

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

Bộ môn: Công nghệ thông tin



BÀI TẬP LỚN

MÔN HỌC

LẬP TRÌNH GAME 3D VỚI UNITY

Sinh viên

: Triệu Văn Đông

Hoàng Thị Hường

Đặng Hoàng Nam

Phạm Như Thanh

Lớp

: K55KMT

Giáo viên hướng dẫn : ThS. Đỗ Duy Cốp

Thái Nguyên – 2023

BÀI TẬP LỚN

MÔN HỌC: LẬP TRÌNH GAME 3D VỚI UNITY

BỘ MÔN : CÔNG NGHỆ THÔNG TIN

Sinh viên: Triệu Văn Đông MSSV: K195480106004

Hoàng Thị Hương MSSV: K195480106013

Đặng Hoàng Nam MSSV: K195480106015

Phạm Như Thanh MSSV: K195480106021

Lớp: K55KMT.01

Ngành: Kỹ thuật máy tính

Ngày giao đề: 30/03/2023

Ngày hoàn thành: 05/06/2023

1. Tên đề tài : Phân tích thiết kế game RPG.....

.....
.....

2. Yêu cầu :

TỔ TRƯỞNG BỘ MÔN

(Ký và ghi rõ họ tên)

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

Thái Nguyên, ngày.....tháng.....năm 2023

GIÁO VIÊN HƯỚNG DẪN

(Ký ghi rõ họ tên)

NHẬN XÉT CỦA GIÁO VIÊN CHẤM

.....

.....

.....

.....

.....

Thái Nguyên, ngày.....tháng.....năm 2023

GIÁO VIÊN CHẤM

(Ký ghi rõ họ tên)

Mục lục

| | |
|--|-----------|
| LỜI MỞ ĐẦU | 5 |
| CHƯƠNG I. TỔNG QUAN VỀ ĐỀ TÀI..... | 6 |
| 1.1. RPG là gì ? | 6 |
| 1.1.1. Khái niệm | 6 |
| 1.1.2. Các yếu tố hình thành thể loại game RPG..... | 7 |
| 1.2. Mô tả trò chơi | 8 |
| 1.3. Yêu cầu đối với sản phẩm..... | 9 |
| 1.4. Đầu vào đầu ra của sản phẩm..... | 9 |
| CHƯƠNG II. CƠ SỞ LÝ THUYẾT..... | 10 |
| 2.1. Tổng quan về Unity | 10 |
| 2.1.1. Unity là gì ? | 10 |
| 2.1.2. Lịch sử của Unity..... | 10 |
| 2.1.3. Tính năng cơ bản của Unity | 11 |
| 2.1.4. Ưu điểm và nhược điểm của Unity | 12 |
| 2.1.5. Một số khái niệm cơ bản trong lập trình Unity..... | 13 |
| 2.2. Microsoft Visual Studio | 14 |
| 2.2.1. Visual Studio là gì ?..... | 14 |
| 2.2.2. Một số tính năng của Visual Studio..... | 15 |
| 2.2.3. Các phiên bản phổ biến của Visual Studio | 17 |
| CHƯƠNG III. XÂY DỰNG CHƯƠNG TRÌNH | 18 |
| 3.1. Giao diện | 18 |
| 3.1.1. Giao diện trò chơi..... | 18 |
| 3.1.2. Nhân vật | 19 |
| CHƯƠNG IV. KẾT LUẬN | 20 |
| 4.1. Nhận xét và kết luận | 20 |
| 4.2. Hướng phát triển của bài tập..... | 20 |
| TÀI LIỆU THAM KHẢO | 21 |

LỜI MỞ ĐẦU

Trong thời đại công nghệ ngày nay, lĩnh vực game đang trở thành một trong những ngành công nghiệp phát triển mạnh mẽ. Trò chơi điện tử không chỉ là một hình thức giải trí, mà còn là một nền tảng cho sự sáng tạo và khám phá. Trong bối cảnh đó, Game RPG từ lâu đã thu hút người chơi bởi sự đa dạng, tính chiến thuật và truyền cảm hứng phiêu lưu mạo hiểm. Chúng cho phép người chơi nhập vai vào một nhân vật, khám phá thế giới ảo và trải nghiệm các cuộc phiêu lưu đầy thách thức. Bằng cách sử dụng công cụ Unity mạnh mẽ, chúng ta sẽ có cơ hội tạo ra một trò chơi đẹp mắt và thú vị, nơi người chơi có thể truyền cảm hứng và khám phá những điều tuyệt vời.

