

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CUỐI KỲ MÔN PHÁT TRIỂN TRÒ CHƠI

GAME OFFLINE THỂ LOẠI CÓ CỐT
TRUYỆN: “RUMMAGE MEMORIES”

Giảng viên hướng dẫn: **ThS. Vũ Đình Hồng**

Sinh viên thực hiện: **Tô Vĩnh Khang - 51800408**

Bùi Quang Khải - 51800785

Du Thuận Long - 51800429

Lớp: 18050203

Khóa: 22

Nhóm: 01

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN CUỐI KỲ MÔN PHÁT TRIỂN TRÒ CHƠI

GAME OFFLINE THỂ LOẠI CÓ CỐT
TRUYỆN: “RUMMAGE MEMORIES”

Giảng viên hướng dẫn: **ThS. Vũ Đình Hồng**

Sinh viên thực hiện: **Tô Vĩnh Khang - 51800408**

Bùi Quang Khải - 51800785

Du Thuận Long - 51800429

Lớp: 18050203

Khóa: 22

Nhóm: 01

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2020

LỜI CẢM ƠN

Phát triển trò chơi là một môn học rất hay và bổ ích, chúng em xin chân thành cảm ơn thầy ThS.Vũ Đình Hồng đã hướng dẫn và tạo dựng những điều kiện tốt nhất để chúng em hoàn thành đồ án cuối kỳ môn Phát triển trò chơi với đề tài game offline có cốt truyện mang tên “Rummage Memories”. Giúp chúng em có thêm kiến thức về những giai đoạn từ lên ý tưởng đến hiện thực game, hiểu biết khái quát hơn về cách sử dụng game engine Unity để xây dựng một game 3D thực tế.

Trong quá trình thực hiện, nhóm vẫn khó tránh khỏi những sai sót không mong muốn, kính mong thầy có thể góp ý và giúp đỡ chúng em. Nhóm xin chân thành cảm ơn thầy.

ĐỒ ÁN CUỐI KỲ NÀY ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Chúng tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của thầy Vũ Đình Hồng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án này còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào chúng tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 01 tháng 11 năm 2020

Tác giả

(ký tên và ghi rõ họ tên)

Tô Vĩnh Khang

(Trưởng nhóm)

Bùi Quang Khải

Du Thuận Long

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(ký và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(ký và ghi họ tên)

TÓM TẮT

Trong xã hội ngày càng phát triển với sự bùng phát mạnh mẽ từ các dịch vụ giải trí, nhu cầu về một game thú vị mới lạ, mang lại cảm giác thử thách người chơi với lối chơi mạo hiểm, lôi cuốn ngày càng được chú trọng. Và từ đó các game 3D ra đời. Để đáp ứng được nhu cầu trong bối cảnh thị trường hiện nay, KKL Software Team đã triển khai thực hiện dự án game 3D offline có cốt truyện với góc nhìn thứ ba mang tên Rummage Memories. Game được triển khai với mục tiêu giúp người chơi xả stress sau những ngày làm việc căng thẳng, tạo cảm giác muốn chinh phục trò chơi, thử thách bản thân cũng như sự khéo léo trong việc di chuyển, sử dụng vũ khí để tấn công tiêu diệt kẻ thù.

Rummage Memories - Nơi những mảnh kí ức còn sót lại trong tiềm thức lấy ý tưởng từ việc một cậu bé mang tên Kalus thuộc tộc người Heran. Tại Công nguyên Phantom, trong lúc di chuyển cùng cha của mình là Qatus trên phi thuyền số hiệu JAS-X thì bị bắn hạ bởi phi thuyền của Binh đoàn tộc thú (Zerg) khiến toàn bộ các thành viên phi thuyền tử nạn. Riêng Kalus còn sống và bị bắt giữ. Cậu bị giam trong căn phòng thí nghiệm. Zerg đã truyền nọc Asla khiến cho cậu rơi vào trạng thái hôn mê liên tục và tiến hành khai thác những mảnh kí ức từ não bộ của cậu để truy tìm đến nơi cất giữ Norm – một loài kén bướm giúp tăng khả năng sinh trưởng và sức mạnh cho tộc thú gấp ba lần. Trong quá trình khai thác não bộ, tiềm thức của Kalus sống dậy và điều khiển ngăn chặn hành vi khai thác của Zerg. Tất cả quá trình này diễn ra trong vòng 7 ngày nhưng với thể giới kí ức mà Kalus đang đối diện là tận 7 năm. Cậu đã đấu tranh, tìm kiếm những sự trợ giúp của các bô lão (NPC) để được quay trở về cánh cổng thức tỉnh và trả thù cho người cha quá cố của mình và dòng tộc Heran.

Về công cụ hỗ trợ phát triển game, nhóm sử dụng game engine Unity - một phần mềm làm game đa nền tảng được phát triển bởi Unity Technologies, hỗ trợ đồ họa 2D và 3D, các chức năng được viết chủ yếu qua ngôn ngữ C#. Game Rummage Memories được triển khai với giao diện đẹp mắt, thân thiện với người chơi, dễ dàng trong các thao tác di chuyển, nhảy để vượt chướng ngại vật, sử dụng vũ khí để tấn công kẻ thù.

Với đề tài game offline có cốt truyện “Rummage Memories”, nhóm mong muốn áp dụng những kiến thức đã học trong trường cùng với việc tìm hiểu nghiên cứu, phân tích tâm lý người chơi, những kiến thức bên ngoài xã hội để xây dựng game 3D offline có cốt truyện này đạt được hiệu quả cao nhất.

MỤC LỤC

TÓM TẮT.....	6
MỤC LỤC.....	8
DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT.....	12
DANH MỤC CÁC BẢNG BIỂU VÀ HÌNH.....	13
Danh mục Hình.....	13
Danh mục Bảng.....	14
CHƯƠNG I: GIỚI THIỆU.....	16
1.1 Giới thiệu đề tài.....	16
1.2 Phương pháp sử dụng.....	16
1.2.1 Game engine - Unity.....	16
1.2.2 Ngôn ngữ lập trình - C#.....	17
1.3 Lý do và mục đích chọn đề tài.....	17
1.4 Thông tin về game.....	17
1.4.1 Tên game.....	17
1.4.2 Bản quyền.....	17
1.4.3 Phiên bản.....	17
1.4.4 Ý tưởng game.....	17
1.4.5 Thể loại.....	18
1.4.6 Đối tượng nhắm đến.....	18
1.4.7 Tóm tắt dòng game.....	18
1.4.8 Giao diện.....	18
1.4.9 Phạm vi dự án.....	19
1.4.9.1 Số địa điểm.....	19
1.4.9.2 Số cấp độ.....	19
1.4.9.3 Số đối tượng.....	19
1.4.9.4 Số vũ khí.....	20

1.4.10 Lối chơi và cơ chế.....	20
1.4.10.1 Lối chơi.....	20
1.4.10.2 Cơ chế.....	20
1.4.11 Thế giới game.....	21
1.4.11.1 Chi tiết về địa điểm.....	21
1.4.11.2 Chi tiết về nhân vật.....	23
1.4.11.3 Chi tiết về cấp độ.....	25
CHƯƠNG II: PHÂN TÍCH YÊU CẦU.....	30
2.1 Xác định các use case và các tác nhân.....	30
2.1.1 Xác định tác nhân.....	30
2.1.2 Xác định các use case.....	30
2.2 Đặc tả game.....	31
CHƯƠNG III: THIẾT KẾ HỆ THỐNG.....	33
3.1 Sơ đồ use case tổng quát.....	33
3.2 Các use case.....	33
3.2.1 Use case Xem menu chính.....	33
3.2.2 Use case Xem menu thất bại.....	34
3.2.3 Use case Xem menu dừng.....	35
3.2.4 Use case Bắt đầu.....	36
3.2.5 Use case Tiếp tục.....	37
3.2.6 Use case Tiếp diễn.....	39
3.2.7 Use case Quay về menu chính.....	40
3.2.8 Use case Chơi lại.....	41
3.2.9 Use case Thoát.....	42
3.2.10 Use case Điều chỉnh âm lượng.....	43
3.3 Sơ đồ trạng thái.....	45
3.3.1 Trạng thái bật pause menu.....	45
3.3.2 Trạng thái bật dead menu.....	46
3.3.3 Trạng thái tấn công.....	47

3.3.4 Trạng thái qua cổng.....	48
3.4 Sơ đồ ERD.....	49
3.5 Sơ đồ PDM.....	49
3.6 Mô hình lược đồ quan hệ.....	50
3.7 Sơ đồ lớp.....	51
CHƯƠNG IV: HIỆN THỰC CODE VÀ KẾT QUẢ ĐẠT ĐƯỢC.....	52
4.1 Hiện thực bằng code.....	52
4.1.1 Về phía người chơi.....	52
4.1.1.1 Di chuyển nhân vật.....	52
4.1.1.2 Quay Camera.....	53
4.1.1.3 Hiệu ứng ảnh động và tốc độ di chuyển của nhân vật.....	54
4.1.2 Về phía hệ thống.....	57
4.1.2.1 Kẻ thù kiểm soát phạm vi lãnh thổ.....	57
4.1.2.2 Tính toán sát thương máu của người chơi.....	60
4.1.2.3 Thanh máu của người chơi.....	62
4.1.2.4 Tấn công bằng vũ khí súng.....	63
4.1.2.5 Thanh nhiệt của vũ khí súng.....	66
4.1.2.6 Thanh máu của quái vật.....	67
4.1.2.7 Quái vật tấn công người chơi.....	69
4.2 Một số hình ảnh trong game.....	71
4.2.1 Bản đồ chính và người chơi.....	71
4.2.2 Bản đồ khu vực đấu.....	71
4.2.2.1 Vòng đấu cấp 1.....	71
4.2.2.2 Vòng đấu cấp 2.....	72
4.2.2.3 Vòng đấu cấp 3.....	72
4.2.2.4 Vòng đấu cấp 4.....	73
4.2.2.5 Vòng đấu cấp 5.....	73
4.2.2.6 Vòng đấu cấp 6.....	74
4.2.2.7 Vòng đấu cấp 7.....	74

4.2.3 NPC.....	75
4.2.3.2 Trưởng làng.....	75
4.2.4 Kẻ thù.....	76
4.2.4.1 Lính canh.....	76
4.2.4.2 Lính hắc cầu.....	76
4.2.4.3 Zombie.....	77
4.2.4.4 Boss.....	77
4.2.5 Menu.....	78
4.2.5.1 Menu chính.....	78
4.2.5.2 Menu thua cuộc.....	78
4.2.5.3 Menu dừng.....	79
4.2.5.4 Menu kết thúc.....	79
CHƯƠNG V: KẾT LUẬN.....	80
PHÂN CÔNG VÀ ĐÁNH GIÁ.....	81
Phân công nhiệm vụ.....	81
Đánh giá thành viên.....	82
TÀI LIỆU THAM KHẢO.....	83

DANH MỤC KÝ HIỆU VÀ CHỮ VIẾT TẮT

KKL: Tên nhóm

HCSDL: Hệ cơ sở dữ liệu

DANH MỤC CÁC BẢNG BIỂU VÀ HÌNH

Danh mục Hình

Hình 1. Sơ đồ Use case tổng quát.....	33
Hình 2. Use case Xem menu chính.....	33
Hình 3. Use case Xem menu thất bại.....	34
Hình 4. Use case Xem menu dừng.....	35
Hình 5. Use case Bắt đầu.....	36
Hình 6. Use case Tiếp tục.....	38
Hình 7. Use case Tiếp diễn.....	39
Hình 8. Use case Quay về menu chính.....	40
Hình 9. Use case Chơi lại.....	41
Hình 10. Use case Thoát.....	42
Hình 11. Use case Điều chỉnh âm lượng.....	43
Hình 12. Trạng thái bật pause menu.....	45
Hình 13. Trạng thái bật dead menu.....	46
Hình 14. Trạng thái tấn công.....	47
Hình 15. Trạng thái qua cổng.....	48
Hình 16. Sơ đồ mô hình quan hệ thực thể - ERD.....	49
Hình 17. Sơ đồ mô hình vật lý - PDM.....	49
Hình 18. Sơ đồ lớp.....	51
Hình 19. Thiết lập lớp cơ sở hoạt ảnh chuyển động nhân vật.....	56
Hình 20. Thiết lập hoạt ảnh chuyển động nhân vật.....	57
Hình 21. Thêm thành phần NavMeshAgent cho quái vật.....	59
Hình 22. Thiết lập thông số lưới điều hướng.....	60
Hình 23. Thiết lập thanh máu nền.....	62
Hình 24. Thiết lập thanh máu hiện tại.....	62

Hình 25. Thiết lập thanh máu quái vật.....	67
Hình 26. Căng chỉnh thanh máu quái vật.....	68
Hình 27. Thiết lập va chạm trên tay quái vật.....	70
Hình 28. Bản đồ chính và người chơi.....	71
Hình 29. Vòng đấu cấp 1.....	71
Hình 31. Vòng đấu cấp 3.....	72
Hình 32. Vòng đấu cấp 4.....	73
Hình 33. Vòng đấu cấp 5.....	73
Hình 34. Vòng đấu cấp 6.....	74
Hình 35. Vòng đấu cấp 7.....	74
Hình 36. NPC Dân làng.....	75
Hình 37. NPC Trưởng làng.....	75
Hình 38. Lính canh.....	76
Hình 39. Lính hắc cầu.....	76
Hình 40. Bầy Zombie.....	77
Hình 41. Boss.....	77
Hình 42. Menu chính.....	78
Hình 43. Menu thua cuộc.....	78
Hình 44. Menu dừng.....	79
Hình 45. Menu kết thúc.....	79

Danh mục Bảng

Bảng 1. Các địa điểm trong game.....	19
Bảng 2. Các cấp độ trong game.....	19
Bảng 3. Các đối tượng trong game.....	20
Bảng 4. Các vũ khí trong game.....	20
Bảng 5. Các use case.....	31

Bảng 6. Đặc tả Use case Xem menu chính.....	34
Bảng 7. Đặc tả Use case Xem menu thất bại.....	35
Bảng 8. Đặc tả Use case Xem menu dừng.....	36
Bảng 9. Đặc tả Use case Bắt đầu.....	37
Bảng 10. Đặc tả Use case Tiếp tục.....	38
Bảng 11. Đặc tả Use case Tiếp diễn.....	39
Bảng 12. Đặc tả Use case Quay về menu chính.....	40
Bảng 13. Đặc tả Use case Chơi lại.....	41
Bảng 14. Đặc tả Use case Thoát.....	42
Bảng 15. Đặc tả Use case Điều chỉnh âm lượng.....	44

CHƯƠNG I: GIỚI THIỆU

1.1 Giới thiệu đề tài

Trong xã hội ngày càng phát triển với sự bùng phát mạnh mẽ từ các dịch vụ giải trí, nhu cầu về một game thú vị mới lạ, mang lại cảm giác thử thách người chơi với lối chơi mạo hiểm, lôi cuốn ngày càng được chú trọng. Và từ đó các game 3D ra đời. Để đáp ứng được nhu cầu trong bối cảnh thị trường hiện nay, KKL Software Team đã triển khai thực hiện dự án game 3D offline có cốt truyện với góc nhìn thứ ba mang tên Rummage Memories. Game được triển khai với mục tiêu giúp người chơi xả stress sau những ngày làm việc căng thẳng, tạo cảm giác muốn chinh phục trò chơi, thử thách bản thân cũng như sự khéo léo trong việc di chuyển, sử dụng vũ khí để tấn công tiêu diệt kẻ thù.

1.2 Phương pháp sử dụng

Nhóm sử dụng nền tảng ngôn ngữ C# và Game engine Unity để hiện thực game.

1.2.1 Game engine - Unity

Unity là một phần mềm làm game đa nền tảng được phát triển bởi Unity Technologies, hỗ trợ đồ họa 2D và 3D, các chức năng được viết chủ yếu qua ngôn ngữ C#. Nó cũng được xem là nền tảng hàng đầu thế giới để tạo và vận hành nội dung Realtime 3D tương tác trong thời gian thực, hỗ trợ tất cả các ứng dụng 3D chính và nhiều định dạng âm thanh, và thậm chí còn hiểu được định dạng .psd của Photoshop. chỉ có thể thả tệp .psd vào một dự án Unity, cho phép nhập và lắp ráp nội dung, viết mã để tương tác với các đối tượng trong game, tạo hoặc nhập hoạt ảnh để sử dụng với hệ thống hoạt ảnh nâng cao và hơn thế nữa.

Trong đó, Unity Asset Store được cho là thị trường tài sản tốt nhất trong thị trường game. Các tác phẩm nghệ thuật, mô hình 3D, tệp hoạt ảnh cho mô hình 3D, hiệu ứng âm thanh và bản nhạc đầy đủ, các hệ thống tạo kịch bản trực quan, các trình tạo bóng nâng cao, kết cấu, hiệu ứng hạt,...

Môi trường phát triển chạy trên nhiều hệ điều hành bao gồm Windows, macOS, Linux. Unity Engine có thể tạo trò chơi nhắm mục tiêu nền tảng PC, Web và di động.

1.2.2 Ngôn ngữ lập trình - C#

C# là một ngôn ngữ lập trình hướng đối tượng (OOP) được phát triển bởi Microsoft, được phát triển dựa trên C++ và Java với việc đáp ứng các nguyên tắc của ngành kỹ thuật phần mềm như kiểm tra chặt chẽ kiểu dữ liệu, kiểm tra giới hạn mảng, phát hiện các trường hợp sử dụng các biến chưa có dữ liệu, và tự động thu gom rác. Nó được sử dụng để phát triển các thành phần của phần mềm theo hướng thích hợp cho việc triển khai trong các môi trường phân tán.

1.3 Lý do và mục đích chọn đề tài

Chúng em chọn đề tài này là vì muốn có thêm kiến thức về game engine Unity và sử dụng ngôn ngữ C# để hiện thực một game 3D. Ngoài ra, loại hình game offline 3D có cốt truyện này đang ngày càng được phát triển và có chỗ đứng trong xã hội. Khi mà nhu cầu đòi hỏi về những game mang tính chinh phục, lôi cuốn, đồ họa đẹp mắt.

