

MỤC LỤC

MỤC LỤC	1
DANH MỤC HÌNH	3
DANH MỤC BẢNG.....	4
TÓM TẮT.....	5
ABSTRACT.....	6
LỜI CẢM ƠN.....	7
DANH MỤC TỪ VIẾT TẮT	8
TỔNG QUAN VỀ UNITY ENGINE.....	9
1. TỔNG QUAN VỀ UNITY.....	9
1.1 Unity là gì ?	9
1.2 Sơ lược về sự hình thành và phát triển của Unity.....	10
1.3 Tính năng của Unity Engine	11
2. TÌM HIỂU THÊM VỀ UNITY ENGINE	13
2.1 Hướng dẫn sử dụng và cài đặt Unity-Engine	13
2.2 Các thành phần trong Unity Editor	16
PHẦN GIỚI THIỆU	18
1. ĐẶT VẤN ĐỀ	18
2. TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ.....	19
3. MỤC TIÊU ĐỀ TÀI.....	19
4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU.....	20
5. NỘI DUNG NGHIÊN CỨU	20
6. NHỮNG ĐÓNG GÓP CHÍNH CỦA ĐỀ TÀI.....	21
PHẦN NỘI DUNG.....	22
CHƯƠNG 1 – MÔ TẢ BÀI TOÁN	22
1. MÔ TẢ CHI TIẾT BÀI TOÁN.....	22
2. PHÂN TÍCH ĐÁNH GIÁ CÁC GIẢI PHÁP CÓ LIÊN QUAN ĐẾN BÀI TOÁN	23
3. TIẾP CẬN GIẢI QUYẾT VẤN ĐỀ	24
4. TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT GIẢI QUYẾT VẤN ĐỀ	25
CHƯƠNG 2 – THIẾT KẾ GIẢI PHÁP.....	26
1. THIẾT KẾ KIẾN TRÚC.....	26

2. THIẾT KẾ DỮ LIỆU	28
3. THIẾT KẾ GIAO DIỆN.....	29
4. THIẾT KẾ THUẬT TOÁN.....	34
4.1 Tạo Background lặp lại:	35
4.2 Tạo kẻ địch:	35
4.3 Kiểm tra va chạm:.....	36
4.4 Giải thuật A*, dùng để tạo AI vạch đường cho kẻ địch tấn công người chơi:	37
CHƯƠNG 3 – CÀI ĐẶT GIẢI PHÁP	38
1. CÀI ĐẶT KIẾN TRÚC	38
2. CÀI ĐẶT ĐỐI TƯỢNG VÀ GIAO DIỆN TRONG GAME	38
3. HÌNH ẢNH VÀ CHUYỂN ĐỘNG CỦA NHÂN VẬT	41
4. ÂM THANH CHO GAME	41
CHƯƠNG 4 – KIỂM THỬ VÀ ĐÁNH GIÁ.....	43
1. MỤC TIÊU KIỂM THỬ	43
2. KỊCH BẢN KIỂM THỬ.....	43
3. KẾT QUẢ KIỂM THỬ	44
PHẦN KẾT LUẬN	45
1. KẾT QUẢ ĐẠT ĐƯỢC	45
2. HẠN CHẾ	45
3. HƯỚNG PHÁT TRIỂN.....	46
TÀI LIỆU THAM KHẢO.....	47

DANH MỤC HÌNH

Hình 0.1: Hình minh họa đa nền	9
Hình 0.2: Các phiên bản của Unity	13
Hình 0.3: Giao diện sử dụng Unity sau khi cài đặt xong	13
Hình 0.4: Giao diện chính khi sử dụng Unity	14
Hình 0.5: Giao diện khi cần build project	15
Hình 0.6: Hình minh họa các thành phần trong Unity	16
Hình 1.1: Sơ đồ GameFlow cơ bản	24
Hình 2.1: Hệ thống phân cấp project cơ bản	26
Hình 2.2: Sơ đồ phân cấp các chức năng của phần mềm	27
Hình 2.3: Sơ đồ phân cấp dữ liệu dùng trong game	28
Hình 2.4: Thiết kế giao diện chính của trò chơi	29
Hình 2.5: Giao diện chính khi vào Game và kèm theo hướng dẫn	30
Hình 2.6: Giao diện khi người chơi ấn phím P	31
Hình 2.7: Giao diện khi người chơi ấn phím U	32
Hình 2.8: Giao diện khi người chơi thua cuộc	33
Hình 2.9: Giao diện khi người chơi chiến thắng	33
Hình 2.10: Thuật toán tạo background lặp lại cho màn chơi	35
Hình 2.11: Thuật toán tạo kẻ địch cho từng lượt tấn công	35
Hình 2.12: Thuật toán kiểm tra va chạm cho nhân vật cũng như của kẻ địch	36
Hình 2.13: Thuật toán tạo background lặp lại cho màn chơi	37
Hình 3.1: Loạt hình ảnh nhân vật duy chuyển được tạo ra thông qua công cụ Animator của Unity ..	41
Hình 3.2: Loạt hình ảnh nhân vật duy chuyển được tạo ra thông qua công cụ Animator của Unity ..	42

DANH MỤC BẢNG

Bảng 2.1: Bảng diễn giải các dữ liệu dùng trong Game.....	28
Bảng 2.2: Bảng danh sách tên các thuật toán trong trò chơi	34
Bảng 3.1: Danh sách các đối tượng trong game.	39
Bảng 4.1: Kết quả kiểm thử theo từng chức năng.	44

TÓM TẮT

Trong thời đại công nghệ thông tin như hiện nay, sản phẩm công nghệ ngày nay càng lúc càng phải chịu sự khắt khe hơn từ phía người dùng. Đặc biệt là về sản phẩm Game được nhận rất nhiều sự đánh giá từ phía các người dùng là “Game thủ”, hay những người chơi thông thường. Ngành công nghiệp Game hiện nay có thể nói là bùng nổ, với tốc độ chóng mặt, đã có rất nhiều game hay và hấp dẫn đã được ra đời trong thời gian qua. Phía sau những Game phát triển và nổi tiếng thì đều có một game engine. Game engine là một công cụ hỗ trợ, một Middleware giúp người phát triển viết Game một cách nhanh chóng và đơn giản, đồng thời cung cấp khả năng tái sử dụng các tài nguyên và mã nguồn cao, do có thể phát triển nhiều game từ một game engine.

Từ xu hướng phát triển và những bất cập trên, đề án này sẽ khảo sát và nghiên cứu về Unity Engine – Một Engine rất phổ biến và không kém mạnh mẽ như hiện nay nhằm thực hiện một trò chơi (Demo) “Sinh tồn ở hành tinh mới” bằng Unity 2D. Chuẩn bị cho định hướng (phát triển game) chuyên nghiệp sau này, góp phần cho việc phát triển game của Việt Nam.

ABSTRACT

In our world with the current technology, so many products are produced and when it's came to the user, that product will suffer the harsh uses from the user. Especially is Game, this product will get many feedback from the users and that users are called "Gamer". With that issue, the game industry is in the explode time and so many famous games in the world are born. Behind a famous game is always has a Game-Engine. Game-Engine is a tool to help the developer to make a game with fully support, to make game faster and easier, so that the developer can make more than one game from a Game-Engine.

From the issues above, this project will survey and research about Unity-Engine, one of most popular Game-Engine in the world to make a game for this project, this game is called "Survive in the new planet" with Unity 2D. This project will help and prepare for the author of this project come to the professional game career in the future, to make Viet Nam game more good and greater.

LỜI CẢM ƠN

Đầu tiên, em xin chân gửi lời cảm ơn và sự tri ân sâu sắc đến thầy cô trường Đại học Cần Thơ. Đặc biệt là các thầy cô khoa Công Nghệ Thông Tin và Truyền Thông đã tạo điều kiện cho em học tập rèn luyện và hoàn thành khóa học, cũng như luận văn tốt nghiệp này.

Bên cạnh đó, em cũng xin gửi lời cảm ơn chân thành đến thầy Trần Công An đã nhiệt tình hướng dẫn, chỉ bảo những điều thiếu sót, giúp em hoàn thành luận văn tốt nghiệp trong thời gian qua.

Đồng thời em xin cảm ơn tất cả bạn bè người thân đã động viên, tạo điều kiện cho em hoàn thành tốt luận văn này.

Trong thời gian học tập, cũng như trong quá trình làm luận văn, mặc dù đã cố gắng hết sức nhưng có thể vẫn còn sai sót, vì vậy rất mong nhận được sự đóng góp ý kiến quý báu của quý Thầy Cô và các bạn.

em xin chân thành cảm ơn.

Cần Thơ ngày 30 tháng 11 năm 2018

Sinh viên

Trương Hoàng Vương

DANH MỤC TỪ VIẾT TẮT

PC: Máy tính cá nhân (Personal Computer).

s: giây (second).

GB: gigabyte.

RAM: Random Access Memory.

VGA: Video Graphics Video.

IOS: (Iphone Operating System).

CPU: Computer Processing Unit.

GHz: Tần suất hoạt động.

TỔNG QUAN VỀ UNITY ENGINE

1. TỔNG QUAN VỀ UNITY

1.1 Unity là gì ?

Đã qua trời thời làm game trên Flash căn bản và buồn chán với những chuyển động thật cứng nhắc. Unity mang lại sức mạnh kỳ diệu cho nhân vật mà chúng ta muốn thể hiện sống động hơn trong không gian 3 chiều đầy huyền ảo. Công nghệ cao này tạo ra một bước đột phá mới về sự khác biệt trong công nghệ làm game hiện nay, mang đến cho người chơi một cảm giác rất khác lạ và hào hứng trong từng chuyển động, tương lai công nghệ này được áp dụng vào game Việt Nam sẽ mở ra một trang mới trong thế giới game 2D, 3D huyền ảo.

Unity là một trong những game engine được giới làm game không chuyên cực kỳ ưa chuộng bởi khả năng tuyệt vời của nó là phát triển trò chơi đa nền. Trình biên tập có thể chạy trên Windows và Mac OS và có thể xuất ra game cho Windows, Mac, Wii, iOS, Android. Game cũng có thể chơi trên trình duyệt web thông qua plugin Unity Web Player. Unity gần đây đã bỏ túc khả năng xuất ra game trên widget cho Mac và cả Xbox 360, PlayStation 3 (4).

Unity được dùng để làm video game, hoặc những nội dung có tính tương tác như thể hiện kiến trúc, hoạt hình 2D, 3D thời gian thực. Unity hao hao với Director, Blender game engine, Virtools hay Torque Game Builder trong khía cạnh dùng môi trường đồ họa tích hợp ở quá trình phát triển game là chính.



Hình 0.1: Hình minh họa đa nền

Chỉ với khoản tiền bỏ ra khá khiêm tốn (1.500 USD) là phiên bản pro đã nằm trong tay chúng ta, dĩ nhiên tại Việt Nam số tiền này vẫn là quá lớn, nhưng thật may là đã có phiên bản Unity Free. Tuy nhiên, nhiều tính năng quan trọng (Network) bị cắt giảm nhưng đó không phải là vấn đề quá lớn nếu muốn phát triển một tựa game tầm trung.

Vào năm 2009, Unity nằm trong top 5 game engine tốt nhất cho việc sản xuất game với chỉ sau 4 năm phát triển, Unity đứng thứ 4, xếp sau Unreal Engine 4, Gamebryo Engine (được VC Studio mua về phát triển SQUAD) và Cry Engine 2. Lượng tài liệu hướng dẫn Unity rất phong phú. Hơn thế nữa nó còn có sẵn một cộng đồng cực lớn với diễn đàn riêng. Bất cứ điều gì không hiểu chúng ta đều có thể thoải mái hỏi và nhận được câu trả lời nhanh chóng, tận tâm.

Quá trình tạo địa hình cũng như truy xuất các phần mềm 3DSMax, Maya, Cinema4D ... rất nhanh chóng. Sức mạnh và sự tiện lợi của Unity là vô cùng lớn.

