**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

KHOA CÔNG NGHỆ THÔNG TIN

C:\Program Files (x86)\Microsoft Office\MEDIA\OFFICE14\Lines\BD21325_.gif

C:\Program Files (x86)\Microsoft Office\MEDIA\CAGCAT10\j0205582.wmf

C:\Program Files (x86)\Microsoft Office\MEDIA\OFFICE14\Lines\BD14997_.gif

BÁO CÁO MÔN HỌC

“LẬP TRÌNH GAME

BẰNG NGÔN NGỮ JAVA”

GV : Nguyễn Trọng Phúc

Sinh viên thực hiện: + Đặng Tuấn Đạt (MSV: 181200376)

+ Nguyễn Việt Anh (MSV: 181202292)

LỚP: Công Nghệ Thông Tin 1

KHOA CÔNG NGHỆ THÔNG TIN

C:\Program Files (x86)\Microsoft Office\MEDIA\OFFICE14\Lines\BD21325_.gif

C:\Program Files (x86)\Microsoft Office\MEDIA\CAGCAT10\j0205582.wmf

C:\Program Files (x86)\Microsoft Office\MEDIA\OFFICE14\Lines\BD14997_.gif

BÁO CÁO MÔN HỌC

“LẬP TRÌNH GAME BẰNG NGÔN NGỮ

LẬP TRÌNH JAVA”

GV :

SVTH: + Đặng Tuấn Đạt

+ Nguyễn Việt Anh

LỚP: Công Nghệ Thông Tin 1

Lời nói đầu

Công nghệ thông tin (CNTT) ngày càng có vai trò qaun trọng trong cuộc sống hàng ngày của chúng ta . Việc ứng dụng CNTT vào các lĩnh vực trong đời sống giúp công việc được tiến hành nhanh chóng và hiệu quả hơn. Có rất nhiều công việc mới phát triển song song với sự phát triển của CNTT, một trong số đó là hiệu ứng game, hướng đi dịch vụ mang lại hiệu quả kinh tế lớn.

Chúng em chọn đề tài” Lập trình game Ai là triệu phú bằng ngôn ngữ java” nhằm tìm hiểu sâu hơn về ngôn ngữ java trong lập trình hướng đối tượng, từ đó viết một ứng dụng củ thể thử nghiệm làm cơ sở củng cố kiến thức và định hướng, kế hoạch xây dựng những ứng dụng game cụ thể , phát triển theo hướng dịch vụ trong tương lai.

1. **Lý do chọn đề tài**
   1. ***Mục tiêu của đề tài***

Hiểu khái niệm phân tích và giải quyết bài toán lập trình hướng đối tượng : Xác định các đối tượng lớp , mỗi quan hệ giữa các lớp trong một hệ thống thông tin .

Mục tiêu đề tài mục đích nhằm nghiên cứu môi trường phát triển , ngôn ngữ java xây dựng những ứng dụng cụ thể.

* 1. ***Phương pháp***

Bài tập lớn được thiết kế theo phương pháp “Lập Trinh Hướng Đối Tượng” (LTHĐT) bằng ngôn ngữ java.Chính vì vậy nó giải quyết được những vướng mắc gặp phải khi thiết kế theo phương pháp lập trình thủ tục thuần túy.

+ Mã chương trình rõ ràng , dễ đọc , dễ hiểu và cô đọng

+ Chương trình được tổ chức thành những Class lắp ghép lại với nhau thành một khối thống nhất

+ Mỗi class gồm nhiều Method đảm nhận các vai trò khác nhau trong chương trình

+ Chương trình có tính mềm dẻo cao

+ Có khả năng tái sử dụng tài nguyên

**2) Phân tích trò chơi**.

***2.1. Tên trò chơi và luật chơi***

- Đây là trò chơi mô phỏng theo trò chơi truyền hình nổi tiếng: “Ai là triệu phú”.

Trò chơi dưới đây sử dụng quy tắc và luật chơi của chương trình trên.Dưới đây là luật cụ thể của trò chơi.