Trong phạm vi bài báo cáo này chúng em xin được trình bày quá trình thực hiện bài tập “Phân tích thiết kế game RPG” từ các yếu tố cốt truyện đến cách xây dựng hệ thống gameplay và các phần tử hình ảnh và âm thanh. Cuối cùng, mặc dù đã cố gắng rất nhiều nhưng do thời gian có hạn, khả năng dịch và hiểu tài liệu chưa tốt nên nội dung bài tập này không thể tránh khỏi những thiếu sót, rất mong được sự chỉ bảo, góp ý của các thầy cô và các bạn.

CHƯƠNG I. TỔNG QUAN VỀ ĐỀ TÀI

1.1. RPG là gì ?

1.1.1. Khái niệm

RPG là viết tắt của cụm từ **Role Playing Game** và nó được gọi là thể loại **game nhập vai**. Hiểu một cách đơn giản nhất thì bạn sẽ hóa thân thành nhân vật chính trong một tựa game RPG nào đó và đi theo sự dẫn dắt của cốt truyện mà game đưa ra để làm các nhiệm vụ - nhận được các món trang bị với chỉ số khủng để nâng cấp nhân vật thật mạnh - mở khóa các kĩ năng hoặc tuyệt chiêu gây sát thương lớn v.v...

Chứng kiến sự phát triển mạnh mẽ của internet và công nghệ, việc chuyển sang đồ họa 3D cùng với nhiều cải tiến về chất lượng đồ họa, đã khiến những trò chơi điện tử trở nên chân thực và chi tiết hơn bao giờ hết. Ngày nay khi công nghệ được phát triển một cách vượt bậc, RPG là thể loại mà các nhà phát triển game AAA hay các nhà phát triển Game Indie muốn hướng đến.



Hình 1.1. Game nhập vai (Role Playing Game)

1.1.2. Các yếu tố hình thành thể loại game RPG

- **Cốt truyện và bối cảnh game:** Các game RPG thường sở hữu cho mình một cốt truyện có chiều sâu và phức tạp, cần người chơi phải liên tục tìm hiểu những manh mối xuyên suốt quá trình chơi. Bối cảnh của game ảnh hưởng trực tiếp đến mạch truyện và lối kể chuyện trong game.
- **Hệ thống nhiệm vụ:** Game sẽ cung cấp cho bạn một chuỗi các nhiệm vụ tuyến tính cần phải thực hiện để hoàn thành trò chơi, tuy nhiên một số game RPG lại sở hữu cốt truyện bán tuyến tính giúp người chơi cảm thấy mình có ảnh hưởng hơn trong cái kết của game. Các nhiệm vụ phụ có thể được thêm vào để giúp người chơi giải đáp thêm một số bí ẩn, khai thác nhiều hơn những khía cạnh khác của game.
- **Khám phá:** Thiết kế bản đồ rộng lớn chính là điểm mạnh của các dòng game RPG nói chung và MMORPG nói riêng. Việc thiết kế bản đồ như vậy khiến người chơi có thêm thời gian để đi khám phá các vùng đất khác nhau, trò chuyện với các NPC và thu thập thêm nhiều trang bị mới.
- **Hệ thống vật phẩm trang bị:** Trang bị là thứ thiết yếu không thể thiếu trong các game RPG, những trang bị như giáp, quần áo, vũ khí cần được người chơi thu thập trong quá trình chơi để tăng sức mạnh cho nhân vật. Vật phẩm hay trang bị cũng có thể bán đi nếu bạn đã sở hữu món đồ mạnh hơn, dùng để quy đổi ra các đơn vị tiền tệ hoặc trao đổi các sản phẩm mới.
- **Sự lựa chọn:** Trong quá trình chơi bạn sẽ phải trải qua rất nhiều sự lựa chọn khác nhau, quyết định của bạn sẽ ảnh hưởng đến lối phát triển nhân vật mà bạn đang chơi. Các kỹ năng, thuộc tính, trang bị sẽ thay đổi dựa trên các lựa chọn này của người chơi.
- **Kinh nghiệm và cấp độ:** Đa phần các game RPG sẽ có phong cách tiêu diệt kẻ địch trên đường để tăng kinh nghiệm và tăng cấp khi tích lũy đủ điểm kinh nghiệm. Sự tăng cấp này giúp nhân vật có thêm kỹ năng hoặc chỉ số cần thiết, giúp tăng trải nghiệm của người chơi.
- **Hệ thống chiến đấu:** Chiến đấu trong RPG rất đa dạng, có thể là phong cách chiến đấu thời gian thực hoặc chiến đấu theo lượt. Nhưng điểm chung là hệ thống chiến đấu của các game RPG rất hấp dẫn người chơi vì tính chiến thuật cao và rất đã mắt.
- **Hệ thống nhân vật và NPC:** Các nhân vật trong game rất đa dạng cho người chơi lựa chọn, sự phát triển của nhân vật là sẽ do bạn lựa chọn để