- **Sức mạnh:** Unity có thể tạo ra được nhiều loại game 2D, 3D đa dạng, dễ sử dụng với người làm game chưa chuyên nghiệp, chất lượng cao, chạy hầu hết trên các hệ điều hành.
- **Sự tiện lợi:** Nếu chúng ta là một người chuyên dùng 3Dmax, hay Maya hoặc phần mềm mã nguồn mở Blender thì quả là thật tuyệt, chúng ta sẽ có một lợi thế lớn khi viết game trên Unity, bởi công việc tạo các mô hình 2D, 3D sẽ trở nên dễ dàng hơn rất nhiều, việc kết hợp giữa người lập trình và người thiết kế các mô hình sẽ nhanh và hiệu quả hơn. Trong Unity chúng ta có thể import trực tiếp các file mô hình đang thiết kế và sẽ thiết kế hoàn thiện tiếp nếu chưa xong trong khi đó công việc import chỉ diễn ra một lần. Không như việc phải dùng các công cụ khác để thực hiện việc viết game, chúng ta sẽ phải xuất chúng ra một dạng nào đó và mỗi lần sửa lại phần mô hình, chúng ta lại phải import lại và như thế là quá mất thời gian trong việc tạo và chỉnh sửa các mô hình theo ý muốn. Ngoài ra Unity còn cho chúng ta trực tiếp tạo các mô hình nếu muốn. Việc đặt các thuộc tính vật lý trong Unity cũng cực kỳ dễ dàng và hỗ trợ sẵn nhiều chức năng.

1.2 Sơ lược về sự hình thành và phát triển của Unity

a. Hình thành

Phần lõi của Unity ban đầu được viết bởi Joachim Ante vào năm 2001. Sau đó công ty được hình thành vào năm 2005 và bắt đầu với phiên bản 1.0. Đến năm 2007, Unity được nâng lên phiên bản 2.0. Unity bắt đầu hỗ trợ iPhone vào năm 2008. Vào tháng 6/2010, Unity

chính thức hỗ trợ Android và cho ra đời phiên bản 3.0 có hỗ trợ Android vào tháng 9/2010 và bây giờ là phiên bản Unity 5. Từ đó có thể thấy là tốc độ phát triển của Unity khá nhanh.

b. Giải thưởng

Unity đã đoạt được nhiều giải lớn với những giải chính sau:

- Năm 2006, Unity đạt “Best Use of Mac OS X Graphics” tại Apple’s WWDC. Đây là lần đầu tiên trong một công cụ phát triển game đạt được chất lượng do giải thưởng uy tín này đưa ra.
- Năm 2009, Unity Technologies có tên trong “Top 5 công ty game của năm” do Gamasutra tổ chức.
- Năm 2010, Unity đoạt giải Best Engine Finalist do Develop Magazine bình chọn, giải Technology Innovation Award của Wall Street Journal ở thể loại phần mềm.

c. Khách hàng

Unity được trên 250.000 người đăng ký sử dụng gồm Bigpoint, Cartoon Network, Coca-Cola, Disney, Electronic Arts, LEGO, Microsoft, NASA, Ubisoft, Warner Bros, các hãng phim lớn nhỏ, các chuyên gia độc lập, sinh viên và những người đam mê.

1.3 Tính năng của Unity Engine

Môi trường phát triển được tích hợp với tính năng kế thừa, khả năng chỉnh sửa đồ họa, chức năng kiểm tra chi tiết, và đặc biệt tính năng xem trước game ngay trong lúc xây dựng (live game preview).

Triển khai được trên nhiều nền tảng:

- Chương trình độc lập trên Windows và Mac OS.
- Trên web, thông qua Unity Web Player plugin cho Internet Explorer, Firefox, Safari, Opera, Chrome, cho cả Windows và Mac OS.
- Trên Mac OS Dashboard widget.
- Cho Nintendo Wii (cần mua license thêm).
- Cho iPhone, iPad application (cần mua thêm license).
- Cho Google Android (cần mua thêm license).
- Cho Microsoft Xbox 360 (cần mua thêm license).
- Cho Sony PlayStation 3 (4) (cần mua thêm license).

Tài nguyên (model, âm thanh, hình ảnh, ...) được tải vào trong Unity và tự động cập nhật nếu tài nguyên có sự thay đổi. Unity hỗ trợ các kiểu định dạng từ 3DS Max, Maya, Blender, Cinema 4D và Cheetah3D.

Graphics engine sử dụng Direct3D (Windows), OpenGL (Mac, Windows), OpenGL ES (iPhone OS) và các API khác trên Wii.

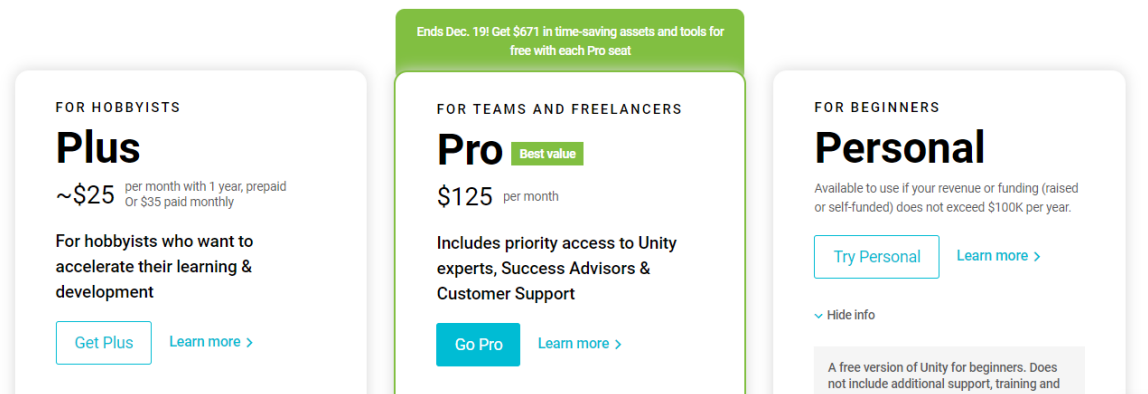
Hỗ trợ bump mapping, reflection mapping, parallax mapping, Screen Space Ambient Occlusion v.v...

Unity Asset Server: Đây là một trong những tính năng mới của Unity, theo đó Unity sẽ cung cấp một hệ thống quản lý theo dạng phiên bản cho tất cả asset và cả script. Đây là một kho chứa các tài nguyên cần thiết cho việc làm game. Khi import cũng như sửa chữa, trạng thái của asset ngay lập tức được cập nhật. Server chạy trên Database OpenSource PostgreSQL và có thể truy cập trên cả Mac lẫn Windows, Linux. Asset Server đòi hỏi một khoản phí phụ trội là \$499 cho mỗi bản copy Unity và một license Unity Pro.

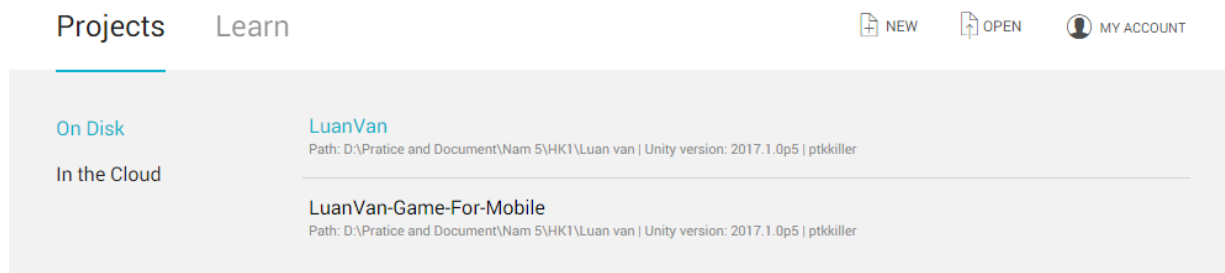
2. TÌM HIỂU THÊM VỀ UNITY ENGINE

2.1 Hướng dẫn sử dụng và cài đặt Unity-Engine

Đầu tiên chúng ta sẽ tải bản Unity cần thiết về máy tính của chính mình, Unity có các phiên bản Plus, Pro và Personal. Sau đó sẽ tải phiên bản cần thiết về máy và tiến hành cài đặt.



Hình 0.2: Các phiên bản của Unity

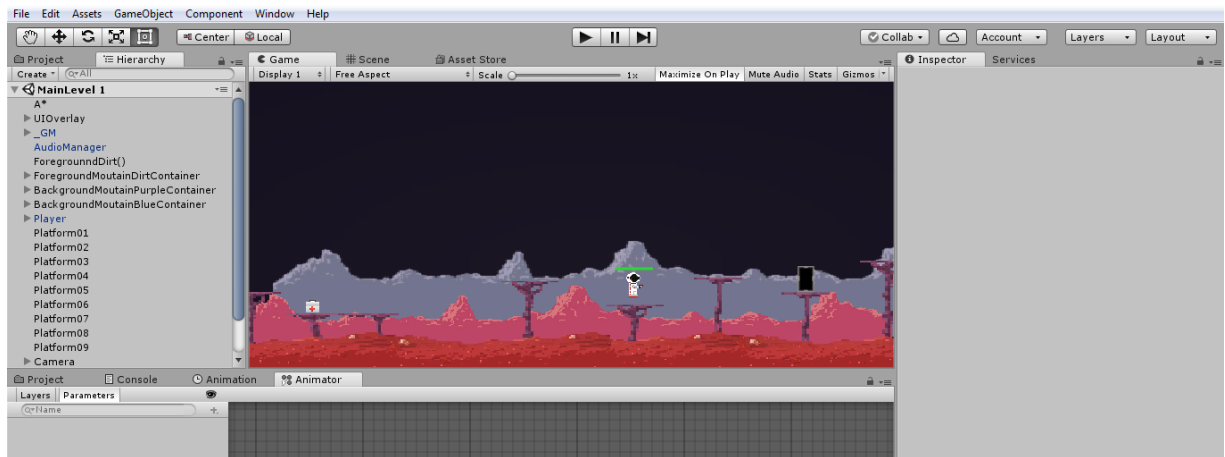


Hình 0.3: Giao diện sử dụng Unity sau khi cài đặt xong

Giao diện sau khi cài đặt gồm có Projects, trong mục Projects hiển thị hai nội dung là “On Disk” và “In the Cloud”, “On Disk” sẽ hiển thị những project mà người dùng đang sử dụng trong ổ đĩa cứng của mình, “In the Cloud” sẽ hiển thị những projects mà chúng ta sử dụng trên Cloud.

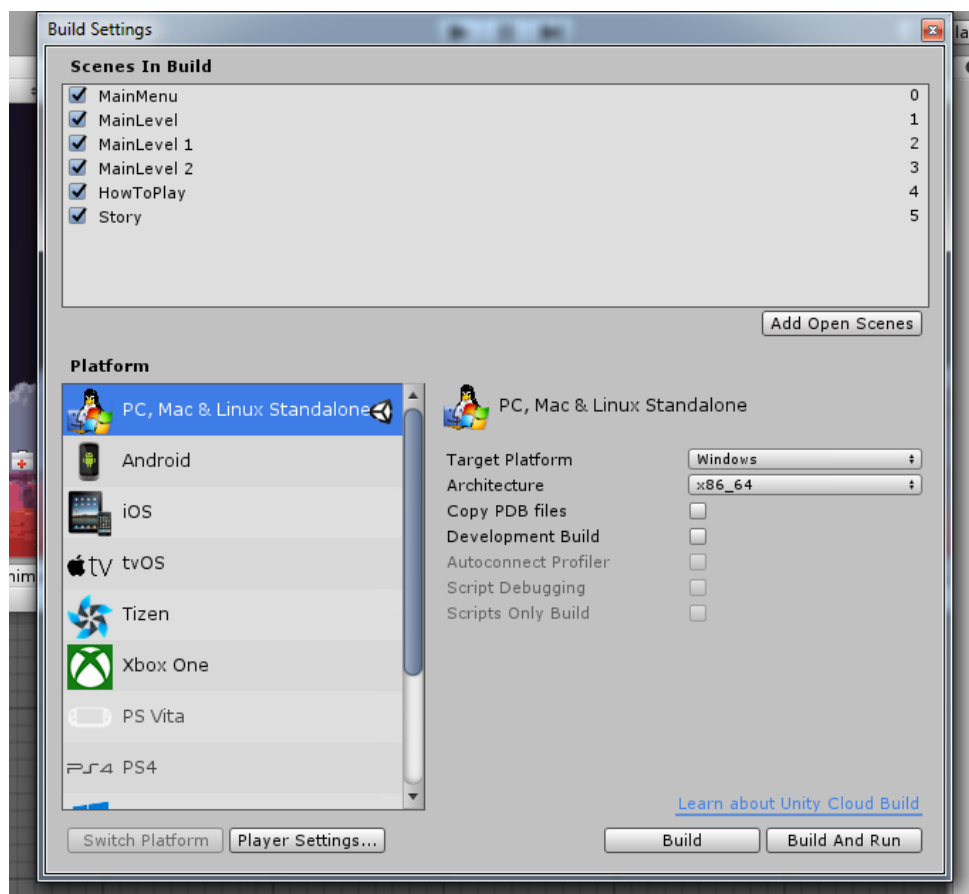
New ở góc trên bên phải là chứng năng tạo một project mới với Unity.

