**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------------------------------------**

****

**ĐỒ ÁN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI: XÂY DỰNG GAME FALCON : SURVIVOR**

**SỬ DỤNG CÔNG CỤ UNITY**

**GVHD**: Ths. Nguyễn Tuấn Tú

**Sinh viên**: Ngô Đình Luân - 2020603876

**Lớp**: 2020DHCNTT03

**Khóa**: K15

**Hà Nội – Năm 2024**

**LỜI CẢM ƠN**

Trong thời đại công nghệ thông tin như hiện nay, sản phẩm công nghệ ngày càng chịu sự đánh giá khắt khe hơn từ phía người dùng, đặc biệt là các sản phẩm về Game nhận được rất nhiều sự quan tâm của người chơi. Ngành công nghiệp Game hiện nay có thể nói là đang trong giai đoạn bùng nổ với tốc độ phát triển chóng mặt. Phía sau sự phát triển ấy là những dự án được đầu tư tâm huyết từ các công ty và Studio với mục đích đem lại cho người dùng không chỉ là trải nghiệm giải trí, mà ngày nay nó đang dần trở thành một lĩnh vực thể thao được nhiều người công nhận.Việc phát triển một tựa game cũng giống như bất kì một dự án công nghệ nào khác, có rất nhiều giai đoạn và quy trình cần được nắm vững và thực hiện một cách khoa học.

Ngoài ra, các Game Engine được coi là công cụ đắc lực giúp các nhà phát triển có thể tự do đưa các ý tưởng của mình trở thành hiện thực qua màn hình máy tính hay điện thoại, một trong số đó là Game Engine vô cùng phổ biến - Unity. Có rất nhiều bộ phận tạo nên một tựa game như 2D artist, 3D artist, quản lý dự án, developer, game designer,… Cũng giống như các công cụ vẽ của họa sĩ, thì với game developer, Unity nói riêng hay các Game Engine nói chung là công cụ hữu hiệu để giúp họ phát triển game một cách thuận tiện và dễ dàng hơn.  
 Với niềm yêu thích các trò chơi từ nhỏ cùng với đó là các kiến thức được tiếp thu trong quá trình học tập, em quyết định lựa chọn đề tài : “Xây dựng game Falcon : Survivor sử dụng công cụ Unity”. Trong quá trình hoàn thiện, em xin cảm ơn sự giúp đỡ của Ths. Nguyễn Tuấn Tú đã hướng dẫn và giúp em hoàn thành đề tài.

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc12760)

[DANH SÁCH CÁC TỪ VIẾT TẮT 5](#_Toc13778)

[DANH SÁCH CÁC BẢNG BIỂU 6](#_Toc4148)

[DANH SÁCH CÁC HÌNH ẢNH 7](#_Toc12707)

[CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI 9](#_Toc6376)

[1.1. Đặt vấn đề 9](#_Toc14580)

[1.2. Mục tiêu và phạm vi đề tài 9](#_Toc29751)

[1.3. Định hướng giải pháp 10](#_Toc28201)

[1.4. Bố cục đồ án 10](#_Toc17232)

[CHƯƠNG 2. THIẾT KẾ TRÒ CHƠI 12](#_Toc26710)

[2.1. Bối cảnh câu chuyện 12](#_Toc901)

[2.2. Cốt truyện chính 12](#_Toc27684)

[2.2.1. Map Aranchu 12](#_Toc14579)

[2.2.2. Map Haru 13](#_Toc24304)

[2.2.3. Map Kanu 14](#_Toc9841)

[2.3. Thiết kế lối chơi 14](#_Toc18294)

[2.3.1. Map Aranchu 14](#_Toc20993)

[2.3.2. Map Kanu 15](#_Toc11521)

[2.3.3. Map Muda 16](#_Toc8650)

[2.4. Chi tiết về game 18](#_Toc4350)

[2.4.1. Súng có thể sử dụng trong game 18](#_Toc32230)

[2.4.2. Các loại vật phẩm hỗ trợ 19](#_Toc24086)

[2.4.3. Thiết kế phần thưởng khi qua màn 21](#_Toc1408)

[2.4.4. Các loại kẻ địch trong game 21](#_Toc16708)

[CHƯƠNG 3. KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU 23](#_Toc6687)

[3.1. Khảo sát hiện trạng 23](#_Toc17544)

[3.1.1. Các thể loại game bắn súng phổ biến nhất hiện nay 23](#_Toc23057)

