

BỘ GIÁO DỤC VÀ ĐÀO TẠO BỘ NÔNG NGHIỆP VÀ PTNT
TRƯỜNG ĐẠI HỌC THỦY LỢI



VŨ TIẾN MẠNH

XÂY DỰNG GAME “XUYÊN KHÔNG THÀNH NINJA”
BẰNG UNITY

ĐỒ ÁN TỐT NGHIỆP

HÀ NỘI, NĂM 2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO BỘ NÔNG NGHIỆP VÀ PTNT
TRƯỜNG ĐẠI HỌC THỦY LỢI

VŨ TIẾN MẠNH

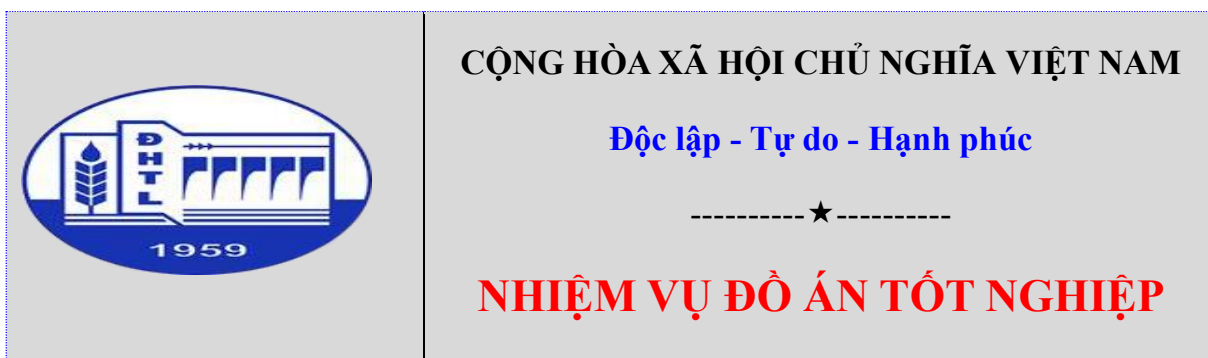
**XÂY DỰNG GAME “XUYÊN KHÔNG THÀNH NINJA” BẰNG
UNITY**

Ngành: Công nghệ thông tin

Mã số: 7480201

NGƯỜI HƯỚNG DẪN ThS. Trương Xuân Nam

HÀ NỘI, NĂM 2024



Họ tên sinh viên: Vũ Tiến Mạnh

Hệ đào tạo: Đại học chính quy

Lớp: 62TH5

Ngành: Công nghệ thông tin

Khoa: Công nghệ thông tin

**1. TÊN ĐỀ TÀI: XÂY DỰNG GAME “XUYÊN KHÔNG THÀNH NINJA”
BẰNG UNITY**

2. NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN:

Nội dung cần thuyết minh	Tỷ lệ
1. Mở đầu	2%
2. Chương I. Khám phá tổng quan về ngành công nghiệp game	4%
3. Chương II. Tìm hiểu quy trình phát triển game sử dụng Unity	44%
4. Chương III. Triển khai quá trình xây dựng game với Unity	15%
5. Chương IV: Phân tích thiết kế hệ thống	19%
6. Chương V: Xây dựng game và demo	6%
7. Chương VI: Kết luận và định hướng phát triển	2%

3. GIÁO VIÊN HƯỚNG DẪN TỪNG PHẦN

Các phần	GVHD
1. Mở đầu	TS. Trương Xuân Nam
2. Chương I. Khám phá tổng quan về ngành công nghiệp game	TS. Trương Xuân Nam
3. Chương II. Tìm hiểu quy trình phát triển game sử dụng Unity	TS. Trương Xuân Nam
4. Chương III. Triển khai quá trình xây dựng game với Unity	TS. Trương Xuân Nam
5. Chương IV: Phân tích thiết kế hệ thống	TS. Trương Xuân Nam
6. Chương V: Xây dựng game và demo	TS. Trương Xuân Nam
7. Chương VI: Kết luận và định hướng phát triển	TS. Trương Xuân Nam

4. NGÀY GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Ngày tháng năm 202...

Trưởng Bộ môn

(Ký và ghi rõ Họ tên)

Giáo viên hướng dẫn chính

(Ký và ghi rõ Họ tên)

Nhiệm vụ Đồ án tốt nghiệp đã được Hội đồng thi tốt nghiệp của Khoa thông qua.

Ngày. . . .tháng. . . .năm 202...

Chủ tịch Hội đồng

(Ký và ghi rõ Họ tên)

Sinh viên đã hoàn thành và nộp bản Đồ án tốt nghiệp cho Hội đồng thi ngày.....
tháng... năm 2024

Sinh viên làm Đồ án tốt nghiệp

(Ký và ghi rõ Họ tên)



TRƯỜNG ĐẠI HỌC THUY LỢI
KHOA CÔNG NGHỆ THÔNG TIN

BẢN TÓM TẮT ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP

Tên đề tài: Xây dựng game “Xuyên không thành ninja” bằng Unity

Sinh viên thực hiện: Vũ Tiến Mạnh

Lớp: 62TH5

Mã sinh viên: 2051063977

Số điện thoại: 0397728656

Email: 2051063977@e.tlu.edu.vn

Giảng viên hướng dẫn: ThS. Trương Xuân Nam

TÓM TẮT ĐỀ TÀI

Ngày nay, với sự phát triển vượt bậc của công nghệ số, game thủ toàn cầu không ngừng tìm kiếm những trò chơi không chỉ giải trí mà còn phải có thử thách, sự hấp dẫn và đặc biệt là tạo cảm giác chân thực. Đó chính là lý do khiến em lựa chọn đề tài game nhập vai ninja trong một thế giới giả tưởng, nơi người chơi vào vai một nhân vật hiện đại bị cuốn vào một không gian cổ xưa của các ninja. Trong hành trình này, người chơi phải học cách thích nghi, sinh tồn, rèn luyện kỹ năng chiến đấu và tìm cách trở về thế giới thực. Khác biệt với nhiều tựa game nhập vai khác, trò chơi này được thiết kế cho chế độ chơi đơn, không phụ thuộc vào kết nối mạng, giúp người chơi có thể tập trung hoàn toàn vào câu chuyện mà không bị gián đoạn bởi các yếu tố kỹ thuật như độ trễ hay cấu hình máy.

Gameplay của trò chơi sẽ cho phép người chơi di chuyển theo nhiều hướng, khám phá môi trường và sử dụng các kỹ năng ninja đặc trưng như ẩn nấp, di chuyển nhanh,

chiến đấu cận chiến với katana hay tấn công từ xa bằng shuriken hoặc cung gỗ. Người chơi cũng có thể nâng cấp kỹ năng dựa trên phong cách của mình, tạo nên một hành trình độc đáo và hấp dẫn. Các NPC (nhân vật không phải người chơi) trong game cùng với thiết kế cảnh quan phong phú sẽ mang lại những thử thách sống động, giúp người chơi hoàn toàn đắm mình vào thế giới ninja đầy kịch tính. Ngoài ra, yếu tố âm thanh và đồ họa được đầu tư kỹ lưỡng, tái hiện bầu không khí huyền bí của thế giới cổ xưa.

Tóm lại, trò chơi nhập vai ninja này là một lựa chọn hoàn hảo cho những ai yêu thích sự phiêu lưu, thử thách và trải nghiệm một thế giới giả tưởng sâu sắc.

Với tất cả những đặc điểm thú vị trên, em quyết định chọn đề tài này để phát triển.

Công nghệ sử dụng:

- Unity.
- Visual Studio

CÁC MỤC TIÊU CHÍNH

Mục tiêu:

- Nhân vật di chuyển theo 2 hướng trước và sau, nhảy lên và camera sẽ di chuyển theo hướng mà nhân vật di chuyển.
- Xây dựng bản đồ game cơ bản gồm: khung cảnh, vật cản, NPC (nhân vật không phải người chơi), quái vật.
- Nhân vật có thể tương tác với NPC và đánh được quái vật.
- Nhân vật có thể đổi vũ khí từ cận chiến như kiếm, đao.. sang đánh xa như cung, phi tiêu... .
- Nhân vật có thể nhặt đồ rơi ra khi đánh quái vật.
- Máu nhân vật trở về 0 nhân vật sẽ chết.
- Quái vật tấn công nhân vật khi phát hiện nhân vật.
- Quái vật có thể sẽ rơi ra vàng, vật phẩm... khi chết.

Kỹ Năng

- Sinh viên tìm hiểu về game học hỏi được những kiến thức lập trình game Unity, C#.
- Sinh viên nắm bắt được quy trình xây dựng tạo ra một game.

KẾT QUẢ DỰ KIẾN

- Xây dựng được kỹ năng viết báo cáo tổng hợp về Unity.
- Thu thập yêu cầu, nghiệp vụ bài toán để đưa ra các phương án giải quyết phù hợp.
- Phát triển trên Unity: Lập trình các chức năng chính bằng C#.
- Kiểm thử và hoàn thiện: Kiểm tra tính ổn định và tính thú vị của trò chơi.
- Xây dựng được một trò chơi thú vị.

LỜI CAM ĐOAN

Tôi xin cam đoan đây là Đồ án tốt nghiệp của bản thân tôi. Các kết quả trong Đồ án tốt nghiệp này là trung thực, và không sao chép từ bất kỳ một nguồn nào và dưới bất kỳ hình thức nào. Việc tham khảo các nguồn tài liệu (nếu có) đã được thực hiện trích dẫn và ghi nguồn tài liệu tham khảo đúng quy định.

Tác giả ĐATN

Vũ Tiến Mạnh

LỜI CẢM ƠN

Đầu tiên em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến thầy Ths. Trương Xuân Nam, ngành Công Nghệ Phần Mềm, khoa Công Nghệ Thông Tin, trường Đại Học Thủy Lợi, người đã tận tình hướng dẫn, chỉ bảo hỗ trợ và cho em những lời khuyên, góp ý trong suốt quá trình thực hiện đồ án tốt nghiệp này những kiến thức quý báu đó không chỉ giúp em hoàn thành đồ án của mình một cách tốt nhất mà còn là hành trang giá trị cho sự nghiệp của em sau này.

Em cũng xin gửi lời cảm ơn đến tập thể thầy cô trong khoa Công Nghệ Thông Tin nói riêng và toàn thể giảng viên trường Đại Học Thủy Lợi nói chung. Những kiến thức và kỹ năng được các thầy cô tận tâm truyền đạt đã tạo cho em nền tảng vững chắc, giúp em có thể tự tin thực hiện đồ án tốt nghiệp cũng như chuẩn bị cho hành trình sự nghiệp sắp tới.

Cuối cùng, em xin bày tỏ lòng biết ơn sâu sắc đến gia đình, bạn bè và những người đồng hành đã luôn bên cạnh, động viên và cổ vũ em trong suốt quãng thời gian học tập và thực hiện đồ án tại Đại Học Thủy Lợi.

Em xin chân thành cảm ơn!

MỤC LỤC

LỜI CAM ĐOAN	i
LỜI CẢM ƠN.....	ii
MỤC LỤC	iii
DANH MỤC CÁC HÌNH ẢNH.....	vi
DANH MỤC BẢNG BIỂU	vii
DANH MỤC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ'	viii
MỞ ĐẦU	1
1. Đặt vấn đề	1
2. Bố cục của đề án	3
CHƯƠNG I: KHÁM PHÁ TỔNG QUAN VỀ NGÀNH CÔNG NGHIỆP GAME	4
1. Lịch sử phát triển của ngành game	4
2. Công cụ phát triển game	4
3. Ngôn ngữ lập trình trong phát triển game.....	5
4. Các thể loại và phong cách game.....	5
5. Xu hướng và tiềm năng của ngành công nghiệp game	6
6. Đối tượng người chơi và nhu cầu thị trường	6
CHƯƠNG II: TÌM HIỂU VỀ QUY TRÌNH PHÁT TRIỂN GAME SỬ DỤNG UNITY	7
1. Lập trình game là gì?	7
2. Quy trình lập trình game cơ bản	7
2.1. Giai đoạn tiền sản xuất (Pre-production)	8
2.2. Giai đoạn sản xuất (Production)	9
2.3. Giai đoạn hậu kỳ (Post-production)	10
3. Cơ hội nghề nghiệp trong ngành lập trình game	10
4. Giới thiệu về Unity	11
5. Các tính năng và ưu điểm của Unity	12
6. Các bước cơ bản trong quy trình phát triển game với Unity	13
7. Tại sao chọn Unity cho phát triển game nhập vai ninja	14

8. Các thành phần trong Unity Editor.....	15
8.1. Cửa sổ Scene	15
8.2. Cửa sổ Hierarchy	15
8.3. Cửa sổ Game	16
8.4. Cửa sổ Project.....	17
8.5. Cửa sổ Inspector	18
9. Các khái niệm cơ bản trong Unity	19
9.1. GameObject	19
9.2. Component.....	19
9.3. Sprite.....	20
9.4. Animation	20
9.5. Key Frame	20
9.6. Prefabs	20
9.7. Sounds	20
9.8. Script.....	20
9.9. Scenes	21
9.10. Assets.....	21
9.11. Camera.....	21
9.12. Transform	21
9.13. Material và Shader.....	21
10. Các hàm API phổ biến trong Unity	21
10.1. MonoBehaviour	21
10.2. Transform	22
10.3. Rigidbody	22
10.4. Collider	22
10.5. Coroutine	22
CHƯƠNG III: TRIỂN KHAI QUÁ TRÌNH XÂY DỰNG GAME VỚI UNITY	23
1. Các yếu tố làm nên game.....	23
2. Phân tích thiết kế game.....	25
2.1. Đồ họa.....	25