Open ở góc trên bên phải là mục dùng để mở một project nào đó mong muốn của người dùng trong ổ đĩa cứng hoặc trên cloud, sau khi mở và sử dụng một project, khi mở lại Unity, project đó sẽ hiển thị như trên hình 0.2.



Hình 0.4: Giao diện chính khi sử dụng Unity

Sau khi đã hoàn thành một project hay muốn xuất project đó thành một ứng dụng vì người sử dụng cần thử nghiệm hay demo trước sản phẩm vì một mục đích cụ thể nào đó, ta chọn Project cần làm việc và chọn File ở góc trái phía trên của ứng dụng và chọn “Build Setting”

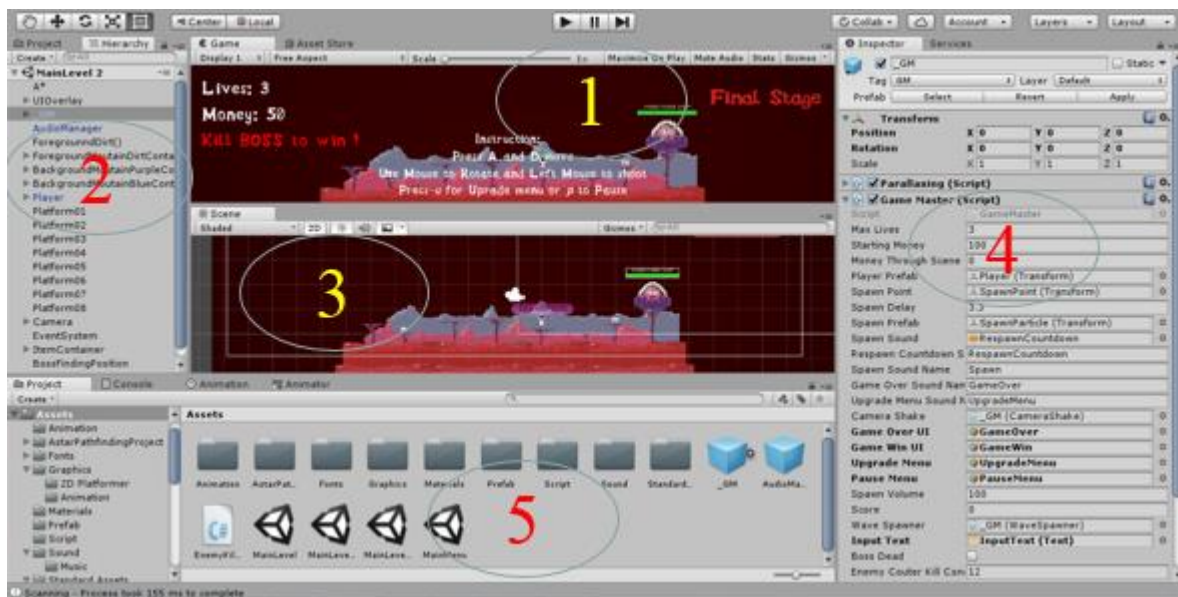


Hình 0.5: Giao diện khi cần build project

Giao diện ở hình 0.4 là giao diện khi cần build thành một sản phẩm, chọn Platform cần thiết ở phía bên trái của hình và chọn Build, lúc này Unity sẽ yêu cầu người sử dụng lựa chọn thư mục cần thiết để lưu trữ.

Sau khi hoàn tất quá trình lưu trữ và build sản phẩm là chúng ta đã có được một ứng dụng game như mong muốn.

2.2 Các thành phần trong Unity Editor



Hình 0.6: Hình minh họa các thành phần trong Unity

1) Cửa sổ Scene

Phần này phần hiển thị các đối tượng trong scenes một cách trực quan, có thể lựa chọn các đối tượng, kéo thả, phóng to, thu nhỏ, xoay các đối tượng.

Có thể thiết lập một số thông số như hiển thị ánh sáng, âm thanh, cách nhìn 2D hay 3D. Khung nhìn Scene là nơi bố trí các Game Object như cây cối, cảnh quan, enemy, player, camera trong game. Sự bố trí hoạt cảnh là một trong những chức năng quan trọng nhất của Unity.

2) Cửa sổ Hierarchy

Tab hierarchy là nơi hiển thị các Game Object trong Scene hiện hành. Khi các đối tượng được thêm hoặc xóa trong Scene, tương ứng với các đối tượng đó trong cửa sổ Hierarchy.

Tương tự trong tab Project, Hierarchy cũng có một thanh tìm kiếm giúp quản lý và thao tác với các Game Object hiệu quả hơn đặc biệt là với các dự án lớn.

3) Cửa sổ Game

Đây là màn hình demo Game, là góc nhìn từ camera trong game. Thanh công cụ trong cửa sổ game cung cấp các tùy chỉnh về độ phân giải màn hình, thông số (Stats), Gizmos, tùy bật tắt các component.

4) Cửa sổ Project

Đây là cửa sổ explorer của Unity, hiển thị thông tin của tất cả các tài nguyên (Assets) trong game.

5) Cửa sổ Inspector

Cửa sổ Inspector hiển thị chi tiết các thông tin về Game Object đang làm việc, kể cả những component được đính kèm và thuộc tính của chính component. Người dùng có thể điều chỉnh, thiết lập mọi thông số và chức năng của Game Object thông qua cửa sổ Inspector.

Mọi thuộc tính thể hiện trong Inspector đều có thể dễ dàng tùy chỉnh trực tiếp mà không cần thông qua một kịch bản định trước.

Các thiết lập của từng component được đặt trong menu. Các bạn có thể click chuột phải, hoặc chọn icon hình bánh răng nhỏ để xuất hiện menu.

Ngoài ra Inspector cũng thể hiện mọi thông số Import Setting của asset đang làm việc như hiển thị mã nguồn của Script hoặc các thông số animation.

PHẦN GIỚI THIỆU

1. ĐẶT VẤN ĐỀ

Trong sự phát triển vượt bậc của nền công nghệ 4.0 hiện nay, sự ra đời và lên ngôi của các nền tảng game trên máy tính hay điện thoại thông minh là tất yếu. Tình trạng này thể hiện và diễn ra rất rõ tại Việt Nam khi mà tổng số lượng thiết bị chơi game tăng cao và người sử dụng càng gia tăng, nhất là giới trẻ hiện nay. Các tựa game hấp dẫn trên thiết bị PC hay điện thoại thông minh cũng lần lượt ra đời với số lượng ngày càng nhiều về nội dung cũng như thể loại. Trong xu thế trên đã tạo ra ngành lập trình game và đã phát triển một cách vượt bậc, mở ra một thị trường làm việc hấp dẫn thu hút đối với các lập trình viên. Vì vậy việc tiếp cận và lập trình game cũng đã trở nên dễ dàng hơn nhờ những công cụ phát triển ứng dụng – Game PC như Unity2D(Unity3D), Unreal-Engine, Cocos2d-x, Corona SDK...

Trong các công cụ phát triển game, Unity là một trong những công cụ phát triển game đa nền tảng và phát triển ứng dụng nhanh, thế nên Unity thường được sử dụng để phát triển game 2D hay 3D trên các thiết bị di động và PC. Ngoài ra, Unity còn cung cấp một hệ thống thư viện phong phú, hỗ trợ đầy đủ và được thiết kế rất tốt. Do đó, các nhà phát triển game có thể dễ dàng xây dựng các thao tác (hành động) trong game chỉ với một vài dòng code đơn giản. Với công cụ Unity, chúng ta có thể chỉ cần viết code một lần, sau đó ta có thể nhanh chóng biên dịch dễ dàng thành ứng dụng cho các nền tảng khác nhau như Windows, Android, IOS v.v... Ngoài ra Unity còn hỗ trợ chuyển đổi các thiết bị có tỉ lệ, kích cỡ và độ phân giải khác nhau một cách dễ dàng.

Từ những nội dung trên, tác giả đã đề xuất đề tài “Phát triển game sinh tồn ở hành tinh mới bằng Unity 2D”.

Game “Sinh tồn ở hành tinh mới” lấy ý tưởng từ game huyền thoại – trò chơi điện tử thể loại “chạy không ngừng – Mario” để vượt qua các chướng ngại vật và tiêu diệt quái vật trên đường nhằm giải cứu công chúa. Trong trò chơi trên người chơi sẽ hóa thân thành 1 trong những người lính đang sinh sống và bảo vệ hành tinh mới mà con người đã tìm được và đột nhiên người chơi sẽ phải đối mặt với người ngoài hành tinh xâm lăng, anh sẽ buộc phải chiến

đấu với người ngoài hành tinh và tìm thủ lĩnh của kẻ địch và tiêu diệt để kết thúc cuộc xâm lăng của kẻ địch.

2. TÓM TẮT LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ

Việc phát game chạy đa nền tảng là một vấn đề không phải là mới, bởi vì hiện nay đã có rất nhiều game sử dụng nhiều bộ game engine khác nhau (Unity, Unreal Engine, Cocos2D , v.v...) được đăng tải liên tục trên website trang chủ của nhà sản xuất hay Stream.

Tựa game huyền thoại “Mario” đã được thế giới chào đón nồng nhiệt, mặc dù có đồ họa không cao nhưng trò chơi này có sức hút rất lớn vì lối chơi vui nhộn, dễ dàng tiếp cận với các đối tượng người chơi khác nhau. Tác giả sử dụng một số thuật toán tương tự để lập trình cho trò chơi này. Mặc dù game “Sinh tồn ở hành tinh mới” dựa trên ý tưởng game “Mario”, nhưng áp dụng trên các đối tượng khác nhau, số lượng đối tượng cũng khác nhau, có sự pha trộn thể loại game bắn súng và giải quyết vấn đề do bản thân tác giả đề xuất.

3. MỤC TIÊU ĐỀ TÀI

Ứng dụng bộ công cụ phát triển game Unity và các thư viện sẵn có (OpenGL, OpenAL, Box2D, v.v...) để xây dựng game “Sinh tồn ở hành tinh mới” với các vấn đề trọng tâm như:

- Bố trí các màn chơi một cách logic từ dễ đến khó.
- Thực hiện nhảy, bắn, quay cánh tay cho nhân vật một cách trung thực.
- Phát hiện nhân vật bắn trúng mục tiêu.
- Xử lý va chạm và hiệu ứng xảy ra khi nhân vật va phải các vật cản.
- Lưu các dữ liệu trong game (thiết lập âm thanh, thành tích, ...).
- Tạo các hiệu ứng cho nhân vật và kẻ địch: hiệu ứng bắn trúng kẻ địch, kẻ địch phát nổ khi bị tiêu diệt, tia lửa khi nhân vật bắn ra.
- Tạo độ chân thật: khi nhân vật bắn thì màn hình chơi game sẽ rung nhẹ, hay khi địch bị tiêu diệt thì màn hình chơi game cũng rung nhẹ, tạo tia lửa khi nhân vật bắn ra không cùng một vị trí.
- Tạo thị sai so với Background khi nhân vật duy chuyển (chẳng hạn như đồi núi khi nhân vật duy chuyển thì background cũng duy chuyển, tạo độ chân thật khi chơi)
- Tạo item: Bình cứu thương, để hồi phục lại Health cho nhân vật

4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

Đối tượng được nghiên cứu là các đối tượng đồ họa, các thuật toán được sử dụng trong game và cách sử dụng công cụ Unity. Tuy nhiên, trong thời gian có hạn và phạm vi luận văn nghiên cứu chỉ tập trung vào nền tảng PC trên hệ điều hành Windows và điện thoại thông minh Android.

5. NỘI DUNG NGHIÊN CỨU

Nội dung cần nghiên cứu: Unity Game-Engine là một trong những đối tượng cần nghiên cứu. Vì Unity cung cấp rất nhiều thư viện và hình ảnh hỗ trợ cơ bản cho người làm game, ngoài ra còn có các bộ thư viện quản lý cảnh(Scene) hay chuyển cảnh là UnityEngine.SceneManagement, bộ thư viện vật lý được hỗ trợ thông qua các Box Collider và Rigidbody được tích hợp trong Unity để tạo ra môi trường vật lý thú vị cho game, cho đến cách phát hiện va chạm giữa các thành phần vật lý với nhau, đồng thời cho phép tùy biến sản phẩm cuối cùng để có thể biên dịch chạy được trên đa nền tảng.