- Luật chơi:

* Có tất cả 15 câu hỏi trong tựa game này,các câu hỏi được xuất hiện ngẫu nhiên theo thứ tự từ dễ đến khó. Các công hỏi chạy từ file, từ câu 1-5 có file lưu riêng, câu hỏi sẽ được lấy ngẫu nhiên từ đây; tương tự với các câu từ 1-10 và các câu từ 11-15. Khi bắt đầu trò chơi thì sẽ thiết lập ngay 15 câu hỏi cho game từ mức độ dễ đến khó.
* Trong trò chơi này, bạn sẽ có 2 sự trợ giúp bao gồm:
  + Trợ giúp 50/50: Trò chơi sẽ loại bỏ đi 2 phương án không chính xác.
  + Hỏi ý kiến khán giả: Trò chơi sẽ mô phỏng câu trả lời của khán giả, sự trợ giúp thường đưa ra câu trả lời đúng và câu trả lời đúng sẽ luôn có tỷ lệ hơn 20%.
  + Mỗi sự trợ giúp chỉ có tác dụng 1 lần trong suốt trò chơi.
* Với mỗi câu hỏi ta sẽ có 4 đáp án trả lời và chỉ có 1 đáp án đúng. Đưa ra câu trả lời bằng cách click vào lựa chọn bạn nghĩ là đúng.
* Ban đầu điểm của bạn sẽ là 0, sẽ tăng dần qua từng câu hỏi đúng.Có 3 mốc quan trọng trong trò chơi là 5/10/15, nếu bạn trả lời sai trước mốc quan trọng này thì trò chơi kết thúc và điểm của bạn là điểm của mốc trước đó
* Trò chơi kết thúc khi bạn trả lời sai hoặc khi bạn hoàn thành được 15 câu hỏi của chương trình. Bạn có chơi game mới hoặc thoát khỏi chương trình.
* Bạn luôn có thể xem lại luật chơi của chương trình trong khi chơi tại nút Help.

***2.2. Xây dựng chương trình***

***2.2.1. Xây dựng lớp đầu tiên của chương trình là Question:***

- Đối tượng quản lý nhỏ nhất của chương trình là từng đối tượng question bao gồm các thuộc tính: ques (câu hỏi) , answer (câu trả lời đúng) , choices (các đáp án của chương trình), score(điểm số của câu hỏi), playerAnswer ( câu trả lời của người chơi )

- Các phương thức của chương trình bao gồm các hàm getter và setter và các hàm tạo: getQues , setQues, getAnswer, setAnswer, getChoices, setChoices, setPlayerAnswer, setScore, getScore, toString.

- Bên cạnh đó là hàm check xem câu trả lời của người chơi đúng không.

***2.2.2. Xây dựng lớp QuestionSet là tập hợp bộ câu hỏi của game:***

-Lớp này cho phép trả về câu hỏi số num của chương trình, đáp án của câu hỏi số num, phương thức để show ra trò chơi.

- Đặc biệt là hàm xây dựng QuestionSet ( xây dựng bộ câu hỏi chương trình), xây dựng các câu hỏi từ 1-5, xây dựng các câu hỏi từ 6-10, xây dựng các câu hỏi từ 11-15: setQuestions(), setFirst(), setSecond(), setLast().

- Lớp này tạo các câu hỏi bằng cách đọc từ file, mỗi file sẽ chứa 15 câu hỏi thuộc từng khúc câu hỏi dễ, trung bình hay khó. Lúc này hàm random sẽ chọn ra 5 câu trong số đó làm câu hỏi cho trò chơi. Để đảm bảo cho việc các câu hỏi khác nhau thì ta tạo một mảng 15 phần tử có giá trị bằng và sau khi random ra phần tử nào rồi thì gán giá trị bằng không hoặc nếu đã xuất hiện sẽ thực hiện random lại

***2.2.3. Lớp Gameplay:***

- Lớp này chỉ có tác dụng kiểm tra xem phần chương trình trước của mình đã hoạt động chính xác như mong muốn chưa.

***2.2.4. Xây dựng giao diện trò chơi:***

- Trong mục này thì sẽ cài đặt các lớp để xây dựng được giao diện chương trình sử dụng Java swing bao gồm: GameJFrame, GameJPanel, GameMenu, Help2Btn, PlayGameUI, GameListener.

- Trong đó, GameJFrame để xây dựng khung chương trình chính; GameJPanel để xây dựng lớp chứa chính của chương trình; GameMenu để xây dựng thanh menu cho chương trình, Help2Btn để xây dựng lớp chứa phần sự trợ giúp thứ hai của chương trình, GameListener là để bắt sự kiện, How ToPlay để xây dựng lớp chứa phần luật chơi.

- Class JPanel còn chứa thêm 1 vài phương thức khác như:

* Phương thức setButton cho phép thay đổi text và màu của button sau mỗi câu hỏi
* Phương thức setAnswerButton cho sự trợ giúp 50:50. Khi sử dụng sự trợ giúp 50:50 sẽ gọi đến hàm này và bỏ đi 2 phương án sai, đồng thời thay đổi màu và setEnable trở thành false không cho người dùng click vào
* Phương thức setButtonCorrect để thay đổi màu cho sự lựa chọn của người chơi. Với mỗi button, xét xem nó có bị enable không ( khi sử dụng trợ giúp ), nếu không thì thay đổi màu thành vàng và chữ thì đen, các lựa chọn không phải của người chơi thì cho nó thành màu ban đầu.
* Phương thức end\_game dùng để gọi giao diện của phần end\_game hiển thị lên trên mainframe.
* Phương thức setNewGame dùng để bắt đầu chơi mới lại trên chính mainframe đó.