2.2. Âm thanh	25
2.3. Giao diện.....	25
2.4. Điều khiển.....	25
2.5. Nhiệm vụ và Màn chơi	25
3. Kiểm thử và bảo trì game	26
3.1. Kiểm thử	26
3.2. Bảo trì	26
CHƯƠNG IV: PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....	27
1. Xác định tác nhân	27
2.Xây dựng biểu đồ Usecase	28
2.1.Mô hình UseCase tổng quát	28
3. Danh sách các Use Case	28
4. Đặc tả UseCase	30
4.1 .Use case tấn công player:	30
4.2. Use case xử lý va chạm vật lý:	32
4.3. Use case điều khiển di chuyển nhân vật:	33
4.4.Use case tiêu diệt NPC:	35
CHƯƠNG V: XÂY DỰNG GAME VÀ DEMO.....	37
CHƯƠNG VI: KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN	42
TÀI LIỆU THAM KHẢO	43

DANH MỤC CÁC HÌNH ẢNH

Hình 2.1: ảnh quy trình cơ bản của lập trình game	8
Hình 2.2: Logo Unity.....	11
Hình 2.3: Các nền tảng hỗ trợ của Unity	13
Hình 2.4: Giao diện của Unity.....	14
Hình 2.5: Cửa sổ Scene	15
Hình 2.6: Cửa sổ Hierarchy	16
Hình 2.7: Cửa sổ Game	17
Hình 2.8: Cửa sổ Project.....	18
Hình 2.9: Cửa sổ Inspector	19
Hình 4.1: Mô hình UseCase tổng quát	28
Hình 4.2: Sơ đồ Activity Use Case tấn công Player.....	31
Hình 4.3: Sơ đồ tuần tự Use Case tấn công Player.....	31
Hình 4.4: Sơ đồ Activity Use Case xử lý va chạm.....	32
Hình 4.5: Sơ đồ tuần tự Use Case xử lý va chạm.....	33
Hình 4.6: Sơ đồ Activity Use Case điều khiển di chuyển nhân vật.....	34
Hình 4.7: Sơ đồ tuần tự Use Case di chuyển nhân vật	35
Hình 4.8: Sơ đồ Activity Use Case tiêu diệt Enemy	36
Hình 4.9: Sơ đồ tuần tự Use Case tiêu diệt Enemy	36
Hình 5.1: Màn hình chính.....	37
Hình 5.2: Màn hình hội thoại	37
Hình 5.3: Màn hình Gameplay	38
Hình 5.4: Màn hình thua.....	38
Hình 5.5: Màn hình dừng game.....	39
Hình 5.6: PLayer.....	39
Hình 5.7: Quái vật loại 1	40
Hình 5.8: Quái vật loại 2	40
Hình 5.9: Quái vật Boss.....	41

DANH MỤC BẢNG BIỂU

Bảng 4.1: Kịch bản Use Case Tấn công player	30
Bảng 4.2: Kịch bản Use Case xử lý va chạm	32
Bảng 4.3: Kịch bản Use Case điều khiển di chuyển nhân vật.....	33
Bảng 4.4: Kịch bản Use Case tiêu diệt Enemy.....	35

DANH MỤC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ

Từ viết tắt	Từ đầy đủ	Nghĩa tiếng việt
NPC	Not Player Character	Nhân vật không phải người chơi
BM	Bộ môn	
GVHD	Giáo viên hướng dẫn	
SV	Sinh viên	

MỞ ĐẦU

1. Đặt vấn đề

Trong thế giới game hiện nay, thể loại nhập vai đã trở thành một trong những thể loại nổi bật và thu hút, mang lại cho người chơi cảm giác sống động và trải nghiệm hoàn hảo khi hóa thân thành các nhân vật hư cấu. Một trong những chủ đề được yêu thích trong thể loại này là game ninja, đưa người chơi vào vai những chiến binh tinh nhuệ trong một bối cảnh cổ xưa đậm chất huyền bí. Trò chơi nhập vai ninja không chỉ khai thác các kỹ năng chiến đấu mà còn đi sâu vào yếu tố phiêu lưu, giải mã các bí mật ẩn giấu, tạo nên một hành trình đầy cuốn hút. Đây là nơi người chơi không chỉ đối mặt với những thử thách cam go mà còn trải nghiệm sự hồi hộp khi di chuyển trong một thế giới giả tưởng, nơi những bí mật và hiểm nguy luôn chực chờ.

Trò chơi ninja với góc nhìn ngang mở ra không gian tương tác độc đáo, cho phép người chơi dễ dàng quan sát và làm chủ từng hành động của nhân vật. Góc nhìn này giúp người chơi không chỉ có khả năng bao quát không gian mà còn tận hưởng từng khung cảnh và cảm nhận tính chân thực của môi trường xung quanh. Từng bước di chuyển, từng pha chiến đấu được thể hiện kiết, rõ nét, giúp người chơi dễ dàng đắm mình vào vai trò của một ninja thực thụ. Với công cụ Unity, trò chơi có thể đạt được chất lượng đồ họa sắc nét và phong cách thiết kế cổ điển nhưng vẫn mang hơi hướng hiện đại, đáp ứng nhu cầu thẩm mỹ và mang lại cảm giác mãn nhãn qua từng khung hình. Những hiệu ứng chuyển động, âm thanh và hình ảnh chân thực khiến cho từng màn chơi trở nên sinh động và hấp dẫn hơn bao giờ hết.

Điểm đặc biệt của trò chơi không chỉ nằm ở các pha hành động hay chiến đấu mà còn ở sự phát triển nhân vật một cách linh hoạt và tự nhiên. Người chơi sẽ được trải nghiệm quá trình nâng cấp kỹ năng của nhân vật ninja thông qua các nhiệm vụ khác nhau, từ những kỹ thuật chiến đấu cận chiến với katana, shuriken cho đến các đòn đánh bất ngờ trong tình huống nguy hiểm. Kỹ năng di chuyển nhanh nhẹn, leo trèo qua các địa hình hiểm trở, và khả năng ẩn nấp là những yếu tố quan trọng giúp nhân

vật vượt qua các chướng ngại và thử thách khó khăn. Sự phát triển nhân vật không ngừng khiến người chơi cảm thấy mình thực sự là một phần của hành trình, từng bước trở nên mạnh mẽ hơn, khéo léo hơn, và có khả năng ứng phó với mọi tình huống. Việc này không chỉ tăng thêm tính thử thách mà còn mang lại cảm giác thành tựu cho người chơi khi thấy sự tiến bộ rõ rệt qua mỗi màn chơi.

Trong thế giới ninja cổ xưa, bối cảnh được xây dựng một cách chi tiết và đa dạng, từ những ngôi làng cổ kính, những rừng rậm sâu thẳm cho đến các đền thờ linh thiêng đầy bí ẩn. Mỗi địa điểm trong trò chơi không chỉ là bối cảnh mà còn đóng vai trò tương tác, giúp người chơi khám phá các con đường ẩn, nơi ẩn nấp hay các cơ hội tấn công bất ngờ. Các yếu tố môi trường này giúp tạo nên những tình huống chiến đấu đa dạng và đầy sáng tạo, cho phép người chơi tận dụng địa hình để vượt qua các thử thách. Những thiết kế môi trường không chỉ là phần nền, mà còn góp phần thúc đẩy chiến thuật của người chơi, từ đó làm tăng thêm tính chân thực và phong phú cho trò chơi.

Một yếu tố cốt lõi làm nên sức hấp dẫn của trò chơi là cốt truyện. Trò chơi ninja này không chỉ dừng lại ở các nhiệm vụ chiến đấu mà còn đi sâu vào các câu chuyện về gia tộc, các di sản cổ xưa, và những bí mật huyền bí. Người chơi sẽ hóa thân thành ninja, thực hiện những nhiệm vụ liên quan đến bảo vệ danh dự của gia tộc, khám phá những kho báu bị lãng quên, và đối đầu với những kẻ thù nguy hiểm, những kẻ luôn muốn nắm giữ những bí mật của thế giới này. Các tình huống trong trò chơi không chỉ là các thử thách mà còn là những phần quan trọng của một câu chuyện sâu sắc, đưa người chơi qua các nhiệm vụ liên tiếp, khiến họ cảm thấy mình không chỉ đang chơi một trò chơi mà còn đang sống trong một hành trình phiêu lưu. Sự kết hợp này giúp người chơi dễ dàng hòa mình vào nhân vật và cảm nhận sâu sắc về vai trò của mình trong thế giới ninja đầy huyền bí.

Không chỉ có cốt truyện và yếu tố nhập vai, trò chơi còn đặc biệt chú trọng vào phần âm thanh và hình ảnh, hai yếu tố không thể thiếu để tạo nên không khí và cảm giác thực sự hồi hộp. Hiệu ứng âm thanh trong game tái hiện sống động từng bước di chuyển của nhân vật, tiếng chạm kiếm đầy kịch tính, tiếng lá cây xào xạc khi ninja ẩn

nấp hay thậm chí là nhịp thở dồn dập khi nhân vật đang gặp nguy hiểm. Âm nhạc nền cũng được lựa chọn kỹ lưỡng để mang đậm không khí cổ điển và bí ẩn, giúp người chơi dễ dàng đắm chìm vào không gian đầy tính nghệ thuật của thế giới ninja. Đồ họa sắc nét, các nhân vật và khung cảnh được thiết kế tỉ mỉ, từ ki tiết của từng ngôi đền bỏ hoang cho đến các khu rừng rậm rạp, khiến người chơi như đang thực sự di chuyển trong một không gian cổ đại đầy lôi cuốn.

Trò chơi nhập vai ninja với góc nhìn ngang là một sự kết hợp hoàn hảo giữa các yếu tố chiến đấu, phiêu lưu, và cốt truyện sâu sắc. Những chi tiết về phát triển nhân vật, các thử thách đa dạng, và sự tương tác với môi trường xung quanh khiến trò chơi trở nên hấp dẫn và tạo nên sự kết nối chặt chẽ giữa người chơi và nhân vật. Mỗi hành động, mỗi thử thách, và mỗi tình huống đều đóng góp vào hành trình khám phá và chinh phục, tạo nên một trải nghiệm độc đáo, thu hút và đầy cảm xúc. Với cách thiết kế này, trò chơi không chỉ là một sản phẩm giải trí mà còn là một hành trình khám phá, nơi người chơi có thể thử thách bản thân, khám phá các bí mật và tìm kiếm danh dự cho nhân vật của mình trong thế giới ninja huyền bí và đầy lôi cuốn.

2. Bố cục của đồ án

Các nhiệm vụ chính trong đồ án bao gồm:

1. Chương I: Khám phá tổng quan về ngành công nghiệp game.
2. Chương II: Tìm hiểu quy trình phát triển game sử dụng Unity.
3. Chương III: Triển khai quá trình xây dựng game với Unity.
4. Chương IV: Thực hiện phân tích và thiết kế hệ thống cho game.
5. Chương V: Hoàn thiện và trình bày demo game.
6. Chương VI: Đưa ra kết luận và định hướng phát triển tương lai.

CHƯƠNG I: KHÁM PHÁ TỔNG QUAN VỀ NGÀNH CÔNG NGHIỆP GAME

1. Lịch sử phát triển của ngành game

Lịch sử phát triển của ngành công nghiệp game là một hành trình đáng kể, kéo dài từ những năm 1950 đến nay. Đầu tiên là các trò chơi đơn giản như Tic-Tac-Toe hay Pong, đến những tựa game arcade cổ điển trong thập niên 70 và 80 như Pac-Man, Space Invaders, đã mở ra thời kỳ huy hoàng cho game điện tử. Sự xuất hiện của các hệ máy chơi game như NES của Nintendo vào thập niên 1980, PlayStation vào những năm 90 đã giúp ngành công nghiệp game trở thành một phần của đời sống giải trí toàn cầu. Với sự phát triển của công nghệ 3D, vào cuối thập niên 90 và đầu 2000, các tựa game với đồ họa sống động và thế giới mở như GTA, The Legend of Zelda đã đem đến những trải nghiệm mới lạ cho người chơi. Ngày nay, ngành công nghiệp game đang chứng kiến sự bùng nổ mạnh mẽ nhờ vào các công nghệ như VR (thực tế ảo), AR (thực tế tăng cường), và trí tuệ nhân tạo, mang lại những trải nghiệm chân thực hơn bao giờ hết. Mỗi giai đoạn lịch sử đều để lại dấu ấn riêng, góp phần định hình cách chúng ta chơi và tiếp cận game hiện nay.

2. Công cụ phát triển game

Các công cụ phát triển game đóng vai trò then chốt trong việc hiện thực hóa ý tưởng của các nhà làm game. Một số công cụ phổ biến hiện nay bao gồm:

- **Unity:** Unity là một trong những công cụ phát triển game đa nền tảng mạnh mẽ nhất hiện nay, nổi bật với khả năng xây dựng các game 2D và 3D. Unity cung cấp một kho thư viện phong phú, tài nguyên cộng đồng, và tính năng mạnh mẽ cho phép nhà phát triển dễ dàng triển khai ý tưởng từ trò chơi đơn giản đến phức tạp.
- **Unreal Engine:** Với khả năng hỗ trợ đồ họa cao cấp, Unreal Engine được nhiều studio lớn lựa chọn để phát triển các trò chơi với chất lượng đồ họa chân thực. Unreal Engine thường được dùng cho các game AAA, với các hiệu ứng vật lý phức tạp và ánh sáng thực tế, mang lại các trò chơi có đồ họa vượt trội.

- **Godot:** Đây là một công cụ mã nguồn mở, mạnh mẽ và hoàn toàn miễn phí, nổi bật với giao diện thân thiện và hỗ trợ tốt cho game 2D. Godot đang ngày càng phổ biến trong cộng đồng indie vì sự tiện dụng và khả năng tùy biến cao. Các công cụ này có sự khác biệt về tính năng, tài nguyên, và cách tiếp cận, nhưng đều cho phép nhà phát triển tận dụng các ưu điểm riêng biệt của chúng để tạo ra sản phẩm ấn tượng, phù hợp với nhu cầu và quy mô dự án.

3. Ngôn ngữ lập trình trong phát triển game

Ngôn ngữ lập trình là một yếu tố thiết yếu trong phát triển game, quyết định cách game hoạt động và phản hồi người chơi.