Thông qua quá trình phát triển game. Đã giúp cho người phát triển game hiểu được một cách tổng quan, sơ bộ về phương thức hoạt động của một số trò chơi đơn giản, những thành phần cơ bản cần có trong một trò chơi. Tìm hiểu cách thêm, sửa, xóa, tạo mới các kết cấu, cấu tạo, xếp đặt màu sắc lên các đối tượng hiển thị trong Unity.

Đối với người phát triển game bằng Unity, ngôn ngữ sử dụng để lập trình để có thể tạo ra những file văn bản(script) là không thể thiếu. Do đó, Unity cho phép sử dụng hai loại ngôn ngữ lập trình cấp cao giúp người phát triển game bằng Unity có thể thao tác là C# và JavaScript.

Tác giả đã chọn ngôn ngữ C# làm ngôn ngữ chính để nghiên cứu, tìm hiểu và phát triển đề tài game của chính mình. Vì ngôn ngữ C#, là một ngôn ngữ thuần OOP được phát triển bởi tập đoàn Microsoft, được hỗ trợ tối đa và đa nền tảng bởi chính nhà sản xuất.

Tìm hiểu cách lập trình OOP, soạn thảo mã kịch bản cho chương trình trên Unity, cách đặt ra quy luật và giải quyết các vấn đề trong trò chơi. Đồng thời tham khảo các runner game, shoot game trên internet, hay qua các video, tìm kiếm các ý tưởng để tạo ra các màn chơi cũng như các hiệu ứng cho game thêm sinh động.

6. NHỮNG ĐÓNG GÓP CHÍNH CỦA ĐỀ TÀI

Đề tài đã đóng góp thêm vào tài liệu nghiên cứu cho sinh viên Đại Học Cần Thơ nói chung và khoa Công Nghệ Thông Tin và Truyền Thông nói riêng trong lĩnh vực lập trình game. Về sau, đề tài này cũng sẽ trở thành tài liệu tham khảo cho các sinh viên khóa tiếp theo khi nghiên cứu về game engine – Unity và ngôn ngữ lập trình C#.

Game “Sinh tồn ở hành tinh mới” đã bổ sung thêm vào kho trò chơi phong phú và đa dạng trên chợ ứng dụng Stream hay Google Play-Store.

Với cách tiếp cận dễ dàng, gameplay quen thuộc của trò chơi, người chơi có thể giải trí nhẹ sau những thời gian làm việc căng thẳng mệt mỏi, nhưng cũng không kém phần lôi cuốn hấp dẫn.

PHẦN NỘI DUNG

CHƯƠNG 1 – MÔ TẢ BÀI TOÁN

1. MÔ TẢ CHI TIẾT BÀI TOÁN

Vấn đề hiện tại là làm sao để có thể tạo ra nhân vật, làm cho nhân vật duy chuyển và có thể nhảy được để có thể nhảy qua các bụi và tiêu diệt kẻ địch tấn công mình, bắn trả lại kẻ địch và tiêu diệt chúng. Bài toán đặt ra có 3 màn chơi, trong từng màn chơi tùy vào mức độ phức tạp mà người lập trình đã đặt ra: như độ dài và độ phức tạp của địa hình, các bụi có độ cao khác nhau, số lượng vật khác nhau qua các màn chơi.

Các yêu cầu đặt ra với bài toán này được chia thành hai nhóm: yêu cầu chức năng và yêu cầu phi chức năng:

Các yêu cầu chức năng:

Có Menu chính để người chơi bắt đầu game. Khi người chơi vào game thì sẽ mặc định là màn chơi đầu tiên, có địa hình cụ thể, nhân vật chính, các kẻ địch tấn công, các bụi địa hình cao thấp khác nhau.

Hướng dẫn cách chơi – người dùng có thể hiểu thêm về cách điều khiển nhân vật theo ý muốn.

Các yêu cầu phi chức năng:

- Giao diện:

- + Đẹp, bắt mắt.
- + Các tông màu phù hợp với các đối tượng.

- Thời gian thực hiện:

- + Thời gian trễ không quá 3s.
- + Game vận hành mượt mà mà không phát sinh lỗi.

- Thiết bị hỗ trợ:

- + PC Pentium 4 2GB, RAM 512MB, VGA 128MB
- + Bộ nhớ trống 50MB
- + Độ phân giải màn hình đề xuất 800x600 hoặc cao hơn

- Ghi chú: Ứng dụng được biên dịch cho hệ điều hành Windows, việc chuyển sang hệ điều hành IOS, Android không tốn quá nhiều thời gian cấu hình.

2. PHÂN TÍCH ĐÁNH GIÁ CÁC GIẢI PHÁP CÓ LIÊN QUAN ĐẾN BÀI TOÁN

Trong nền công nghiệp game hiện đại, việc lập trình game chơi trên command-line, hay các cách phát triển và tạo ra một game theo các cách cổ điển đã không còn hữu dụng. Vì vậy các nhà sản xuất phần mềm đã bắt đầu phát triển những công cụ hỗ trợ cho người làm game, các công cụ gọi là Game-Engine.

Game-Engine cho phép người sử dụng tạo ra một game một cách nhanh chóng và hiệu quả nhờ những bộ công cụ và thư viện được hỗ trợ sẵn. Như là thư viện quản lý cảnh(Scene) và chuyển cảnh UnityEngine.SceneManagement, hệ trợ xử lý va chạm vật lý RigidBody, Box-Collider. Đồng thời cho phép sử dụng những mã văn bản(script) để sử dụng vào các mục đích phát triển của người làm game.

Đối với Game-Engine, cần phải có một ngôn ngữ lập trình phù hợp mới có thể sử dụng một cách hoàn chỉnh những ý muốn và mục đích của người làm game.

Vì vậy hướng giải quyết là Sử dụng công cụ làm game Unity và ngôn ngữ lập trình C#.

Unity có nhiều ưu điểm vượt trội, Unity vận hành rất mượt mà, ngay cả trên những máy tính có cấu hình khiêm tốn, không những vậy Unity còn giúp dễ dàng điều chỉnh các độ phân giải, tỉ lệ màn hình khác nhau. Unity cho phép tạo âm thanh/yếu tố khá đơn giản. Đáp ứng được mong muốn và nhu cầu của người phát triển game.

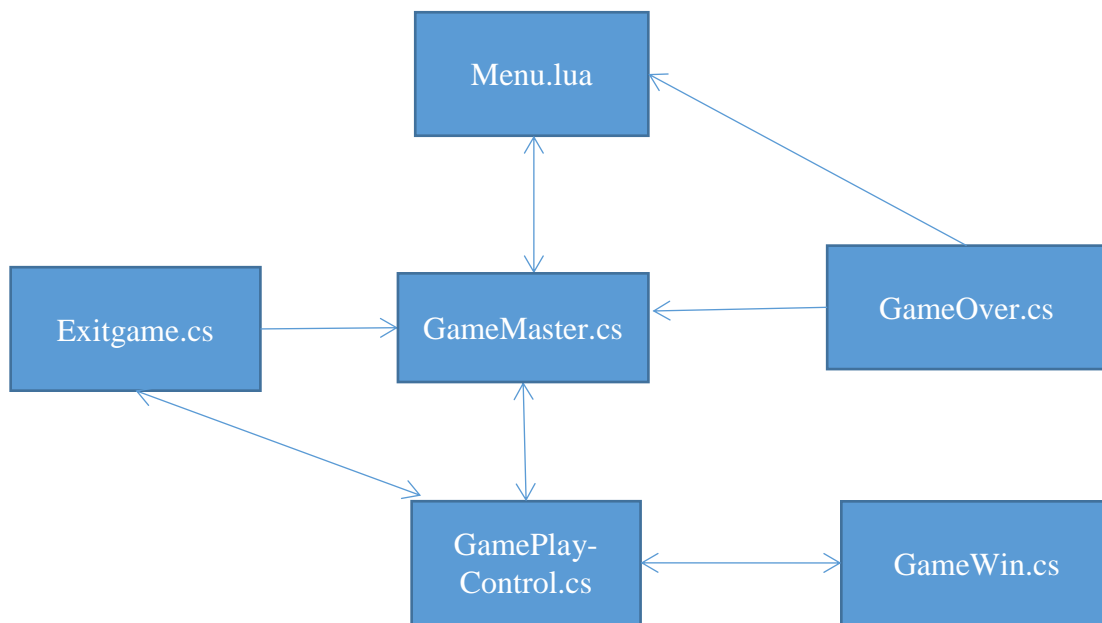
C# là một ngôn ngữ lập trình được phát triển bởi Microsoft. Để cài đặt ngôn ngữ C# dùng cho việc lập trình, có thể sử dụng Visual Studio, vì đây là một bộ IDE cho phép biên dịch chương trình do chính Microsoft phát triển và nhà sản xuất còn cho phép tùy chọn gói cài đặt phù hợp cho người dùng lựa chọn. Ngoài ra ngôn ngữ C# cũng thuộc vào loại những ngôn ngữ nhanh nhất, kết hợp tốt với các ngôn ngữ khác, chẳng hạn như C/C++, Java .

Sử dụng ngôn ngữ C# có thể giúp cài đặt các kiến trúc, thiết kế dữ liệu và thiết kế thuật toán trong game. Vậy nên tác giả đã chọn hướng giải quyết này.

3. TIẾP CẬN GIẢI QUYẾT VẤN ĐỀ

Về cơ bản, một ứng dụng game cần có một luồng trình tự nhất định, nó giúp ta hiểu rõ hơn về cách mà game hoạt động. Với Unity, ứng dụng sẽ kiểm tra xem các file script của ta có bị lỗi Compile không và sẽ thực hiện chạy xử lý.

Để có thể dễ dàng trong việc quản lý và sửa lỗi ta cần phân loại game ra thành những thành phần khác nhau.



Hình 1.1: Sơ đồ GameFlow cơ bản.

Trong sơ đồ trên, tập tin GameMaster.cs sẽ chịu trách nhiệm điều khiển, quản lý và thiết lập các thành phần trong game. Giúp nối kết các sự kiện được thiết lập trong quá trình chơi game của người chơi.

Ngoài ra, khi có vấn đề về sự cố xảy ra trong quá trình vận hành và kiểm thử, sẽ dễ dàng cho người phát triển khi sửa lỗi.

4. TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT GIẢI QUYẾT VẤN ĐỀ

Về cơ bản phải giải quyết được các chuyển động của nhân vật và hoàn thành cơ bản về sự duy chuyển của nhân vật, cũng như bắn trả lại kẻ địch đang tấn công người chơi. Sau đó tạo ra các màn chơi được thiết lập từ dễ đến khó, tạo độ hấp dẫn cho game.

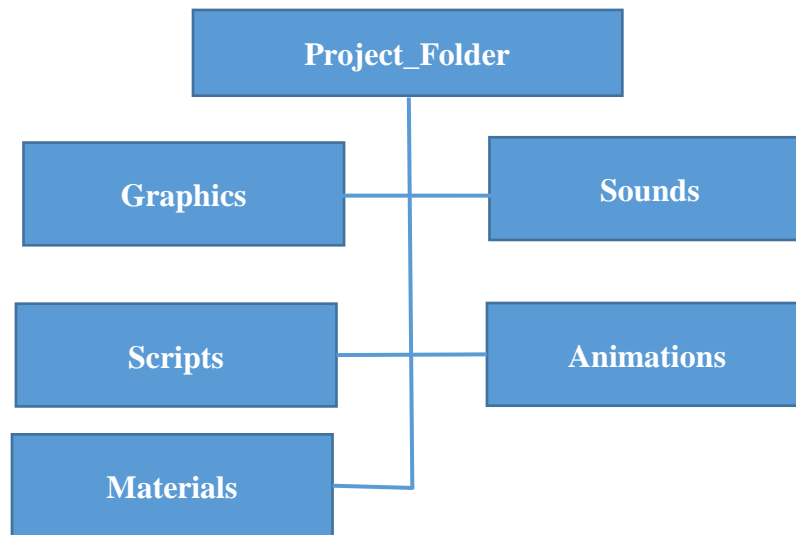
Vì để tạo được một game cần phải có công cụ làm game hay còn gọi là những Engine giúp đỡ người phát triển làm game và các thư viện hỗ trợ tối đa trong việc phát triển game. Do đó tác giả đã lựa chọn Unity làm công cụ phát triển game và giải quyết vấn đề về lựa chọn game-engine để phát triển game. Đồng thời Unity cung cấp hầu như đầy đủ các thư viện cần thiết để người dùng có thể dễ dàng làm việc. Sử dụng ngôn ngữ C# để lập trình và thao tác trên Unity. Vì C# là một trong hai ngôn ngữ lập trình được sử dụng trong Unity.