1.4 Thông tin về game

1.4.1 Tên game

Rummage Memories - Nơi những mảnh kí ức còn sót lại trong tiềm thức.

1.4.2 Bản quyền

Bản quyền thuộc về KKL Software Team

1.4.3 Phiên bản

Phiên bản 1.0 - Ban hành ngày 01/11/2020

Tác giả: KKL Software Team

1.4.4 Ý tưởng game

Rummage Memories - Nơi những mảnh kí ức còn sót lại trong tiềm thức lấy ý tưởng từ việc một cậu bé mang tên Kalus thuộc tộc người Heran. Tại Công nguyên Phantom, trong lúc di chuyển cùng cha của mình là Qatus trên phi thuyền số hiệu JAS-X thì bị bắn hạ bởi phi thuyền của Binh đoàn tộc thú (Zerg) khiến toàn bộ các thành viên phi thuyền tử nạn. Riêng Kalus còn sống và bị bắt giữ. Cậu bị giam trong căn phòng thí

nghiệm. Zerg đã truyền nọc Asla khiến cho cậu rơi vào trạng thái hôn mê liên tục và tiến hành khai thác những mảnh kí ức từ não bộ của cậu để truy tìm đến nơi cất giữ Norm – một loài kén bướm giúp tăng khả năng sinh trưởng và sức mạnh cho tộc thú gấp ba lần. Trong quá trình khai thác não bộ, tiềm thức của Kalus sống dậy và điều khiển ngăn chặn hành vi khai thác của Zerg. Tất cả quá trình này diễn ra trong vòng 7 ngày nhưng với thể giới kí ức mà Kalus đang đối diện là tận 7 năm. Cậu đã đấu tranh, tìm kiếm những sự trợ giúp của các bô lão (NPC) để được quay trở về cánh cổng thức tỉnh và trả thù cho người cha quá cố của mình và dòng tộc Heran.

1.4.5 Thể loại

Hành động, phiêu lưu, thử thách, chinh phục.

1.4.6 Đối tượng nhắm đến

Độ tuổi: 12+

1.4.7 Tóm tắt dòng game

Mở đầu game người chơi sẽ thấy màn hình hiển thị menu chính và chọn lựa chức năng New game để bắt đầu game, ngoài ra còn nhiều tính năng mở rộng khác. Người chơi sẽ bắt đầu game tại vị trí Sảnh và di chuyển lần lượt để tìm đến các cổng và tiêu diệt từng tên quái vật của tộc Zerg. Kết quả về các vòng đấu vẫn được lưu giữ lại, người chơi sẽ tiếp tục hành trình chinh phục của mình, giải thoát chính mình khỏi giấc mơ mà Kalus đang chìm đắm. Đối với nút “Exit Game” người chơi sẽ thoát khỏi ứng dụng. “Volume” sẽ giúp người chơi thay đổi âm lượng cho phù hợp với sở thích. Nếu người chơi bị quái vật tiêu diệt, màn hình sẽ hiển thị menu thất bại để người chơi có thể chơi lại. Trong quá trình chơi, người chơi có thể dừng game để nghỉ ngơi.

1.4.8 Giao diện

Giao diện cơ bản của game: Góc nhìn thứ ba

Hình ảnh trong game: Nhân vật Kalus tại Sảnh (Hall) lần lượt di chuyển đến 7 cổng (Gateway). Tương ứng với mỗi cổng là những bối cảnh vòng đấu khác nhau cũng như mức độ khó tăng dần lên. Trong quá trình di chuyển cần sự trợ giúp của NPC trưởng làng và dân làng để được hỗ trợ.

1.4.9 Phạm vi dự án

1.4.9.1 Số địa điểm

Địa điểm	Mô tả
Hall	Sảnh (giao diện ban đầu khi vào game)
Gateway 1	Cổng game thứ nhất (cấp độ Siêu Dễ)
Gateway 2	Cổng game thứ hai (cấp độ Dễ)
Gateway 3	Cổng game thứ ba (cấp độ Trung Bình)
Gateway 4	Cổng game thứ tư (cấp độ Khó)
Gateway 5	Cổng game thứ năm (cấp độ Siêu Khó)
Gateway 6	Cổng game thứ sáu (cấp độ Siêu Siêu Khó)
Gateway 7	Cổng game thứ bảy (cấp độ Kinh Hoàng)

Bảng 1. Các địa điểm trong game

1.4.9.2 Số cấp độ

Cấp độ	Mô tả
1	Siêu Dễ
2	Dễ
3	Trung Bình
4	Khó
5	Siêu Khó
6	Siêu Siêu Khó
7	Kinh Hoàng

Bảng 2. Các cấp độ trong game

1.4.9.3 Số đối tượng

Nhân vật	Mô tả
Kalus	Người chơi chính (Player)
Village Chief	Trưởng làng (NPC)

Villager	Dân làng (NPC)
Qatus	Cha của Kalus (NPC)
Zerg	Phản diện chính (Enemy)

Bảng 3. Các đối tượng trong game

1.4.9.4 Số vũ khí

Tên	Mô tả
Dao găm	Cận chiến
Súng nhỏ	Tấn công tầm xa (Sát thương thấp - giới hạn 20 viên)
Súng lớn	Tấn công tầm xa (Sát thương cao - giới hạn 50 viên)

Bảng 4. Các vũ khí trong game

1.4.10 Lối chơi và cơ chế

1.4.10.1 Lối chơi

- **Tiến trình game:** Tiến trình của người chơi sẽ được ghi nhận mỗi khi hoàn thành từng nhiệm vụ ở từng cấp độ. Khi hoàn thành tất cả các nhiệm vụ ở các cấp độ thì người chơi sẽ chiến thắng.
- **Cấu trúc nhiệm vụ:** Bản đồ game của từng cấp độ sẽ tạo ra những loại quái vật khác nhau tùy theo mức độ khó của cấp độ đó. Các quái vật sẽ được bố trí sẵn trong từng vòng đấu. Nhiệm vụ của người chơi là phải tiêu diệt các quái vật trong màn chơi.
- **Mục tiêu:** Ở mỗi cấp độ, người chơi sẽ phải tiêu diệt hết tất cả các loại quái vật xuất hiện trong màn chơi và hoàn thành nhiệm vụ, tiến đến cấp độ tiếp theo.
- **Môi trường:** Được thiết kế khá ổn và có thể điều hướng người chơi một cách tự nhiên, giúp người chơi có thể dễ dàng hiểu và nắm bắt được lối chơi của game.

1.4.10.2 Cơ chế

- **Vật lý:** Các chuyển động va chạm tương ứng

- Chuyển động:
 - + Chuyển động chung: Di chuyển (Move), Nhàn rỗi (Idle)
 - + Chuyển động khác: Tấn công (Firing)
- Hành động:
 - + Điều khiển:
 - Di chuyển: tổ hợp phím [\leftarrow \uparrow \downarrow \rightarrow] hoặc tổ hợp phím [W,A,S,D]
 - Nhảy: phím [Space]
 - Chạy: phím [Shift]
 - + Tùy chọn khác:
 - Khi người chơi truy cập ứng dụng game Rummage Memories sẽ hiển thị Menu chính (Main Menu) gồm Bắt đầu (New Game), Tiếp tục (Load Game), Thoát (Exit Game), Điều chỉnh âm lượng (Volume).
 - Khi đang chơi game, người chơi sẽ có các tính năng tùy chọn khi nhấn nút ESC thì trò chơi sẽ dừng lại tại thời điểm đó và hiển thị Menu Dừng (Pause Menu) gồm các chức năng như Tiếp diễn (Continue), Quay về menu chính (Back Main Menu), Exit Game.
 - Khi người chơi chết sẽ hiển thị màn hình Menu Thất Bại (Lose Menu) gồm những chức năng Chơi Lại (Retry), Back Main Menu, Exit Game.
 - + Phát lại và lưu trữ: Cơ sở dữ liệu

1.4.11 Thế giới game

1.4.11.1 Chi tiết về địa điểm

- **Địa điểm 1:**
 - + Đặc tả chung: Gateway 1 là một cánh cổng không gian đưa người chơi đến thử thách đầu tiên trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách
 - + Cấp độ để tham gia địa điểm: Cấp độ 1

- + Liên kết đến các địa điểm khác: Cổng đầu tiên được mở sau khi bắt đầu trò chơi
- **Địa điểm 2:**
 - + Đặc tả chung: Gateway 2 là một cánh cổng không gian đưa người chơi đến thử thách thứ hai trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.
 - + Cấp độ để tham gia địa điểm: Cấp độ 2.
 - + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 1.
- **Địa điểm 3:**
 - + Đặc tả chung: Gateway 3 là một cánh cổng không gian đưa người chơi đến thử thách thứ ba trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.
 - + Cấp độ để tham gia địa điểm: Cấp độ 3.
 - + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 2.
- **Địa điểm 4:**
 - + Đặc tả chung: Gateway 4 là một cánh cổng không gian đưa người chơi đến thử thách thứ tư trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.
 - + Cấp độ để tham gia địa điểm: Cấp độ 4.
 - + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 3.
- **Địa điểm 5:**
 - + Đặc tả chung: Gateway 5 là một cánh cổng không gian đưa người chơi đến thử thách thứ tư trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.

- + Cấp độ để tham gia địa điểm: Cấp độ 5.
- + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 4.
- **Địa điểm 6:**
 - + Đặc tả chung: Gateway 6 là một cánh cổng không gian đưa người chơi đến thử thách thứ tư trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.
 - + Cấp độ để tham gia địa điểm: Cấp độ 6.
 - + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 5.
- **Địa điểm 7:**
 - + Đặc tả chung: Gateway 7 là một cánh cổng không gian đưa người chơi đến thử thách thứ tư trong quá trình tìm lại kí ức.
 - + Đặc trưng vật lý: Nhân vật đi xuyên vào cánh cổng để dịch chuyển đến địa điểm thử thách.
 - + Cấp độ để tham gia địa điểm: Cấp độ 7.
 - + Liên kết đến các địa điểm khác: Mở sau khi vượt qua thử thách ở Gateway 6.

1.4.11.2 Chi tiết về nhân vật

- **Nhân vật 1:**
 - + Câu chuyện nền: Kalus là con trai duy nhất của Qatus tộc người Heran. Cậu sống trong một ngôi làng hòa bình và yên tĩnh cùng với cha. Theo như lời tiên tri cổ xưa của tộc Heran, cậu chính là người nắm giữ bí mật về nơi cất giấu Norm. Sau khi thoát khỏi Zerg, cậu đi tìm sức mạnh và trả thù cho cha mình.
 - + Chi tiết nhân vật: Ngày sinh: 10-10-2301
Giới tính: Nam
Tộc: Heran

- + Tính cách nhân vật: Kalus là một cậu trẻ con có tính cách hơi cao ngạo nhưng lại tốt bụng và thông minh. Được cha mình rèn luyện từ bé cậu mang trong mình lòng dũng cảm và chính nghĩa.
- + Mối liên hệ với cốt truyện: Nhân vật chính
- + Mối liên hệ với các nhân vật khác: Con trai Qatus, bị Zerg săn đuổi.

- **Nhân vật 2:**

- + Câu chuyện nền: Qatus – cha của Kalus, thời trẻ ông là một người lính, sau khi trở về nhà ông bắt đầu trồng trọt và trở thành một nông dân.
- + Chi tiết nhân vật: Ngày sinh: 12-01-2267
Giới tính: Nam
Tộc: Heran
- + Tính cách nhân vật: Mạnh mẽ và luôn làm mọi thứ theo sở thích
- + Mối liên hệ với cốt truyện: Qatus bị tộc Zerg sát hại trong lúc chúng tấn công phi thuyền để bắt giữ Katus
- + Mối liên hệ với các nhân vật khác: Qatus là cha của Kalus

- **Nhân vật 3:**

- + Câu chuyện nền: Village Chief – trưởng làng, người đưa ngôi làng tộc Heran đến với thời kì hòa bình và phát triển
- + Chi tiết nhân vật: Ngày sinh: 10-05-2210
Giới tính: Nam
Tộc: Heran
- + Tính cách nhân vật: Trưởng làng là một người thông thái và nhân từ. Khát vọng hòa bình đã khiến ông đứng lên dẫn dắt tộc Heran vượt qua khó khăn.
- + Mối liên hệ với cốt truyện: Người lãnh đạo của tộc Heran
- + Mối liên hệ với các nhân vật khác: Người hướng dẫn cho Kalus tìm lại chính mình

- **Nhân vật 4:**

- + Câu chuyện nền: Villager – dân làng, nhóm người sinh sống trong ngôi làng trên đồi núi và khát khao hòa bình thuộc tộc Heran.
- + Chi tiết nhân vật: Ngày sinh: 02-02-2200

Giới tính: Nam

Tộc: Heran

- + Tính cách nhân vật: Dân làng là nhóm người hiền lành chất phác cùng với khát vọng hòa bình giống trưởng làng.
- + Mối liên hệ với cốt truyện: Nhóm người đi cùng trưởng làng lấy lại hòa bình cho tộc Heran
- + Mối liên hệ với các nhân vật khác: Người hướng dẫn cho Kalus tìm lại chính mình

- **Nhân vật 5:**

- + Câu chuyện nền: Zerg nói chung là quái thú có tư duy và trí tuệ. Chúng luôn khao khát và tìm kiếm sức mạnh nên đi chinh chiến khắp nơi trong vũ trụ. Sau khi nghe được tin tức về Norm chúng tiến đến tấn công phi thuyền tộc Heran.
- + Chi tiết nhân vật: Ngày sinh: xxx
Giới tính: xxx
Tộc: Zerg
- + Tính cách nhân vật: Tộc Zerg hiếu chiến và khao khát sức mạnh, sẵn sàng chiếm đoạt thứ chúng muốn.
- + Mối liên hệ với cốt truyện: Phản diện chính
- + Mối liên hệ với các nhân vật khác: Zerg tấn công phi thuyền tộc Heran và bắt Kalus để tìm tung tích của Norm.

1.4.11.3 Chi tiết về cấp độ

- **Cấp độ 1:**

- + Tóm tắt: Cấp độ “Siêu Dễ” giúp người chơi hiểu rõ hơn về game thông qua việc tiêu diệt một vài tên quái yếu ớt, trả nhiệm vụ và bước sang cổng game tiếp theo.
- + Tài liệu giới thiệu: Kalus thức tỉnh trong một vùng đất hoang tàn (Hall), cậu đi khắp nơi để tìm hiểu vì sao mình ở đây và cậu gặp được một dân làng ven đường (Villager). Cậu được hướng dẫn đi đến cổng game thứ

nhất (Gateway 1) để biết xem chuyện gì đang xảy ra. Tại đây, cậu phải đối mặt với 3 tên quái là lính canh của bộ tộc Zerg.

- + Mục tiêu: Gặp được NPC dân làng nhận nhiệm vụ tiêu diệt 3 tên quái lính canh. Biết về cách sử dụng súng lớn.
- + Mô tả vật lý: Kalus khi bước vào Gateway 1 sẽ sang một bản đồ thu nhỏ (Mini map). Lần lượt di chuyển để gặp từng tên quái Lính Canh của Zerg trong những con đường mòn trong rừng rậm.
- + Bản đồ: Bối cảnh bản đồ “Khu Rừng Âm U”.
- + Phần cốt lõi để chiến thắng: Tiêu diệt 3 tên quái lính canh.
- + Những cuộc gặp gỡ: NPC dân làng (Villager). 3 tên quái lính canh ở Khu Rừng Âm U.

- **Cấp độ 2:**

- + Tóm tắt: Cấp độ “Dễ” này người chơi đã biết được mình cần làm gì nhưng sẽ tăng số lượng quái vật lên gấp đôi cấp độ Siêu Dễ. Đồng thời tốc độ tấn công nhanh và sát thương của chúng cũng được tăng lên 15%.
- + Tài liệu giới thiệu: Kalus phải đối mặt với 6 tên quái nhưng lần này là lính Hắc Cầu của bộ tộc Zerg với sự hung hăng và điên loạn. Chúng tấn công rất nhanh và sát thương từ bộ răng và vuốt bén.
- + Mục tiêu: Gặp được NPC dân làng nhận nhiệm vụ tiêu diệt 6 tên quái lính Hắc Cầu bằng súng lớn.
- + Mô tả vật lý: Kalus khi bước vào Gateway 2 sẽ sang một bản đồ thu nhỏ (Mini map). Lần lượt di chuyển để gặp từng tên quái là lính Hắc Cầu trong hoang mạc khô cằn trơ trụi.
- + Bản đồ: Bối cảnh bản đồ “Sa Mạc Hoang Tàn”.
- + Phần cốt lõi để chiến thắng: Tiêu diệt 6 tên quái lính Hắc Cầu. Biết cách kiểm soát số lượng đạn của súng lớn (50 viên).
- + Những cuộc gặp gỡ: NPC dân làng (Villager). 6 tên quái lính Hắc Cầu ở hoang mạc.

- **Cấp độ 3:**

- + Tóm tắt: Cấp độ “Trung Bình” này người chơi sẽ gặp Bầy Zombie hung bạo. Tốc độ của chúng rất rất nhanh và sát thương cao hơn 25%.