[3.1.2. Ưu và nhược điểm của các thể loại game bắn súng 23](#_Toc24)

[3.2. Tổng quan chức năng 24](#_Toc11084)

[3.2.1. Biểu đồ use case tổng quát 24](#_Toc7750)

[3.2.2. Biểu đồ use case phân rã vào cửa hàng súng 26](#_Toc16965)

[3.2.3. Biểu đồ use case phân rã mở kho Player 28](#_Toc3958)

[3.3. Đặc tả chức năng 29](#_Toc11399)

[3.3.1. Đặc tả use case mua súng 29](#_Toc30420)

[3.3.2. Đặc tả use case nâng cấp súng 30](#_Toc25883)

[3.3.3. Đặc tả usecase chiến đấu 31](#_Toc23981)

[3.4. Yêu cầu phi chức năng 32](#_Toc26869)

[3.4.1. Yêu cầu về kỹ thuật 32](#_Toc27284)

[3.4.2. Yêu cầu về thiết kế 32](#_Toc13315)

[CHƯƠNG 4. CÔNG NGHỆ SỬ DỤNG 34](#_Toc704)

[4.1. Ngôn ngữ lập trình C# và Mono frameWork 34](#_Toc5018)

[4.1.1. Giới thiệu 34](#_Toc12101)

[4.1.2. Lý do lựa chọn 34](#_Toc24909)

[4.2. Công cụ hỗ trợ phát triển trò chơi Unity 35](#_Toc11533)

[4.2.1. Giới thiệu 35](#_Toc4466)

[4.2.2. Tổng quan về các thành phần trong unity 35](#_Toc1481)

[4.2.3. Lý do lựa chọn 36](#_Toc6882)

[4.3. Công nghệ Navmesh 36](#_Toc16684)

[4.3.1. Giới thiệu 37](#_Toc10685)

[4.3.2. Một số khái niệm cơ bản 37](#_Toc31019)

[4.3.3. Lý do lựa chọn 37](#_Toc23858)

[CHƯƠNG 5. THỰC NGHIỆM VÀ ĐÁNG GIÁ 39](#_Toc12570)

[5.1. Thiết kế kiến trúc 39](#_Toc4250)

[5.1.1. Object 39](#_Toc3808)

[5.1.2. Class 39](#_Toc29293)

*[5.1.3.](#_Toc20119)* [Đặc tính cơ bản của OOP 39](#_Toc20119)

[5.2. Thiết kế chi tiết 41](#_Toc32727)

[5.2.1. Thiết kế giao diện 41](#_Toc19533)

[5.2.2. Thiết kế lớp 45](#_Toc29174)

[5.3. Xây dựng ứng dụng 56](#_Toc7372)

[5.3.1. Thư viện và công cụ sử dụng 56](#_Toc21000)

[5.3.2. Kết quả đạt được 58](#_Toc13706)

[5.3.3. Minh họa các chức năng chính 59](#_Toc21525)

[5.4. Kiểm thử 64](#_Toc25905)

[5.5. Triển khai 66](#_Toc21462)

[CHƯƠNG 6. CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT 68](#_Toc10598)

[6.1. Hiệu năng của trò chơi 68](#_Toc28140)

[6.1.1. Vấn đề gặp phải 68](#_Toc5038)

[6.1.2. Quá trình giải quyết vấn đề 69](#_Toc16282)

[6.1.3. Kết quả đạt được 71](#_Toc16043)

[6.2. Cách di chuyển của Enemy 71](#_Toc27052)

[6.2.1. Vần đề gặp phải 71](#_Toc7681)

[6.2.2. Quá trình giải quyết vấn đề 72](#_Toc5931)

[6.2.3. Kết quả đạt được 73](#_Toc1699)

[CHƯƠNG 7. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 74](#_Toc3848)

[7.1. Kết luận 74](#_Toc31327)

[7.2. Hướng phát triển 74](#_Toc20467)

[TÀI LIỆU THAM KHẢO 76](#_Toc22746)

**DANH SÁCH CÁC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **Từ viết tắt** | **Ý nghĩa** |
| Enemy | Kẻ địch người sẽ tấn công căn cứ |
| GameObject | Thành phần đối tượng trong một cảnh của trò chơi. |
| GamePlay | GamePlay để chỉ những cách thức người chơi trải nghiệm game. |
| Map | Bản đồ |
| UI | Giao diện |
| Scene | Một cảnh trong game |
| Unity | Tên công cụ phát triển trò chơi |