Sử dụng ngôn ngữ C# có thể giúp cài đặt các kiến trúc, thiết kế dữ liệu và thiết kế thuật toán trong game.

CHƯƠNG 2 – THIẾT KẾ GIẢI PHÁP

1. THIẾT KẾ KIẾN TRÚC

Một project Unity được phân cấp tổ chức như sau:



Hình 2.1: Hệ thống phân cấp project cơ bản

Project_Folder: Là thư mục cha của game, chứa các thư mục con Graphic, Sounds, Scripts, Animations, Materials.

Graphics: Thư mục chứa các hình ảnh hiển thị trong game, chẳng hạn như: Background(Khung nền), nhân vật, kẻ địch, tia đạn, vũ khí .v.v...

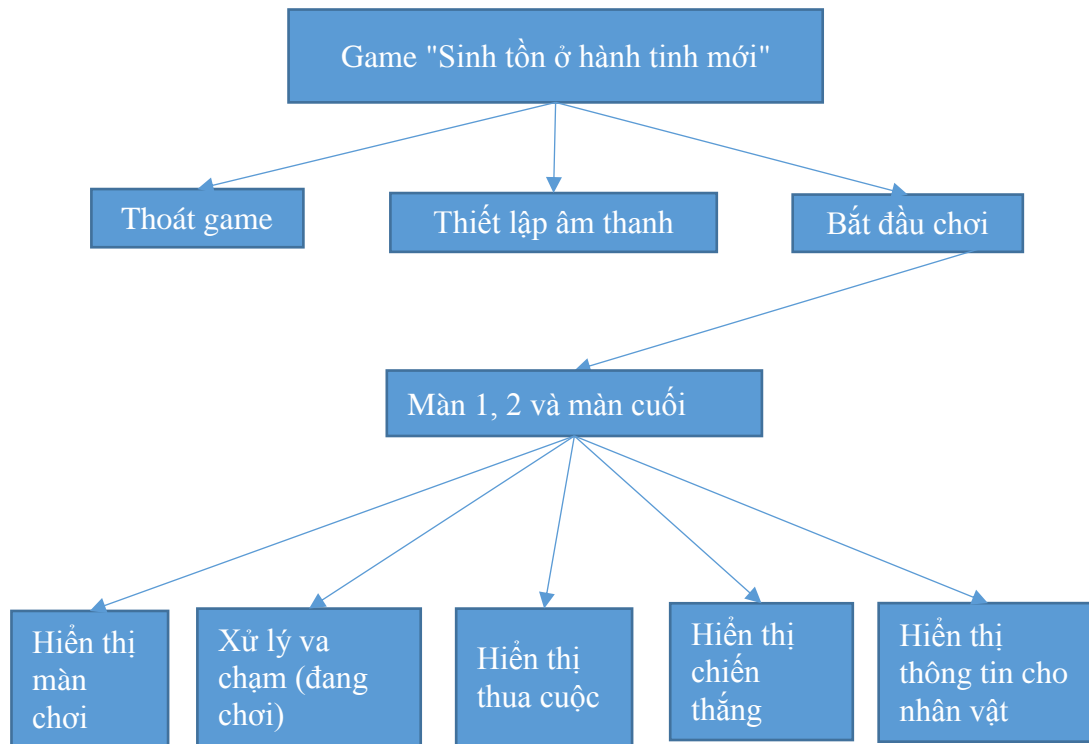
Sounds: Thư mục chứa âm thanh khi bắt đầu chơi game.

Scripts: Thư mục chứa các file script mã nguồn được sử dụng trong game.

Animations: Thư mục chứa các file thành phần cấu tạo nên chuyển động và ràng buộc chuyển động của nhân vật.

Materials: Thư mục chứa các file refab(thành phần) cần thiết trong game.

Một sơ đồ phân cấp các chức năng cơ bản khi vào game được thể hiện như sau:



Hình 2.2: Sơ đồ phân cấp các chức năng của phần mềm.

Sau khi game được kích hoạt và hoạt động, ngay lập tức sẽ thiết lập âm thanh cho game, người chơi sẽ có lựa chọn là bắt đầu chơi hoặc thoát game.

Nếu người chơi ấn vào thoát game, ứng dụng sẽ đóng lại. Còn ngược lại sẽ bắt đầu màn chơi đầu tiên.

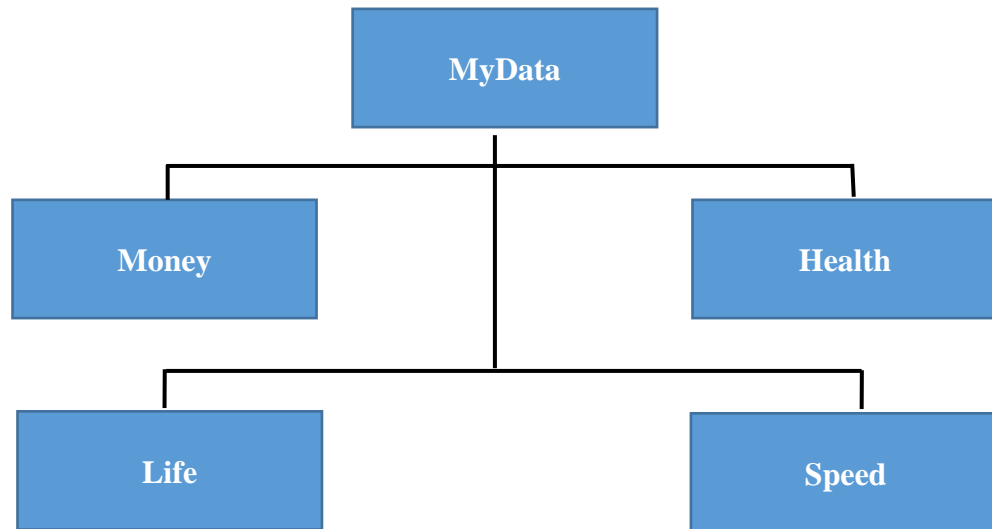
Khi màn chơi đầu tiên bắt đầu thì các chức năng sau sẽ được kích hoạt:

- + Hiển thị màn chơi: cho phép người chơi nhìn thấy màn chơi hiện tại của chính mình.
- + Xử lý va chạm (đang chơi): Người chơi có thể đứng trên mặt đất, hay va vào vật thể hoặc kẻ địch.
- + Hiển thị thua cuộc: Thua cuộc khi người chơi không thể tiếp tục.
- + Hiển thị chiến thắng: Khi người chơi đã hạ được BOSS cuối cùng.

+ Hiển thị thông tin cho nhân vật: Người chơi có thể nhìn thấy thông tin của điều kiện qua màn khi chạm vào cánh để thực hiện qua màn hay thông tin số mạng còn lại, hay số tiền kiếm được.

2. THIẾT KẾ DỮ LIỆU

Một sơ đồ phân cấp dữ liệu trong game được thể hiện như sau:



Hình 2.3: Sơ đồ phân cấp dữ liệu dùng trong game

Bảng 2.1: Bảng diễn giải các dữ liệu dùng trong Game

Tên dữ liệu		Giải thích
MyData	Money	Số tiền người chơi kiếm được khi tiêu diệt kẻ địch
	Health	Chỉ số sức sống của người chơi
	Life	Số mạng mà người chơi có được
	Speed	Tốc độ duy chuyển của người chơi

3. THIẾT KẾ GIAO DIỆN

Giao diện game gồm có:

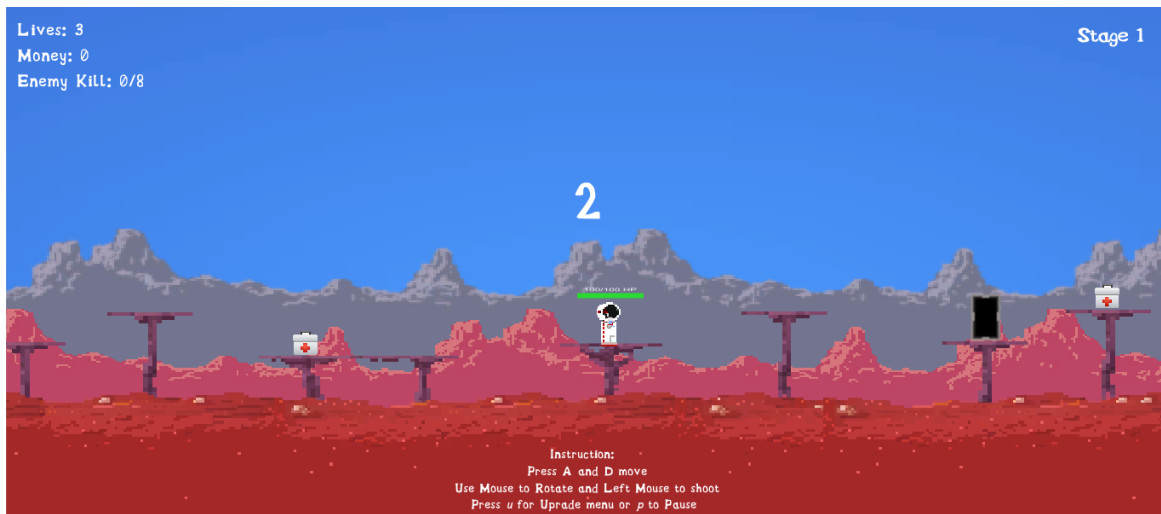
- + Menu mở đầu game.
- + Giao diện chính khi vào chơi.
- + Giao diện game khi ấn Pause(tạm dừng game).
- + Giao diện game khi ấn U để bật upgrade menu.
- + Giao diện khi người chơi thua cuộc.
- + Giao diện khi người chơi thắng cuộc.



Hình 2.4: Thiết kế giao diện chính của trò chơi.

Giải thích:

- + Giao diện chính khi vào game cho phép người chơi lựa chọn dựa vào hai nút(button) được thiết kế sẵn, là “Play” và “Quit” cho phép người chơi ấn “Play” để bắt đầu chơi và “Quit” để thoát game.



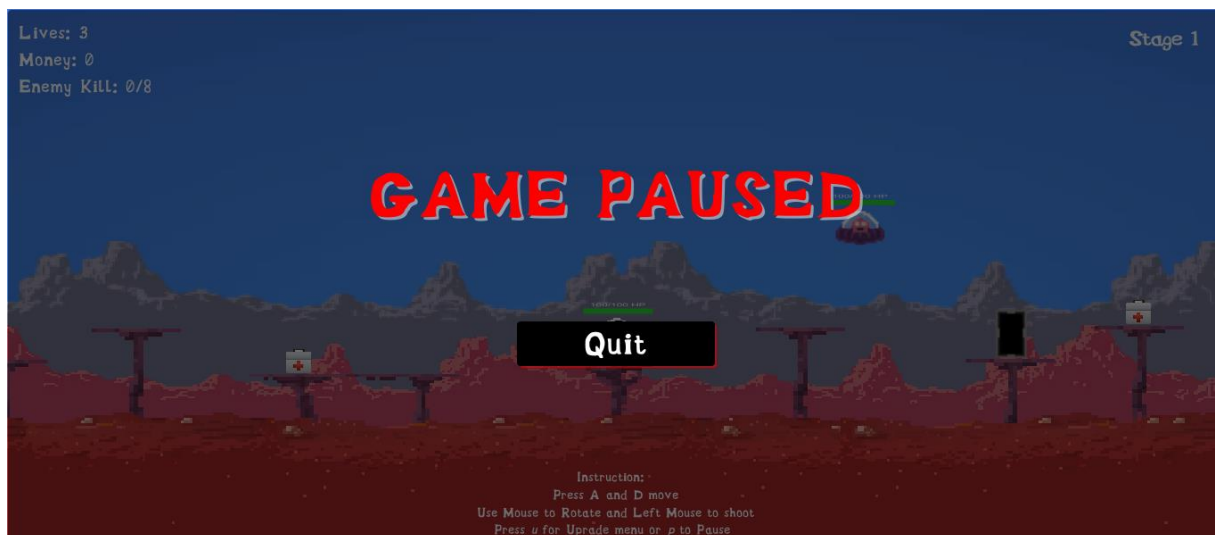
Hình 2.5: Giao diện chính khi vào Game và kèm theo hướng dẫn

Giải thích:

+ Khi vào được giao diện chính của game, người chơi sẽ thấy được nhân vật chính mà mình sẽ điều khiển kèm theo thanh máu(Health bar) để chống lại kẻ địch xâm lăng, đồng thời sẽ thấy được tổng thể các ảnh nền phía sau(background) của game. Bên cạnh đó người chơi sẽ thấy được các thông tin cơ bản của nhân vật chính ở góc trái phía trên màn hình hiển thị các thông tin như: số mạng(Lives) hiện tại, số tiền(Money) kiếm được khi bắn hạ quái vật, kèm theo là một trong những điều kiện dùng để qua màn là số lượng kẻ địch cần giết(Enemy Kill).

