông tin của ván game dựa trên ID được truyền vào. | GameDAO.cs (20) |
| 6 | Player(string playerName, int playCount, int winCount, int loseCount)  Input: playerName, playCount, winCount, loseCount  Output: không có | Khởi tạo đối tượng Player mới với thông tin mặc định được truyền vào. | Player.cs (22) |
| 7 | Player(int playCount, int winCount, int loseCount, int playerID)  Input: playCount, winCount, loseCount, playerID  Output: không có | Khởi tạo đối tượng Player mới để tiến hành cập nhật dựa trên ID và các giá trị tương ứng được truyền vào. | Player.cs (29) |
| 8 | Add(Player p)  Input: p  Output: không có | Thêm một đối tượng Player mới vào cơ sở dữ liệu. | PlayerDAO.cs (15) |
| 9 | Update(Player p)  Input: p  Output: không có | Cập nhật đối tượng Player tương ứng với ID được truyền vào. | PlayerDAO.cs (20) |
| 10 | checkName(string name)  Input: name  Output: ID của người chơi có tên là {name} | Kiểm tra xem tên mà người chơi nhập vào có tồn tại trong bảng cơ sở dữ liệu hay chưa. | PlayerDAO.cs (25) |
| 11 | getPlayCount(string name)  Input: name  Output: số lần chơi | Lấy tổng số lần chơi của người chơi có name tương ứng. | PlayerDAO.cs (38) |
| 12 | getWinCount(string name)  Input: name  Output: số lần thắng | Lấy tổng số lần thắng của người chơi có name tương ứng. | PlayerDAO.cs (51) |
| 13 | getLoseCount(string name)  Input: name  Output: số lần thua | Lấy tổng số lần thua của người chơi có name tương ứng. | PlayerDAO.cs (64) |
| 14 | LoadList(string sqlStr)  Input: sqlStr  Output: Datatable chứa dữ liệu được lấy ra | Lấy danh sách dữ liệu từ cơ sở dữ liệu dựa trên câu lệnh SQL được truyền vào. | DBConnection.cs (15) |
| 15 | Process(string sqlStr)  Input: sqlStr  Output: không có | Thực hiện một câu lệnh SQL trên cơ sở dữ liệu, như thêm, cập nhật hoặc xóa dữ liệu. | DBConnection.cs (35) |
| 15 | GetID(string sqlStr)  Input: sqlStr  Output: ID của bản ghi vừa được thêm vào. | Lấy ID của bản ghi mới được thêm vào cơ sở dữ liệu. | DBConnection.cs (55) |
| 16 | NumberGuessingGame()  Input: không có  Output: không có | Khởi tạo một trò chơi mới, xác định người chơi, và hiển thị giao diện để bắt đầu một ván chơi mới của ứng dụng đoán số. | NumberGuessingGame.cs (19) |
| 16 | getPlayerInfo(string pname)  Input: pname  Output: đối tượng playertemp chứa các thông tin đã lấy từ cơ sở dữ liệu | Lấy thông tin về một người chơi từ cơ sở dữ liệu dựa trên tên người chơi tương ứng. | NumberGuessingGame.cs (46) |
| 17 | txtInput\_Click(object sender, EventArgs e)  Input: sender, e  Output: không có | Làm trống textbox mỗi khi người dùng click vào | NumberGuessingGame.cs (56) |
| 18 | loadNum(int n1, int n2)  Input: n1, n2  Output: không có | Set số ngẫu nhiên trong khoảng từ n1 đến n2. | NumberGuessingGame.cs (60) |
| 19 | IsValidNumber(string input)  Input: input  Output: true/false | Kiểm tra người dùng nhập vào có phải là số hợp lệ hay không. | NumberGuessingGame.cs (67) |
| 20 | EndGame(string winorlose)  Input: winorlose  Output: không có | Hiển thị hộp thoại thông báo kết thúc ván chơi, xử lí cập nhật, thêm mới các đối tượng. | NumberGuessingGame.cs (77) |
| 21 | btnCheck\_Click(object sender, EventArgs e)  Input: sender, e  Output: không có | Xử lý sự kiện khi người chơi click nút "Check", gọi đến các hàm kiểm tra, khởi tạo bên trên và xử lí thông báo hiển thị khi người dùng nhập dự đoán của mình. | NumberGuessingGame.cs (97) |
| 22 | update(string winorlose)  Input: winorlose  Output: không có | Cập nhật thông tin người chơi và trò chơi trong cơ sở dữ liệu. | NumberGuessingGame.cs (191) |
| 23 | btnStop\_Click(object sender, EventArgs e)  Input: sender, e  Output: không có | Xử lí sự kiện khi người chơi nhấn nút “Stop”, hiển thị thông báo và cập nhật thông tin của các đối tượng. | NumberGuessingGame.cs (199) |