**DANH SÁCH CÁC BẢNG BIỂU**

[Bảng 2.1. Bảng giới thiệu súng 17](#_Toc167972996)

[Bảng 2.2. Bảng giới thiệu vật phẩm phụ trợ 18](#_Toc167972997)

[Bảng 2.3.Bảng phần thưởng khi qua màn 19](#_Toc167972998)

[Bảng 2.4.Các loại Enemy trong game 19](#_Toc167972999)

[Bảng 3.1.Mô tả chức năng trong Use case tổng quát 24](#_Toc167973000)

[Bảng 3.2.Mô tả chức năng trong use case phân rã chiến đấu của Player 25](#_Toc167973001)

[Bảng 3.3.Mô tả chức năng trong use case phân rã chiến đấu của Enemy 26](#_Toc167973002)

[Bảng 3.4.Mô tả chức năng trong use case phân rã mở kho Player 26](#_Toc167973003)

[Bảng 3.5. Bảng đặc tả Use Case mua súng 27](#_Toc167973004)

[Bảng 3.6.Bảng đặc tả Use Case nâng cấp súng 28](#_Toc167973005)

[Bảng 3.7.Bảng đặc tả Use Case chiến đấu 29](#_Toc167973006)

[Bảng 5.1.Thông tin thiết kế giao diện 39](#_Toc167973007)

[Bảng 5.2.Đặc tả lớp PlayerController 45](#_Toc167973008)

[Bảng 5.3.Bảng đặc tả PlayerMovement 46](#_Toc167973009)

[Bảng 5.4.Đặc tả lớp Weapon 48](#_Toc167973010)

[Bảng 5.5.Danh sách thư viện và công cụ sử dụng 57](#_Toc167973011)

[Bảng 5.6.Thông tin project 57](#_Toc167973012)

[Bảng 5.7.Kiểm thử về độ nguy hiểm của các kẻ địch có trong hệ thống 63](#_Toc167973013)

[Bảng 5.8.Kiểm thử về sức mạnh của từng loại vũ khí trong game 64](#_Toc167973014)

[Bảng 5.9.Kiểm thử về các chức năng của Player 65](#_Toc167973015)

[Bảng 5.10.Cấu hình tối thiểu 65](#_Toc167973016)

# DANH SÁCH CÁC HÌNH ẢNH

[Hình 3.1: Biểu đồ Use Case tổng quát. 21](#_Toc167973657)

[Hình 3.2: Biểu đồ Use case phân rã vào cửa hàng súng 22](#_Toc167973658)

[Hình 3.3: Biểu đồ Use Case phân rã chiến đấu của Enemy 23](#_Toc167973659)

[Hình 3.4: Biểu đồ use case phân rã mở kho Player 24](#_Toc167973660)

[Hình 4.1: Hình ảnh minh họa Navmesh [13] 33](#_Toc167973661)

[Hình 5.1: Thiết kế giao diện màn hình chính 38](#_Toc167973662)

[Hình 5.2: Thiết kế giao diện điểm danh hàng ngày 39](#_Toc167973663)

[Hình 5.3: Thiết kế giao diện cửa hàng súng 40](#_Toc167973664)

[Hình 5.4: Thiết kế giao diện trong màn hình chiến đấu 40](#_Toc167973665)

[Hình 5.5: Biểu đồ lớp liên quan đến các chức năng mà player có thể thực hiện khi chiến đấu 42](#_Toc167973666)

[Hình 5.6: Biểu đồ thể hiện trình tự mua súng 47](#_Toc167973667)

[Hình 5.7: Biểu đồ thực thể liên kết 48](#_Toc167973668)

[Hình 5.8: Biểu đồ các lớp dữ liệu 49](#_Toc167973669)

[Hình 5.9: Ví dụ về lưu trữ dữ liệu sử dụng ScriptTableObject 51](#_Toc167973670)

[Hình 5.10: Ví dụ về lưu trữ dữ liệu sử dụng PlayerPrefs 52](#_Toc167973671)

[Hình 5.11: Ví dụ về lưu trữ dữ liệu vào trực tiếp đối tượng trên Sence 53](#_Toc167973672)

[Hình 5.12: Màn hình chính của game 56](#_Toc167973673)

[Hình 5.13: Màn hình chiến đầu của map 1 57](#_Toc167973674)

[Hình 5.14: Màn hình nâng cấp súng 58](#_Toc167973675)

[Hình 5.15: Màn hình quản lý kho súng của người chơi 59](#_Toc167973676)