+ Phía góc phải phía trên là thông tin màn chơi mà người chơi đang chơi.

+ Ở giữa phía dưới màn hình là thông tin dùng để hướng dẫn người chơi khi vào chơi gồm có các thông tin như: duy chuyển trái phải bằng phím A hoặc D, sử dụng chuột để điều khiển hướng bắn của vũ khí và chuột trái để bắn, ấn phím U để bật menu nâng cấp (menu upgrade) và ấn phím P để dừng game(Pause).

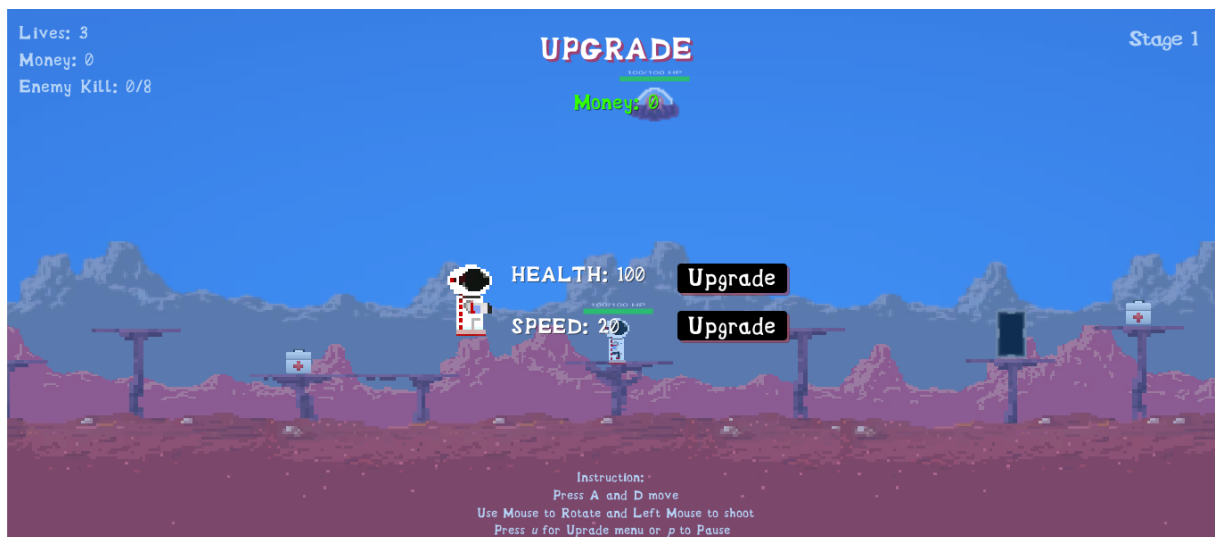


Hình 2.6: Giao diện khi người chơi ấn phím P

Giải thích:

+ Sau khi ấn vào phím P(Pause) để dừng game, game sẽ dừng tạm thời và người chơi sẽ không thể dùng các nút khác để duy chuyển, hoặc bắn trả kẻ địch đang tấn công người chơi. Đồng thời kẻ địch cũng sẽ hoàn toàn dừng lại và không tấn công được người chơi. Được sử dụng khi người chơi muốn tạm dừng game để làm việc và khi cần giải trí thì sẽ không bỏ lỡ màn chơi mà mình đang chơi.

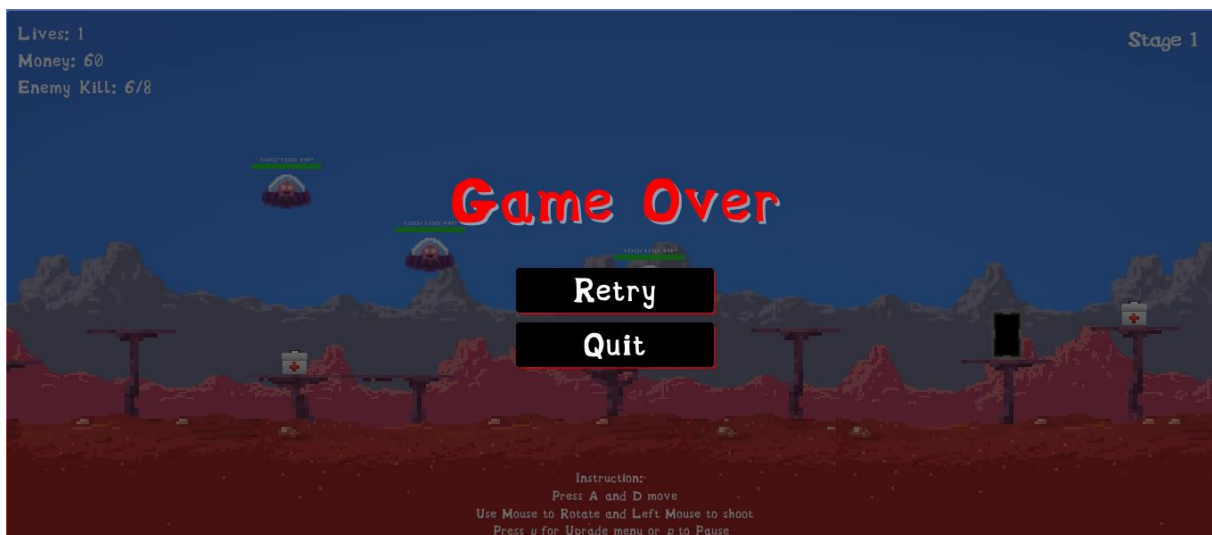
+ Được cung cấp thêm nút(button) “Quit” dùng để thoát game khi cần thiết.



Hình 2.7: Giao diện khi người chơi ấn phím U

Giải thích:

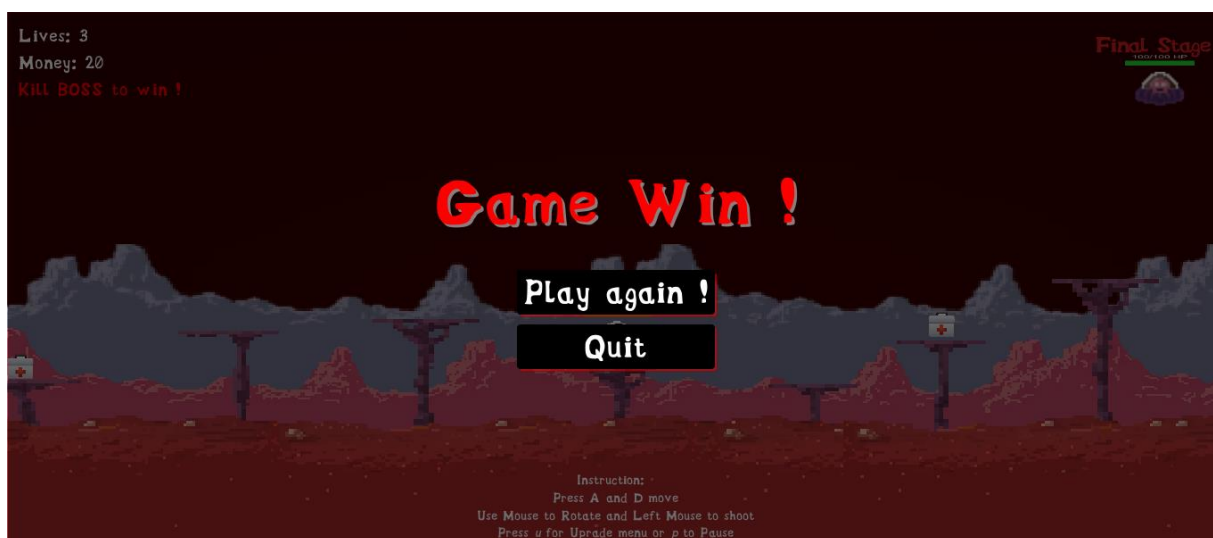
- + Sau khi ấn phím U để bật menu nâng cấp(upgrade menu), game sẽ dừng tạm thời và người chơi sẽ không thể dùng các nút khác để duy chuyển, hoặc bắn trả kẻ địch đang tấn công người chơi. Đồng thời kẻ địch cũng sẽ hoàn toàn dừng lại và không tấn công được người chơi.
- + Được sử dụng khi người chơi muốn muốn nâng cấp hay nâng cao tốc độ duy chuyển của nhân vật, hay máu của nhân vật lên.



Hình 2.8: Giao diện khi người chơi thua cuộc

Giải thích:

- + Là giao diện khi người chơi không thể tiếp tục game do không còn mạng(Lives) nữa.
- + khi giao diện này xuất hiện người chơi sẽ có hai lựa chọn là chơi lại hoặc thoát game.



Hình 2.9: Giao diện khi người chơi chiến thắng

Giải thích:

- + Là giao diện khi người chơi đã thắng game khi đã hạ được “BOSS” của kẻ địch.
- + Người chơi sẽ có hai lựa chọn là chơi lại game(Play again) hoặc là thoát game(Quit).