- **C#:** Được sử dụng chủ yếu trong Unity, C# là ngôn ngữ lập trình dễ học, dễ tiếp cận với các lập trình viên mới. Với cú pháp rõ ràng và tính năng hỗ trợ mạnh mẽ, C# giúp lập trình viên dễ dàng viết các đoạn mã hiệu quả cho các trò chơi từ đơn giản đến phức tạp.
 - **C++:** Thường được sử dụng trong Unreal Engine, C++ cho phép lập trình viên kiểm soát sâu hơn về bộ nhớ và hiệu suất, giúp tối ưu hóa các trò chơi với yêu cầu cao về đồ họa và tốc độ xử lý.
 - **JavaScript và Python:** JavaScript phổ biến trong các game chạy trên nền tảng web, còn Python với cú pháp thân thiện thường được dùng cho các ứng dụng prototyping hoặc game casual.
- Mỗi ngôn ngữ có những ưu điểm riêng, và việc lựa chọn ngôn ngữ phụ thuộc vào yêu cầu của dự án cũng như công cụ phát triển mà lập trình viên sử dụng.

4. Các thể loại và phong cách game

Trong ngành công nghiệp game, có nhiều thể loại và phong cách khác nhau nhằm đáp ứng nhu cầu đa dạng của người chơi.

- **Game bắn súng:** Thể loại phổ biến với các trò chơi có góc nhìn thứ nhất và thứ ba, cho phép người chơi điều khiển nhân vật và sử dụng các loại vũ khí để hoàn thành mục tiêu.

- **Game phiêu lưu:** Đòi hỏi người chơi khám phá thế giới, giải đố và tương tác với nhân vật.
- **Game nhập vai:** Người chơi hóa thân thành nhân vật chính, trải nghiệm cốt truyện và phát triển nhân vật qua từng thử thách.
- **Phong cách chơi góc nhìn ngang:** Với góc nhìn ngang, người chơi có thể dễ dàng quan sát toàn cảnh và môi trường xung quanh nhân vật, đặc biệt phù hợp với các trò chơi hành động, phiêu lưu. Góc nhìn ngang tạo nên một trải nghiệm mượt mà và dễ theo dõi.

5. Xu hướng và tiềm năng của ngành công nghiệp game

Hiện nay, ngành game đang hướng đến các xu hướng mới như VR, AR, và trí tuệ nhân tạo (AI). VR và AR đưa người chơi vào thế giới game chân thực, tương tác cao, trong khi AI cải thiện trải nghiệm qua việc tạo ra các đối thủ và môi trường thông minh hơn. Game di động cũng phát triển mạnh mẽ, giúp game tiếp cận với nhiều người chơi ở mọi lứa tuổi. Những công nghệ này không chỉ nâng cao trải nghiệm của người chơi mà còn mở ra tiềm năng lớn cho ngành công nghiệp game phát triển trong tương lai.

6. Đối tượng người chơi và nhu cầu thị trường

Đối tượng người chơi ngày càng đa dạng, từ trẻ nhỏ đến người lớn, từ người chơi thông thường đến game thủ chuyên nghiệp. Nhu cầu thị trường hiện nay tập trung vào trải nghiệm chơi mượt mà, cốt truyện hấp dẫn và gameplay sáng tạo. Các trò chơi tích hợp yếu tố xã hội, cho phép người chơi giao tiếp và kết nối với nhau, cũng được ưa chuộng.

CHƯƠNG II: TÌM HIỂU VỀ QUY TRÌNH PHÁT TRIỂN GAME SỬ DỤNG UNITY

1. Lập trình game là gì?

Lập trình game không chỉ là việc viết mã đơn thuần mà còn là cả một nghệ thuật sáng tạo, nơi những ý tưởng và khái niệm được biến thành những trò chơi sống động trên các thiết bị khác nhau như **PC, điện thoại di động, máy chơi game cầm tay, và các nền tảng trực tuyến**. Lập trình game bao gồm nhiều giai đoạn và công đoạn khác nhau từ lên ý tưởng, thiết kế đồ họa, phát triển gameplay, lập trình AI cho đến thử nghiệm và phát hành.

Những người đảm nhiệm vai trò phát triển game được gọi là **Game Developers**. Họ chính là “nhạc trưởng” của dàn nhạc công nghệ, biến các ý tưởng tưởng chừng như bất khả thi thành hiện thực. Công việc của họ không chỉ đòi hỏi sự am hiểu sâu sắc về lập trình mà còn cần có kỹ năng về **thiết kế game, tư duy logic**, và đặc biệt là **khả năng sáng tạo không giới hạn**. Từ những dòng mã, họ tạo nên các yếu tố như **nhân vật, bối cảnh, hiệu ứng ánh sáng, âm thanh**, và đặc biệt là **trải nghiệm người chơi**.

Thông qua việc sử dụng các công nghệ như **Unity, Unreal Engine**, cùng với các ngôn ngữ lập trình như **C#, C++, Python**, các lập trình viên game có thể tạo ra **thế giới ảo** sống động với đồ họa và âm thanh chân thực. Việc lập trình các yếu tố như **AI cho nhân vật, các vật lý va chạm, và các hiệu ứng đặc biệt** giúp mang đến những trải nghiệm hấp dẫn, thu hút người chơi khám phá và trải nghiệm game.

2. Quy trình lập trình game cơ bản

Để phát triển một tựa game hoàn chỉnh từ ý tưởng đến sản phẩm cuối cùng, đội ngũ phát triển game thường tuân thủ một quy trình tiêu chuẩn gồm **ba giai đoạn chính: Tiền sản xuất (Pre-production), Sản xuất (Production) và Hậu kỳ (Post-production)**. Mỗi giai đoạn đều đóng vai trò quan trọng và cần thiết để đảm bảo chất lượng và thành công của dự án.



Hình 2.1: ảnh quy trình cơ bản của lập trình game

2.1. Giai đoạn tiền sản xuất (Pre-production)

Giai đoạn tiền sản xuất là **bước nền tảng** cho mọi dự án game, nơi mọi ý tưởng được định hình và phát triển trước khi chuyển sang giai đoạn sản xuất thực sự. Đây là giai đoạn đòi hỏi sự chuẩn bị kỹ lưỡng và có sự tham gia của nhiều bộ phận khác nhau:

- **Phát triển ý tưởng và cốt truyện:** Đây là bước khởi đầu quan trọng, nơi nhóm phát triển cùng nhau brainstorm để đưa ra **một ý tưởng độc đáo** cho trò chơi. Với tựa game ninja xuyên không mà bạn đang phát triển, cốt truyện có thể xoay quanh một **ninja bị cuốn vào cuộc hành trình xuyên không gian và thời gian**, chiến đấu với kẻ thù để bảo vệ thế giới khỏi các thế lực đen tối.
- **Thiết kế nhân vật và bối cảnh:** Sau khi xác định được cốt truyện, đội ngũ thiết kế sẽ tiến hành phác thảo các **nhân vật chính, kẻ thù, và các bối cảnh** trong game. Điều này bao gồm từ **trang phục của ninja**, các loại vũ khí như

phi tiêu, kiếm katana, cho đến các môi trường như rừng rậm, thành phố cổ đại và các màn chơi đầy thử thách.

- **Xây dựng tài liệu thiết kế game (Game Design Document - GDD):** Tài liệu này là **kim chỉ nam** cho toàn bộ quá trình phát triển game, mô tả chi tiết các yếu tố như **cơ chế gameplay, hệ thống điều khiển, và các nhiệm vụ chính**. GDD giúp đảm bảo rằng mọi thành viên trong nhóm phát triển đều hiểu rõ về **tầm nhìn của dự án**.
- **Lập kế hoạch chi phí và thời gian:** Việc lập kế hoạch rõ ràng về **thời gian phát triển, nguồn lực và chi phí** giúp dự án diễn ra suôn sẻ và đúng tiến độ. Đối với game ninja của bạn, việc ước tính chi phí cho việc phát triển các yếu tố như **đồ họa pixel art, âm thanh, và hiệu ứng đặc biệt** là rất quan trọng.

2.2. Giai đoạn sản xuất (Production)

Giai đoạn sản xuất là nơi tất cả các ý tưởng được hiện thực hóa, và đây là giai đoạn **tiêu tốn nhiều thời gian và nguồn lực nhất**:

- **Lập trình gameplay và cơ chế điều khiển:** Tại giai đoạn này, nhóm lập trình sẽ triển khai các yếu tố gameplay như **di chuyển, nhảy, và tấn công** cho nhân vật chính. Đối với game ninja của bạn, người chơi có thể sử dụng các **phím A và D để di chuyển, Space để nhảy, và chuột hoặc phím khác để tung đòn tấn công**.
- **Tạo đồ họa và hiệu ứng âm thanh:** Các nghệ sĩ đồ họa sẽ xây dựng các **nhân vật, kẻ thù, bối cảnh, và hiệu ứng đặc biệt** như **ánh sáng, khói lửa, và các đòn tấn công** của ninja. Đồng thời, các kỹ sư âm thanh sẽ thiết kế **nhạc nền, hiệu ứng âm thanh cho các kỹ năng đặc biệt, và âm thanh khi nhân vật bị trúng đòn**.
- **Phát triển AI và tương tác NPC:** Các lập trình viên AI sẽ xây dựng các **hành vi thông minh cho NPC**, giúp chúng có thể **phát hiện người chơi, tấn công hoặc phòng thủ** tùy theo tình huống. Sử dụng **State Machine** để quản lý trạng thái của các NPC giúp game trở nên chân thực và đầy thử thách.

- **Kiểm thử và tối ưu hóa:** Trước khi game hoàn thiện, đội ngũ **Game Tester** sẽ kiểm tra toàn bộ trò chơi để phát hiện và sửa chữa các lỗi phát sinh. Quá trình kiểm thử bao gồm **kiểm tra tính tương thích trên nhiều thiết bị**, đảm bảo trò chơi hoạt động mượt mà và không có lỗi gây gián đoạn trải nghiệm.

2.3. Giai đoạn hậu kỳ (Post-production)

Sau khi trò chơi hoàn tất các công đoạn sản xuất, nhóm phát triển sẽ bước vào giai đoạn **hậu kỳ**, nơi tập trung vào việc **quảng bá, ra mắt, và bảo trì trò chơi**:

- **Marketing và chiến dịch quảng bá:** Để đảm bảo game tiếp cận đúng đối tượng người chơi, đội ngũ Marketing sẽ sử dụng các **chiến dịch truyền thông** trên các nền tảng như **YouTube, Facebook, và TikTok**. Đồng thời, các video **trailer và teaser** sẽ được phát hành nhằm tạo sự chú ý và kích thích sự mong đợi từ người chơi.
- **Phát hành phiên bản beta và tiếp nhận phản hồi:** Để cải thiện chất lượng trước khi phát hành chính thức, một số nhà phát triển thường tung ra phiên bản **beta** cho một số người chơi thử nghiệm. Những phản hồi từ người chơi giúp phát hiện **lỗi tiềm ẩn** và tối ưu hóa trải nghiệm người dùng.
- **Cập nhật và bảo trì trò chơi:** Sau khi trò chơi được phát hành chính thức, việc **bảo trì và cập nhật** định kỳ là cần thiết để giữ cho trò chơi luôn mới mẻ và hấp dẫn. Các bản cập nhật có thể bao gồm **nhiệm vụ mới, sự kiện đặc biệt**, và **các cải tiến về hiệu suất** nhằm giữ chân người chơi lâu dài.

3. Cơ hội nghề nghiệp trong ngành lập trình game

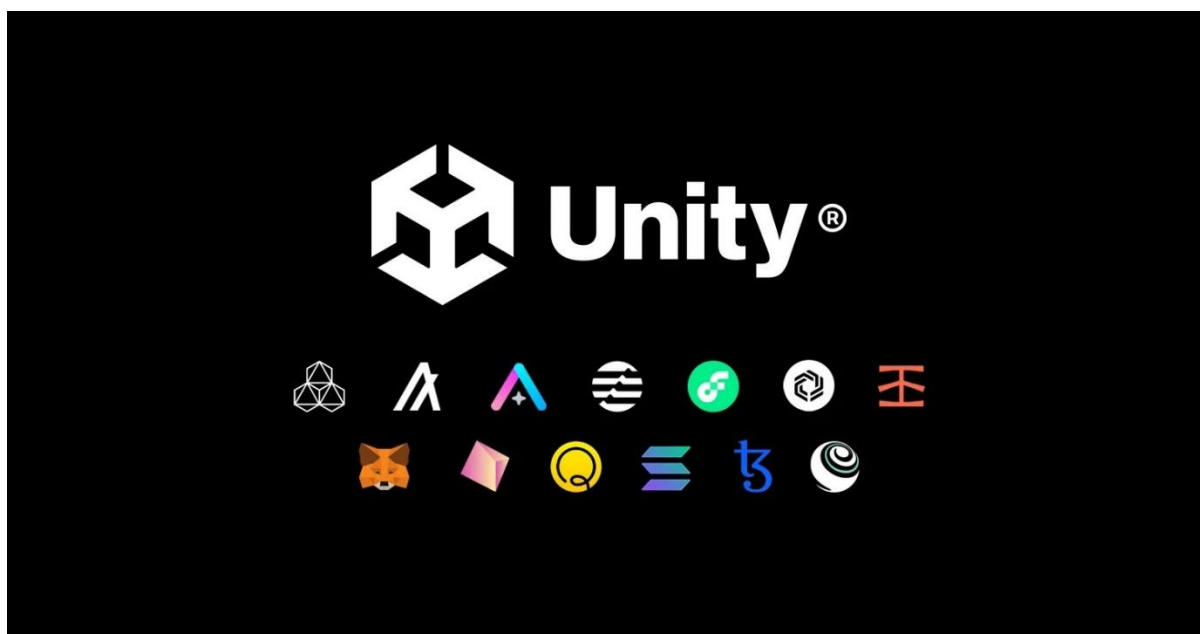
Ngành lập trình game hiện nay đang **tăng trưởng mạnh mẽ**, không chỉ ở Việt Nam mà còn trên toàn thế giới. Việt Nam hiện có hơn **12 triệu game thủ**, và nhu cầu về trò chơi ngày càng tăng cao. Các công ty như **Gameloft, VNG, và GlassEgg** liên tục tuyển dụng các vị trí **Game Developer, Game Artist, và Game Tester**. Nếu bạn có **kỹ năng lập trình tốt, hiểu biết về thiết kế đồ họa**, và khả năng **ngoại ngữ**, bạn

hoàn toàn có thể tìm kiếm cơ hội làm việc tại **các công ty nước ngoài** với mức lương hấp dẫn.