- Class GameListener có thêm phương thức setResultLbl để sử dụng cho sự trợ giúp hỏi ý kiến khán giả.

***2.2.4. File câu hỏi:***

- Cấu trúc file là câu hỏi, câu trả lời đúng, điểm câu hỏi đó, các câu trả lời của chương trình, các dữ liệu được ngăn cách với nhau bởi dấu ‘/’.

- Bao gồm 3 file tương ướng với 3 giai đoạn của trò chơi, đường dẫn file là đường dẫn tương đối.

***2.2.5.Một số lớp khác:***

- MyException để thêm các exception của chương trình

- Trong package model có class GamePlay là chương trình chơi trước khi xây dựng giao diện

**3) Phần tâm đắc nhất**

- Dưới đây sẽ là phần trình bày về cách hoạt động của các quyền trợ giúp

***3.1. Sự trợ giúp 50/50:***

- Khi bắt đầu game thì bộ câu hỏi sẽ được tạo ra. Khi người chơi chọn sử dụng sự trợ giúp 50/50 thì ta sẽ so sánh kết quả đúng của câu hỏi đó so sánh với các câu trả lời của chương trình. Từ đó ta có được 3 câu trả lời là sai, bây giờ ta sử dụng thư viện random trong java để xử lý.

* Gọi 2 số random ra là num1 và num2.Gọi các button chứa câu trả lời của chương trình lần lượt là 1,2,3,4. Giả sử button chứa câu trả lời đúng là x.
* Nếu num1 = num2 hoặc num1 = x hoặc năm 2 = x thì tiến hành random lại.
* Sau quá trình random thì ta đặt cho các button được random ra thành màu đỏ và đặt setEnable để không cho phép chọn nút đó nữa.
* Sau đó lưu lại biến boolean helpBtn1 là true ( đã được sử dụng ) và biến quesHelp1 để biết câu nào đã sử dụng sự trợ giúp này – điều này giúp cho thuật toán sự trợ giúp thứ 2 phía dưới.

***3.2. Sự trợ giúp hỏi ý kiến khán giả:***

- Trợ giúp này sẽ chia ra làm 2 trường hợp:

* Trường hợp 1:
* Trường hợp này xảy ra khi mà sự trợ giúp 50/50 chưa được sử dụng hoặc là ( sự trợ giúp 50/50 đã được sử dụng và câu sử dụng sự trợ giúp trên khác câu đang xét )
  + Gọi các button chứa câu trả lời của chương trình lần lượt là 1,2,3,4. Giả sử button chứa câu trả lời đúng là x.
  + Button x sẽ luôn có tỷ lệ xuất hiện ban đầu là 20.
  + Ta tiến hành random cho các button, ở đây tỷ lệ lớn nhất còn lại là 80% chia cho 4 button.Như vậy là tỷ lệ của button x sẽ thường là tỷ lệ lớn nhất trong 4 đáp án của chương trình. Tuy nhiên không phải lúc nào cũng đúng do có thể có trường hợp tỷ lệ của 1 button trở nên quá thấp khiến tỷ lệ của button sai cao hơn tỷ lệ của button x.
* Trường hợp 2:
* Trường hợp này xảy ra khi đã sử dụng sự trợ giúp câu 50/50 trên chính câu sử dụng sự trợ giúp này.
  + Khi này ta chỉ cần xác định câu trả lời trong 2 câu còn lại và thực hiện random tỷ lệ cho 2 button.
  + Button x trong trường hợp này sẽ có sẵn 40% tỷ lệ chính xác để đảm bảo khả năng đưa ra trợ giúp chính xác là cao

***4. Giao diện trò chơi:***

- Giao diện trò chơi tương đối đơn giản, khi vào game thì màn hình trò chơi sẽ hiện ra luôn



- Khi trò chơi kết thúc thì sẽ đưa ra màn hình kết thúc

A screenshot of a cell phone

Description automatically generated

- Dưới đây là phần hiển thị khi bạn chọn how to play để xem luật chơi của chương trình.

A screenshot of a cell phone

Description automatically generated

- Và phần hiển thị của sự trợ giúp hỏi ý kiến khán giả trong cả 2 trường hợp:



A screenshot of a cell phone

Description automatically generated