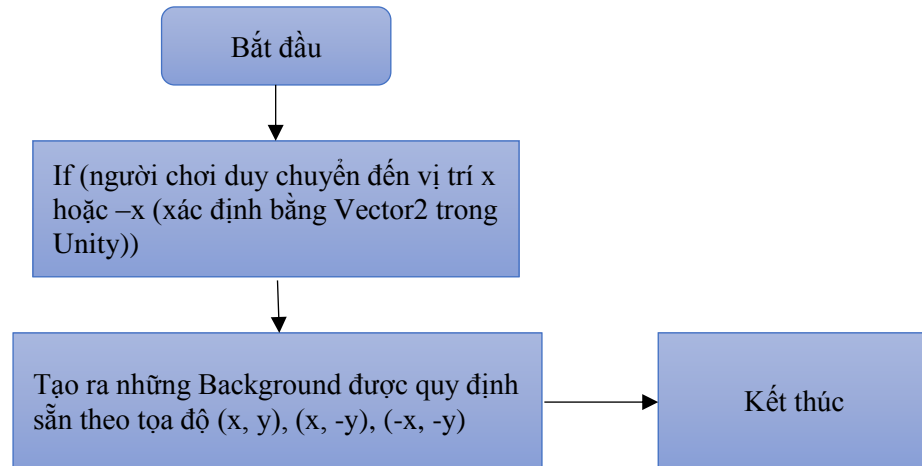
4. THIẾT KẾ THUẬT TOÁN

Danh sách thuật toán cần thiết trong trò chơi

Bảng 2.2: Bảng danh sách tên các thuật toán trong trò chơi

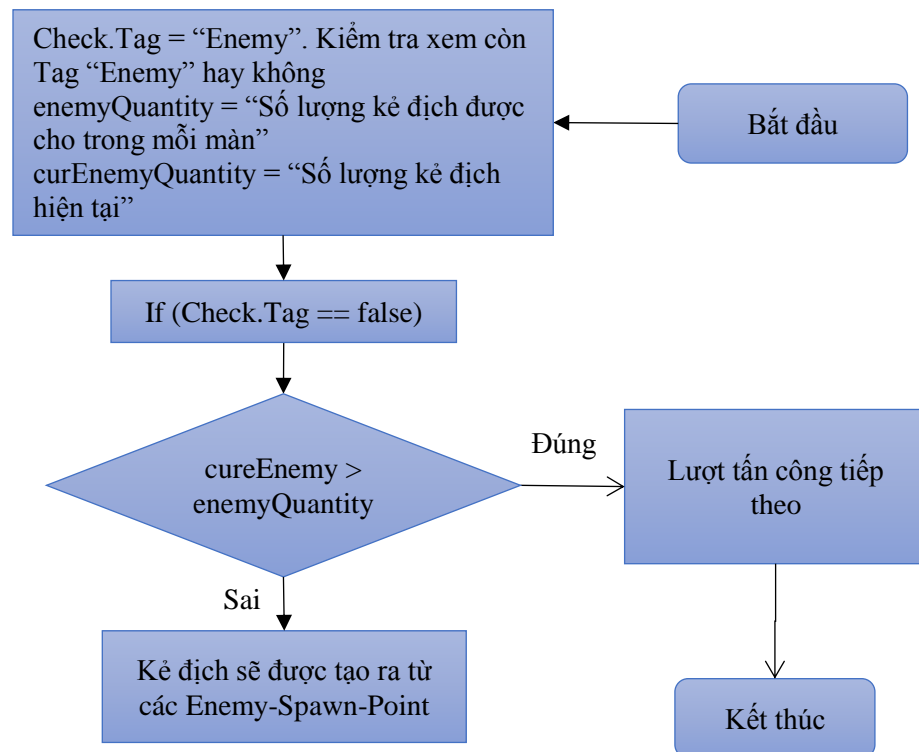
STT	Tên thuật toán	Ghi chú
1	Tạo background lặp lại.	Khi người chơi duy chuyển đến một khoảng cách nhất định thì background sẽ tự động được thêm vào, giúp cho người chơi không có cảm giác đi tràn màn hình, hoặc ra khỏi vùng phủ background
2	Tạo kẻ địch	Thực hiện tạo kẻ địch xuất hiện ra ngẫu nhiên theo các điểm quy định sẵn theo từng đợt tấn công
3	Kiểm tra va chạm.	Kiểm tra va chạm giữa nhân vật với các đối tượng khác trong trò chơi. Hoặc giữa các vật thể khác nhau, chẳng hạn như đạn bắn ra từ vũ khí của người chơi
4	Bắt sự kiện kết thúc	Kết thúc sự kiện khi hoàn tất một sự kiện và các quá trình thực hiện giải thuật.
4	Giải thuật A*, dùng để tạo AI vạch đường cho kẻ địch tấn công người chơi	Sử dụng giải thuật A*, tìm đường đi ngắn nhất từ kẻ tấn công đến kẻ bị tấn công và duy chuyển đến gây sát thương cho kẻ bị tấn công.

4.1 Tạo Background lặp lại:



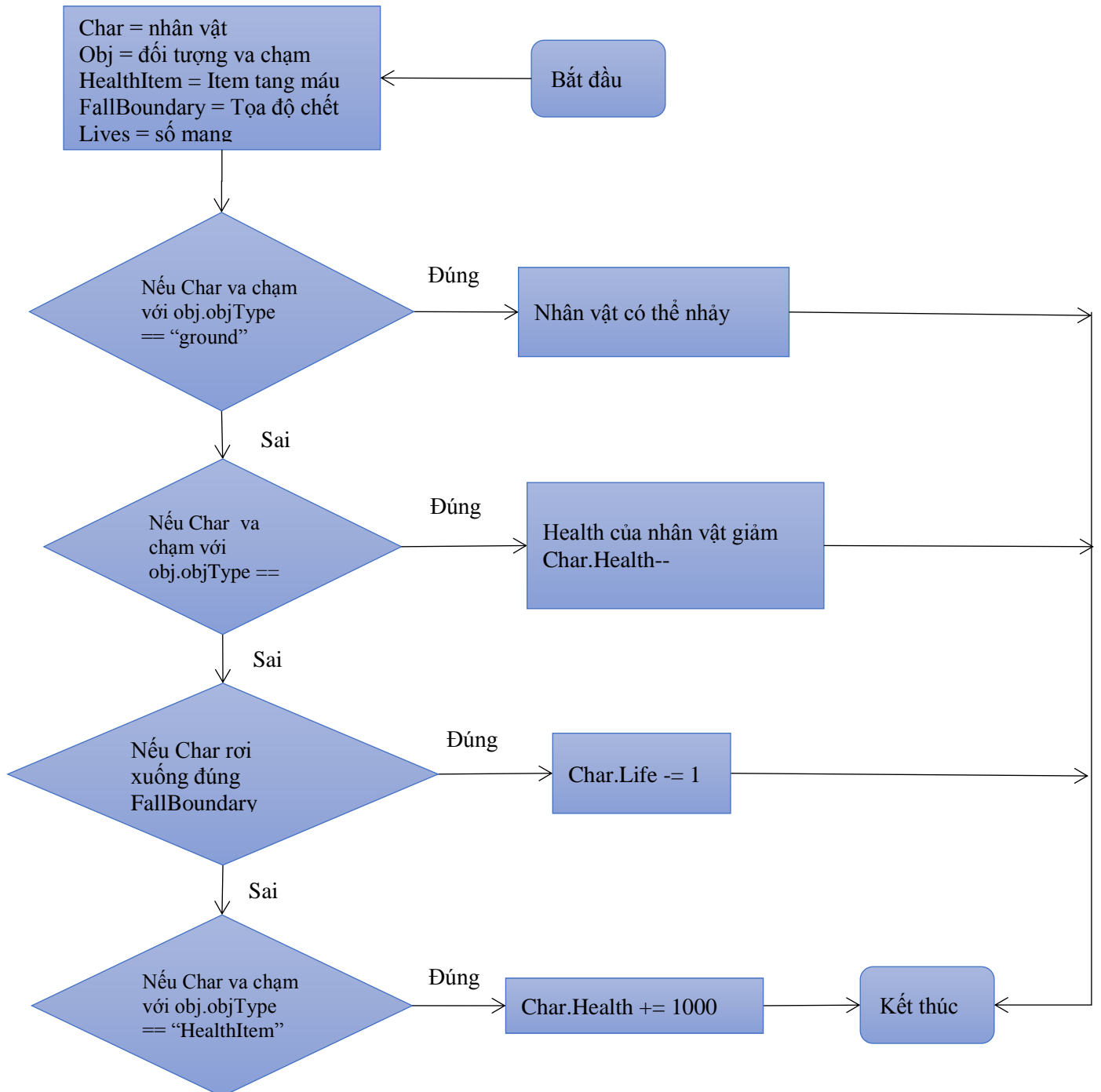
Hình 2.10: Thuật toán tạo background lặp lại cho màn chơi

4.2 Tạo kẻ địch:



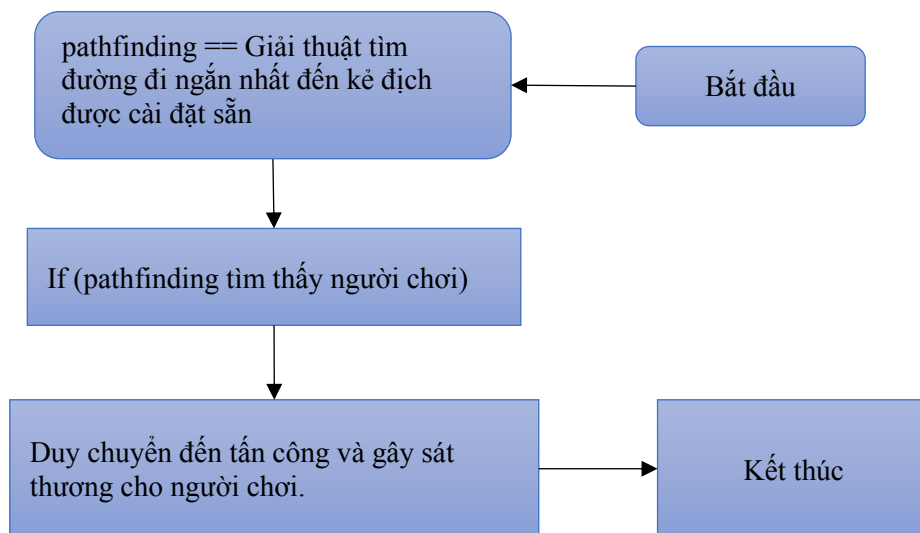
Hình 2.11: Thuật toán tạo kẻ địch cho từng lượt tấn công

4.3 Kiểm tra va chạm:



Hình 2.12: Thuật toán kiểm tra va chạm cho nhân vật cũng như của kẻ địch

4.4 Giải thuật A*, dùng để tạo AI vạch đường cho kẻ địch tấn công người chơi:



Hình 2.13: Thuật toán A* áp dụng cho kẻ địch để tấn công người chơi

CHƯƠNG 3 – CÀI ĐẶT GIẢI PHÁP

1. CÀI ĐẶT KIẾN TRÚC

Khi thiết kế, phát triển và cài đặt một game, việc sắp xếp các file và thư mục một cách gọn gàng là điều không thể thiếu, vì nếu không làm vậy, sẽ tạo ra khó khăn cho người phát triển game khi gặp sự cố hay đang thao tác và làm việc. Nên cấu trúc dự án được cài đặt dựa trên cấu trúc đã thiết kế. Vì như vậy chúng ta sẽ dễ quản lý các file mã nguồn một cách hiệu quả và dễ dàng tìm kiếm vào thao tác khi cần thiết.

Tạo một thư mục cha, thư mục này sẽ chứa những file vận hành và thư mục con được phân bố theo từng mục đích khác nhau.

Các thư mục con trong thư mục cha Project theo như cấu trúc đã thiết kế:









- + Sound (folder): thư mục chứa các tập tin âm thanh và nhạc nền
- + Graphic (folder): thư mục chứa những hình ảnh sẽ được sử dụng trong game
- + Animation (folder): thư mục chứa các file chuyển động của nhân vật
- + Scripts (folder): thư mục chứa các tập tin mã nguồn của game.






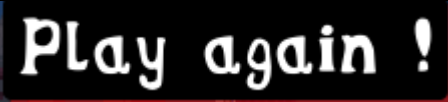

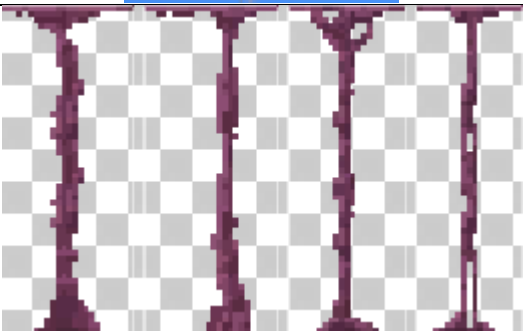
2. CÀI ĐẶT ĐỐI TƯỢNG VÀ GIAO DIỆN TRONG GAME

Trong bất cứ game nào, những đối tượng và thành phần cần thiết để thiết kế và tạo ra giao diện là điều không thể thiếu, vì vậy cần phải có dữ liệu chứa các hình ảnh dùng để tạo ra các đối tượng và giao diện cho game.

Dưới đây là danh sách các đối tượng có trong game “Sinh tồn ở hành tinh mới” được mô tả trong bảng dưới đây. Bao gồm tên của đối tượng, hình ảnh và vai trò trong trò chơi.

Bảng 3.1: Danh sách các đối tượng trong game.

STT	Tên đối tượng	Hình ảnh	Vai trò
1	Nhân vật chính		Nhân vật chính trong game, có nhiệm vụ bắn hạ kẻ địch tiến lại gần và truy tìm BOSS của kẻ địch để kết thúc cuộc chiến.
2	Kẻ địch		Kẻ địch dùng để ngăn cản và tiêu diệt người chơi
3	BackGround		BackGround, dùng để khung nền phía sau và trang trí cho game
4	BackGround		BackGround, dùng để khung nền phía sau và trang trí cho game
5	BackGround		BackGround, dùng để khung nền phía sau và trang trí cho game
6	BackGround		BackGround, dùng để khung nền phía sau và trang trí cho game
7	Tia lửa khi bắn		Khi nhân vật sử dụng vũ khí để tấn công thì sẽ xuất hiện tia lửa
8	Lối đi sang màn khác		Khi nhân vật tích lũy đủ số tiền cần thiết thì sẽ qua được màn chơi mới

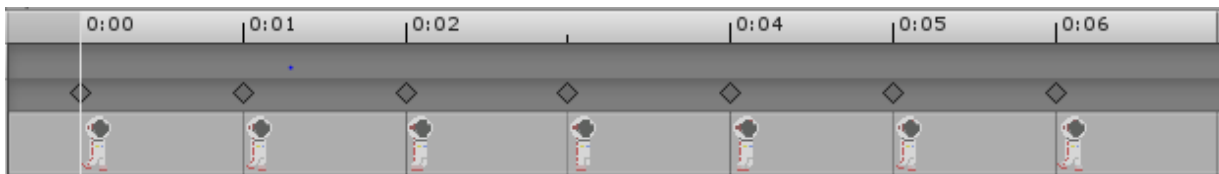
STT	Tên đối tượng	Hình ảnh	Vai trò
9	Medicine Box		Dùng để phục hồi Health cho nhân vật khi nhân vật lấy được.
10	Game Over		Khi người chơi thua sẽ xuất hiện ra
11	Retry		Button cho phép người chơi chơi lại từ đầu
12	Quit		Button cho phép người chơi thoát khỏi ứng dụng game
13	Game Win		Khi người chơi, chơi thắng game sẽ xuất hiện
14	Play again		Tác dụng như Retry
15	Particle		Khi kẻ bị tiêu diệt sẽ phát nổ. Là một hiệu ứng đơn giản được Unity cung cấp cho người sử dụng.
16	Platform		Để tạo nơi cho nhân vật duy chuyển

3. HÌNH ẢNH VÀ CHUYỂN ĐỘNG CỦA NHÂN VẬT

Khi làm game, hình ảnh nhân vật là không thể thiếu, nhưng để cho nhân vật sống động hơn cũng là một điều quan trọng không kém. Hướng giải quyết là sử dụng công cụ Animator trong Unity.