4. Giới thiệu về Unity

Unity là một công cụ phát triển trò chơi (game engine) mạnh mẽ được công ty Unity Technologies phát triển, ra mắt lần đầu vào năm 2005. Ban đầu, Unity được thiết kế dành riêng cho các nhà phát triển game trên hệ điều hành Mac OS. Tuy nhiên, với sự phát triển vượt bậc và nhu cầu mở rộng, Unity đã nhanh chóng trở thành công cụ đa nền tảng, hỗ trợ phát triển game trên nhiều hệ điều hành như Windows, iOS, Android, WebGL, và nhiều nền tảng console khác.

Unity được yêu thích nhờ vào khả năng hỗ trợ đồ họa 2D, 3D mạnh mẽ, cùng với tính năng kéo và thả (drag-and-drop) tiện dụng, giúp các nhà phát triển nhanh chóng xây dựng, thử nghiệm và triển khai game của mình mà không cần quá nhiều thao tác phức tạp. Hơn nữa, Unity cung cấp một kho tài nguyên phong phú, từ hình ảnh, âm thanh, đến các đoạn mã, giúp tối ưu hóa quy trình phát triển và tạo ra những sản phẩm chất lượng cao.



Hình 2.2: Logo Unity

5. Các tính năng và ưu điểm của Unity

Unity không chỉ đơn giản là một công cụ lập trình, mà còn là nền tảng toàn diện cho phát triển trò chơi. Một số tính năng nổi bật bao gồm:

- **Hỗ trợ đồ họa 2D và 3D:** Unity cung cấp khả năng tạo đồ họa chất lượng cao, phù hợp với cả game 2D và 3D, tạo ra những cảnh quan, nhân vật sống động và chi tiết.
- **Giao diện trực quan:** Với giao diện kéo-thả, Unity cho phép người dùng dễ dàng thiết kế, thay đổi, và bố trí các thành phần trong game mà không cần sử dụng quá nhiều dòng mã, giúp tiết kiệm thời gian phát triển.
- **Tính năng đa nền tảng:** Unity có khả năng xuất bản trò chơi trên nhiều nền tảng khác nhau chỉ từ một mã nguồn duy nhất, giúp tối ưu hóa quy trình và tiết kiệm chi phí.
- **Cộng đồng lớn và tài nguyên phong phú:** Unity có cộng đồng người dùng rộng lớn, các khóa học, diễn đàn và kho tài nguyên Asset Store, cung cấp mô hình, âm thanh và công cụ hỗ trợ phong phú để người dùng có thể dễ dàng tùy chỉnh theo nhu cầu.



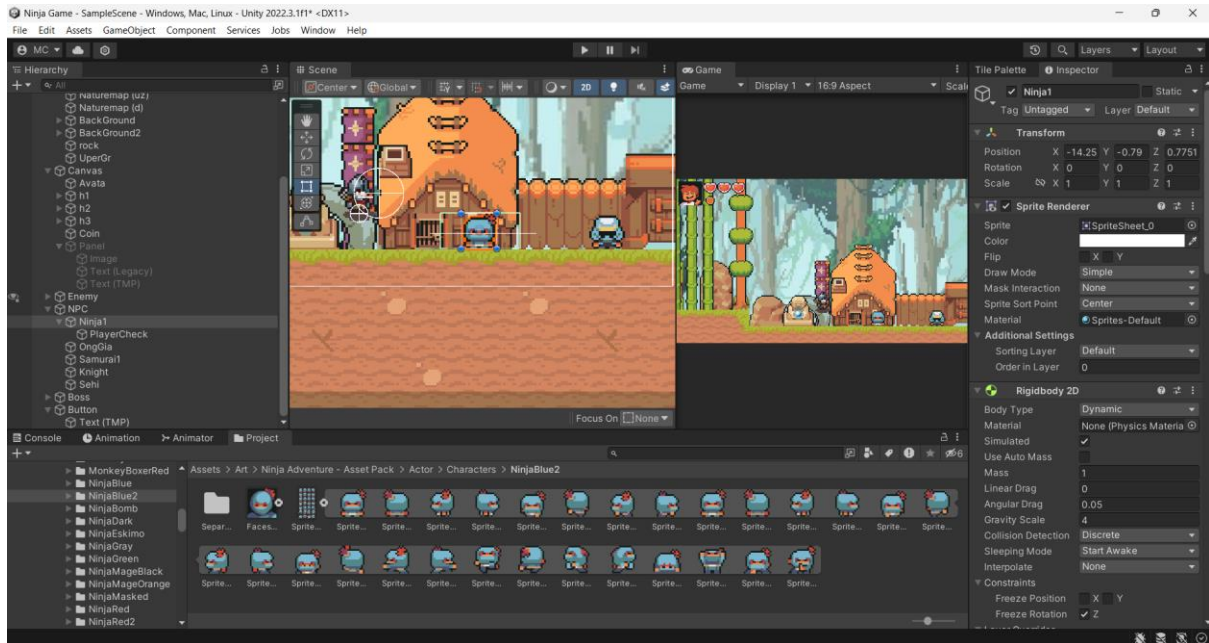
Hình 2.3: Các nền tảng hỗ trợ của Unity

6. Các bước cơ bản trong quy trình phát triển game với Unity

Phát triển game với Unity bao gồm nhiều bước từ lập kế hoạch, thiết kế, đến lập trình và thử nghiệm. Quy trình này có thể được chia thành các giai đoạn chính sau:

- **Lên ý tưởng và kịch bản:** Đây là bước khởi đầu, nơi ý tưởng và câu chuyện của trò chơi được hình thành. Lên ý tưởng về bối cảnh, nhân vật và các nhiệm vụ sẽ giúp xây dựng nền tảng cốt lõi cho game.
- **Thiết kế và mô hình hóa:** Dựa trên ý tưởng, đội ngũ thiết kế sẽ tạo ra bản phác thảo, mô hình 2D hoặc 3D của các nhân vật, cảnh quan và các đối tượng trong game.
- **Lập trình gameplay:** Sử dụng ngôn ngữ C#, các nhà phát triển sẽ xây dựng hệ thống logic, cơ chế điều khiển nhân vật, và các tính năng chính của game.
- **Kiểm tra và tối ưu hóa:** Sau khi hoàn thành game, giai đoạn kiểm thử và tối ưu hóa sẽ giúp phát hiện và khắc phục lỗi, điều chỉnh để trò chơi hoạt động mượt mà trên nhiều thiết bị.

- **Phát hành và bảo trì:** Khi hoàn thành các bước trên, game sẽ được phát hành trên các nền tảng mà Unity hỗ trợ. Sau khi ra mắt, game vẫn cần được bảo trì và cập nhật để cải thiện trải nghiệm người chơi và khắc phục các vấn đề phát sinh.



Hình 2.4: Giao diện của Unity

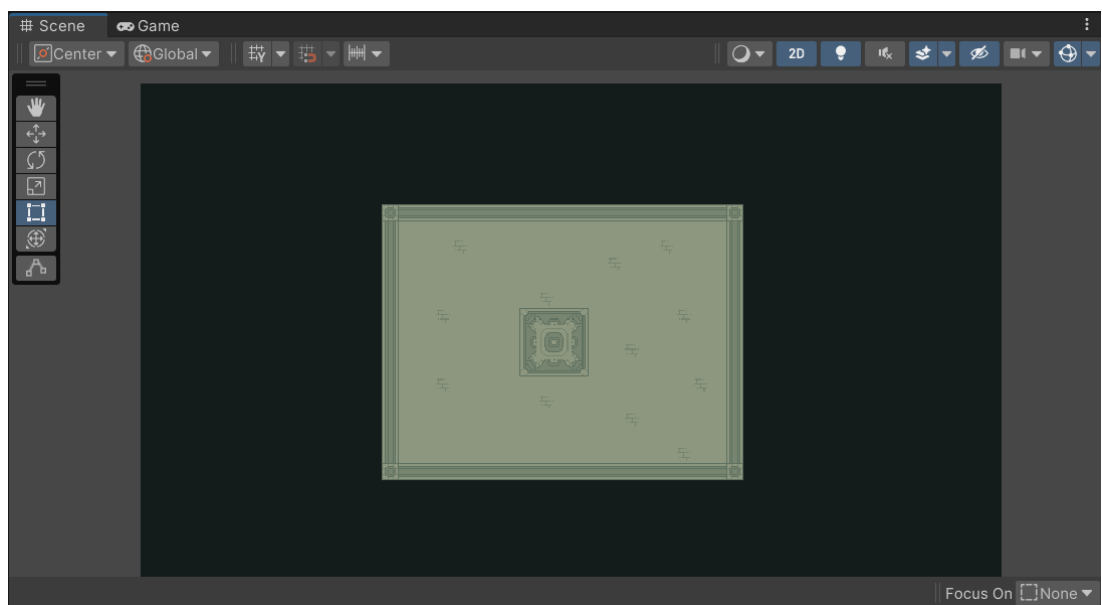
7. Tại sao chọn Unity cho phát triển game nhập vai ninja

Unity không chỉ cung cấp công cụ phát triển mà còn giúp các nhà làm game biến ý tưởng thành hiện thực. Đối với dự án game nhập vai ninja, Unity cung cấp sự hỗ trợ tuyệt vời để xây dựng một thế giới sống động với đồ họa sắc nét, âm thanh chân thực và các cơ chế chơi hấp dẫn. Các tính năng của Unity như hỗ trợ đồ họa mạnh mẽ, kho tài nguyên phong phú và khả năng tùy chỉnh cao sẽ giúp người chơi dễ dàng nhập vai, hòa mình vào cuộc phiêu lưu kịch tính của nhân vật ninja trong thế giới cổ xưa.

8. Các thành phần trong Unity Editor

8.1. Cửa sổ Scene

- Cửa sổ Scene cho phép bạn trực quan hóa tất cả các đối tượng trong scene hiện tại của bạn, cho phép bạn lựa chọn, kéo thả, phóng to, thu nhỏ và xoay các đối tượng một cách dễ dàng.
- Bạn có thể thay đổi chế độ xem giữa 2D và 3D, tùy chỉnh ánh sáng, bóng đổ và âm thanh. Đặc biệt, Scene View là nơi bạn sẽ bố trí các GameObject như cây cối, nhân vật, vật phẩm, camera... Đây là một trong những chức năng quan trọng nhất trong Unity giúp tạo ra hoạt cảnh cho game của bạn.

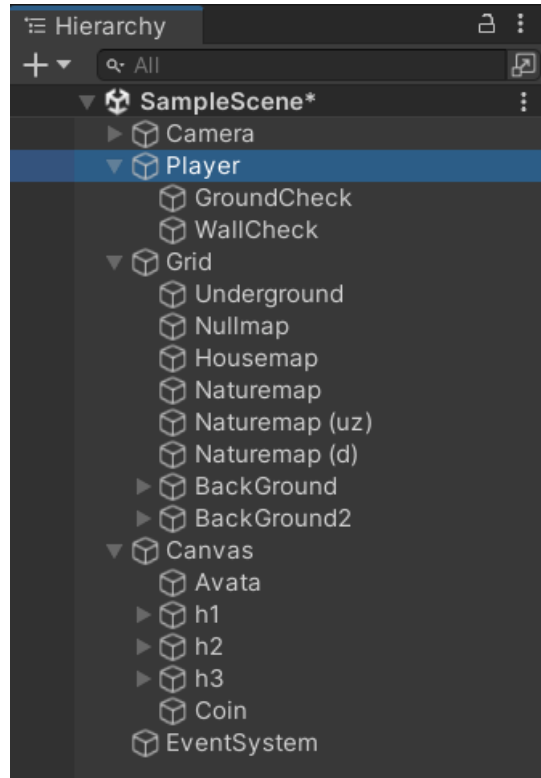


Hình 2.5: Cửa sổ Scene

8.2. Cửa sổ Hierarchy

- Tab Hierarchy là danh sách tất cả các GameObject hiện có trong Scene. Mỗi khi bạn thêm hoặc xóa một đối tượng, nó sẽ xuất hiện hoặc biến mất trong cửa sổ này.

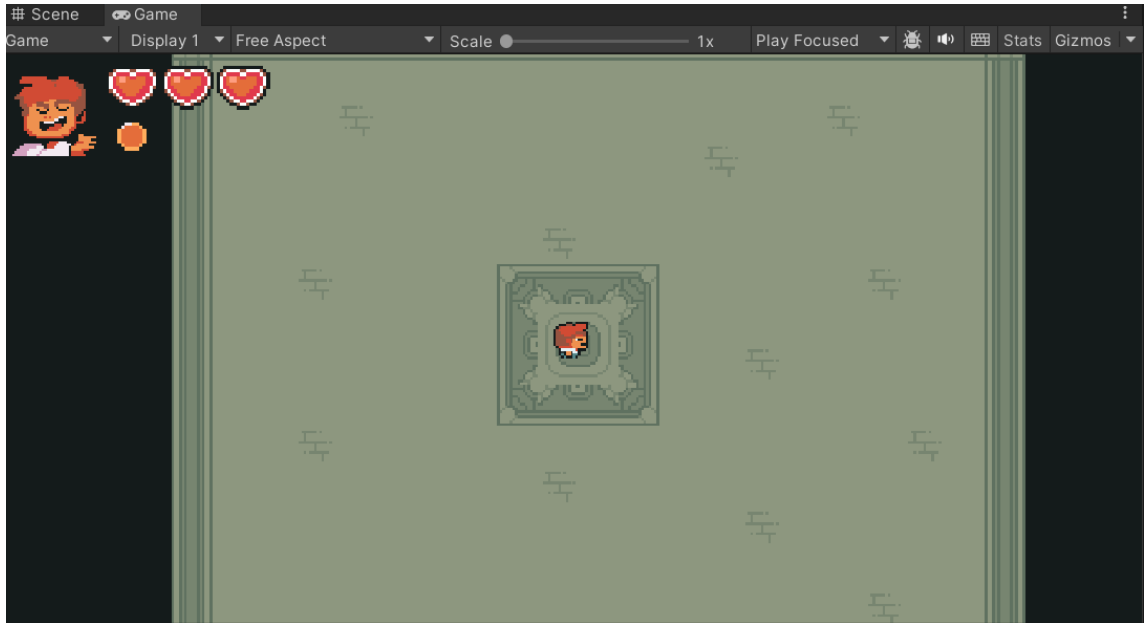
- Để dễ dàng quản lý các dự án lớn, Hierarchy cung cấp thanh tìm kiếm, cho phép bạn nhanh chóng tìm và tương tác với các đối tượng, đặc biệt hữu ích khi có nhiều đối tượng cùng tồn tại.