định hình tính cách, chỉ số và các yếu tố khác của nhân vật. NPC trong game đóng vai trò là cầu nối cho người chơi đến với các nút thắt trong cốt truyện chính, hoặc cũng có thể là nhân vật cung cấp cho người chơi một sức mạnh đặc biệt, một nhiệm vụ hoặc giúp đỡ người chơi trong game.



Hình 1.2. Hệ thống vật phẩm trang bị trong game RPG

1.2. Mô tả trò chơi

Trò chơi được làm theo thể loại game nhập vai vượt hầm ngục. Sau khi bấm nút vào chơi game, người chơi được đưa đến một hầm ngục khởi đầu. Ở đây người chơi sẽ học cách thao tác cơ bản của game như di chuyển, tấn công quái vật, phá những hộp gỗ,... Sau khi hoàn thành màn hướng dẫn, người chơi đi qua cánh cửa để đến hầm ngục chính. Tại đây người chơi sẽ lần lượt vượt qua các hầm ngục với nhiều loại quái vật và bẫy, thu thập các vật phẩm như vũ khí, bình thuốc,..v.v.. Mỗi khi tiêu diệt được quái vật, người chơi sẽ nhận được một ít vàng dùng để nâng cấp nhân vật và vũ khí.

Khi bị dính đòn của quái vật, người chơi sẽ mất máu. Khi lượng máu này trở về 0, người chơi sẽ chết hay GameOver. Sau khi chết người chơi sẽ bị đưa về nơi save game. Khi người chơi bấm nút Menu dưới thanh máu, một cửa sổ hiện lên giúp ta xem được chỉ số nhân vật như lượng máu, sát thương gây ra, kinh nghiệm để lên cấp, vũ khí hiện đang sử dụng,... Người chơi có thể bỏ ra một lượng vàng để nâng cấp vũ khí, khiến sát thương gây ra tăng.

1.3. Yêu cầu đối với sản phẩm

- Game phải có dung lượng không quá lớn, tốc độ xử lý nhanh.
- Giao diện game dễ nhìn, thân thiện với người sử dụng.
- Công việc tính toán (va chạm, tấn công, điểm số, ...) phải thực hiện chính xác, không có sai sót.
- Đồ hoạ, âm thanh chất lượng

1.4. Đầu vào đầu ra của sản phẩm

- **Đầu vào (inputs):** Các thao tác từ người chơi như bấm chuột, gõ bàn phím.
- **Đầu ra (outputs):** Hình ảnh trò chơi được hiển thị lên màn hình, âm thanh trò chơi được phát qua loa.

CHƯƠNG II. CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về Unity

2.1.1. Unity là gì ?

Unity là một game engine đa nền tảng được phát triển bởi công ty Unity Technologies, chủ yếu được dùng để phát triển video game cho 21 nền tảng bao gồm máy tính, game consoles (như PlayStation) và điện thoại,...

Hơn 50% số lượng game trên thị trường được sản xuất bởi Unity. Một vài tựa game vô cùng nổi tiếng được tạo ra bởi Unity có thể được kể đến như Pokémon Go, Hearthstone, Ori And The Blind Forest, Monument Valley, Axie Infinity,... Độ “phủ sóng” của Unity rất rộng, có thể được áp dụng phổ biến trong nhiều dòng game khác nhau từ game “hạng nặng” Triple A (AAA) cho đến game giáo dục đơn giản.