Lợi thế của Animator là có thể xuất ra nhiều Frame hình ảnh lần lượt nhau. Animator trong Unity có tính năng tạo chuyển động cho nhân vật thông qua các Sprite qua công cụ Animator, công cụ này giúp làm cho nhân vật chuyển động sống động và chân thật hơn khi nhân vật duy chuyển trái, phải, hoặc nhảy lên.

Sprite là tên gọi cho những hình ảnh đã được định dạng và thiết kế dành riêng cho Unity để có thể xử lý. Lợi thế của Sprite là có thể tùy chỉnh tùy theo pixel của hình ảnh và cắt ảnh mà không bị vướng khung nền ở phía sau.



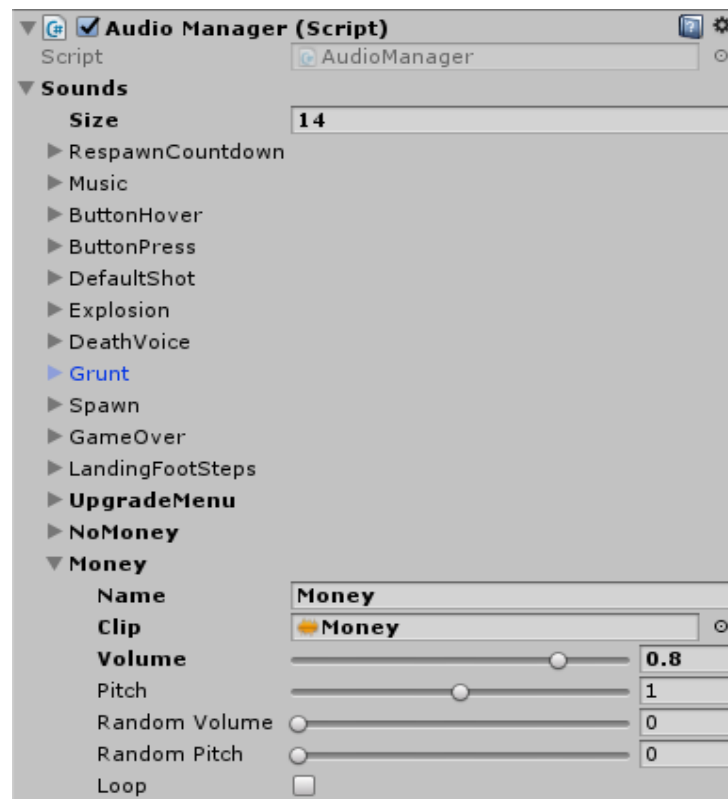
Hình 3.1: Loạt hình ảnh nhân vật duy chuyển được tạo ra thông qua công cụ Animator của Unity

4. ÂM THANH CHO GAME

Âm thanh của game là một trong những yếu tố cực kỳ quan trọng trong bất cứ game nào. Vì khi một game có âm thanh, mọi thứ trở nên chân thật, lôi cuốn và sống động hơn.

Do đó Unity cho phép người làm game có thể lồng ghép âm thanh vào một cách trực tiếp theo nhiều cách khác nhau như: sử dụng một các file âm thanh và trực tiếp kéo thả vào ứng dụng, hoặc có thể sử dụng mã văn bản(script) để thao tác và sử dụng các file âm thanh.

Tác giả đã chọn cách sử dụng chọn cách sử dụng mã can bản(script) để thao tác. Vì điều cho phép thao tác linh hoạt và có thể tùy chọn bật tắt âm thanh một cách nhanh chóng hiệu quả và không tốn quá nhiều thời gian, một điều không thể thiếu là có thể dễ dàng sửa chữa và thêm bớt âm thanh



Hình 3.2: Loạt hình ảnh nhân vật duy chuyển được tạo ra thông qua công cụ Animator của Unity

Giải thích:

- + Như trên hình ta có file Audio Manager (Script) là file mã nguồn phụ trách âm thanh cho game.
- + Trong đó Size là số lượng file âm thanh trong game.
- + Money ở cuối hình. Gồm có thêm các thông tin như Name dùng để đặt tên cho danh mục của vị trí âm thanh đó, Clip là file âm thanh đưa vào, Pitch dùng để chỉ độ lệch theo một số cố định, Random Volume dùng để tạo ra âm thanh ngẫu nhiên theo độ lớn của âm thanh dựa vào Pitch, Random Pitch dùng để tạo ra độ lệch ngẫu nhiên theo độ lớn của âm thanh.
- + Các độ lệch âm thanh thường dùng cho tiếng nổ, va chạm hoặc bắn trả kẻ địch nhằm tạo ra sự sống động hơn trong âm thanh.

CHƯƠNG 4 – KIỂM THỬ VÀ ĐÁNH GIÁ

1. MỤC TIÊU KIỂM THỬ

Nhằm xác định được sản phẩm đã hoạt động bình thường hay chưa, có còn xuất hiện lỗi chương trình hay không, quy hoạch lại các lỗi không mong muốn và khắc phục các lỗi.

Bên cạnh đó, ta cần kiểm tra lại tính chính xác của từng chức năng có trong chương trình của từng đối tượng. Xem xét lại những chức năng thực hiện có đúng như yêu cầu đề ra hay không.

2. KỊCH BẢN KIỂM THỬ

Kịch bản 1: Bắt đầu vào game, xem có lỗi hay vấn đề gì bất thường xảy ra hay không. Thực hiện duy chuyển và sử dụng vũ khí tiêu diệt kẻ địch.

Kịch bản 2: Thực hiện thao tác như: chơi lại, chơi tiếp và quay về giao diện chính.

Kịch bản 3: Vào màn chơi, thực hiện va chạm giữa các nhân vật và các chương ngại vật, quái vật, xem kết quả khi va chạm. Thực hiện các thao tác như: Thua Game, chơi lại, quay về giao diện chính

Kịch bản 4: Thực hiện chơi từ đầu đến cuối game xem còn điều gì không ổn không. Khi đến màn cuối cùng giết chết con BOSS cuối, có xuất hiện hiệu ứng thể hiện người chơi chiến thắng hay không.

Ghi chú: Môi trường kiểm thử

- Thiết bị: PC ToShiBa
- Cấu hình: Ram 8Gb, 500GB. CPU 1.8GHz. Winhdows7.
- Kết quả: Hoạt động ổn định.

3. KẾT QUẢ KIỂM THỬ

Bảng danh sách kết quả kiểm thử cho trò chơi

Bảng 4.1: Kết quả kiểm thử theo từng chức năng.

STT	Mục đích	Kết quả sản phẩm	Kết luận
1	Kiểm tra xem, nhân vật có thực hiện được một cú nhảy hay không. Khi nhân vật đang nhảy, thì ta có thể thực hiện nhảy liên tiếp khi chưa chạm đất hay không. Kiểm tra xem nhân vật có thể nhảy thấp và nhảy cao được hay không.	Nhân vật chỉ có thể nhảy lên khi đang chạm đất. Nhân vật có thể nhảy 2 lần và không thể hơn. Nhân vật chỉ có thể nhảy được một độ cao nhất định khi ta nhảy lên.	Thành công
2	Kiểm tra xem có thể chuyển sang màn khác khi đã hội đủ điều kiện cần thiết hay không. Chẳng hạn như: kiểm đủ số tiền cần thiết và số lượng kẻ địch cần tiêu diệt.	Chuyển cảnh(scene) đang chơi sang cảnh(scene) tiếp theo. Các phím chức năng hoạt động theo đúng yêu cầu.	Thành công
3	Kiểm tra va chạm giữa nhân vật và các đối tượng khác trong trò chơi, hay đạn bắn ra của nhân vật tới các đối tượng khác trong trò chơi.	Nhân vật và đạn của nhân vật chạm được với các đối tượng khác trong trò chơi. Khi va chạm với vật cản thì nhân vật sẽ bị cản lại và không duy chuyển được.	Thành công
4	Kiểm tra tổng thể game, bằng cách	Nhân vật và đạn của nhân vật chạm được với các đối tượng khác trong trò chơi. Khi va chạm với vật cản thì nhân vật sẽ bị cản lại và không duy chuyển được.	Thành công

PHẦN KẾT LUẬN

1. KẾT QUẢ ĐẠT ĐƯỢC

Về lý thuyết, tác giả đã tìm hiểu và tự học được một ngôn ngữ lập trình mới, bổ sung vốn kiến thức lập trình cho bản thân. Bên cạnh đó, tác giả còn học hỏi được rất nhiều những tiện ích khi sử dụng Unity-Engine để có thể phát triển một ứng dụng hoạt động trên PC, Android và thiết bị chạy hệ điều hành khác. Từ đó, bản thân tác giả có thể tự tạo dự án và phát triển các ứng dụng hoặc game một cách nhanh chóng.

Về chương trình, hoàn thành khóa luận, tác giả đã hoàn thành được một trò chơi bao gồm 3 màn chơi, được sắp xếp theo thứ tự từ dễ đến khó, với những chức năng đáp ứng được yêu cầu đề ra. Sản phẩm vận hành tốt trên nền tảng Windows và Android phiên bản 5.0 Lollipop trở về sau.

Khi thực hiện kiểm tra trên thiết bị thật, game hoạt động mượt mà, không xuất hiện tình trạng giật lag, âm thanh phát ra chính xác, không bị ngắt, giao diện thân thiện, dễ sử dụng, dễ thao tác. Sản phẩm còn góp phần làm phong phú hơn cho kho tàng trò chơi giải trí.

2. HẠN CHẾ

Với đề tài làm game của mình tác giả đã hoàn thành được mục tiêu đề ra và hoàn thành game với những nội dung đã được nêu. Nhưng hiện tại trò chơi vẫn còn ít màn chơi, chưa thật sự đủ để làm khó người chơi và làm người chơi hài lòng.

Trò chơi theo hướng truyền thống, mang tính giải trí nhẹ, chưa thực sự hấp dẫn đông đảo số lượng người chơi.

3. HƯỚNG PHÁT TRIỂN

Tác giả sẽ tiếp tục phát triển game, bổ sung các hình ảnh chi tiết, các hiệu ứng âm thanh, thêm các phần tiếp theo, bổ sung thêm câu chuyện nhằm tăng tính hấp dẫn cho game.

Thêm các màn chơi và thiết kế thêm nhiều hình ảnh hơn, nhất là hình ảnh về vũ khí, hình dáng nhân vật và kẻ địch.

Thiết kế thêm nhiều vũ khí cho người chơi, nhằm tăng tính hấp dẫn và đa dạng trong các lối chơi.

Thiết kế và phát triển kẻ địch trở nên đa dạng hơn nữa, thêm vào các kẻ địch tấn công tầm xa và các hiệu ứng cho kẻ địch.

Xây dựng cốt truyện trở nên có cơ sở hơn và hấp dẫn hơn, để cho người chơi có một mục tiêu và mục đích rõ ràng khi nhập vai vào một nhân vật trong game.

TÀI LIỆU THAM KHẢO

[1] **TS. Huỳnh Xuân Hiệp, Th.S Phan Phương Lan.** GIÁO TRÌNH Nhập môn CÔNG NGHỆ PHẦN MỀM. Nhà xuất bản Đại học Cần Thơ 2011.

[2] Website hướng dẫn chính thức về lý thuyết scripting của Unity. Truy cập lần cuối 4/12/2018. Website: <https://docs.unity3d.com/ScriptReference/>

[3] Website hướng dẫn chính bằng video chính thức của Unity từ dễ đến khó. Truy cập lần cuối 3/12/2018. Website: <https://unity3d.com/learn/tutorials/s/scripting>