## Thiết kế cơ sở dữ liệu

Mô hình thực thể kết hợp

A diagram of a computer

Description automatically generated

Hình 3.3.1

Bảng 3.3.1

|  |  |  |
| --- | --- | --- |
| TT | Tên bảng | Mục đích |
| 1 | Games | Lưu trữ thông tin về các ván game. |
| 2 | Player | Lưu trữ thông tin về các ván người chơi, ở đây người chơi không trùng tên với nhau. |

## Bảng mô tả các field trong table Games

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | gameID | Int | ID duy nhất của mỗi ván chơi, dùng để cập nhật thông tin. |
| 2 | playerID | Int | ID duy nhất của người chơi, dùng để cập nhật thông tin. |
| 3 | guessCount | Int | Tổng số lần đoán của người chơi trong một ván. |
| 4 | Number | Int | Số cần đoán của ván game. |
| 5 | WinLose | Nvarchar(50) | Kết quả thắng thua của ván game. |

Bảng 3.4.1

## **.**Bảng mô tả các field trong table Players

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mục đích |
| 1 | playerID | Int | ID duy nhất của người chơi, dùng để cập nhật thông tin. |
| 2 | playerName | Varchar(255) | Tên của người chơi, mỗi tên tương ứng với 1 ID. |
| 3 | playCount | Int | Tổng số lần chơi của người chơi. |
| 4 | winCount | Int | Tổng số lượt thắng của người chơi. |
| 5 | loseCount | Int | Tổng số lượt thua của người chơi. |

Bảng 3.5.1

## Thiết kế giao diện

Bảng 3.6.1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TT | | Màn hình/Cửa sổ/Dialog | Mục đích | Giải thích ngắn gọn các quyết định thiết kế khi thiết kế màn hình (Người thiết kế ở trên 1 dòng. Sau dòng đó là phần giải thích. | | | |
| 1 | Welcome dialog.  Hình 3.6.1 | | | | Cho phép người dùng nhập tên để bắt đầu chơi | Sử dụng InputBox để người chơi có thể thấy rõ mình đã nhập những gì. |
| 2 | Cửa sổ Game    Hình 3.6.2 | | | | Cửa sổ để người dùng nhập vào và kiểm tra dự đoán của mình | Sử dụng label để trực quan hóa quá trình chơi, đồng thời không cho phép người chơi chỉnh sửa. Textbox cho khả năng kiểm soát việc nhập dữ liệu vào, 2 nút stop và check tạo trải nghiệm tương tác khi nhấn vào sẽ có phản hồi tương ứng. |
| 3 | Dialog thông báo nhập quá thấp    Hình 3.6.3 | | | | Để gợi ý cho người chơi biết số dự đoán vừa nhập là quá thấp. | Cung cấp thông tin thông qua một hộp thoại đơn giản nhưng lại dễ gây chú ý. |
| 4 | Dialog thông báo là rất gần số đúng nhưng hơi cao    Hình 3.6.4 | | | | Gợi ý cho người chơi biết số dự đoán vừa nhập là đã gần đúng, chỉ cần thấp xuống một tí | Cung cấp thông tin gợi ý thông qua một hộp thoại đơn giản nhưng lại dễ gây chú ý. |
| 5 | Dialog thông báo nhập là khá gần nhưng cần cao hơn    Hình 3.6.5 | | | | Gợi ý cho người chơi biết số dự đoán vừa nhập là khá gần, nhưng phải cao hơn nữa. | Cung cấp thông tin gợi ý thông qua một hộp thoại đơn giản nhưng lại dễ gây chú ý. |