Hình 2.6: Cửa sổ Hierarchy

8.3. Cửa sổ Game

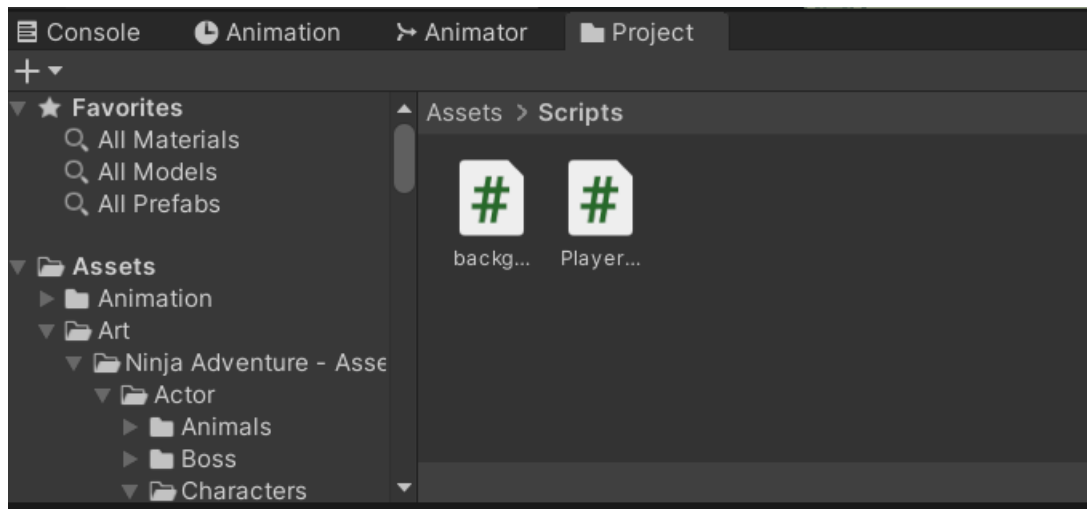
- Cửa sổ Game thể hiện những gì mà người chơi sẽ thấy từ góc nhìn của camera trong game. Đây là nơi bạn có thể kiểm tra và xem trước trò chơi của mình.
- Các công cụ trong cửa sổ Game cho phép bạn thay đổi độ phân giải màn hình, xem các thông số kỹ thuật (stats) và bật/tắt các tính năng như gizmos hoặc các thành phần khác để tối ưu trải nghiệm người dùng.



Hình 2.7: Cửa sổ Game

8.4. Cửa sổ Project

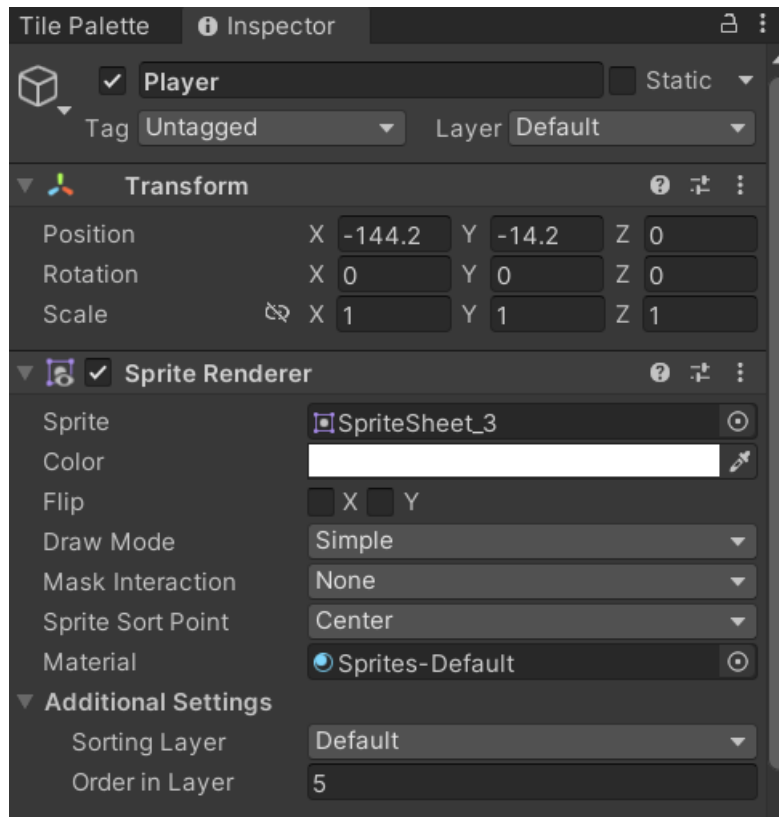
- Cửa sổ Project giống như trình quản lý file của Unity, hiển thị tất cả tài nguyên (Assets) trong dự án.
- Cột bên trái hiển thị cấu trúc cây thư mục, giúp bạn tổ chức các Assets hiệu quả, trong khi khung bên phải hiển thị nội dung của thư mục bạn chọn. Bạn có thể tạo các thư mục mới và tài nguyên bằng cách nhấp chuột phải hoặc nhấn nút "Create".
- Ngoài ra, bạn có thể sử dụng mục Favorites để đánh dấu và truy cập nhanh các tài nguyên thường dùng.



Hình 2.8: Cửa sổ Project

8.5. Cửa sổ Inspector

- Cửa sổ Inspector cung cấp thông tin chi tiết về GameObject đang làm việc, bao gồm các **component** và thuộc tính của nó. Bạn có thể tùy chỉnh trực tiếp các thông số này mà không cần thông qua mã lập trình.
- Inspector cũng hiển thị cài đặt của các asset như script, hoạt ảnh và âm thanh, giúp bạn kiểm soát chúng dễ dàng hơn.



Hình 2.9: Cửa sổ Inspector

9. Các khái niệm cơ bản trong Unity

9.1. GameObject

- Mọi đối tượng trong trò chơi đều được coi là một **GameObject**, từ nhân vật, xe cộ cho đến cây cối, nhà cửa... GameObject chỉ là vỏ bọc, không có chức năng gì cho đến khi được gắn các thành phần (**Component**).

9.2. Component

- Các thành phần (Component) cung cấp chức năng cho GameObject. Ví dụ, **Sprite Renderer** để hiển thị hình ảnh, **Animator** để xử lý hoạt ảnh, **Collider** để phát hiện va chạm, **Rigidbody** để tạo hiệu ứng vật lý, và **Script** để điều khiển hành vi.

9.3. Sprite

- Sprite là hình ảnh 2D đại diện cho các đối tượng trong game, chẳng hạn như nhân vật hoặc nền tảng.

9.4. Animation

- Animation là sự kết hợp của nhiều Sprite để tạo ra các chuyển động mượt mà. Mỗi hình ảnh riêng lẻ gọi là một **Frame**, và tập hợp các Frame tạo thành một hoạt ảnh hoàn chỉnh.

9.5. Key Frame

- Key Frame là các điểm chính trong quá trình tạo hoạt ảnh, nơi các thay đổi quan trọng về hình dạng, vị trí hoặc các thuộc tính khác của đối tượng xảy ra.

9.6. Prefabs

- Prefabs giúp bạn tạo ra các bản sao đối tượng nhanh chóng. Khi bạn cần sử dụng nhiều đối tượng giống nhau, ví dụ như các viên gạch hoặc đồng xu trong game Mario, bạn chỉ cần tạo một Prefab và tái sử dụng nó.

9.7. Sounds

- Âm thanh đóng vai trò quan trọng trong việc tạo cảm xúc và không khí cho game. Unity hỗ trợ quản lý âm thanh dễ dàng thông qua các **Audio Source** và **Audio Listener**.

9.8. Script

- Các Script chứa mã nguồn, thường viết bằng ngôn ngữ C#, để điều khiển và xử lý hành vi của các GameObject.

9.9. Scenes

- Scene là một tập hợp các GameObject đại diện cho một màn chơi hoặc môi trường trong game. Mỗi scene có thể được tải hoặc chuyển đổi khi cần.

9.10. Assets

- Assets bao gồm tất cả những tài nguyên cần thiết cho dự án game như hình ảnh, âm thanh, hoạt ảnh, mã nguồn và mô hình 3D.

9.11. Camera

- Camera trong Unity là một GameObject đặc biệt, xác định tầm nhìn của người chơi. Nó có thể được di chuyển hoặc xoay để tạo các góc nhìn khác nhau.

9.12. Transform

- Transform xác định vị trí, tỉ lệ và góc quay của một đối tượng trong Scene. Đây là một trong những thành phần quan trọng nhất vì nó quyết định đối tượng nằm ở đâu và nhìn như thế nào.

9.13. Material và Shader

- **Material** xác định cách hiển thị của bề mặt vật thể, trong khi **Shader** là các đoạn mã quy định cách ánh sáng và màu sắc tương tác với bề mặt.

10. Các hàm API phổ biến trong Unity

10.1. MonoBehaviour

- Là lớp cơ sở mà mọi Script trong Unity cần kế thừa để có thể gắn vào GameObject. Các hàm như Start(), Update() và FixedUpdate() giúp bạn điều khiển các hành vi của đối tượng theo từng khung hình.

10.2. Transform

- Thành phần giúp bạn thay đổi vị trí, góc quay và kích thước của GameObject trong Scene.

10.3. Rigidbody

- Dùng để áp dụng hiệu ứng vật lý như trọng lực hoặc va chạm cho các GameObject.

10.4. Collider

- Xác định các vùng để phát hiện va chạm. Bạn có thể sử dụng **Box Collider**, **Sphere Collider**... tùy theo hình dạng của đối tượng.

10.5. Coroutine

- Coroutine cho phép bạn tạm dừng thực thi mã mà không cần dừng toàn bộ ứng dụng, giúp thực hiện các tác vụ như đợi một khoảng thời gian (WaitForSeconds) hoặc tải tài nguyên không đồng bộ.

CHƯƠNG III: TRIỂN KHAI QUÁ TRÌNH XÂY DỰNG GAME VỚI UNITY

1. Các yếu tố làm nên game

Để tạo nên một trò chơi 2D hấp dẫn và cuốn hút, cần phải chú trọng đến nhiều yếu tố quan trọng. Dưới đây là các yếu tố chính góp phần làm nên sự thành công của game **xuyên không ninja**:

- **Đồ họa (Graphics):**

- Game sử dụng đồ họa **2D pixel art**, mang đậm phong cách Nhật Bản cổ đại. Các nhân vật ninja, kẻ địch, và môi trường được thiết kế với chi tiết sắc nét nhưng vẫn giữ được vẻ cổ điển. Đồ họa 2D giúp tối ưu hóa hiệu năng game, phù hợp với nhiều thiết bị.
- Mỗi màn chơi có một bối cảnh khác nhau như **rừng trúc**, **làng mạc hoang tàn**, **hang động tối tăm**, và **lâu đài bỏ hoang**. Các chi tiết như đèn lồng treo, cầu gỗ bắc qua sông, và cánh cổng torii đỏ nổi bật trong khung cảnh tạo cảm giác bí ẩn.
- Các công cụ như Photoshop và Aseprite được sử dụng để thiết kế các sprite nhân vật và hiệu ứng động cho các hành động như chém kiếm, phi tiêu, và sử dụng kỹ năng đặc biệt.

- **Âm thanh (Sound):**

- Âm thanh đóng vai trò quan trọng trong việc tạo nên không khí của game. Nhạc nền sử dụng các nhạc cụ truyền thống Nhật Bản như **sáo shakuhachi**, **đàn koto**, và **trống taiko**, mang lại cảm giác thần bí và căng thẳng.
- Hiệu ứng âm thanh sống động như tiếng kiếm chém vào không khí, âm thanh của phi tiêu khi bay đi, tiếng bước chân nhẹ nhàng khi nhân vật di chuyển và cả tiếng gió rít qua những cánh rừng u ám.
- Công cụ như **FL Studio** và **Audacity** được sử dụng để biên tập và phối âm cho các hiệu ứng và nhạc nền, nhằm đảm bảo chất lượng âm thanh sống động và lôi cuốn.

- **Giao diện (UI):**

- Giao diện đơn giản nhưng trực quan với các yếu tố được sắp xếp hợp lý để người chơi dễ dàng nắm bắt thông tin. Thanh máu, số phi tiêu còn lại, và các kỹ năng đặc biệt của ninja sẽ được hiển thị rõ ràng ở góc màn hình.
- Các menu như **menu chính**, **menu tạm dừng**, **menu cài đặt** đều thiết kế tối giản với tông màu phù hợp chủ đề của game. Hình nền của menu có thể là hình ảnh các ninja đang lén lút hoặc đang chiến đấu dưới ánh trăng.
- **Điều khiển (Controls):**
 - Trò chơi hỗ trợ điều khiển linh hoạt trên **bàn phím** và **tay cầm**, cũng như **chạm cảm ứng** trên các thiết bị di động. Các phím điều khiển chính:
 - Di chuyển trái/phải: **Phím A/D** (hoặc phím mũi tên).
 - Nhảy: **Phím Space**.
 - Điều khiển cảm ứng trên di động bao gồm các nút ảo được bố trí hợp lý trên màn hình để người chơi dễ thao tác.
- **Nhiệm vụ (Quests):**
 - Trò chơi bao gồm nhiều loại nhiệm vụ khác nhau như:
 - **Tiêu diệt địch thủ:** Người chơi phải tiêu diệt các quái vật, lính canh, và boss để mở đường đi tiếp.
 - **Thu thập vật phẩm:** Tìm kiếm các vật phẩm như phi tiêu, máu, và cuộn giấy cổ giúp tăng kỹ năng và sức mạnh của nhân vật.
 - **Khám phá:** Mỗi màn chơi có các khu vực bí mật và lối đi ẩn để người chơi khám phá nhằm mở khóa những vật phẩm đặc biệt.
- **Kỹ năng (Skills):**
 - Nhân vật chính sở hữu nhiều kỹ năng đặc biệt:
 - **Tàng hình:** Giúp ninja ẩn mình khỏi tầm nhìn của địch trong một khoảng thời gian.
 - **Tăng tốc:** Tăng tốc độ di chuyển, giúp nhân vật vượt qua chướng ngại vật hoặc truy đuổi địch nhanh hơn.