Hình 2.1. Unity

2.1.2. Lịch sử của Unity

Ngày nay, con người dành khá nhiều thời gian giải trí bên những chiếc smartphone cùng những tựa game yêu thích. Trong số đó có không ít trò chơi được lập trình dựa trên engine Unity 3D đã ra đời cách đây hơn một thập kỉ. Trải qua thời gian phát triển lâu dài và luôn update công nghệ mới, giờ đây Unity 3D đã trở thành lựa chọn số 1 cho bất cứ lập trình viên nào muốn xây dựng một tựa

game có thể sử dụng đa nền tảng, chi phí rẻ và dễ thao tác. Tuy rất phổ biến những thực tế ít ai biết được nguồn gốc và lịch sử phát triển của engine.

Unity được phát triển bởi công ty Unity Technologies, thành lập bởi David Helgason, Joachim Ante và Nicholas Francis tại Copenhagen, Đan Mạch vào năm 2004. Ban đầu, Unity được thiết kế để phát triển game trên nền tảng Mac OS X, nhưng sau đó đã được mở rộng để hỗ trợ Windows và nền tảng di động.

Phiên bản đầu tiên của Unity, gọi là Unity 1.0, được ra mắt vào tháng 6 năm 2005. Nó chỉ hỗ trợ phát triển game 3D trên máy tính Mac OS X và được sử dụng chủ yếu cho các trò chơi trên web.

Sau đó, Unity đã phát triển và mở rộng để hỗ trợ nhiều nền tảng hơn, bao gồm Windows, iOS, Android và các nền tảng game console như Xbox và PlayStation. Unity cũng được mở rộng để hỗ trợ phát triển game 2D, AR/VR và các ứng dụng không phải game.

Năm 2012, Unity công bố phiên bản miễn phí của nó, Unity Free, để thu hút các nhà phát triển game indie và người mới bắt đầu trong lĩnh vực phát triển game. Phiên bản miễn phí đã giúp Unity trở thành một trong những engine game phổ biến nhất hiện nay.

2.1.3. Tính năng cơ bản của Unity

- Tạo giao diện UI của Game như tạo drop bar, textbox,...
- Hiển thị mô hình 3D, 2D – hệ thống vật lý 2D, 3D
- Networking: Hỗ trợ tạo game chơi nhiều người cùng lúc
- Hỗ trợ nền tảng đặc biệt mới: AR – Augmented reality (Thực tế tăng cường), VR – Virtual reality (Thực tế ảo)
- AI – Hỗ trợ con bot trong màn hình game, hỗ trợ package tạo nên bot trong game
- Hỗ trợ hiển thị chữ bằng font đặc biệt

2.1.4. Ưu điểm và nhược điểm của Unity

Ưu điểm:

- **Editor:** Với Editor, nhà phát triển không cần thiết phải viết Code để sắp đặt các đối tượng trong Game như những Engine khác mà Developer có thể kéo thả, thay đổi vị trí của từng đối tượng trong Game trực tiếp trên Editor.
- **Đa nền tảng:** Việc bạn tạo ra Game mà Game đó có thể chạy được trên hầu hết những hệ điều hành quan trọng như Desktop (Mac, Window và Linux) hay Mobile (iOS, Android) hoặc Web (WebGL) thì cũng đã tiết kiệm công sức cũng như chi phí rất nhiều cho doanh nghiệp.
- **Miễn phí:** Và yếu tố cuối cùng chính là chi phí. Với Unity, miễn phí là một điểm thu hút rất nhiều Developer chọn làm việc với game engine này. Tuy nhiên, với các game được tạo ra miễn phí thì bắt buộc phải có Logo Unity trong Game.

Nhược điểm:

- **Dung lượng Unity game bundle khá lớn:** So với những game engine khác, Unity sản xuất game có dung lượng nặng nên đây là một điểm trừ lớn. Thậm chí, game web do Unity sản xuất có thể có dung lượng lên đến cả trăm MB nên web chạy không nổi. Chính vì thế, cũng cùng một game đó thì game mobile lại chạy tốt trong khi game web lại giật, lag.
- **Các phiên bản cập nhật:** Unity cho ra mắt nhiều phiên bản cập nhật liên tục trong một năm và nhiều năm liên tục. Nếu phiên bản cập nhật được ra mắt trong cùng một năm thì vẫn có thể sử dụng song song nhiều phiên bản cùng năm. Còn nếu trong trường hợp bản cập nhật khác năm, và trong nội bộ một team, hoặc giữa team phát triển và khách hàng, sử dụng các phiên bản Unity khác năm với nhau thì khi push code, đẩy code lên cho member thì sẽ bị lỗi hình ảnh, script, code,... thậm chí là mất hết tiến độ. Tuy nhiên, điều khó là nếu như

bạn nhận nhiều dự án cùng một lúc mà mỗi khách hàng lại sử dụng một phiên bản Unity khác nhau để phát triển game thì bạn bắt buộc phải thay đổi qua lại.