- + Tài liệu giới thiệu: Kalus phải đối mặt với 10 tên Zombie của bộ tộc Zerg với sự hung hăng máu lửa bởi khả năng di chuyển rất nhanh và điên loạn.
- + Mục tiêu: Tiêu diệt 10 tên Zombie bằng súng lớn.
- + Mô tả vật lý: Kalus khi bước vào Gateway 3 sẽ sang một bản đồ thu nhỏ (Mini map). Lần lượt di chuyển để gặp từng tên Zombie trong thị trấn đầy tuyết.
- + Bản đồ: Bối cảnh bản đồ “Thị Trấn Băng Giá”.
- + Phần cốt lõi để chiến thắng: Tiêu diệt 10 tên Zombie. Biết cách kiểm soát số lượng đạn của súng lớn (50 viên).
- + Những cuộc gặp gỡ: 10 tên Zombie ở thị trấn băng giá

- Cấp độ 4:

- + Tóm tắt: Cấp độ “Khó” này người chơi sẽ vẫn gặp Bầy Zombie hung bạo cùng đám Hắc Cầu hung hăng.
- + Tài liệu giới thiệu: Kalus phải đối mặt với 8 tên Zombie và 6 tên Hắc Cầu của bộ tộc Zerg với sự hung hăng máu lửa bởi khả năng di chuyển rất nhanh và điên loạn.
- + Mục tiêu: Tiêu diệt 8 tên Zombie và 6 tên Hắc Cầu bằng súng nhỏ.
- + Mô tả vật lý: Kalus khi bước vào Gateway 4 sẽ sang một bản đồ thu nhỏ (Mini map). Lần lượt di chuyển để gặp từng tên Zombie và lính Hắc Cầu trong bãi đất trống ven thị trấn.
- + Bản đồ: Bối cảnh bản đồ một “Bãi Đất Trống Ven Thị Trấn”.
- + Phần cốt lõi để chiến thắng: Tiêu diệt 8 tên Zombie và 6 tên Hắc Cầu bằng súng nhỏ. Biết cách kiểm soát số lượng đạn của súng nhỏ (10 viên).
- + Những cuộc gặp gỡ: 8 tên Zombie , 6 lính Hắc Cầu ở bãi đất trống ven thị trấn.

- Cấp độ 5:

- + Tóm tắt: Cấp độ “Siêu Khó” này người chơi sẽ gặp gã Bụng Phệ chuyên ăn thịt người với vũ khí là một khúc gỗ lớn.

- + Tài liệu giới thiệu: Kalus phải đối mặt với 1 tên quái to lớn là gã Bụng Phệ của bộ tộc Zerg với sự hung hăng và điên loạn. Nó sử dụng khúc gỗ lớn để tấn công Kalus. Tuy nhiên, gã Bụng Phệ di chuyển rất chậm chạp.
- + Mục tiêu: Tiêu diệt gã Bụng Phệ bằng súng nhỏ.
- + Mô tả vật lý: Kalus khi bước vào Gateway 5 sẽ sang một bản đồ thu nhỏ (Mini map). Bối cảnh là một căn phòng kín đen tối rùng rợn. Một gã Bụng Phệ trên tay cầm một cây gỗ đang chờ đợi cậu.
- + Bản đồ: Bối cảnh bản đồ một “Căn Phòng Kín”.
- + Phần cốt lõi để chiến thắng: Di chuyển khéo léo và kiểm soát lượng đạn của súng nhỏ (10 viên)
- + Những cuộc gặp gỡ: 1 gã Bụng Phệ.

- **Cấp độ 6:**

- + Tóm tắt: Cấp độ “Siêu Siêu Khó” này người chơi sẽ phải đối đầu với quái vật Bụng Phệ chuyên ăn thịt người cùng bầy thuộc hạ của hắn.
- + Tài liệu giới thiệu: Kalus phải đối mặt với 1 tên quái to lớn là gã Bụng Phệ cùng bầy thuộc hạ của bộ tộc Zerg với sự hung hăng và điên loạn. Sự chậm chạp của gã Bụng Phệ đã được bù đắp bằng số lượng thuộc hạ rất nhanh nhẹn quanh hắn.
- + Mục tiêu: Gặp NPC Trưởng làng, Tiêu diệt gã Bụng Phệ và bầy thuộc hạ.
- + Mô tả vật lý: Kalus khi bước vào Gateway 6 sẽ sang một bản đồ thu nhỏ (Mini map). Bối cảnh là nơi lần đầu tiên cậu chìm vào hôn mê ... Khu Rừng Âm U. Một gã Bụng Phệ trên tay cầm một cây gỗ cùng đám thuộc hạ là bọn lính canh rất đông đang chờ đợi cậu.
- + Bản đồ: Bối cảnh bản đồ một “Khu Rừng Âm U”.
- + Phần cốt lõi để chiến thắng: Di chuyển thật nhanh tiêu diệt bọn thuộc hạ và gã Bụng Phệ một cách khéo léo bằng chiếc dao găm
- + Những cuộc gặp gỡ: NPC Trưởng làng. 1 gã quái vật Bụng Phệ và 5 lính canh, Bầy zombie

- **Cấp độ 7:**

- + Tóm tắt: Cấp độ “Ám Ảnh Kinh Hoàng” là cấp độ cuối cùng của trò chơi. Người chơi sẽ bước vào Đầm Lầy Quỷ Dị. Nơi cất giữ kén bướm (Norm) trong kí ức của cậu cũng chính là con đường để cậu quay trở về thế giới thực tại. Kết thúc chuỗi ngày hôn mê.
- + Tài liệu giới thiệu: Kalus mở bức thư của Trưởng làng , trong đó có chỉ đường đến Norm. Nhưng điều trắc trở chính là bộ tộc Zerg đã can thiệp vào kí ức cậu, tiếp cận đến gần Norm trong Đầm Lầy Quỷ Dị. Nhiệm vụ của cậu là tiêu diệt toàn bộ chúng, bảo vệ Norm. Tuy nhiên, trên tay cậu chỉ còn lại là một chiếc dao găm nhỏ.
- + Mục tiêu: Tiêu diệt toàn bộ quái còn lại trong cấp độ cuối cùng.
- + Mô tả vật lý: Kalus khi bước vào Gateway 7 sẽ sang một bản đồ thu nhỏ (Mini map). Một đầm lầy rừng rợn hiện ra. Kalus cần vượt qua bọn Zerg xấu xa để đến nơi cất giữ Norm.
- + Bản đồ: Bối cảnh bản đồ một “Đầm Lầy Quỷ Dị”.
- + Phần cốt lõi để chiến thắng: Khéo léo tiêu diệt toàn bộ tộc thú Zerg bằng dao găm.
- + Những cuộc gặp gỡ: 1 gã quái vật Bụng Phệ, 5 lính canh , 2 lính Hắc Cầu, Bầy Zombie, NPC người cha quá cố (Qatus).

CHƯƠNG II: PHÂN TÍCH YÊU CẦU

2.1 Xác định các use case và các tác nhân

2.1.1 Xác định tác nhân

Sau khi tìm hiểu, thu thập thông tin từ một số loại Game offline 3D có cốt truyện, chúng em đã đúc kết được tác nhân của game Rummage Memories chỉ có một - đó là người chơi (Kalus)

2.1.2 Xác định các use case

Dựa vào yêu cầu đặc ra, và với các tác nhân tương ứng chúng em xác định use case cho game offline này gồm:

ID	Use case	Mô tả
VMM	Xem menu chính (Main Menu)	Xem các chức năng của game chính
VLM	Xem menu thất bại (Lose Menu)	Xem các chức năng của game thất bại
VPM	Xem menu dừng (Pause Menu)	Xem các chức năng của game dừng
SG	Bắt đầu (New Game)	Chọn và bắt đầu chơi game
CG	Tiếp tục (Load Game)	Chọn và tiếp tục game

EG	Thoát (Exit Game)	Chọn và thoát game
RP	Chơi lại (Retry)	Chọn và chơi lại vòng đấu đó
BMM	Quay về menu chính (Back Main Menu)	Chọn và điều hướng sang menu chính
CN	Tiếp diễn (Continue)	Chọn và game vẫn tiếp diễn
AV	Điều chỉnh âm lượng (Volume)	Chọn và kéo để thay đổi âm lượng nhạc nền

Bảng 5. Các use case

2.2 Đặc tả game

Game offline 3D có cốt truyện - Rummage Memories cung cấp dịch vụ cho đối tượng người chơi trên 12 tuổi. Do đó, bối cảnh game cần có sự hấp dẫn, lôi cuốn, mang nét cổ điển đan xen hiện đại, nhân vật và quái vật thể hiện rõ những đặc tính riêng và linh hoạt nhất có thể, nhạc nền sống động. Các chức năng hệ thống phải được tổng hợp tại phần menu game. Giao diện game dễ sử dụng, luật chơi đơn giản nhưng không kém phần thử thách. Người chơi sau khi nhấn vào ứng dụng Rummage Memories sẽ vào trực tiếp menu game mà không cần tài khoản đăng nhập.

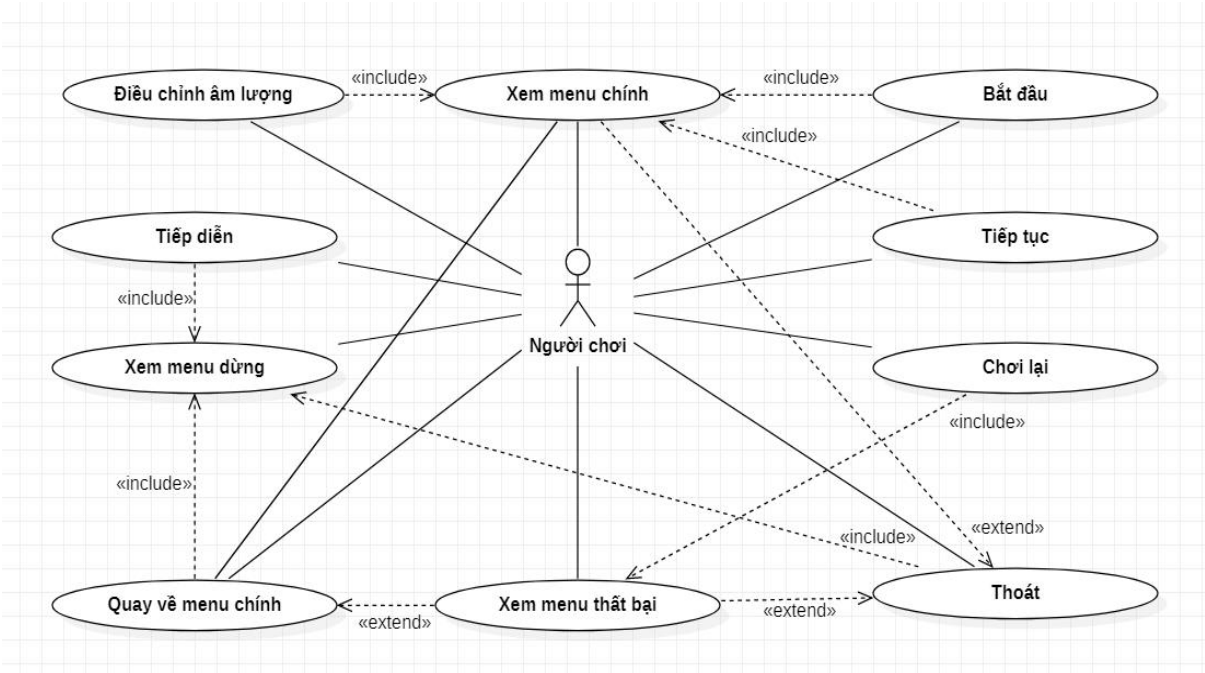
Game cho phép một số chức năng như Xem menu chính (Main Menu) về các chức năng của game như Bắt đầu (New Game), Tiếp tục (Load Game), Thoát (Exit Game), Điều chỉnh âm lượng (Volume). Riêng chức năng Xem menu khi thua cuộc (Lose menu) gồm sẽ được hiển thị sau khi người chơi bị giết chết.

Đối với chức năng Bắt đầu (New Game), người chơi sẽ vào trực tiếp game ở vị trí Sảnh (Hall) trong bản đồ chính và thông tin về trạng thái của các vòng đấu ở từng cổng (Gateways) sẽ được đặt về 0. Đối với chức năng Tiếp tục (Load Game) thì người vẫn được chuyển đến vị trí Hall nhưng thông tin về diễn biến trận đấu vẫn được lưu trữ và trích xuất ra từ HCSDL. Đối với chức năng Thoát (Exit Game), người chơi sẽ thoát hoàn toàn ứng dụng và thông tin về điểm số và cấp độ cũng sẽ được lưu lại. Đối với chức năng Điều chỉnh âm lượng (Volume) sẽ được hiển thị dưới dạng thanh kéo ngang trong khoảng từ 0 đến 100 tương ứng với số phần trăm âm lượng nhạc nền game. Khi chết sẽ hiển thị menu thua cuộc với các chức năng như Chơi Lại (Retry), quay lại menu chính (Back Main Menu), Exit Game. Ngoài ra còn có menu dừng khi người chơi nhấn ESC với chức năng như Tiếp diễn (Continuous), Back Main Menu, Exit Game.

Một số sự kiện trong game như quái vật tộc Zerg, Cổng game (Gateway) sẽ được bố trí theo từng cấp độ. NPC trưởng làng, NPC dân làng sẽ được đặt tại 2 nơi rất xa Sảnh. Đối với quái vật tộc Zerg sẽ gồm có Lính Canh, Lính Hắc Cầu và Gã Bụng Phệ. NPC sẽ giúp người chơi biết mình cần làm gì tiếp theo. Đối với Cổng game sẽ điều hướng qua các cấp độ tiếp theo.

CHƯƠNG III: THIẾT KẾ HỆ THỐNG

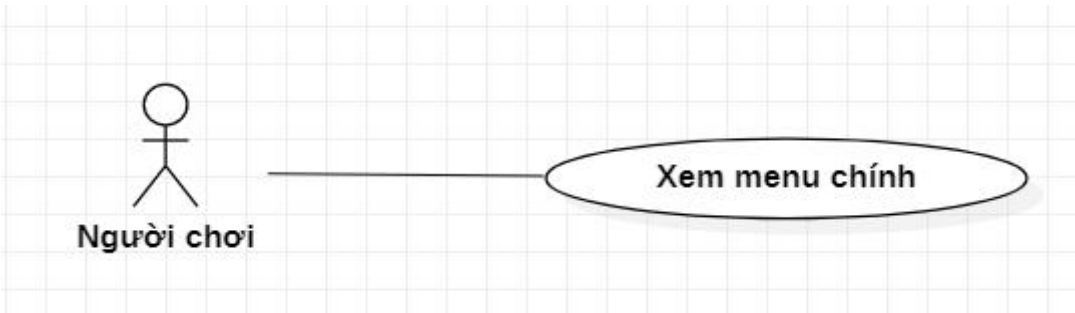
3.1 Sơ đồ use case tổng quát



Hình 1. Sơ đồ Use case tổng quát

3.2 Các use case

3.2.1 Use case Xem menu chính



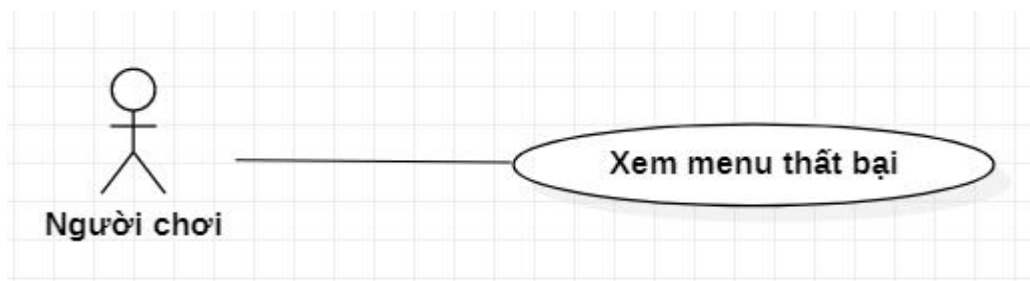
Hình 2. Use case Xem menu chính

Use Case Name	Xem menu chính
Scenario	Ứng dụng Rummage Memories được mở

Triggering event	Truy cập ứng dụng Rummage Memories	
Brief description	Người chơi dùng chức năng này để thao tác chọn lựa cho các chức năng khác trước khi vào game.	
Actors	Người chơi	
Related Use Case	Bắt đầu, Tiếp tục, Điều chỉnh âm lượng, Thoát	
Stakeholder	Không	
Precondition	Truy cập được vào ứng dụng game	
Postcondition	Không	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn vào nút ứng dụng game Rummage Memories.	1.1 Hệ thống hiển thị lên giao diện menu chính để lựa chọn
Exception condition	Không	

Bảng 6. Đặc tả Use case Xem menu chính

3.2.2 Use case Xem menu thất bại



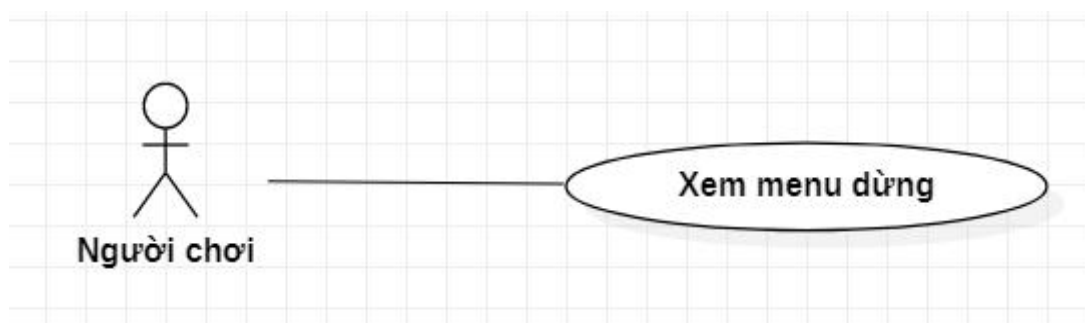
Hình 3. Use case Xem menu thất bại

Use Case Name	Xem menu thất bại
Scenario	Khi người chơi bị chết

Triggering event	Máu người chơi bằng 0	
Brief description	Menu thất bại được hiển thị ra khi người chơi bị chết thông qua máu bằng 0. Người chơi dùng chức năng này để thao tác chọn lựa cho các chức năng khác trước khi chết.	
Actors	Người chơi	
Related Use Case	Chơi lại, Quay về menu chính, Thoát	
Stakeholder	Không	
Precondition	Máu người chơi bằng 0	
Postcondition	Không	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi bị giết	1.1 Hệ thống hiển thị lên giao diện menu thất bại
Exception condition	Không	