- **Phản đòn:** Kỹ năng phòng thủ giúp phản lại đòn tấn công của kẻ thù.
- Kỹ năng có thể được nâng cấp khi thu thập đủ điểm kinh nghiệm từ các màn chơi.

2. Phân tích thiết kế game

2.1. Đồ họa

- Game sử dụng phong cách **pixel art 2D** để mang lại sự hoài cổ nhưng vẫn tạo cảm giác mới lạ. Các **hiệu ứng ánh sáng động** như ánh trăng chiếu xuyên qua tán lá, hoặc lửa bập bùng từ những ngọn đuốc tạo thêm chiều sâu cho màn chơi.

2.2. Âm thanh

- Âm thanh không chỉ là nhạc nền mà còn được tích hợp vào các sự kiện như **tiếng gầm rú của boss**, **âm thanh khi mở kho báu** hoặc **hiệu ứng âm thanh khi sử dụng kỹ năng đặc biệt**, tạo nên sự hấp dẫn và chân thực.

2.3. Giao diện

- Giao diện người dùng được thiết kế để hiển thị các chỉ số **thanh máu**, **năng lượng kỹ năng**, và **số lượng phi tiêu** còn lại. Thanh trạng thái sẽ thay đổi màu sắc khi nhân vật bị thương nặng hoặc ở trạng thái nguy hiểm.

2.4. Điều khiển

- Điều khiển trong game được tối ưu hóa cho cả **PC** và **thiết bị di động**. Người chơi có thể tùy chỉnh các phím bấm để phù hợp với sở thích cá nhân.

2.5. Nhiệm vụ và Màn chơi

- Mỗi màn chơi đều có **bối cảnh độc đáo** và yêu cầu người chơi hoàn thành các nhiệm vụ như vượt qua **bẫy gai**, **dò tìm lối thoát** và tiêu diệt **boss cuối** để mở ra màn chơi mới.

3. Kiểm thử và bảo trì game

3.1. Kiểm thử

- **Kiểm thử chức năng:** Đảm bảo rằng các kỹ năng và vũ khí của nhân vật hoạt động đúng cách. Kiểm tra các trường hợp như khi nhân vật nhảy qua chướng ngại vật, đâm kiếm vào kẻ thù, hay sử dụng kỹ năng tàng hình để tránh bị phát hiện.
- **Kiểm thử giao diện:** Đảm bảo các menu và thông báo hiển thị đúng và dễ đọc, đặc biệt trên các thiết bị có kích thước màn hình khác nhau.
- **Kiểm thử tương thích:** Kiểm tra game trên các hệ điều hành **Windows, macOS, iOS, và Android** để đảm bảo tính tương thích và mượt mà.

3.2. Bảo trì

- **Bảo trì sửa lỗi:** Sau khi phát hành, cập nhật thường xuyên để sửa các lỗi phát sinh dựa trên phản hồi của người chơi.
- **Bảo trì cải tiến:** Tối ưu hóa tốc độ tải game, cải thiện chất lượng đồ họa và âm thanh.
- **Bảo trì mở rộng:** Mở rộng nội dung như thêm các **màn chơi mới, kỹ năng đặc biệt, và boss khủng hơn** để tăng thêm thử thách cho người chơi.

CHƯƠNG IV: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

1. Xác định tác nhân

Player:

Người chơi sẽ nhập vai một **ninja** có khả năng di chuyển linh hoạt và thực hiện các kỹ năng chiến đấu. Để điều khiển nhân vật ninja, người chơi sử dụng **các phím vật lý trên bàn phím** khi chơi trên PC:

- **A**: Di chuyển sang bên trái
- **D**: Di chuyển sang bên phải
- **Space**: Nhảy qua các chướng ngại vật hoặc tránh né đòn tấn công

Ngoài ra, các **nút bổ trợ** khác trên bàn phím có thể được sử dụng để **tấn công** (ví dụ: phím J để chém bằng kiếm), **ném phi tiêu** (phím K), và kích hoạt **kỹ năng đặc biệt** như **tàng hình** hoặc **lướt nhanh** để né tránh kẻ địch. Hệ thống điều khiển được tối ưu hóa để mang lại trải nghiệm mượt mà và chính xác, giúp người chơi có thể thực hiện các thao tác liên tục và phản ứng nhanh chóng với những tình huống bất ngờ trong game.

Enemy:

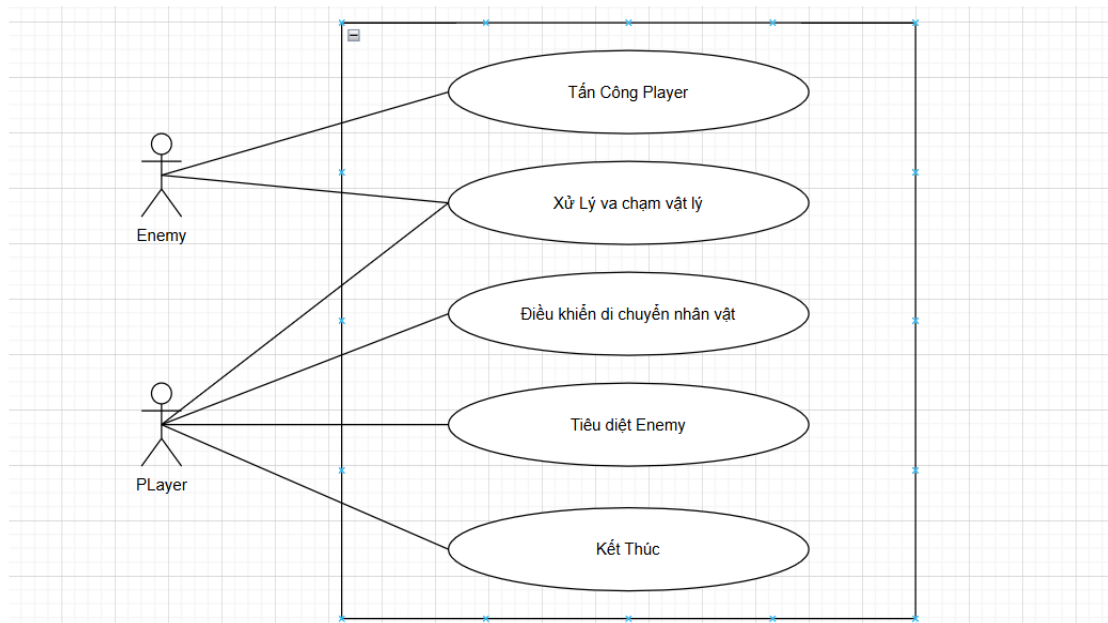
Các **kẻ địch** trong game bao gồm các **samurai**, **ninja đối địch**, và **lính canh**. Hệ thống trí tuệ nhân tạo (AI) điều khiển hành vi của kẻ địch thông qua **state machine** với các trạng thái như:

- **Tuần tra**: Di chuyển theo lộ trình có sẵn, cảnh giác với các dấu hiệu khả nghi.
- **Phát hiện**: Khi ninja của người chơi tiến vào **tầm nhìn** của địch, chúng sẽ chuyển sang trạng thái báo động.
- **Truy đuổi**: Kẻ địch sẽ đuổi theo và tấn công nếu phát hiện người chơi trong phạm vi.
- **Tấn công**: Khi tiếp cận gần ninja của người chơi, kẻ địch sẽ thực hiện các đòn tấn công như **chém kiếm**, **ném phi tiêu**, hoặc sử dụng **kỹ năng đặc biệt**.

Các kẻ địch cũng có khả năng **nhận diện môi trường** và **tránh va chạm với chướng ngại vật**, giúp chúng trở nên thông minh hơn và tạo ra những thử thách khó khăn cho người chơi.

2. Xây dựng biểu đồ Usecase

2.1. Mô hình UseCase tổng quát



Hình 4.1: Mô hình UseCase tổng quát

3. Danh sách các Use Case

- **Tấn công người chơi:** Các Enemy (như NPC, ninja đối địch, samurai hoặc lính canh) sẽ có hành vi di chuyển chủ động và tấn công nhân vật của người chơi khi phát hiện sự hiện diện của họ. Khi vào phạm vi tấn công, Enemy sẽ sử dụng các đòn tấn công như chém kiếm, ném phi tiêu, hoặc sử dụng các kỹ năng đặc biệt nhằm gây sát thương cho người chơi. Enemy sẽ tiếp tục truy đuổi cho đến khi người chơi thoát khỏi tầm nhìn hoặc bị tiêu diệt.
- **Xử lý va chạm vật lý:** Hệ thống xử lý va chạm trong game sẽ đảm bảo phản hồi chính xác khi Enemy hoặc nhân vật của người chơi va chạm với các đối tượng trên bản đồ. Điều này bao gồm cả những tình huống như khi Enemy

hoặc người chơi chạm vào các chướng ngại vật (ví dụ: tường, thùng gỗ, đá hoặc container). Khi va chạm xảy ra, hệ thống sẽ xác định hành động phù hợp như đẩy lùi, dừng lại hoặc gây sát thương dựa trên tính chất của vật thể bị va chạm.

- **Điều khiển di chuyển nhân vật:** Người chơi có thể điều khiển nhân vật ninja của mình bằng các phím vật lý trên bàn phím. Khi người chơi nhập lệnh:
 - A để di chuyển sang bên trái
 - D để di chuyển sang bên phải
 - Space để nhảy vượt qua chướng ngại vật hoặc né tránh đòn tấn công
 - Hệ thống sẽ nhận diện các input đầu vào, từ đó di chuyển và chuyển hướng nhân vật một cách mượt mà, đảm bảo trải nghiệm game chân thực và chính xác. Việc điều khiển cũng bao gồm thay đổi tư thế nhân vật, ví dụ như cúi người, lăn tròn hoặc nhảy cao để vượt qua chướng ngại vật.
- **Tiêu diệt Enemy:** Người chơi có thể tiêu diệt kẻ địch bằng cách sử dụng các kỹ năng chiến đấu, bao gồm việc tấn công bằng kiếm, ném phi tiêu hoặc sử dụng kỹ năng đặc biệt. Khi kẻ địch bị tấn công, hệ thống sẽ tính toán sát thương gây ra dựa trên loại vũ khí hoặc kỹ năng mà người chơi sử dụng. Enemy có thể bị tiêu diệt sau một số lần trúng đòn nhất định và khi đó, chúng sẽ rơi ra các vật phẩm hỗ trợ hoặc điểm kinh nghiệm giúp người chơi nâng cấp nhân vật.
- **Kết thúc màn chơi:** Khi người chơi hoàn thành các nhiệm vụ và tiêu diệt boss cuối cùng của màn chơi hiện tại, hệ thống sẽ kết thúc màn chơi và chuyển sang giao diện chiến thắng. Tại đây, dữ liệu về tiến độ của người chơi sẽ được cập

nhập vào cơ sở dữ liệu, bao gồm thông tin như màn chơi đã hoàn thành, điểm số, và cấp độ hiện tại. Người chơi cũng có thể nhận được phần thưởng như trang bị, kỹ năng mới hoặc vật phẩm đặc biệt để chuẩn bị cho các màn chơi tiếp theo với độ khó cao hơn.