2.1.5. Một số khái niệm cơ bản trong lập trình Unity

- **GameObject:** Một trong những đối tượng cụ thể được sử dụng trong game là khái niệm của game object. Có thể là nhân vật, đồ vật,...
- **Component:** Một GameObject thường sẽ được cấu tạo từ nhiều thành phần nên nó có thể là các hình ảnh, những hành động của nhân vật, mã điều khiển,... Functions thường kế thừa từ MonoBehaviour class và có thể ghi đè bên trong những class còn có thể thực hiện cho những sự kiện quan trọng nào đó. Hai sự kiện này thường được dùng phổ biến trong một Component đó là: **Start()** tiến hành chạy thêm 1 lần duy nhất trước khi hàm update. **Update()** là sẽ thực hiện sau một vòng lặp chạy liên tục. Nó sẽ được gọi một lần dành cho mỗi khung hình (thường là 25 khung hình mỗi giây). Cứ mỗi thứ như vậy gọi sẽ được gọi là một component của GameObject.
- **Script:** Script là dạng tập tin có chứa những đoạn mã nguồn được sử dụng với mục đích khởi tạo cũng như xử lý đối tượng trong game. Đối với Unity thì bạn có thể sử dụng C#, Java Script, BOO để thực hiện lập trình Script. Ngoài ra, bạn cũng có thể điều chỉnh thuộc tính cho component khác thông qua script. Và script thường được xem như là “não” của con người bởi nó có thể điều khiển được những bộ phận còn lại.



Hình 2.2. Giao diện Unity Editor

2.2. Microsoft Visual Studio

2.2.1. Visual Studio là gì ?

Microsoft Visual Studio là một môi trường phát triển tích hợp (IDE) từ Microsoft. Microsoft Visual Studio còn được gọi là "Trình soạn thảo mã nhiều người sử dụng nhất thế giới", được dùng để lập trình C++ và C# là chính. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.



Hình 2.3. Microsoft Visual Studio

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C, C++ và C++/CLI, VB.NET, C# và F#. Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS.

2.2.2. Một số tính năng của Visual Studio

- **Hỗ trợ nhiều ngôn ngữ lập trình:** Tính năng này giúp phát hiện bất kỳ lỗi hoặc tham chiếu ngôn ngữ chéo (cross-language reference) nào một cách dễ dàng.
- **Intelli-Sense:** Là một tính năng giúp phát hiện có bất kỳ đoạn code nào bị bỏ sót hay không, tự động thực thi cú pháp biến (variable syntaxes) và khai báo biến (variable declarations). Ví dụ: Nếu một biến nào đó đang được sử dụng trong chương trình và người dùng quên khai báo, intellisense sẽ khai báo biến đó cho người dùng.
- **Hỗ trợ đa nền tảng:** Visual Studio hoạt động trên cả 3 nền tảng Windows, Linux, Mac.

- **Tiện ích mở rộng và Hỗ trợ:** Các extension hay tiện ích mở rộng của Visual Studio giúp tăng tốc quá trình phát triển ứng dụng mà không ảnh hưởng đến hiệu suất của editor.
- **Repository:** Visual Studio được kết nối với Git hoặc có thể được kết nối với bất kỳ Repository nào khác.
- **Code editor:** Visual Studio có class Code editor tốt nhất hỗ trợ nhiều chức năng. Nó cho phép bookmark trong code để kết hợp Quick Navigation. Visual Studio cũng có chức năng Incremental Search, Regex Search, Multi-item Clipboard và Task-list.
- **Web-Support:** Các ứng dụng web có thể được xây dựng và hỗ trợ trong Visual Studio.
- **Hỗ trợ Terminal:** Visual Studio hỗ trợ Terminal hoặc Console tích hợp giúp người dùng không cần chuyển đổi giữa hai màn hình.
- **Hỗ trợ Git:** Tài nguyên có thể được lấy từ Github Repo trực tuyến và ngược lại giúp tiết kiệm thời gian và công sức.
- **Debugger:** Đây là một tính năng hữu ích cho phép nhà phát triển kiểm tra trạng thái của chương trình và phát hiện bug ở đâu. Bạn cũng có thể xem source code chương trình của mình bằng cách sử dụng các công cụ gỡ lỗi của debugger.
- **Windows Forms Designer:** tạo các ứng dụng GUI bằng Windows Forms.
- **Class designer:** Class designer cho phép chỉnh sửa các class bao gồm các thành viên và quyền truy cập của chúng bằng cách sử dụng mô hình UML.
- **Web designer:** Visual Studio cũng hỗ trợ một trình soạn thảo và thiết kế trang web, cho phép tạo các trang web bằng cách kéo và thả các widget.
- **Mapping Designer:** Mapping Designer được LINQ to SQL sử dụng để thiết kế mapping giữa các lược đồ thông tin và từ đó các class sẽ đóng gói dữ liệu.
- **Data Designer:** Data Designer được sử dụng để chỉnh sửa các lược đồ thông tin, cũng như các bảng được viết, khóa chính và khóa ngoại và các ràng buộc (constraint).