Bảng 7. Đặc tả Use case Xem menu thất bại

3.2.3 Use case Xem menu dừng



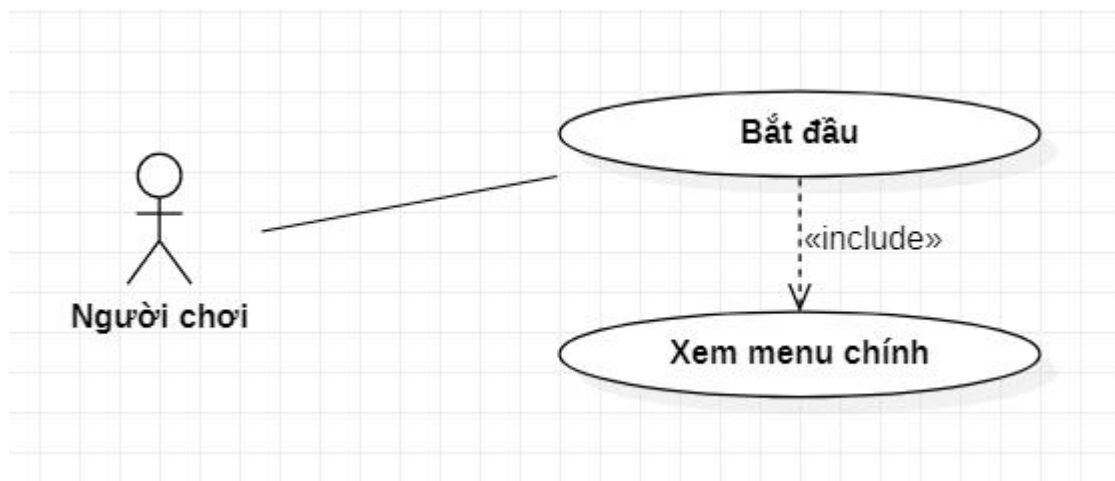
Hình 4. Use case Xem menu dừng

Use Case Name	Xem menu dừng
Scenario	Khi người chơi muốn dừng trận đấu để nghỉ ngơi hoặc muốn thực hiện một số thao tác khác

Triggering event	Người chơi nhấn nút ESC	
Brief description	Khi người chơi nhấn nút ESC. Menu dừng sẽ hiển thị lên và trận đấu được dừng lại. Ở đây, người chơi có thể dùng một số thao tác khác được tích hợp sẵn.	
Actors	Người chơi	
Related Use Case	Tiếp diễn, Quay về menu chính, Thoát	
Stakeholder	Không	
Precondition	Truy cập được vào game	
Postcondition	Không	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn nút ESC để dừng trận đấu	1.1 Hệ thống hiển thị lên giao diện menu dừng
Exception condition	Không	

Bảng 8. Đặc tả Use case Xem menu dừng

3.2.4 Use case Bắt đầu

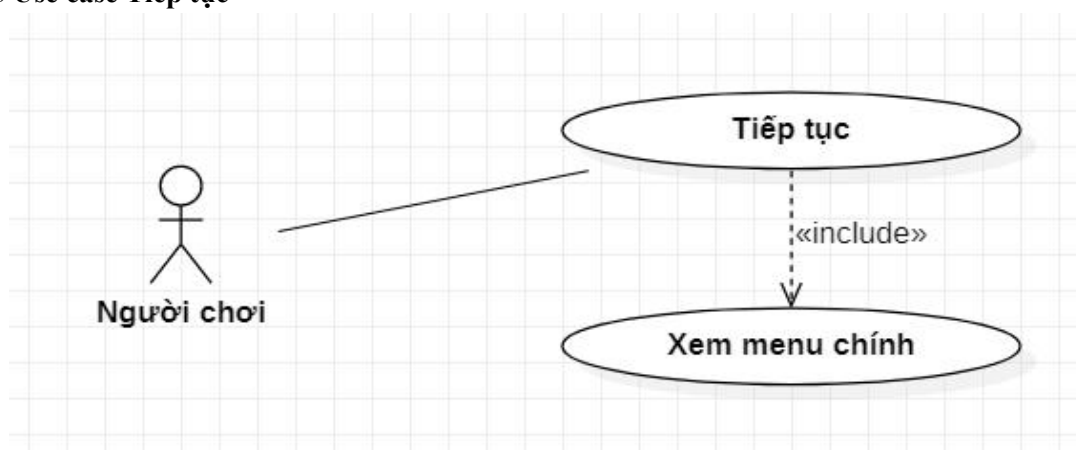


Hình 5. Use case Bắt đầu

Use Case Name	Bắt đầu	
Scenario	Nhấn vào nút Start Game trong menu	
Triggering event	Trang Xem menu chính đã được hiển thị xong	
Brief description	Người chơi dùng chức năng này để truy cập vào game	
Actors	Người chơi	
Related Use Case	Xem menu chính	
Stakeholder	Không	
Precondition	Menu game đã load lên đầy đủ	
Postcondition	Chỉ nhấn bắt đầu game được khi vào đã giao diện menu chính	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn vào nút Start Game	1.1 Hệ thống sẽ điều hướng sang scene chính (main scene), trạng thái tiến trình game được đặt về 0.
Exception condition	Không	

Bảng 9. Đặc tả Use case Bắt đầu

3.2.5 Use case Tiếp tục

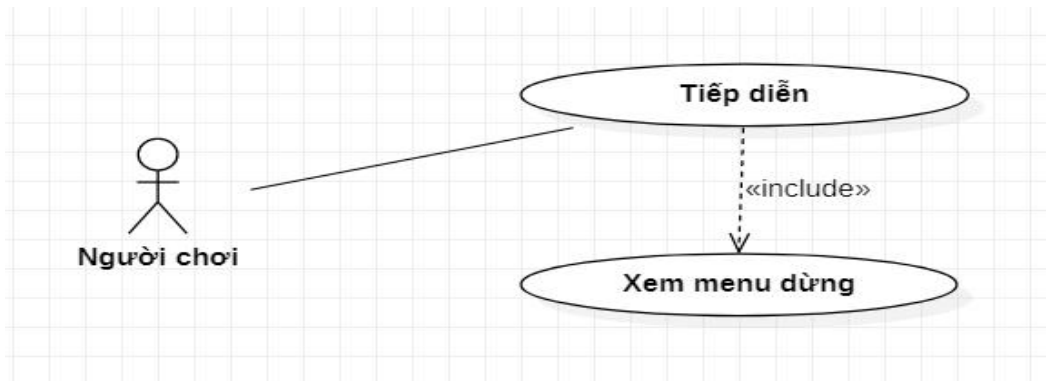


Hình 6. Use case Tiếp tục

Use Case Name	Tiếp tục	
Scenario	Nhấn vào nút Load Game trong menu chính	
Triggering event	Trang Xem menu chính đã được hiển thị xong	
Brief description	Người chơi dùng chức năng này để truy cập vào game với trạng thái đã được lưu trước đó.	
Actors	Người chơi	
Related Use Case	Xem menu chính	
Stakeholder	Không	
Precondition	Menu chính đã load lên đầy đủ	
Postcondition	Chỉ nhấn tiếp tục game được khi vào đã giao diện menu chính	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn vào nút Load Game	1.1 Hệ thống sẽ điều hướng sang scene chính (main scene), trạng thái của game tại lần cuối cùng sẽ được trích xuất từ database.
Exception condition	Không	

Bảng 10. Đặc tả Use case Tiếp tục

3.2.6 Use case Tiếp diễn

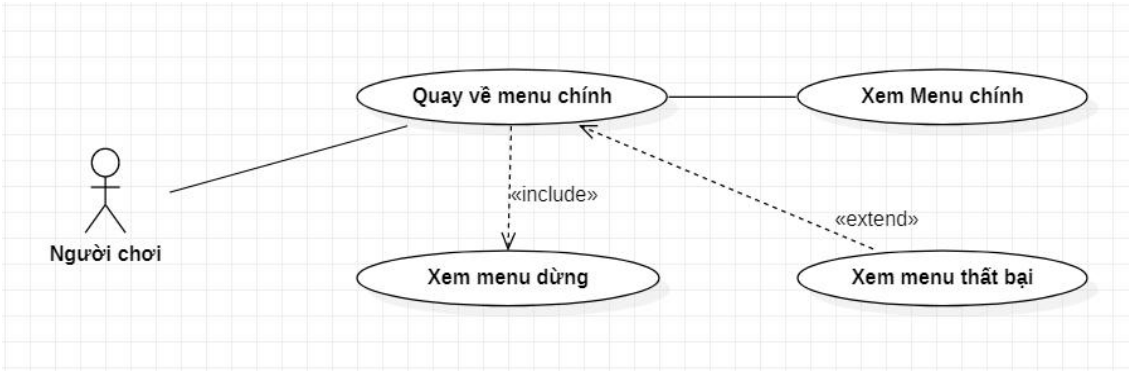


Hình 7. Use case Tiếp diễn

Use Case Name	Tiếp diễn	
Scenario	Nhấn vào nút Continue trong menu dừng	
Triggering event	Trang Xem menu dừng đã được hiển thị xong	
Brief description	Người chơi dùng chức năng này để tiếp tục chơi game với trạng thái đã được lưu trước đó.	
Actors	Người chơi	
Related Use Case	Xem menu dừng	
Stakeholder	Không	
Precondition	Menu dừng đã load lên đầy đủ	
Postcondition	Chỉ nhấn tiếp tục game được khi vào đã giao diện menu chính	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn vào nút Continue	1.1 Hệ thống sẽ tắt menu dừng
Exception condition	Không	

Bảng 11. Đặc tả Use case Tiếp diễn

3.2.7 Use case Quay về menu chính

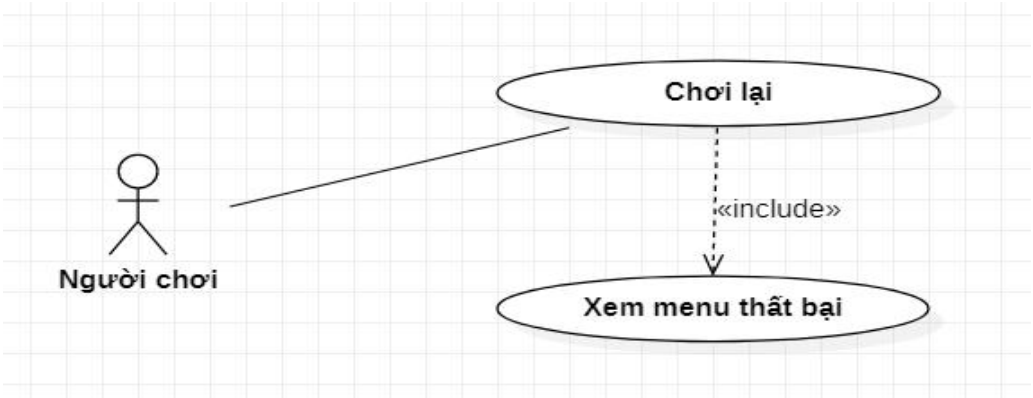


Hình 8. Use case Quay về menu chính

Use Case Name	Quay về menu chính	
Scenario	Nhấn vào nút Back Main Menu trong menu dừng hoặc menu thất bại	
Triggering event	Xem menu dừng/Xem menu thất bại đã được hiển thị xong	
Brief description	Người chơi dùng chức năng này để quay trở về menu chính	
Actors	Người chơi	
Related Use Case	Xem menu dừng, Xem menu chính, Xem menu thất bại	
Stakeholder	Không	
Precondition	Menu dừng đã load lên đầy đủ	
Postcondition	Chỉ nhấn tiếp tục game được khi vào đã giao diện menu chính	
Flow of activities	Người dùng	Hệ thống
	1.Người chơi nhấn vào nút Continue	1.1 Hệ thống sẽ điều hướng sang menu chính
Exception condition	Không	

Bảng 12. Đặc tả Use case Quay về menu chính

3.2.8 Use case Chơi lại

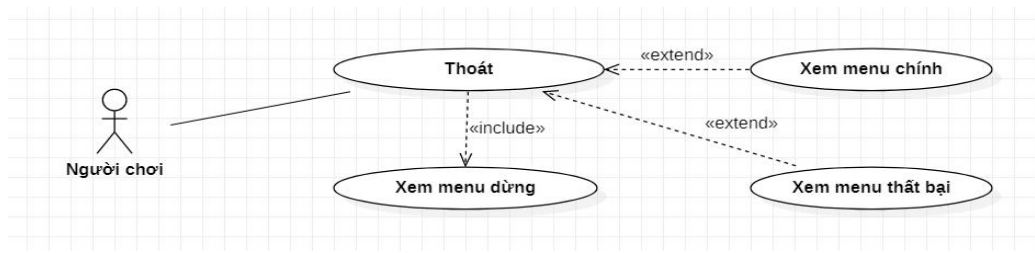


Hình 9. Use case Chơi lại

Use Case Name	Chơi lại	
Scenario	Nhấn vào nút Retry trong menu thất bại	
Triggering event	Xem menu thất bại đã được hiển thị xong	
Brief description	Người chơi dùng chức năng này để chơi lại vòng đấu	
Actors	Người chơi	
Related Use Case	Xem menu thất bại	
Stakeholder	Không	
Precondition	Menu thất bại đã load lên đầy đủ	
Postcondition	Chỉ nhấn tiếp tục game được khi vào đã giao diện menu chính	
Flow of activities	Người dùng	Hệ thống
	1.Người chơi nhấn vào nút Retry	1.1 Hệ thống cho người chơi chơi lại vòng đấu trước đó
Exception condition	Không	

Bảng 13. Đặc tả Use case Chơi lại

3.2.9 Use case Thoát

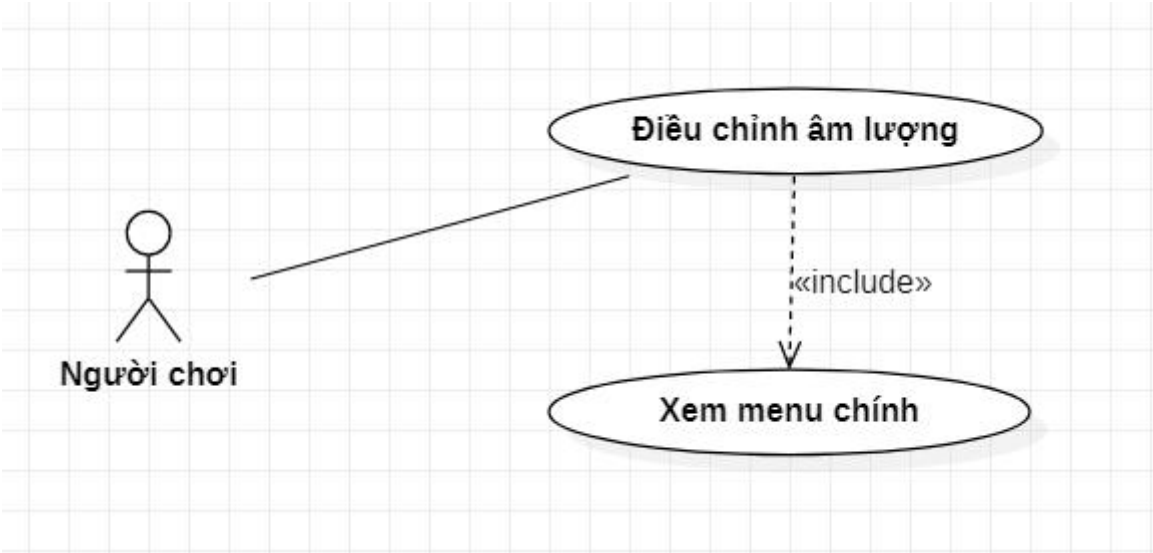


Hình 10. Use case Thoát

Use Case Name	Thoát game	
Scenario	Trong trường hợp người chơi ở menu chính hoặc menu thất bại, người chơi chỉ việc nhấn Exit Game để thoát. Trong trường hợp người chơi có chủ ý dừng trò chơi thông qua phím ESC với menu dừng, phím Exit Game có chức năng tương tự.	
Triggering event	Xem menu chính/Xem menu thất bại/Xem manu dừng đã được hiển thị	
Brief description	Người chơi dùng chức năng này để thoát khỏi ứng dụng	
Actors	Người chơi	
Related Use Case	Xem menu chính, Xem menu thất bại, Xem menu dừng	
Stakeholder	Không	
Precondition	Menu chính/Menu thất bại/Menu dừng đã load lên đầy đủ	
Postcondition	Chỉ nhấn thoát game được khi vào giao diện menu game	
Flow of activities	Người dùng	Hệ thống
	1.Người chơi nhấn vào nút Exit Game	1.1 Hệ thống sẽ chuyển người chơi ra khỏi ứng dụng
Exception condition	Không	