4. Đặc tả UseCase

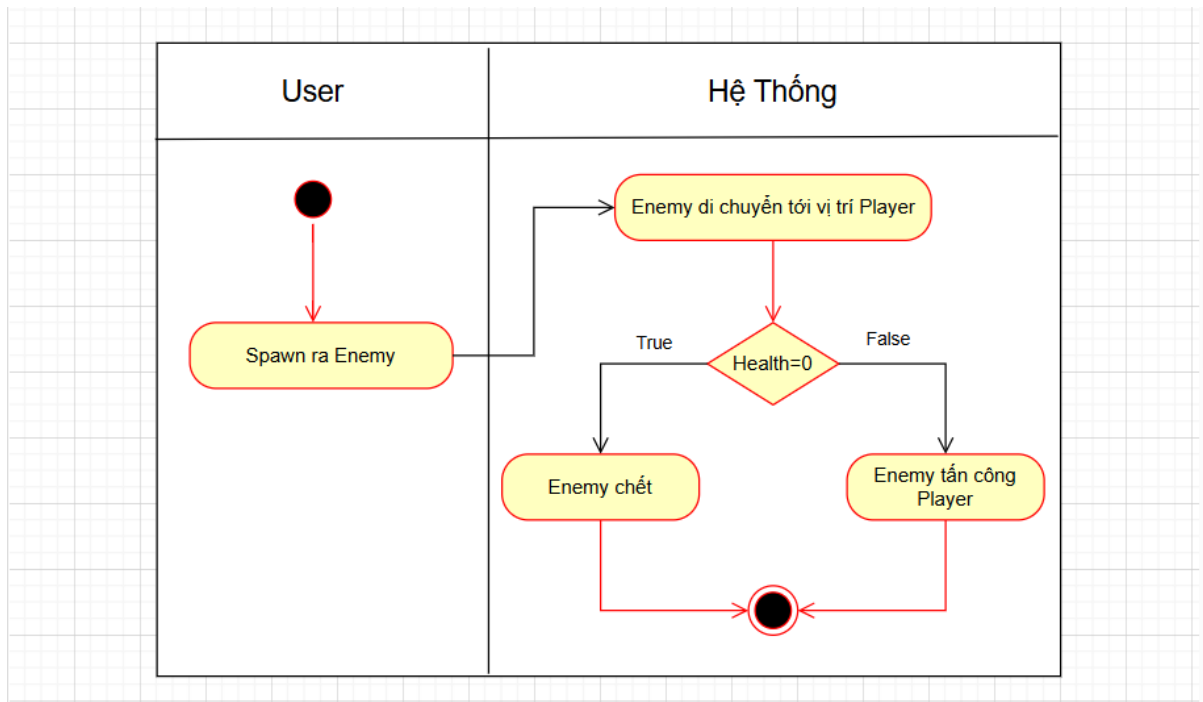
4.1. Use case tấn công player:

Kịch bản Use Case:

Bảng 0.1: Kịch bản Use Case Tấn công player

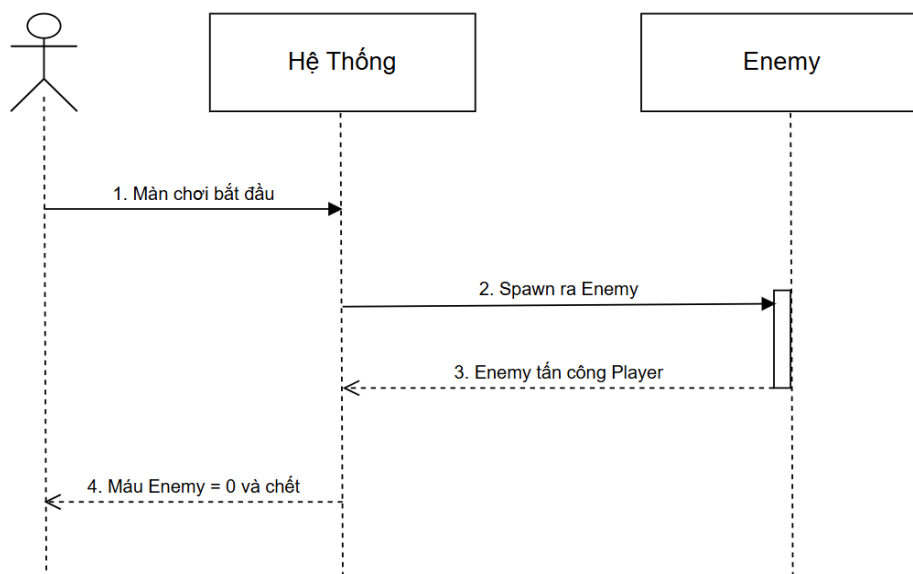
Mô tả	NPC tấn công player
Tác nhân	NPC
Tiền điều kiện	Màn chơi bắt đầu
Hậu điều kiện	NPC di chuyển về phía player
Luồng sự kiện	NPC luôn di chuyển về phía player Khi đến gần player sẽ tấn công và làm cho player mất máu
Luồng ngoại lệ	1a. NPC là quái vật, đạo tặc... của màn chơi thì sẽ cố định tại một vị trí và tấn công player 1b. NPC hết máu 1b.1: NPC chết

Sơ đồ Activity:



Hình 0.2: Sơ đồ Activity Use Case tấn công Player

Sơ đồ tuần tự:



Hình 0.3: Sơ đồ tuần tự Use Case tấn công Player

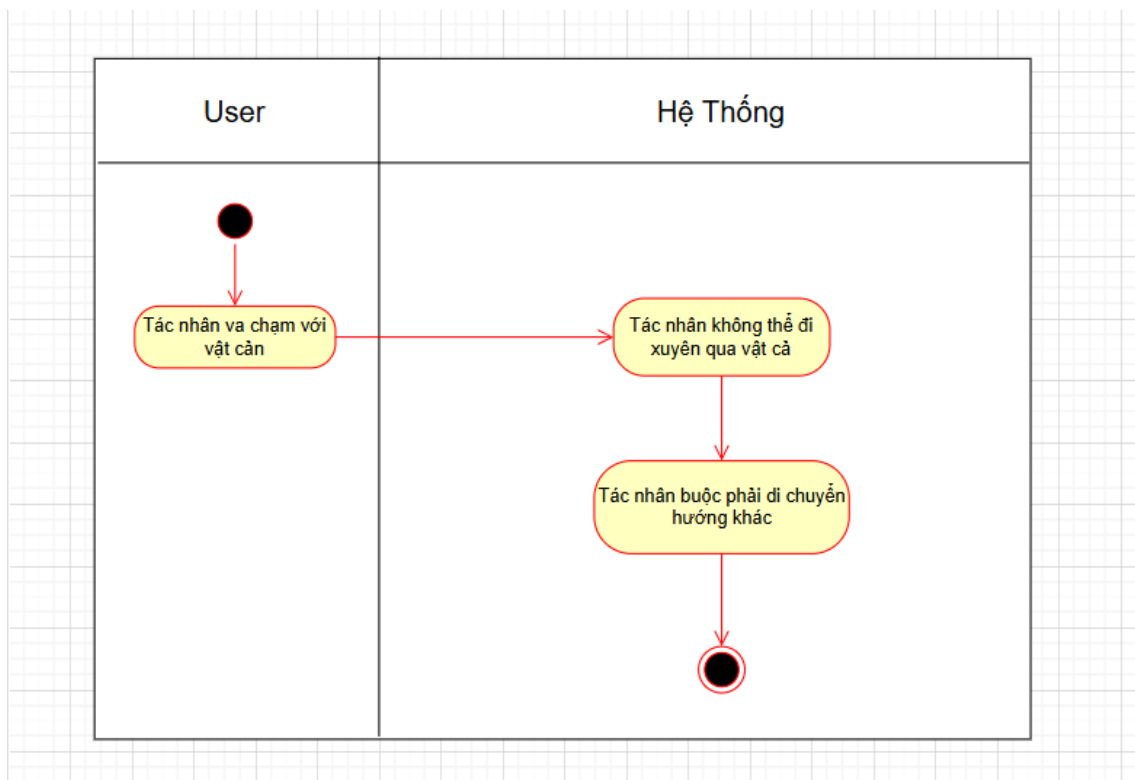
4.2. Use case xử lý va chạm vật lý:

Kịch bản Use Case:

Bảng 0.2: Kịch bản Use Case xử lý va chạm

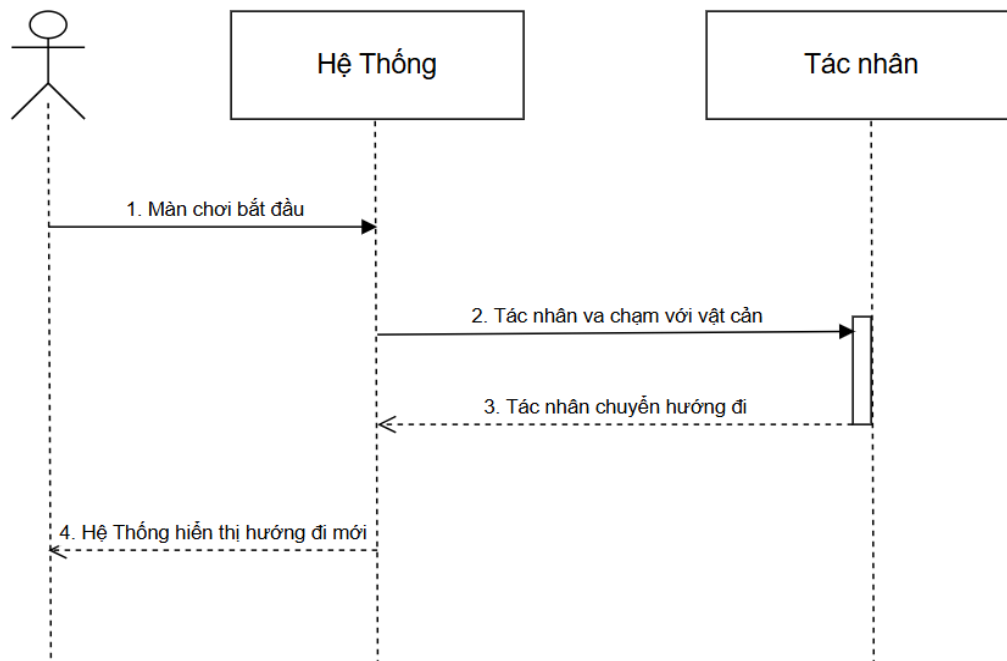
Mô tả	Xử lý va chạm vật lý
Tác nhân	Player, Enemy
Tiền điều kiện	Màn chơi bắt đầu
Hậu điều kiện	Va chạm vật lý được xử lý
Luồng sự kiện	<ol style="list-style-type: none">1. Tác nhân va chạm với chướng ngại vật.2. Tác nhân phải đi đường khác không có chướng ngại vật.

Sơ đồ Activity:



Hình 0.4: Sơ đồ Activity Use Case xử lý va chạm

Sơ đồ tuần tự:



Hình 0.5: Sơ đồ tuần tự Use Case xử lý va chạm

4.3. Use case điều khiển di chuyển nhân vật:

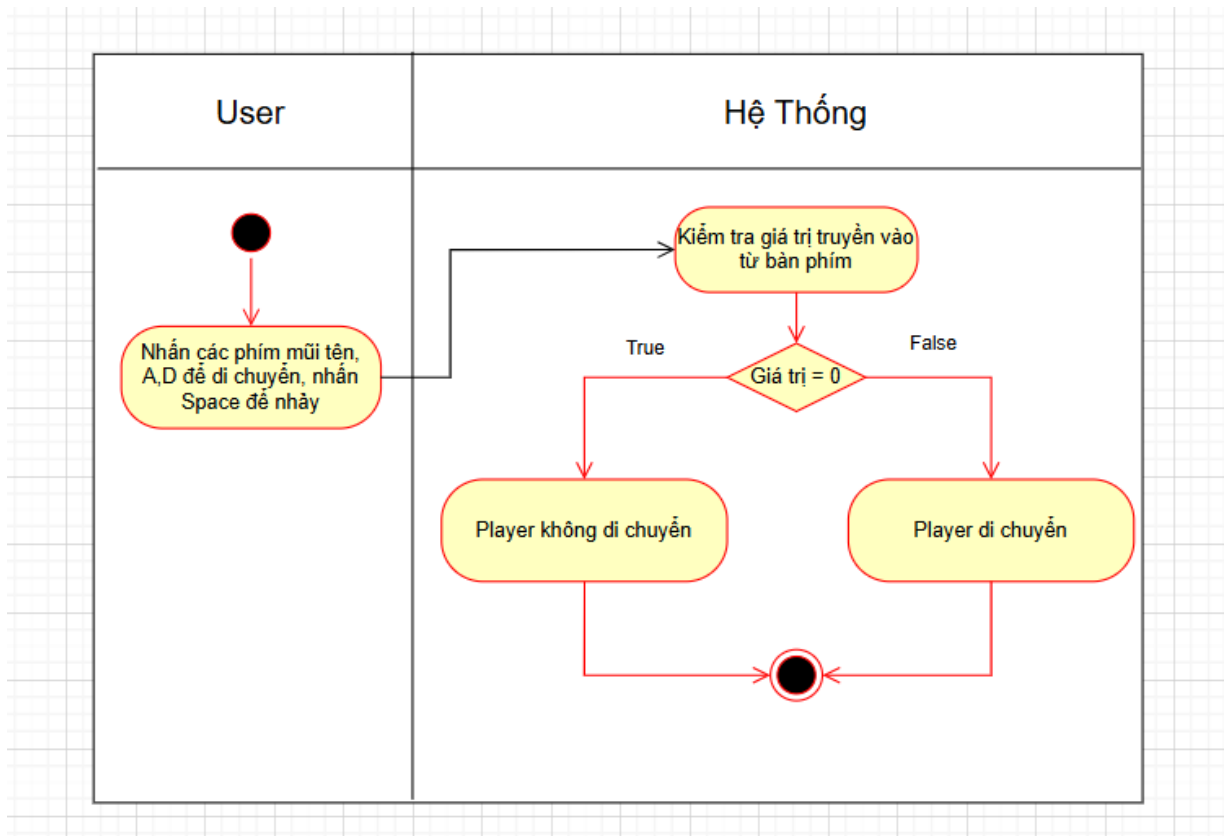
Kịch bản Use Case:

Bảng 0.3: Kịch bản Use Case điều khiển di chuyển nhân vật

Mô tả	Điều khiển di chuyển nhân vật
Tác nhân	Player
Tiền điều kiện	Màn chơi bắt đầu
Hậu điều kiện	Di chuyển nhân vật theo ý của mình
Luồng sự kiện	Nhân vật di chuyển theo 2 hướng trái và phải. Người chơi sử dụng bàn phím (mũi tên/ A,D,Space).