1. **CÀI ĐẶT VÀ KIỂM THỬ**

Bảng 4.1

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tình huống | Mục đích | Giải thích ngắn gọn quyết định lựa chọn tình huống và dữ liệu đầu vào |
| 1 | Dữ liệu vào: số âm hoặc chữ, kí tự khác số  Kết quả dự kiến: Thông báo yêu cầu nhập lại | Kiểm tra dự đoán của người dùng phải là một số hợp lệ trong khoảng từ 1 – 100 để có thể xử lí dữ liệu. | Đảm bảo không xảy ra lỗi trong quá trình xử lí và đảm bảo tính nhất quán cho trò chơi. |
| 2 | Dữ liệu vào: Nhập tên người chơi rỗng.  Kết quả dự kiến: Thông báo yêu cầu nhập tên lại | Xác định tên người chơi để theo dõi kết quả và thống kê. | Kiểm tra tình huống khi người chơi không nhập tên (pname rỗng) để đảm bảo thông báo yêu cầu nhập tên và yêu cầu nhập lại. |
| 3 | Dữ liệu vào: Người chơi đoán quá số lần quy định (10 lần)  Kết quả dự kiến: Thông báo người chơi thua cuộc | Xác định xem người chơi đã đoán đúng hay sai và cập nhật kết quả. | Kiểm tra tình huống khi người chơi đoán đúng trong số lần quy định (count <= 10) để thay đổi thông báo và cập nhật số lần chiến thắng. Kiểm tra khi người chơi đoán sai quá 10 lần để xử lý thất bại và cập nhật số lần thất bại. |
| 4 | Dữ liệu vào: để textbox rỗng và nhấn nút “Check!”  Kết quả dự kiến: Thông báo yêu cầu người chơi nhập lại một số hợp lệ | Xác định xem người chơi đã đoán hay chưa để cập nhất thông tin. | Đảm bảo rằng ứng dụng không gặp lỗi khi người chơi nhấn "Check" mà không nhập số dự đoán. |
| 5 | Dữ liệu vào: kết thúc game khi cần  Kết quả dự kiến: Thông báo kết quả và cập nhật thêm 1 lượt thua vào cơ sở dữ liệu | Cho phép người chơi kết thúc ván chơi và ứng dụng khi cần. | iểm tra tình huống khi người chơi nhấn nút "Stop" để kết thúc ván chơi và hiện thông báo kết quả theo yêu cầu. |

1. **KẾT LUẬN**

Dự án game đoán số đã đạt được một số mục tiêu quan trọng và mang lại một trải nghiệm đơn giản, nhưng hấp dẫn cho người chơi. Trong quá trình phát triển ứng dụng, nhiều khó khăn đã xuất hiện, nhưng qua đó, em cũng đã học được nhiều và áp dụng những giải pháp sáng tạo để vượt qua thách thức.

Mức Độ Hoàn Thành Mục Tiêu:

* Hoàn Thành Cơ Bản:

1. Đạt được: Hoàn thành cơ bản của trò chơi với khả năng sinh số ngẫu nhiên, kiểm tra đoán đúng/sai, cập nhật số lần đoán, số lần chơi, số lần thắng, số lần thua của người chơi, đồng thời cũng cập nhật được chi tiết số lần đoánvà số cần đoán trong 1 ván game, kết quả thắng thua.
2. Khó Khăn và Khắc Phục: Gặp khó khăn trong việc quản lý trạng thái game và xử lý sự kiện người chơi. Đã giải quyết bằng cách sử dụng mô hình đối tượng chặt chẽ và kiến trúc rõ ràng.

* Giao Diện Người Dùng (UI):

1. Đạt được: Thiết kế giao diện đơn giản, dễ sử dụng với các nút chức năng và hiển thị thông tin.
2. Khó khăn và khắc phục: Gặp thách thức trong việc xử lý sự kiện và cập nhật giao diện khiến UI phản hồi không luôn nhất quán. Đã sửa lỗi và cải thiện tính ổn định thông qua việc tự tìm hiểu và nghiên cứu.

Ý tưởng phát triển đồ án

* Nâng cao chức năng
  1. Mở rộng vùng đoán: Thêm tính năng cho phép người chơi chọn khoảng số cần đoán để tăng độ khó.
  2. Chế độ nhiều người chơi: Thêm chế độ cho nhiều người chơi cùng tham gia và cạnh tranh.
* Thúc Đẩy Tương Tác:

1. Tương tác đa dạng hơn: Bổ sung âm thanh, hiệu ứng hình ảnh để tạo ra trải nghiệm tương tác hấp dẫn hơn.
2. Thông kê và đua top tranh hạng: Thêm chức năng thống kê điểm số và xếp hạng để kích thích sự cạnh tranh.

Ưu Điểm và Hạn Chế:

* Ưu Điểm:

1. Giao diện đơn giản và thân thiện.
2. Tính ổn định và khả năng mở rộng cao.
3. Dễ dàng chơi và hiểu.
4. Tích hợp cơ sở dữ liệu để lưu trữ và quản lí thông tin

* Hạn Chế:
  1. Thiếu một số tính năng nâng cao và có thể tồn đọng một số lỗi chưa biết.
  2. Giao diện có thể được cải thiện về mặt thẩm mỹ.

Tổng Quan:

Dù có những thách thức, đồ án đã đạt được một mức độ hoàn thành tương đối cao, mang lại một trải nghiệm chơi game đơn giản và thú vị. Điều quan trọng là em đã học và ôn lại nhiều thứ từ quá trình phát triển và xác định rõ hướng phát triển tiếp theo để làm cho ứng dụng trở nên hoàn thiện hơn.