Bảng 14. Đặc tả Use case Thoát

3.2.10 Use case Điều chỉnh âm lượng



Hình 11. Use case Điều chỉnh âm lượng

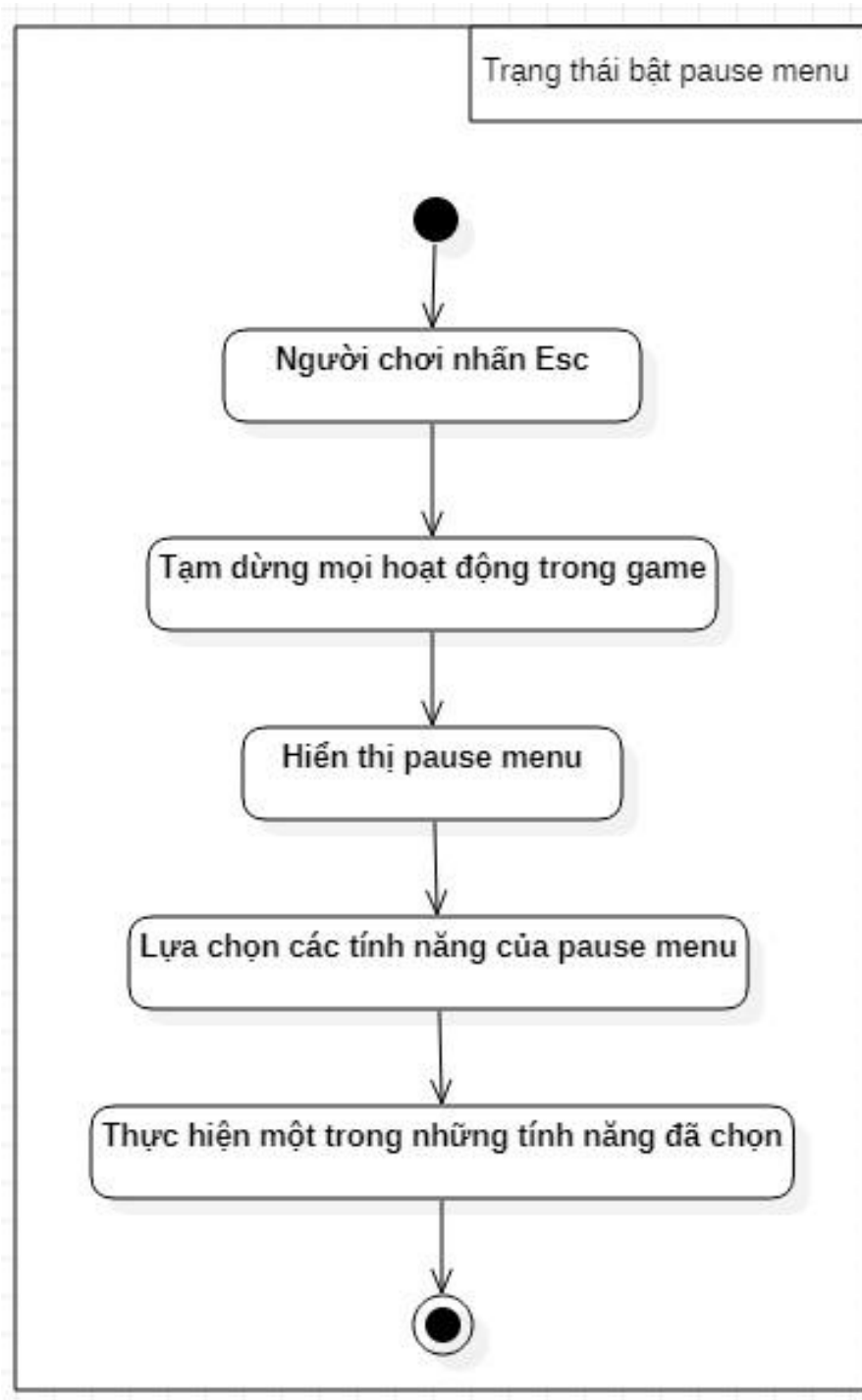
Use Case Name	Điều chỉnh âm lượng	
Scenario	Nhấn vào nút Volume trong menu chính và thanh cuộn ngang được hiển thị	
Triggering event	Trang Xem menu chính đã được hiển thị	
Brief description	Người chơi dùng chức năng này để điều chỉnh tăng giảm âm lượng nhạc nền game	
Actors	Người chơi	
Related Use Case	Xem menu chính	
Stakeholder	Không	
Precondition	Menu chính đã load lên đầy đủ	
Postcondition	Chỉ có thể điều chỉnh âm lượng được khi vào giao diện menu chính và hiển thị thanh cuộn ngang	
Flow of activities	Người dùng	Hệ thống
	1. Người chơi nhấn vào nút Volume	1.1 Hệ thống sẽ hiển thị thanh cuộn ngang để điều chỉnh âm

		lượng trong khoảng từ 0 đến 100 tương ứng với số phần trăm âm lượng nhạc nền.
Exception condition	Không	

Bảng 15. Đặc tả Use case Điều chỉnh âm lượng

3.3 Sơ đồ trạng thái

3.3.1 Trạng thái bật pause menu



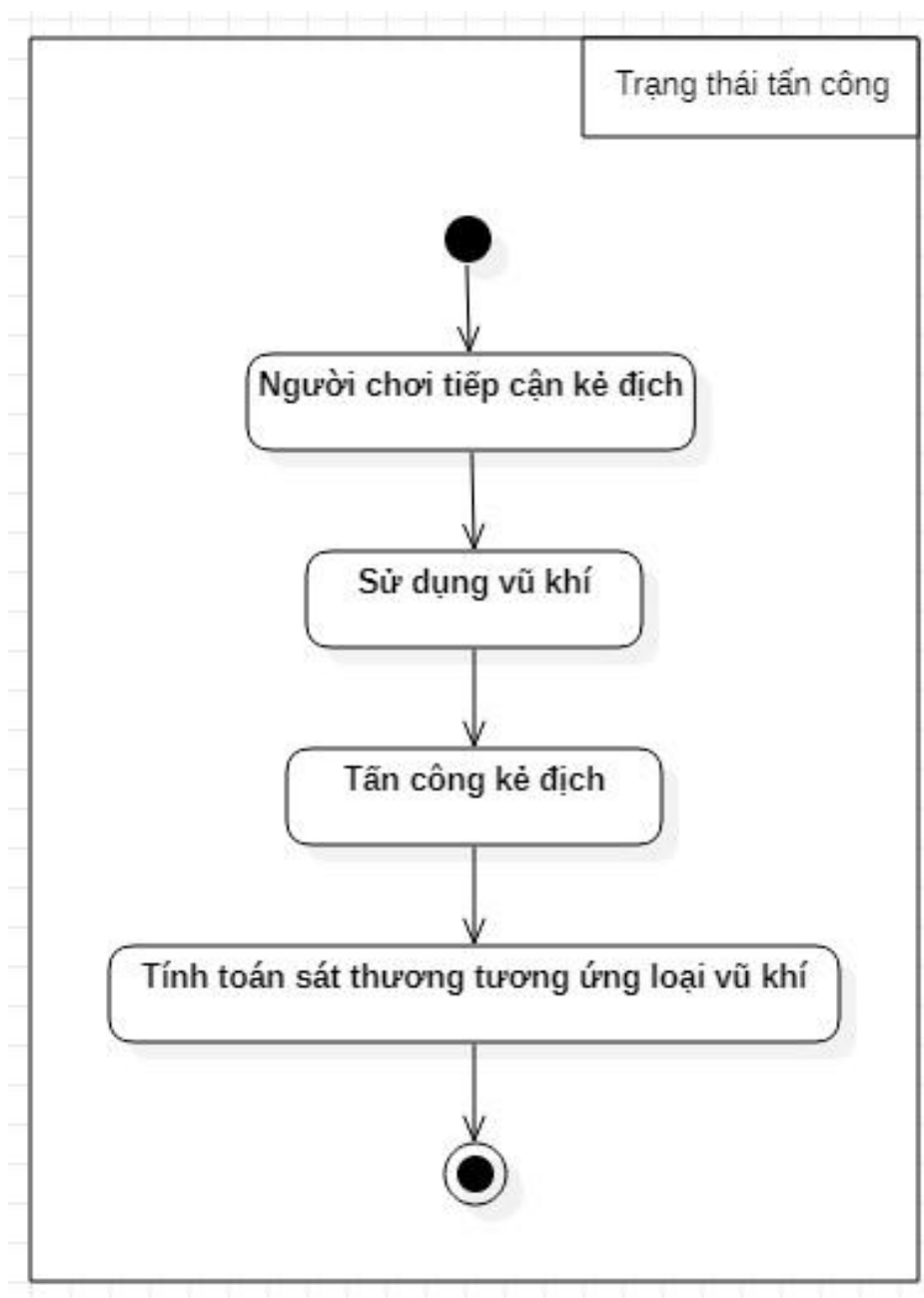
Hình 12. Trạng thái bật pause menu

3.3.2 Trạng thái bật dead menu



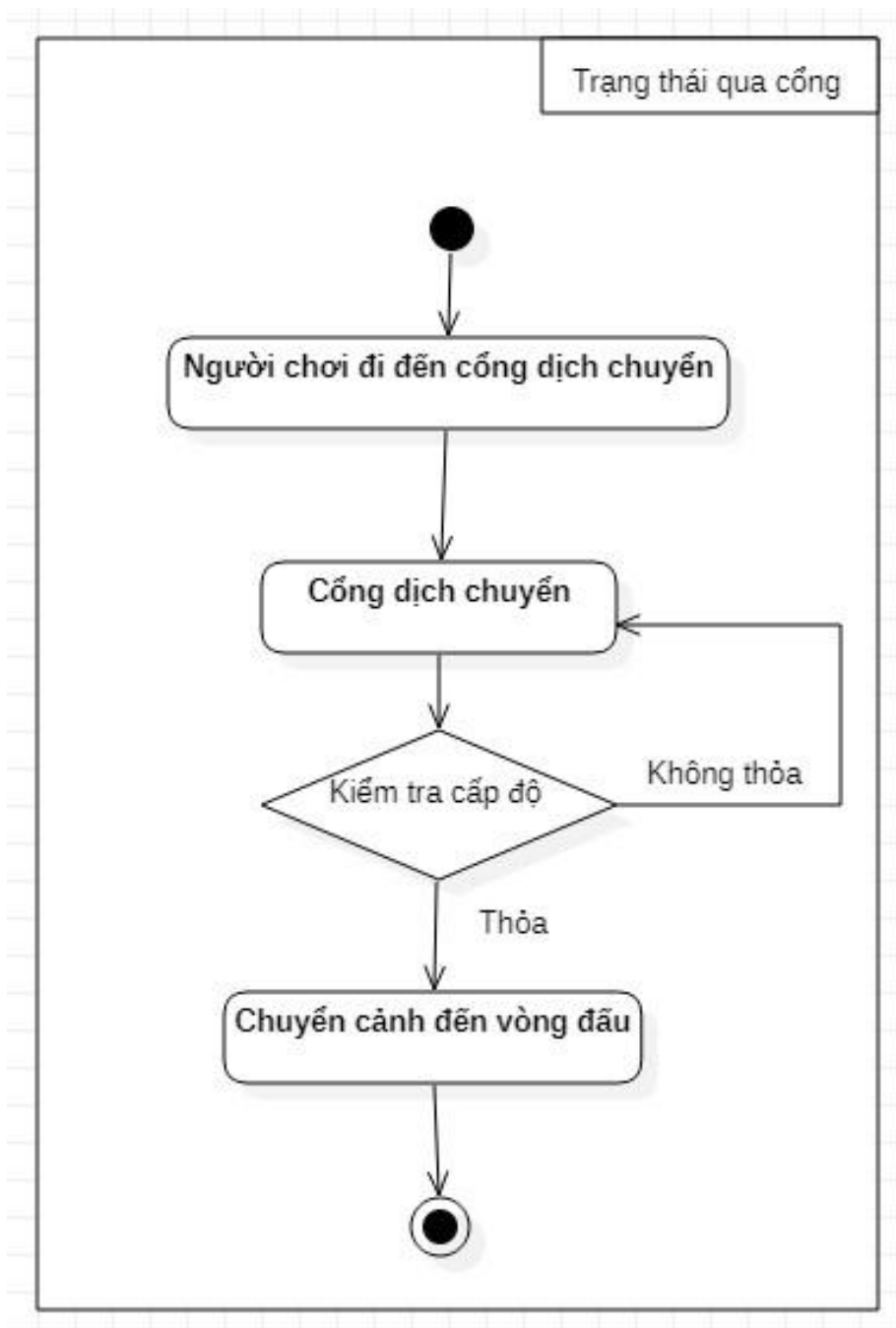
Hình 13. Trạng thái bật dead menu

3.3.3 Trạng thái tấn công



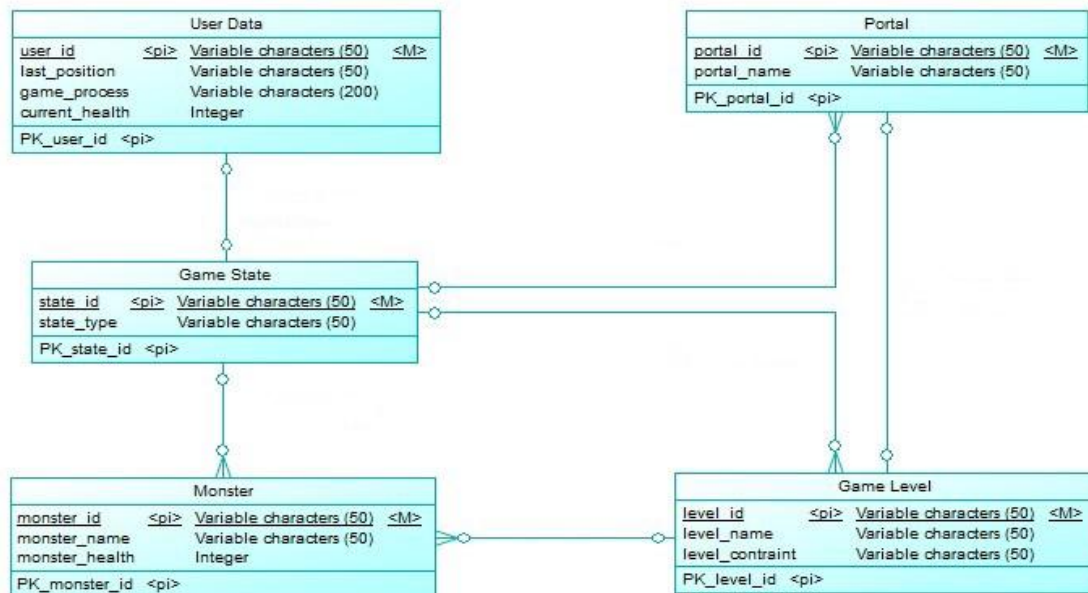
Hình 14. Trạng thái tấn công

3.3.4 Trạng thái qua cổng



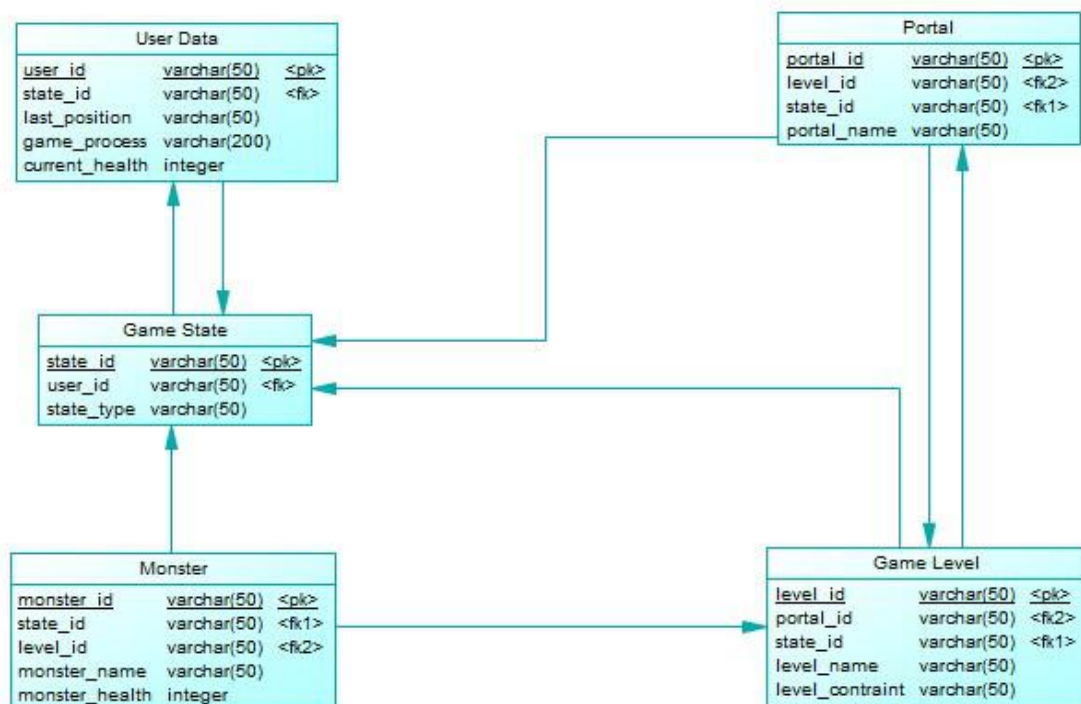
Hình 15. Trạng thái qua cổng

3.4 Sơ đồ ERD



Hình 16. Sơ đồ mô hình quan hệ thực thể - ERD

3.5 Sơ đồ PDM



Hình 17. Sơ đồ mô hình vật lý - PDM

Diễn giải:

- Bảng *User Data* được sử dụng để lưu trữ các thông tin của người chơi trong quá trình chơi game như lưu lại vị trí cuối cùng của người chơi, phần trăm máu hiện tại, tiến trình chơi game. Người chơi có thể tải lại dữ liệu tiến trình mỗi khi vào lại game.
- Bảng *Game State* được sử dụng để lưu trữ thông tin về tất cả các trạng thái của mọi thứ trong game như người chơi, kẻ địch, cổng dịch chuyển, cấp độ của game.
- Bảng *Monster* được sử dụng để lưu trữ thông tin về các kẻ địch cũng như quái vật có trong game. Bao gồm trạng thái, xuất hiện ở cấp độ nào, tên và máu của chúng.
- Bảng *Game Level* được sử dụng để lưu trữ thông tin về các cấp độ của game, bao gồm trạng thái, nằm ở cổng dịch chuyển nào, tên cấp độ và ràng buộc để chơi được cấp độ đó.
- Bảng *Portal* được sử dụng để lưu trữ thông tin về các cổng dịch chuyển, bao gồm trạng thái, cấp độ tương ứng và tên cổng.