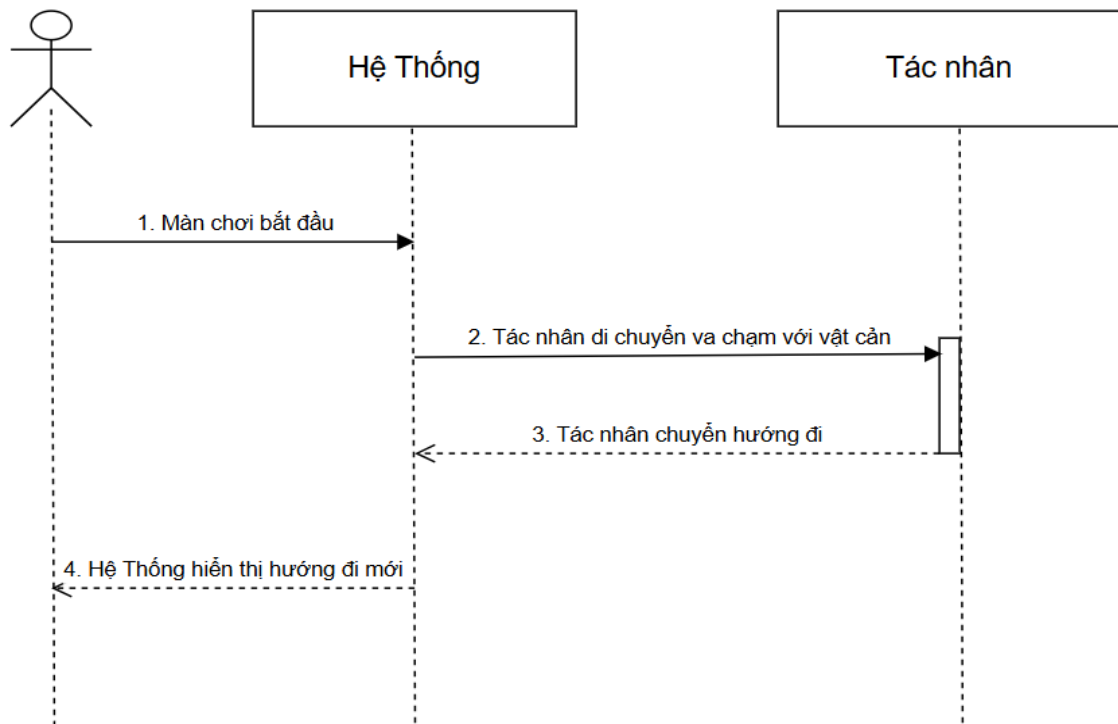
	3. Nhân vật sẽ di chuyển theo hướng mong muốn, trái – phải hoặc nhấn Space để nhảy lên
Luồng ngoại lệ	2a. Khi người chơi bị tiêu diệt thì nhân vật sẽ không di chuyển được nữa.

Sơ đồ Activity:



Hình 0.6: Sơ đồ Activity Use Case điều khiển di chuyển nhân vật

Sơ đồ tuần tự:



Hình 0.7: Sơ đồ tuần tự Use Case di chuyển nhân vật

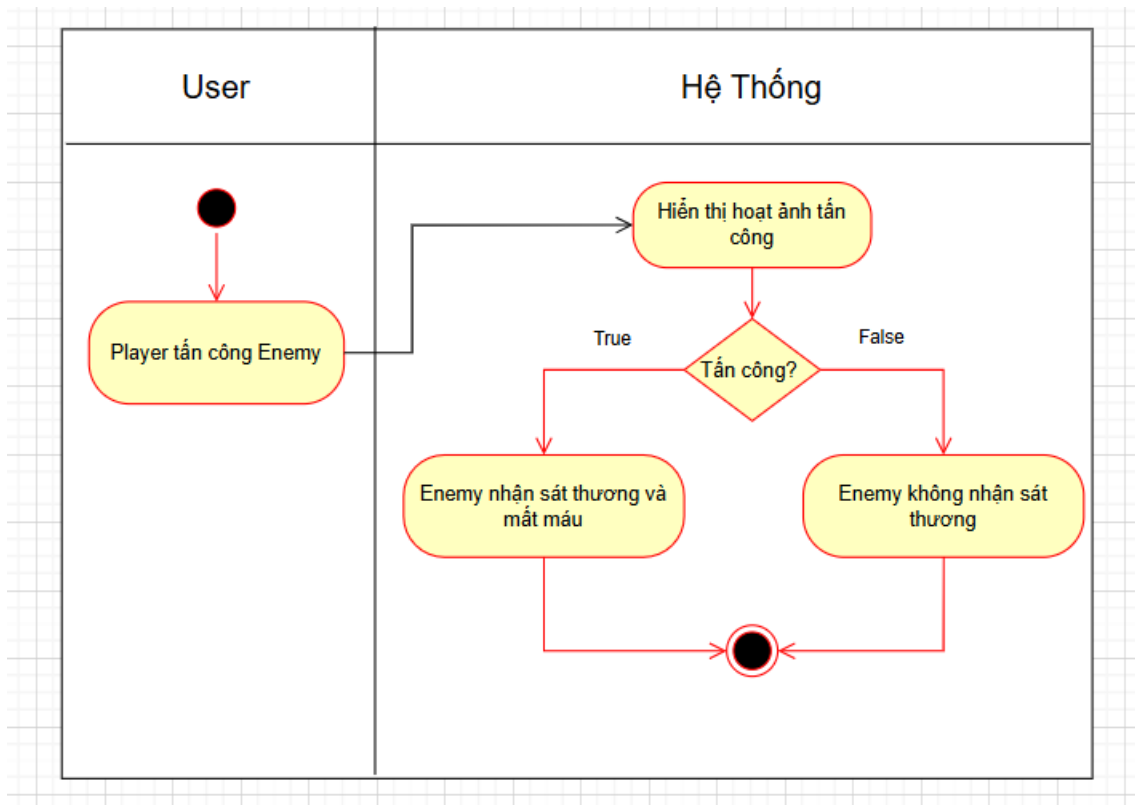
4.4. Use case tiêu diệt NPC:

Kịch bản Use Case:

Bảng 0.4: Kịch bản Use Case tiêu diệt Enemy

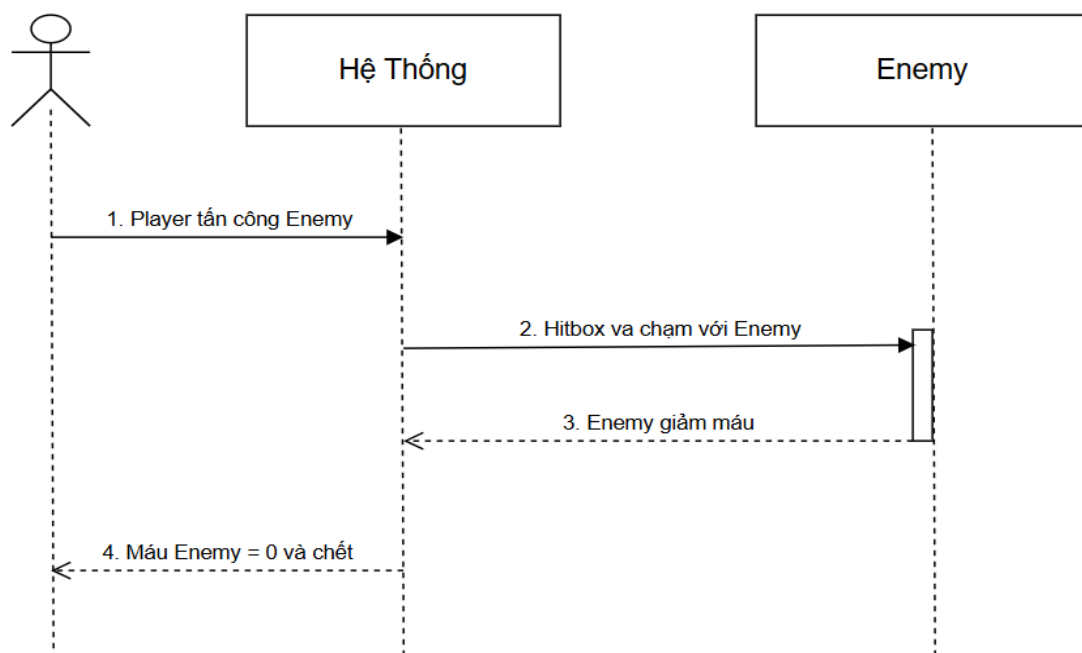
Mô tả	Tiêu diệt Enemy
Tác nhân	Player
Tiền điều kiện	Màn chơi bắt đầu
Hậu điều kiện	Enemy còn sống
Luồng sự kiện	Player tấn công Enemy. Máu của Enemy giảm dần. Enemy chết
Luồng ngoại lệ	1a. Player tấn công hụt

Sơ đồ Activity:



Hình 0.8: Sơ đồ Activity Use Case tiêu diệt Enemy

Sơ đồ tuần tự:



Hình 0.9: Sơ đồ tuần tự Use Case tiêu diệt Enemy

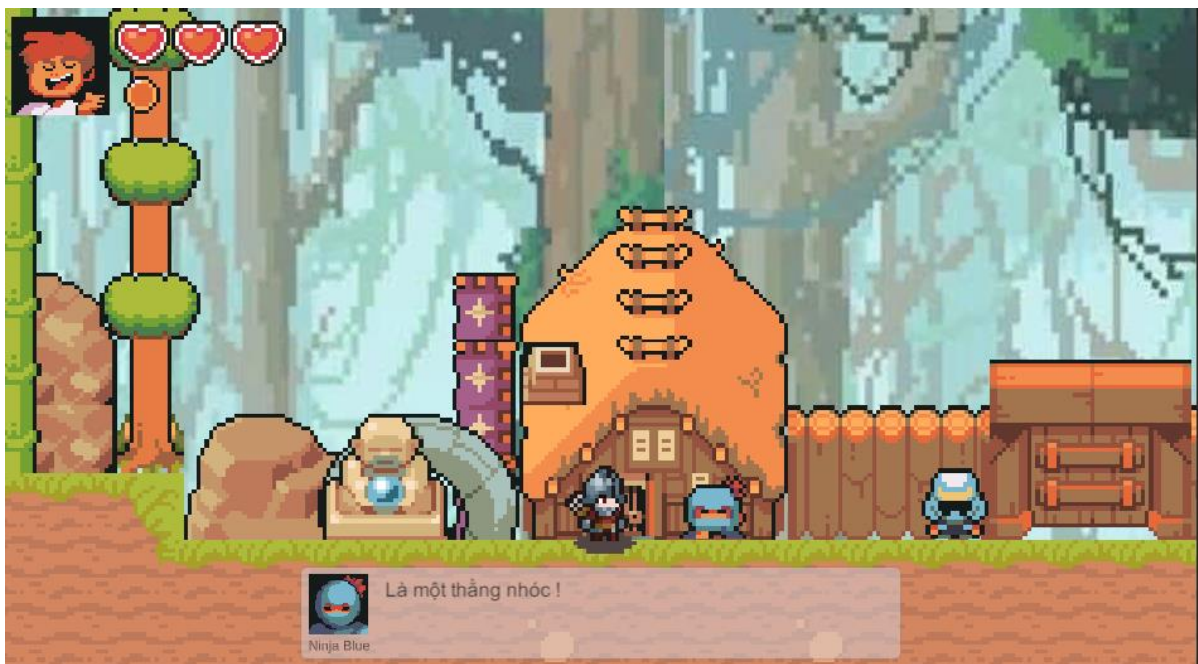
CHƯƠNG V: XÂY DỰNG GAME VÀ DEMO

Màn hình chính:



Hình 0.1: Màn hình chính

Màn hình hội thoại:



Hình 0.2: Màn hình hội thoại

Màn hình GamePlay:



Hình 0.3: Màn hình GamePlay

Màn hình thua:



Hình 0.4: Màn hình thua

Màn hình dừng game:



Hình 0.5: Màn hình dừng game

Người chơi trong game:



Hình 0.6: PLayer

Quái vật loại 1:



Hình 0.7: Quái vật loại 1

Quái vật loại 2:



Hình 0.8: Quái vật loại 2

Quái vật Boss:



Hình 0.9: Quái vật Boss

CHƯƠNG VI: KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN

Qua quá trình nghiên cứu và thực hiện đồ án, em đã tìm hiểu và áp dụng các kiến thức cơ bản trong lĩnh vực phát triển trò chơi, từ việc thiết kế ý tưởng, xây dựng nội dung, đến triển khai và kiểm thử. Tựa game "*Xuyên Không Thành Ninja*" đã được xây dựng với những yếu tố cơ bản như cốt truyện hấp dẫn, lối chơi lôi cuốn, hệ thống điều khiển thân thiện và giao diện trực quan.

Mặc dù còn nhiều hạn chế về mặt thời gian và kỹ thuật, sản phẩm đã thể hiện được tiềm năng phát triển và đáp ứng một phần mục tiêu ban đầu. Đây là một bước khởi đầu quan trọng để em tiếp tục học hỏi và nâng cao kỹ năng trong lĩnh vực lập trình trò chơi.

Để hoàn thiện hơn và nâng cao giá trị của trò chơi "*Xuyên Không Thành Ninja*", em dự định triển khai các hướng phát triển sau:

- **Mở rộng nội dung trò chơi:** Thêm nhiều màn chơi mới, nhân vật, và nhiệm vụ đa dạng hơn để nâng cao trải nghiệm của người chơi.
- **Tối ưu hóa hiệu suất:** Cải thiện khả năng xử lý và tối ưu hóa hiệu suất đồ họa để trò chơi hoạt động mượt mà hơn trên nhiều thiết bị.
- **Tích hợp tính năng mạng:** Xây dựng chế độ chơi nhiều người (multiplayer) để người chơi có thể tương tác và thi đấu cùng nhau.
- **Phát triển AI thông minh:** Cải thiện trí thông minh nhân tạo (AI) của kẻ thù để tạo ra thách thức lớn hơn và tăng tính hấp dẫn.
- **Hỗ trợ đa nền tảng:** Đưa trò chơi lên nhiều nền tảng khác nhau như Android, iOS, và PC để tiếp cận đối tượng người chơi rộng rãi hơn.
- **Cập nhật phản hồi từ người chơi:** Thu thập ý kiến phản hồi để tiếp tục cải tiến trò chơi, đảm bảo đáp ứng được mong đợi của cộng đồng.

Em xin chân thành cảm ơn sự hướng dẫn và hỗ trợ tận tình từ các thầy cô và bạn bè trong suốt quá trình thực hiện đồ án. Những ý kiến đóng góp quý báu sẽ là động lực giúp em không ngừng phát triển và hoàn thiện hơn nữa trong tương lai.

TÀI LIỆU THAM KHẢO

- * Lịch sử video game – Wikipedia tiếng Việt.
- * Lịch sử phát triển của ngành công nghiệp game.
- * Trò chơi điện tử – Wikipedia tiếng Việt.
- * Unity - Game Engine.
- * Unreal Engine - Game Engine.
- * GameMaker Studio - Game Engine.
- * Python - Programming Language.
- * C# - Programming Language.
- * JavaScript - Programming Language.
- * C++ - Programming Language.
- * GML - Programming Language.