2.2.3. Các phiên bản phổ biến của Visual Studio

- **Visual Studio Community:** Đây là phiên bản miễn phí có đầy đủ tính năng, có thể mở rộng. Phiên bản dành cho sinh viên, nhà phát triển nguồn mở và cá nhân, để tạo các ứng dụng hiện đại cho Android, IOS, Windows, cũng như các ứng dụng web và dịch vụ đám mây. Mục đích chính của nó là cung cấp hỗ trợ Ecosystem (hàng nghìn tiện ích mở rộng) và Language (có thể lập trình bằng C#, C++, HTML, JavaScript, Python, v.v.).
- **Visual Studio Professional:** Đây là phiên bản thương mại của Visual Studio, hỗ trợ XML và XSLT editing và cả công cụ như Server Explorer, tích hợp với Microsoft SQL Server. Với phiên bản này, người dùng được dùng thử miễn phí và sau đó cần trả phí để tiếp tục sử dụng. Mục đích chính của phiên bản này là cung cấp các công cụ dành cho nhà phát triển chuyên nghiệp để xây dựng bất kỳ loại ứng dụng nào, các tính năng mạnh mẽ như CodeLens cải thiện năng suất của team, các công cụ lập kế hoạch dự án (Agile, biểu đồ, ...) và các lợi ích của Subscriber như phần mềm Microsoft, cùng với Azure, Pluralsight, v.v.
- **Visual Studio Enterprise:** Đây là một giải pháp tích hợp end-to-end cho các team thuộc bất kỳ quy mô nào với nhu cầu mở rộng cũng như yêu cầu chất lượng khắt khe. Với phiên bản này, người dùng được dùng thử miễn phí 90 ngày và sau đó cần trả phí để tiếp tục sử dụng. Lợi ích chính của phiên bản này là có khả năng mở rộng tốt và cung cấp phần mềm chất lượng cao.