3.6 Mô hình lược đồ quan hệ

`UserData(user_id, last_position, game_process, current_health, state_id)`

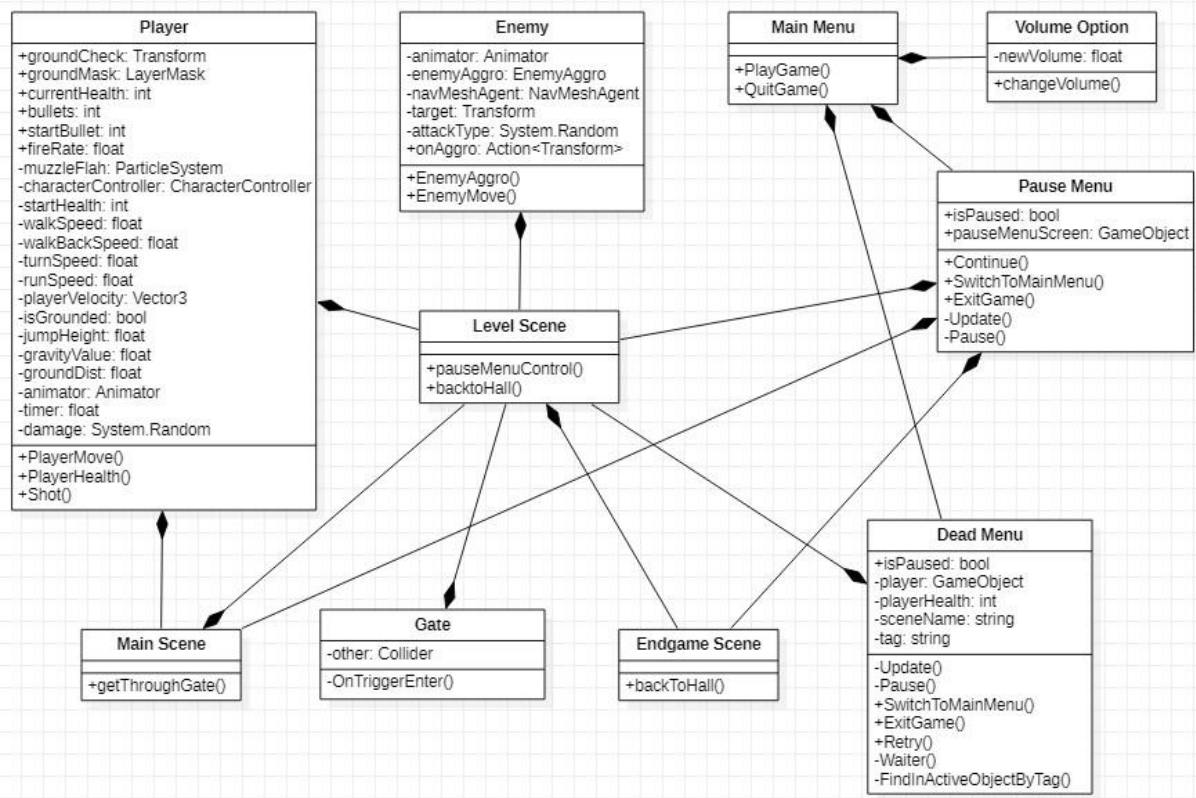
`GameState(state_id, state_type, user_id)`

`Monster(monster_id, monster_name, monster_health, state_id, level_id)`

`Portal(portal_id, portal_name, state_id, level_id)`

`GameLevel(level_id, level_name, level_constraint, state_id, portal_id)`

3.7 Sơ đồ lớp



Hình 18. Sơ đồ lớp

CHƯƠNG IV: HIỆN THỰC CODE VÀ KẾT QUẢ ĐẠT ĐƯỢC

4.1 Hiện thực bằng code

4.1.1 Về phía người chơi

4.1.1.1 Di chuyển nhân vật

Để có thể di chuyển nhân vật ta tạo một file script PlayerMove và thêm vào phần component của nhân vật

Di chuyển nhân vật theo hướng ngang dọc:

```
var vertical = Input.GetAxis("Vertical") ;  
var horizontal = Input.GetAxis("Horizontal");  
Vector3 movement;  
movement = transform.forward * vertical + transform.right * horizontal;  
characterController.Move( movement * speedUsing * Time.deltaTime);
```

Dùng chuột để xoay nhân vật:

```
var mouseX = Input.GetAxis("Mouse X");  
transform.Rotate(Vector3.up, mouseX * turnSpeed);
```

Dùng phím space để nhảy:

```
private Vector3 playerVelocity;  
private bool isGrounded;  
private float jumpHeight = 2f;  
private float gravityValue = -9.81f;  
public Transform groundCheck;  
public LayerMask groundMask;  
private float groundDist = 0.4f;
```

```

isGrounded = Physics.CheckSphere(groundCheck.position, groundDist,
groundMask);
if (isGrounded && playerVelocity.y < 0)
{
    playerVelocity.y = -2f;
}
if (Input.GetButtonDown("Jump") && isGrounded)
{
    playerVelocity.y += Mathf.Sqrt(jumpHeight * -2.0f * gravityValue);
}
playerVelocity.y += gravityValue * Time.deltaTime;
characterController.Move(playerVelocity * Time.deltaTime);

```

4.1.1.2 Quay Camera

Để có thể xoay camera theo chiều dọc bằng chuột ta cần thêm component CameraController cho Virtual Camera với nội dung:

```

public class CameraController : MonoBehaviour
{
    private CinemachineComposer composer;
    private float sens = 0.25f;
    void Start()
    {
        composer =
GetComponent<CinemachineVirtualCamera>().GetCinemachineComponent<
CinemachineComposer>();
    }
    void Update()
    {
        float mouseY = Input.GetAxis("Mouse Y") * sens;
        composer.m_TrackedObjectOffset.y += mouseY;
    }
}

```

```

        composer.m_TrackedObjectOffset.y =
        Mathf.Clamp(composer.m_TrackedObjectOffset.y, 0, 10);
    }
}

```

Lưu ý: Để tạo được Virtual Camera cho nhân vật cần phải thêm Packet Cinemachine cho Unity

4.1.1.3 Hiệu ứng ảnh động và tốc độ di chuyển của nhân vật

Với PlayerMove ta có thể di chuyển nhân vật nhưng vẫn chưa có hiệu ứng chuyển động cho nhân vật. Để làm được điều đó ta cần bổ sung thêm đoạn mã sau:

```

float speedUsing = walkSpeed;
if (isRunning && horizontal > 0 && vertical >= 0
    && !gameObject.CompareTag("PlayerPistol"))
{
    animator.SetFloat("right", 2);
    speedUsing = runSpeed;
}
else if (isRunning && horizontal < 0 && vertical >= 0
    && !gameObject.CompareTag("PlayerPistol"))
{
    animator.SetFloat("right", -2);
    speedUsing = runSpeed;
}
else { animator.SetFloat("right", horizontal); }
if (isRunning && vertical > 0 && !gameObject.CompareTag("PlayerPistol"))
{
    animator.SetFloat("speed", 2);
    speedUsing = runSpeed;
}

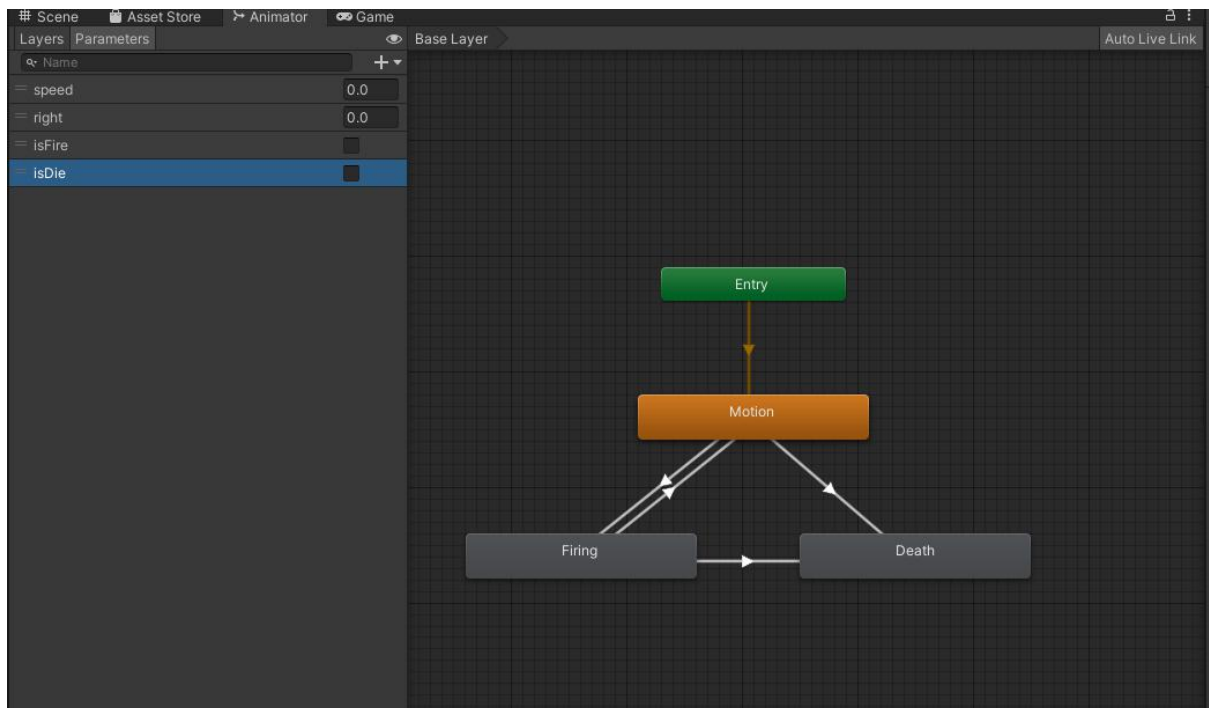
```

```

else { animator.SetFloat("speed", vertical); } //walking
if (vertical < 0) //walking back
{
    speedUsing = walkBackSpeed;
}
if (isShooting)
{
    animator.SetBool("isFire", true);
}
else { animator.SetBool("isFire", false); }
bool mouseLeft = Input.GetKeyDown(KeyCode.Mouse0);
bool mouseRight = Input.GetKeyDown(KeyCode.Mouse1);
if (mouseLeft && transform.tag == "PlayerKnife")
{
    animator.SetTrigger("knife left");
}
else if (mouseRight && transform.tag == "PlayerKnife")
{
    animator.SetTrigger("knife right");
}

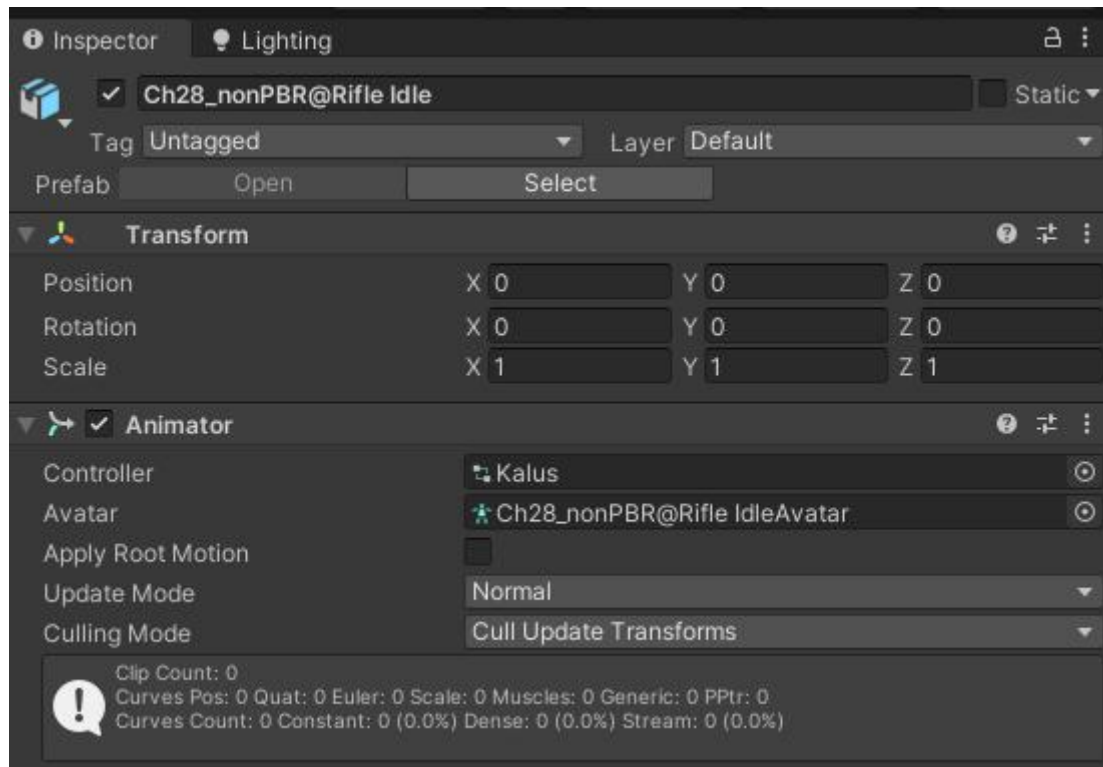
```

Để có thể điều khiển được các hiệu ứng chuyển động của nhân vật, ta phải tạo một Animator Controller với các thuộc tính như hình bên dưới.



Hình 19. Thiết lập lớp cơ sở hoạt ảnh chuyển động nhân vật

Cài đặt các hoạt ảnh chuyển động của nhân vật cho phù hợp. Ta có thể tìm và tải một số các hoạt ảnh của nhân vật miễn phí trên Mixamo của Adobe. Sau đó thêm Animator Controller vừa tạo vào nhân vật.



Hình 20. Thiết lập hoạt ảnh chuyển động nhân vật

4.1.2 Về phía hệ thống

4.1.2.1 Kẻ thù kiểm soát phạm vi lãnh thổ

Các quái vật muốn nhận biết được người chơi cần phải có một Collider. Khi người chơi bước vào sẽ trở thành target. Đoạn mã để nhận biết người chơi khi bước vào lãnh thổ:

```
public class EnemyAggro : MonoBehaviour
{
    public event Action<Transform> OnAggro = delegate { };
    private void OnTriggerEnter(Collider other)
    {
        var player = other.GetComponent<PlayerMove>();
        if(player != null)
        {
```

```

        OnAggro(player.transform);
    }
}
}

```

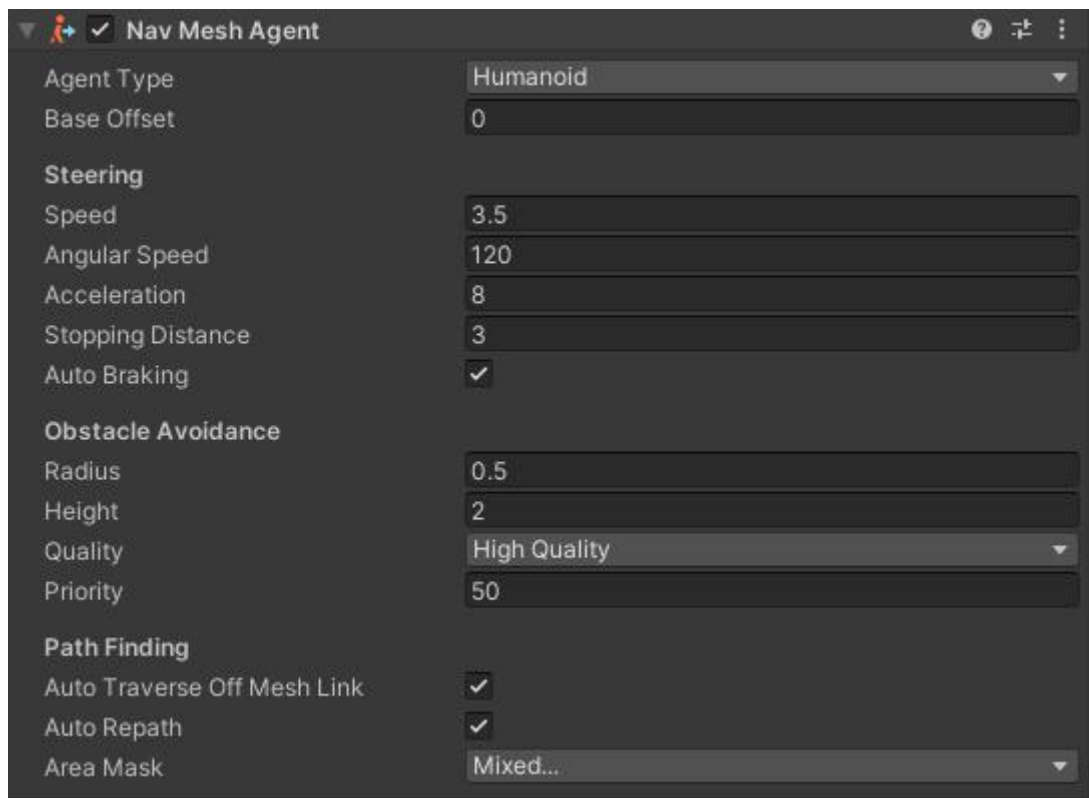
Đoạn mã để quái vật di chuyển và tiếp cận người chơi:

```

enemyAggro = GetComponent<EnemyAggro>();
private void EnemyAggro_OnAggro(Transform target)
{
    this.target = target;
}
if (target != null)
{
    navMeshAgent.SetDestination(target.position);
}

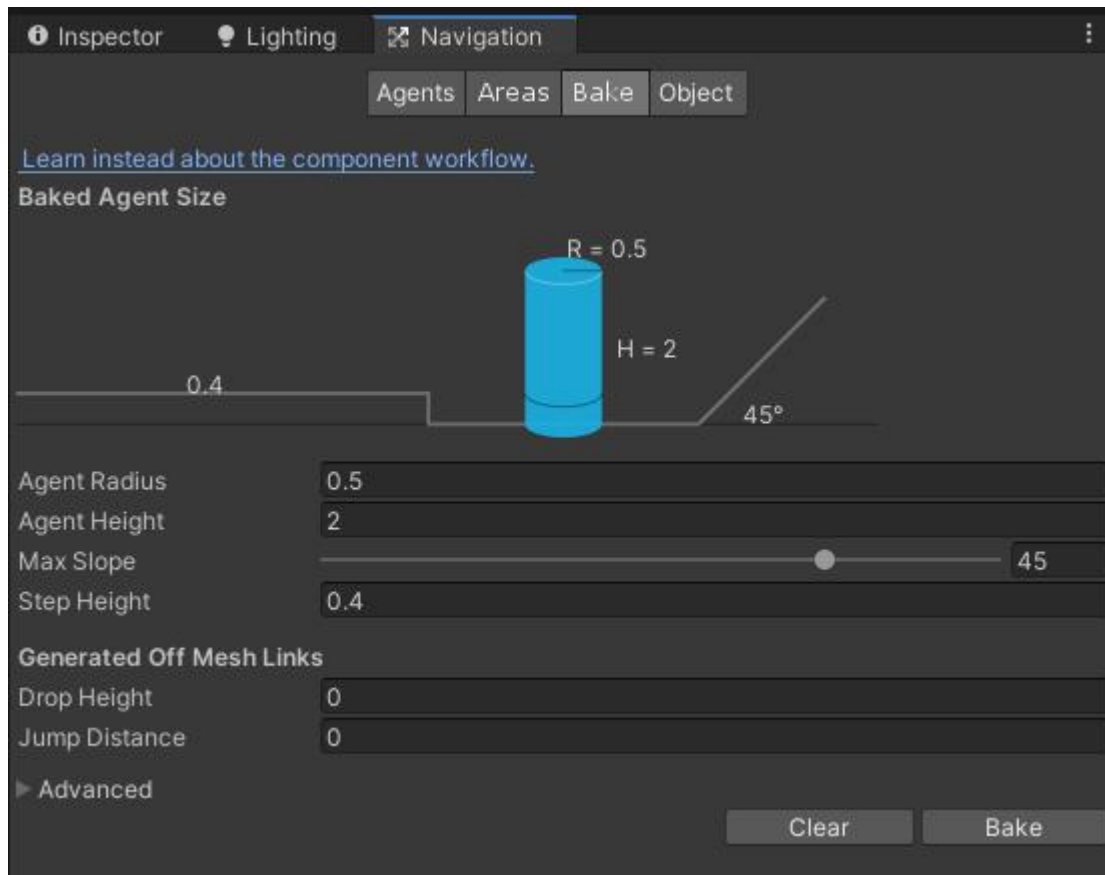
```

Để có thể làm cho quái vật tự động di chuyển theo người chơi ta cần thêm component NavMeshAgent cho quái vật.



Hình 21. Thêm thành phần NavMeshAgent cho quái vật

Sau đó cài đặt Navigation cho môi trường. Chọn Window > AI > Navigation. Chọn thẻ Bake và nhấn Bake.



Hình 22. Thiết lập thông số lưới điều hướng

4.1.2.2 Tính toán sát thương máu của người chơi

```
public class PlayerHealth : MonoBehaviour
{
    private int startHealth = 100;
    [SerializeField]
    public int currentHealth;
    private CharacterController characterController;
    private Animator animator;
    public event Action<float> OnHealthPercentChange = delegate { };
    void Awake()
    {
        characterController = GetComponent<CharacterController>();
        animator = GetComponentInChildren<Animator>();
    }
}
```

```

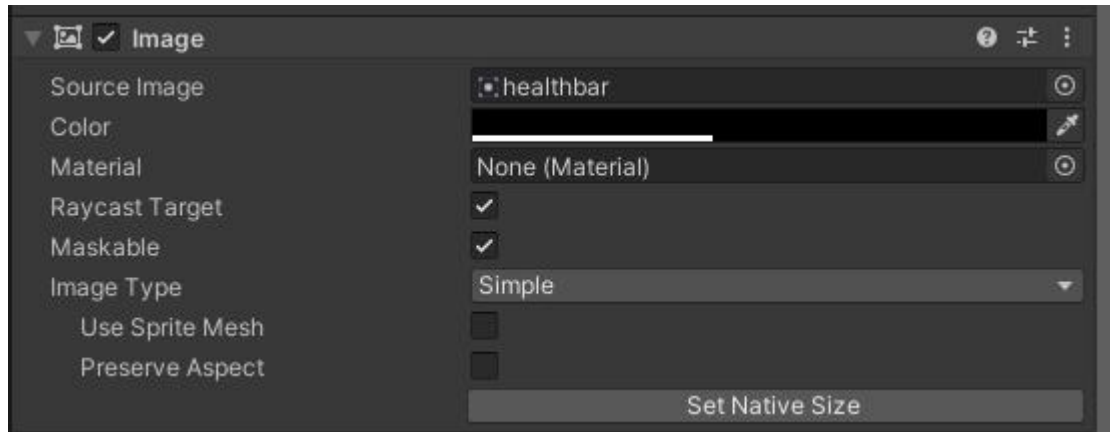
    }
    private void OnEnable()
    {
        currentHealth = startHealth;
    }
    public void TakeDamage(int amount)
    {
        currentHealth -= amount;
        float currentHealthPercent = (float)currentHealth / startHealth;
        OnHealthPercentChange(currentHealthPercent);
        if (currentHealth <= 0)
        {
            Die();
        }
    }
    private void Die()
    {
        animator.SetBool("isDie", true);

        StartCoroutine(waiter());
    }
    IEnumerator waiter()
    {
        yield return new WaitForSeconds(3);
        gameObject.SetActive(false);
    }
}

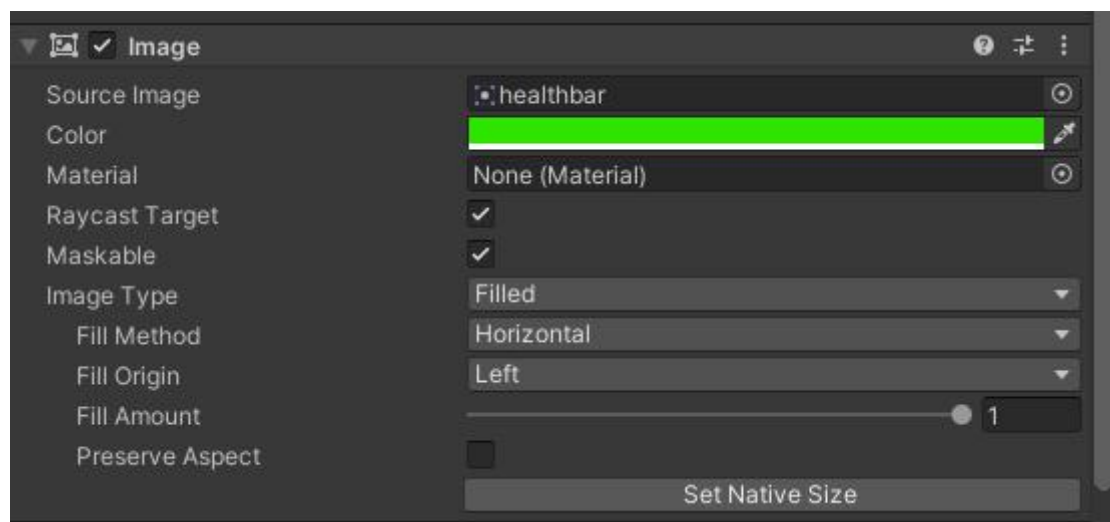
```

4.1.2.3 Thanh máu của người chơi

Để tạo thanh máu cho nhân vật cùng với hiệu ứng của thanh máu ta cần tạo một Canvas chứa hai panel thanh máu với một là panel tĩnh làm nền và còn lại là panel động biểu thị cho lượng máu còn lại của nhân vật.



Hình 23. Thiết lập thanh máu nền



Hình 24. Thiết lập thanh máu hiện tại

Sau đó tạo một component `PlayerHealthBar` cho Canvas dùng để tính toán giá trị máu hiện tại và hiển thị bằng cách điều chỉnh giá trị của panel nền.

```
public class PlayerHealthBar : MonoBehaviour
```

```

{
    [SerializeField]
    private Image image;
    private float updateSpeed = 0.5f;
    private void Awake()
    {
        GetComponentInParent<PlayerHealth>().OnHealthPercentChange +=
        HandleHealthChangePlayer;
    }
    private void HandleHealthChangePlayer(float pct)
    {
        StartCoroutine(ChangeToPercent(pct));
    }
    private IEnumerator ChangeToPercent(float pct)
    {
        float pre = image.fillAmount;
        float elapsed = 0f;
        while (elapsed < updateSpeed)
        {
            elapsed += Time.deltaTime;
            image.fillAmount = Mathf.Lerp(pre, pct, elapsed / updateSpeed);
            yield return null;
        }
        image.fillAmount = pct;
    }
}

```

4.1.2.4 Tấn công bằng vũ khí súng

Để tấn công quái vật bằng súng ta cần tạo thêm một file script Shot để tính toán va chạm của đạn với quái. Đồng thời kiểm soát lượng đạn trong vũ khí của người chơi. Ngoài ra còn tạo thêm một số hiệu ứng hình ảnh và âm thanh khi bắn của nhân vật.

```

public class Shot : MonoBehaviour
{
    public int bullets = 100;
    public int startBullets = 100;
    public float fireRate;
    private System.Random damage = new System.Random();
    private float timer;
    public event Action<float> OnAmmoPercentChange = delegate { };
    [SerializeField]
    private ParticleSystem muzzleFlash;
    void OnEnable()
    {
        if (gameObject.CompareTag("Player1") ||
gameObject.CompareTag("Player2") || gameObject.CompareTag("Player3"))
        {
            startBullets = 50;
            bullets = startBullets;
            fireRate = 0.1f;
        }
        if (gameObject.CompareTag("PlayerPistol"))
        {
            startBullets = 10;
            bullets = startBullets;
            fireRate = 0.7f;
        }
    }
    void Update()
    {
        timer += Time.deltaTime;
        if (bullets > 0)

```



```

{
    if (timer >= fireRate)
    {
        if (Input.GetKey(KeyCode.Mouse0))
        {
            var audio = GetComponent<AudioSource>();
            audio.Play(0);
            timer = 0;
            fireGun();
            bullets -= 1;
            float currentAmmoPercent = (float)bullets / startBullets;
            OnAmmoPercentChange(currentAmmoPercent);
        }
    }
}
else
{
    StartCoroutine(ReloadAmmo());
}
}

private void fireGun()
{
    muzzleFlah.Play();
    Ray ray = Camera.main.ViewportPointToRay(Vector3.one * 0.5f);
    Debug.DrawRay(ray.origin, ray.direction * 100, Color.yellow, 2f);
    RaycastHit hitInfo;
    if(Physics.Raycast(ray, out hitInfo, 100))
    {
        bool enemy = hitInfo.collider.gameObject.tag == "enemy";
        var health = hitInfo.collider.GetComponent<Health>();
        if (health != null && enemy)
    }

```

```

        {
            health.TakeDamage(damage.Next(30,40));
        }
    }
}

IEnumerator ReloadAmmo()
{
    yield return new WaitForSeconds(5);
    bullets = startBullets;
    float currentAmmoPercent = (float)bullets / startBullets;
    OnAmmoPercentChange(currentAmmoPercent);
}
}

```

4.1.2.5 Thanh nhiệt của vũ khí súng

Để kiểm soát lượng đạn của súng ta thêm một thanh trạng thái tương tự như cách làm thanh máu nhân vật, thanh trạng thái cho biết độ nóng của súng vào màn hình chính. Đoạn mã sau sẽ tính toán và hiển thị trạng thái của thanh nhiệt độ:

```

public class AmmoBar : MonoBehaviour
{
    [SerializeField]
    private Image image;
    private float updateSpeed = 0.5f;
    private void Awake()
    {
        GetComponentInParent<Shot>().OnAmmoPercentChange +=
        HandleAmmoChange;
    }
    private void HandleAmmoChange(float pct)
    {
        StartCoroutine(ChangeToPercent(pct));
    }
}

```

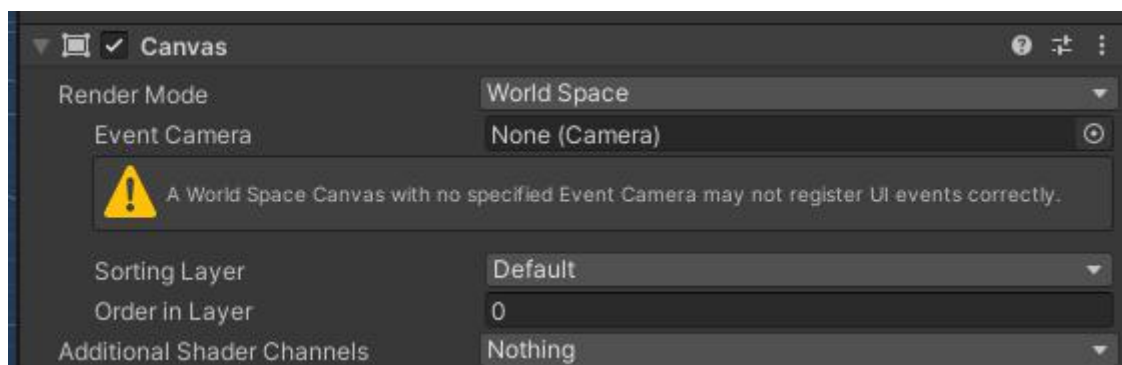
```

    }
    private IEnumerator ChangeToPercent(float pct)
    {
        float pre = image.fillAmount;
        float elapsed = 0f;
        while (elapsed < updateSpeed)
        {
            elapsed += Time.deltaTime;
            image.fillAmount = Mathf.Lerp(pre, pct, elapsed / updateSpeed);
            yield return null;
        }
        image.fillAmount = pct;
    }
}

```

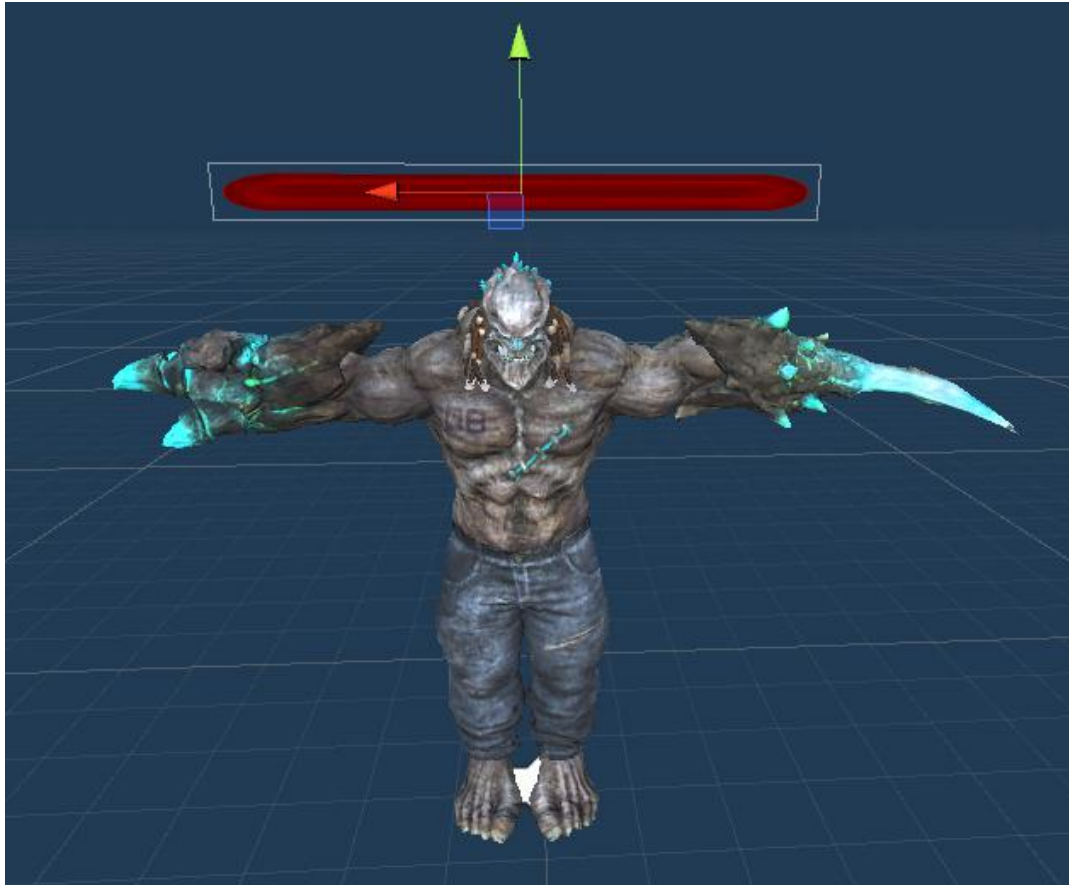
4.1.2.6 Thanh máu của quái vật

Ta tạo một Canvas cho quái vật với hai panel tương tự như khi tạo nhân vật nhưng với Render Mode: World Space.



Hình 25. Thiết lập thanh máu quái vật

Điều chỉnh kích thước và vị trí phù hợp với quái vật



Hình 26. Căn chỉnh thanh máu quái vật

Sau đó thêm component HealthBar cho Canvas để điều khiển thanh máu

```
public class HealthBar : MonoBehaviour
{
    [SerializeField]
    private Image image;
    private float updateSpeed = 0.5f;

    private void Awake()
    {
        GetComponentInParent<Health>().OnHealthPercentChange +=
        HandleHealthChangeEnemy;
    }
    private void HandleHealthChangeEnemy(float pct)
```

```

    {
        StartCoroutine(ChangeToPercent(pct));
    }
    private IEnumerator ChangeToPercent(float pct)
    {
        float pre = image.fillAmount;
        float elapsed = 0f;
        while (elapsed < updateSpeed)
        {
            elapsed += Time.deltaTime;
            image.fillAmount = Mathf.Lerp(pre, pct, elapsed / updateSpeed);
            yield return null;
        }
        image.fillAmount = pct;
    }
}

```

Ngoài ra ta cần thêm đoạn mã sau để cho thanh máu của quái vật luôn quay về phía người chơi:

```

private void LateUpdate()
{
    transform.LookAt(Camera.main.transform);
    transform.Rotate(0, 180, 0);
}

```

4.1.2.7 Quái vật tấn công người chơi

Để có thể nhận biết được khi quái vật tấn công trúng người chơi ta thêm một collider vào vị trí gây sát thương của quái kết hợp với OnTriggerEnter để nhận biết va chạm với người chơi.