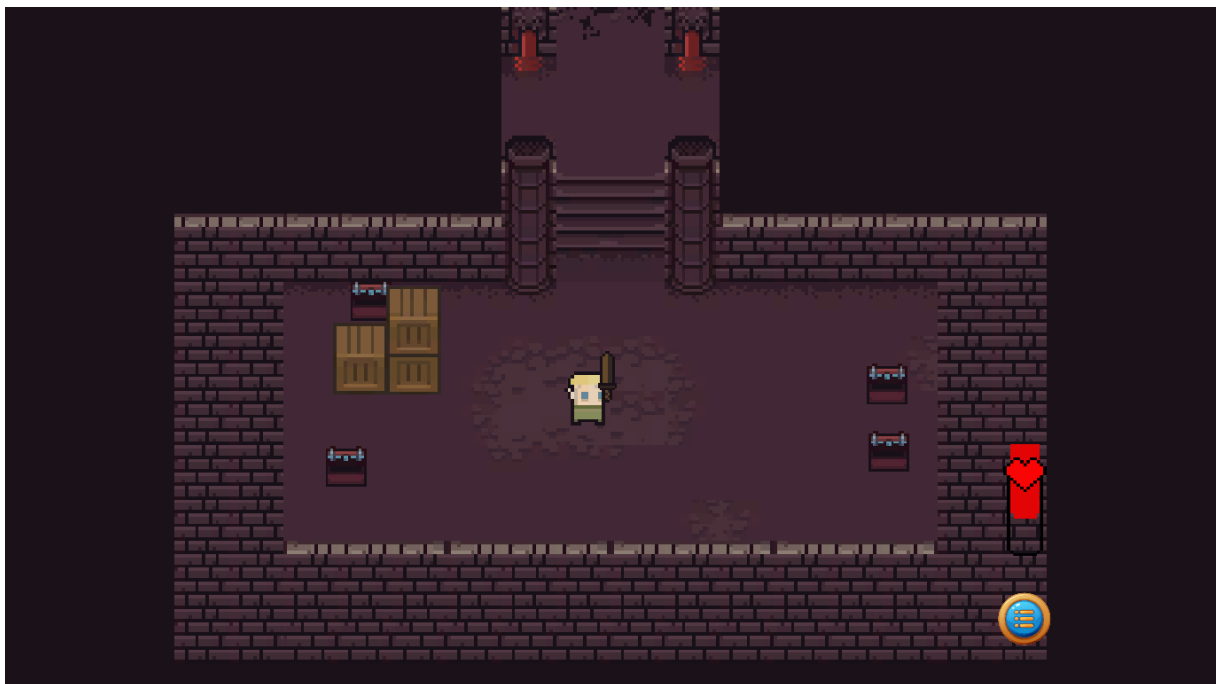
CHƯƠNG III. XÂY DỰNG CHƯƠNG TRÌNH

3.1. Giao diện

3.1.1. Giao diện trò chơi



Hình 3.1. Giao diện Menu trò chơi



Hình 3.2. Giao diện Màn 1 của trò chơi



Hình 3.3. Giao diện Màn 2 của trò chơi

3.1.2. Nhân vật



Hình 3.4. Nhân vật

CHƯƠNG IV. KẾT LUẬN

4.1. Nhận xét và kết luận

Như vậy, việc phát triển “**Game RPG**” là một hành trình đầy thách thức và đồng thời mang lại những trải nghiệm sáng tạo và thú vị. Qua quá trình phân tích thiết kế, triển khai và tối ưu hóa, chúng em đã tạo ra một trò chơi đầy màu sắc và hấp dẫn, nơi người chơi có thể nhập vai vào một thế giới tưởng tượng, khám phá những địa điểm kỳ bí, chiến đấu với quái vật và hoàn thành các nhiệm vụ thú vị. Trong quá trình xây dựng và thiết kế thì sau đây là một số những ưu nhược điểm của đề tài:

- **Ưu điểm:** Hệ thống đã hoàn thiện các bước cơ bản như phân tích thiết kế, cài đặt kiểm thử, bước đầu tạo nên một trò chơi hoàn chỉnh. Trò chơi đáp ứng được những chức năng cơ bản, có dung lượng nhẹ, các lỗi của trò chơi hầu hết đã được xử lý.
- **Nhược điểm:** Việc phân tích vẫn còn chưa được trực quan, hình ảnh trong trò chơi vẫn chưa được mượt mà. Các chức năng của trò chơi cần được hoàn thiện thêm.

4.2. Hướng phát triển của bài tập

Mở rộng nội dung: Xây dựng thêm các khu vực mới, nhiệm vụ phụ, và câu chuyện phong phú để người chơi có thể khám phá và trải nghiệm thêm nhiều điểm đặc biệt trong trò chơi.

Nâng cấp hệ thống chiến đấu: Cải thiện hệ thống chiến đấu bằng cách thêm các kỹ năng mới, vũ khí và trang bị.

Cải thiện hệ thống tiến cấp nhân vật: Xây dựng một hệ thống tiến cấp phong phú và đa dạng, cho phép người chơi nâng cấp kỹ năng, thu thập trang bị và tăng cường khả năng của nhân vật.

TÀI LIỆU THAM KHẢO

<https://itnavi.com.vn/blog/lap-trinh-unity>

<https://itviec.com/blog/unity-la-gi/>

<https://codelearn.io/sharing/lap-trinh-game-voi-unity-nhu-the-nao-phan-1>

https://www.youtube.com/watch?v=b8YUfee_pzc&list=RDCMUCsaXQNLxeHvwJdDUrICGufA&start_radio=1

<https://visualstudio.microsoft.com/fr/>

<https://viettelstore.vn/tin-tuc/game-nhap-vai-rpg-hay-cho-pc>

<https://github.com/ChancePenner/Happy-Village-Scary-Dungeons>