Hình 27. Thiết lập va chạm trên tay quái vật

Người chơi nhận sát thương:

```
private void OnTriggerEnter(Collider other)
{
    if (other.gameObject.tag == "Punch")
    {TakeDamage(10);}
}
```

Quái vật tấn công:

```
var str = "Attack" + attackType.Next(1, 3);
if (navMeshAgent.remainingDistance < 3)
{animator.SetTrigger(str);}
```

4.2 Một số hình ảnh trong game

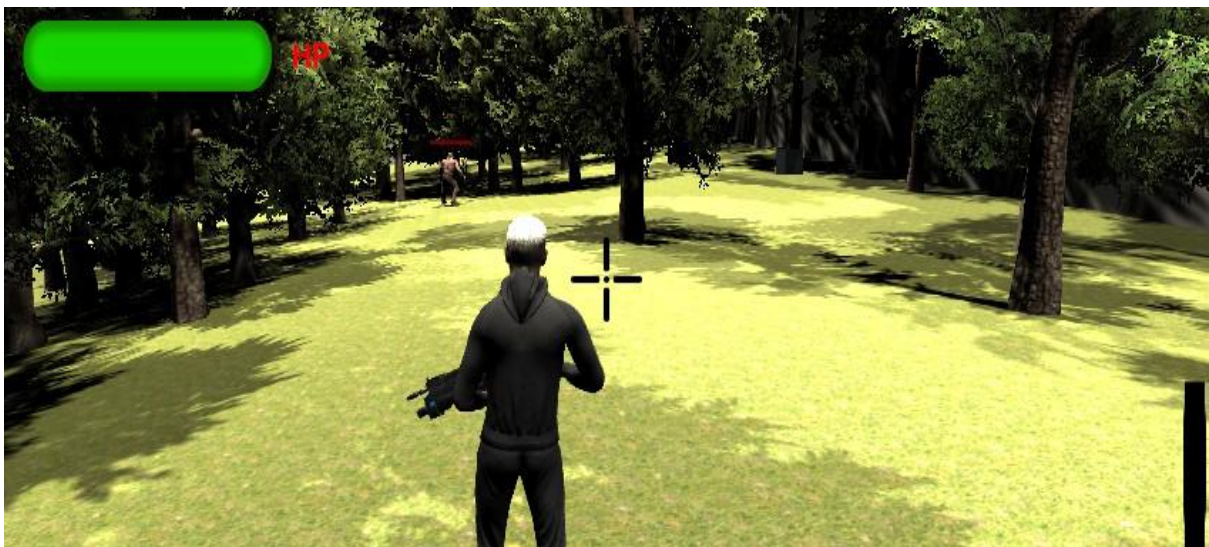
4.2.1 Bản đồ chính và người chơi



Hình 28. Bản đồ chính và người chơi

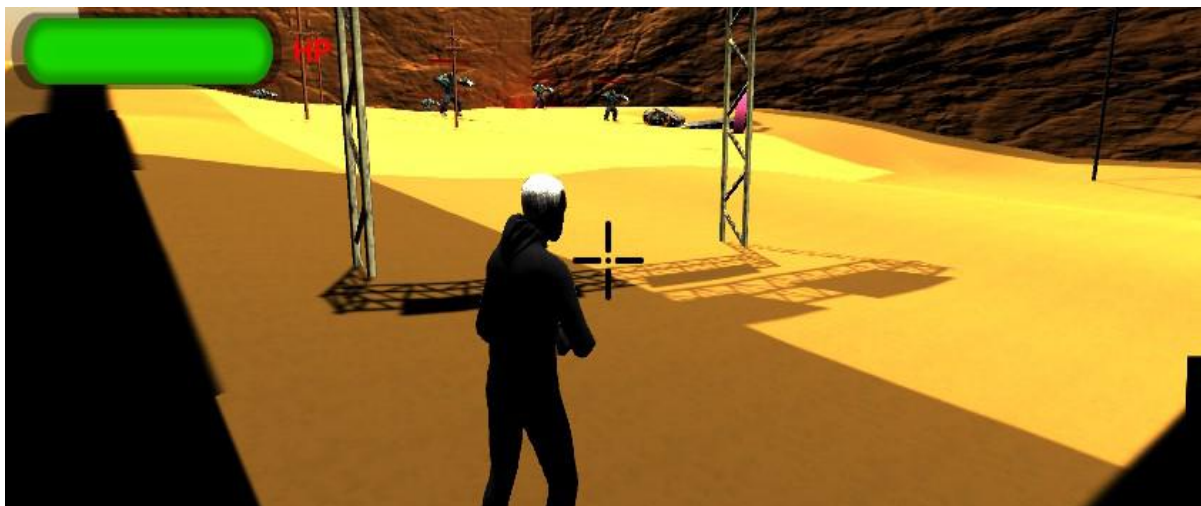
4.2.2 Bản đồ khu vực đấu

4.2.2.1 Vòng đấu cấp 1



Hình 29. Vòng đấu cấp 1

4.2.2.2 Vòng đấu cấp 2



Hình 30. Vòng đấu cấp 2

4.2.2.3 Vòng đấu cấp 3



Hình 31. Vòng đấu cấp 3

4.2.2.4 Vòng đấu cấp 4



Hình 32. Vòng đấu cấp 4

4.2.2.5 Vòng đấu cấp 5



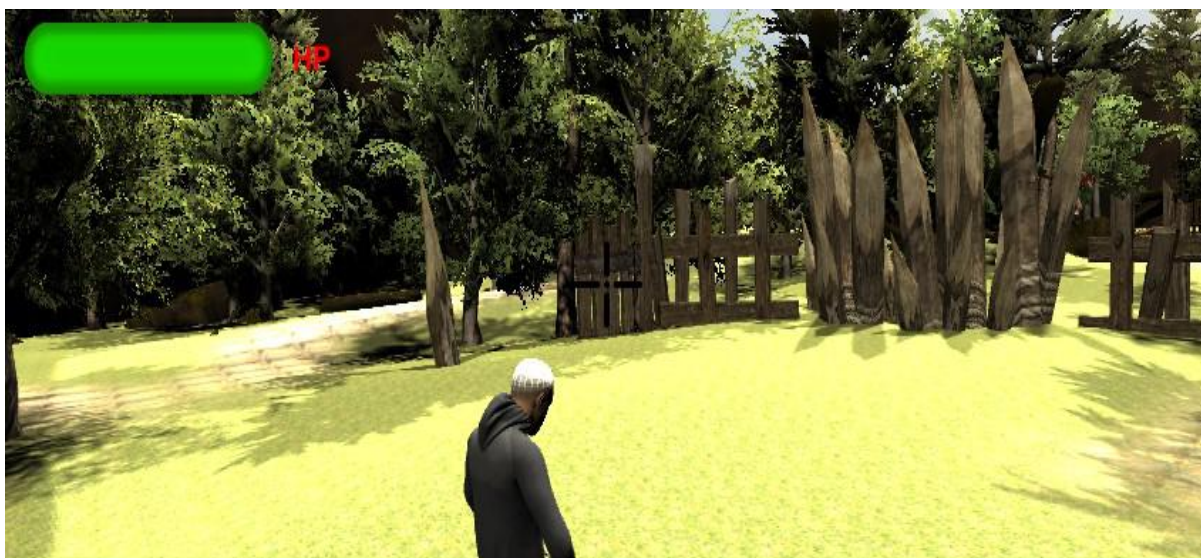
Hình 33. Vòng đấu cấp 5

4.2.2.6 Vòng đấu cấp 6



Hình 34. Vòng đấu cấp 6

4.2.2.7 Vòng đấu cấp 7



Hình 35. Vòng đấu cấp 7

4.2.3 NPC

4.2.3.1 Dân làng



Hình 36. NPC Dân làng

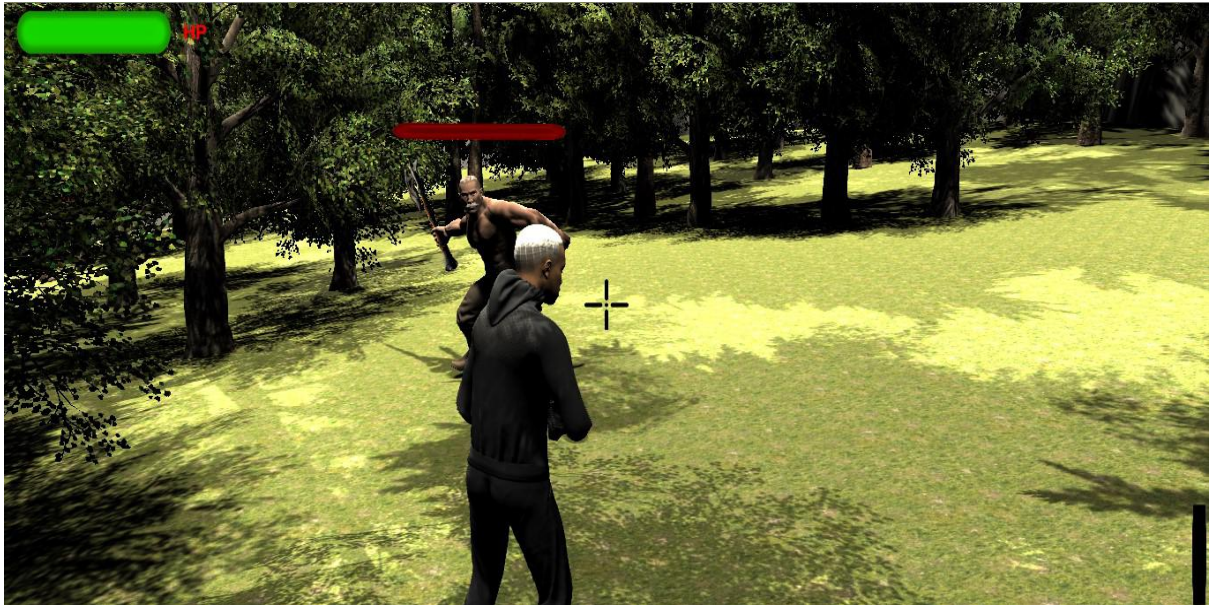
4.2.3.2 Trưởng làng



Hình 37. NPC Trưởng làng

4.2.4 Kẻ thù

4.2.4.1 Lính canh



Hình 38. Lính canh

4.2.4.2 Lính hắc cầu



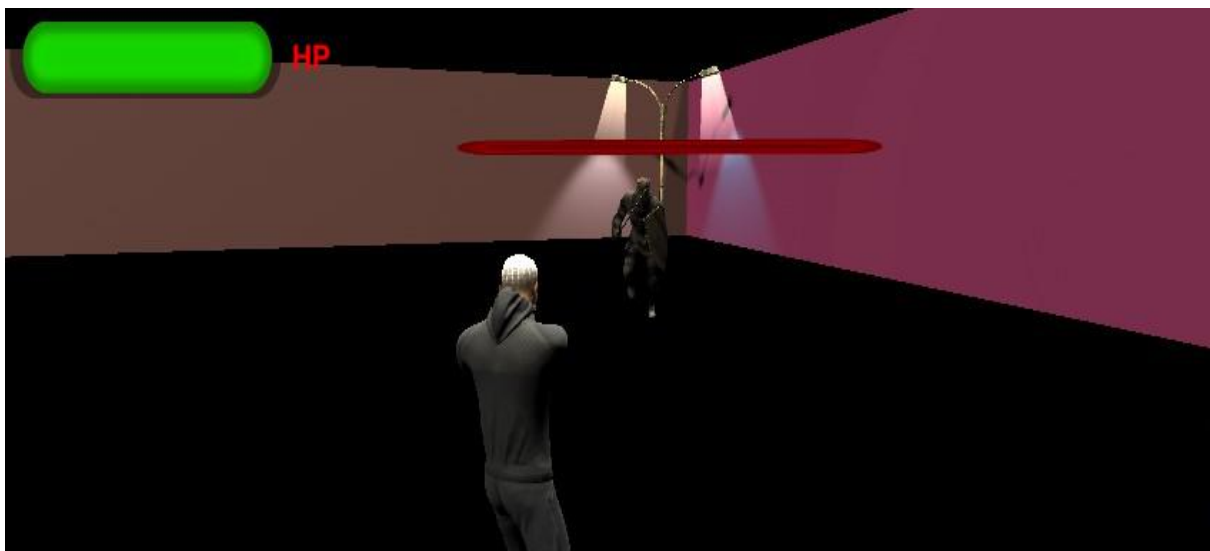
Hình 39. Lính hắc cầu

4.2.4.3 Zombie



Hình 40. Bầy Zombie

4.2.4.4 Boss



Hình 41. Boss

4.2.5 Menu

4.2.5.1 Menu chính



Hình 42. Menu chính

4.2.5.2 Menu thua cuộc



Hình 43. Menu thua cuộc

4.2.5.3 Menu dừng



Hình 44. Menu dừng

4.2.5.4 Menu kết thúc



Hình 45. Menu kết thúc

CHƯƠNG V: KẾT LUẬN

Bài báo cáo này đã giới thiệu tổng quát nhất về game offline 3D có cốt truyện - Rummage Memories. Thông qua các use case, đặc tả usecase cùng với các sơ đồ hoạt động, sơ đồ lớp đã minh họa cho việc hiện thực một ứng dụng game offline 3D một cách đơn giản nhất, dễ hiểu nhất. Đồng thời cũng giới thiệu qua về một game engine rất có ích trong việc phát triển game là Unity. Với sức mạnh trong việc hỗ trợ đồ họa, hoạt ảnh, âm thanh, build trên nhiều nền tảng,... giúp tiết kiệm thời gian rất nhiều trong việc thiết kế và triển khai thực hiện một game 3D. Cú pháp ngôn ngữ C# quen thuộc dễ sử dụng.

Tuy nhiên, với kiến thức còn hạn hẹp và chưa có nhiều thời gian nên việc triển khai hệ cơ sở dữ liệu cho game này vẫn chưa thực hiện được. Do đó, một số tính năng như gắn cờ để lưu trữ trạng thái game trước khi thoát, nút load game hay dead menu vẫn chưa thể hoạt động đúng hoàn toàn như ý tưởng trò chơi ban đầu đề ra.

Trong tương lai, với thời gian thoải mái hơn thì nhóm sẽ hoàn thiện game có cả hệ cơ sở dữ liệu để các tính năng được hoạt động đúng với ý tưởng trò chơi đề ra. Ngoài ra, nhóm còn mong muốn phát triển một số game 3D mới lạ, đánh thẳng nhu cầu thị trường hiện nay, tìm hiểu chi tiết hơn về tâm lý người chơi để đưa ra những trải nghiệm tốt nhất cho người chơi.

PHÂN CÔNG VÀ ĐÁNH GIÁ

Phân công nhiệm vụ

Họ và tên	Thời gian	Phân công nhiệm vụ
Tô Vĩnh Khang	16/09/2020-23/09/2020	Thảo luận nhóm và Lên ý tưởng. Phân công công việc và trình bày bố cục từng phần của bài báo cáo. Viết Game Concept.
	23/09/2020-01/10/2020	Thiết kế 7 vòng đấu ứng với 7 cấp độ. Chuyển scenes 7 vòng đấu. Tìm hiểu về cách thức hoạt động của nhân vật.
	01/10/2020-01/11/2020	Trình bày báo cáo Chương I, Chương II, Chương III (3.1,3.2), Chương V. Chỉnh sửa, bổ sung ý tưởng cho Game Concept.
	01/11/2020-29/11/2020	Cập nhật tiến độ đồ án. Tổng hợp tất cả nội dung. Chỉnh sửa và cập nhật mục lục. Nộp bài báo cáo và source code lên hệ thống.
Bùi Quang Khải	16/09/2020-23/09/2020	Thảo luận nhóm và Lên ý tưởng. Viết Game Concept.
	23/09/2020-15/10/2020	Thiết kế map chính, chuyển scenes các menu. Tìm hiểu về cách thức hoạt động của nhân vật. Chỉnh sửa, bổ sung ý tưởng cho Game Concept.
	15/10/2020-29/11/2020	Trình bày báo cáo Chương III (3.3-3.7).

Du Thuận Long	16/09/2020- 23/09/2020	Thảo luận nhóm và Lên ý tưởng. Viết Game Concept.
	23/09/2020- 15/10/2020	Tìm hiểu các chuyển động nhân vật và thiết kế nhân vật chính, vũ khí cho nhân vật, lính quái. Chỉnh sửa, bổ sung ý tưởng cho Game Concept.
	15/10/2020- 29/11/2020	Trình bày báo cáo Chương IV.

Bảng phân công nhiệm vụ

Đánh giá thành viên

Tên thành viên	Tô Vĩnh Khang	Bùi Quang Khải	Du Thuận Long
Số lượng công việc được giao	33.33%	33.33%	33.34%
Mức độ hoàn thành công việc	100%	100%	100%

Bảng đánh giá các thành viên trong nhóm

TÀI LIỆU THAM KHẢO

Tài liệu Tiếng Việt:

- [1] Vũ Đình Hồng, 2019, Slide Bài giảng môn Phát triển Trò chơi, Đại học Tôn Đức Thắng

Tài liệu Tiếng Anh:

- [2] Steve Rabin, 2010, Introduction to Game Development, 2nd Edition, Cengage Learning.
- [3] Jesse Schell, 2008, The Art of Game Design - A Book of Lenses, 1st Edition, Morgan Kaufmann.

Tài liệu Internet:

- [4] <https://unity.com>