[Hình 5.16: Màn hình người chơi chọn map chơi 60](#_Toc167973677)

[Hình 6.1: Hình ảnh mở rương sinh quá nhiều vàng 65](#_Toc167973678)

[Hình 6.2: Hình ảnh hiệu ứng thủng tường, hiệu ứng bắn súng 66](#_Toc167973679)

[Hình 6.3: Ví dụ về cách di chuyển cũ của enemy 68](#_Toc167973680)

[Hình 6.4: Hình minh họa map 1 đã sử dụng Navmesh 69](#_Toc167973681)

[Hình 6.5: Hình minh họa gắn NavmeshAgent vào cho đối tượng 70](#_Toc167973682)

**TÓM TẮT NỘI DUNG ĐỒ ÁN**

Hiện tại, việc giải quyết nhu cầu giải trí đang trở thành một vấn đề quan trọng trong cuộc sống. Tuy nhiên, vẫn chưa có nhiều sản phẩm giải trí đáp ứng đầy đủ nhu cầu giải trí của người dùng, đặc biệt là những trò chơi có tính tập trung cao. Do đó, em quyết định phát triển một game 3D offline để giải quyết vấn đề này. Game có cốt truyện hấp dẫn và đầy thử thách. Người chơi sẽ nhập vai vào một điệp viên và hoàn thành nhiệm vụ được chỉ định bởi chỉ huy.

Nhiệm vụ đòi hỏi sự tập trung cao độ của người chơi, khi phải đối mặt với các thử thách và mối nguy hiểm khác nhau trong quá trình chiến đấu để hoàn thành nhiệm vụ và thoát khỏi căn cứ. Hướng tiếp cận mà Em lựa chọn để phát triển game này là sử dụng công nghệ Unity để tạo ra một môi trường chân thực và hấp dẫn cho người chơi. Những tài nguyên trong trò chơi như hình ảnh, âm thanh, đồ họa sẽ được lấy từ hai nguồn là Unity Asset Store hoặc tham khảo các tài nguyên miễn phí được cung cấp trên internet. Sau khi đã hoàn thành thì sản phẩm sẽ được phát hành lên cửa hàng của itch.io. Kết quả của ĐATN là một game 3D offline đầy hấp dẫn và đáp ứng đầy đủ nhu cầu giải trí của người dùng, đặc biệt là những người yêu thích những trò chơi có tính tập trung cao. Em tin rằng sản phẩm này sẽ mang lại nhiều giá trị cho người chơi và cộng đồng game thủ nói chung.

# GIỚI THIỆU ĐỀ TÀI

## Đặt vấn đề

Trong thời đại công nghệ 4.0, game bắn súng 3D ngày càng trở nên phổ biến và thu hút được sự quan tâm của đông đảo người chơi trên toàn thế giới. Tuy nhiên, việc tạo ra một game bắn súng 3D chất lượng, hấp dẫn và phù hợp với sở thích của người chơi là một thách thức lớn đối với các nhà phát triển game.[1]

Với mong muốn nghiên cứu và phát triển một game bắn súng 3D, đồ án tốt nghiệp này đặt ra vấn đề: Làm thế nào để tạo ra một game bắn súng 3D chất lượng, hấp dẫn và thu hút được sự quan tâm của người chơi?

Vấn đề này là rất cấp thiết và quan trọng bởi vì game bắn súng 3D không chỉ đòi hỏi kỹ năng lập trình và thiết kế đồ họa mà còn phải đảm bảo được tính hấp dẫn, sáng tạo, đồng thời tương thích với nhiều thiết bị khác nhau. Nếu giải quyết được vấn đề này, chúng ta có thể tạo ra một sản phẩm game bắn súng 3D đạt chất lượng cao, thu hút được sự quan tâm của người chơi và có thể mang lại lợi nhuận từ nhưng con game của mình.

Vì vậy, phần tiếp theo của đồ án tốt nghiệp này sẽ tập trung vào việc nghiên cứu và phát triển game bắn súng 3D để đưa ra giải pháp cho vấn đề trên.

## Mục tiêu và phạm vi đề tài

Trong phần này, em sẽ trình bày về mục tiêu và phạm vi đề tài của đồ án. Đầu tiên, em sẽ tổng quan lại các sản phẩm game bắn súng 3D hiện tại và đánh giá các hạn chế của chúng.

Hiện nay, trên thị trường đã có rất nhiều sản phẩm game bắn súng 3D phong phú và đa dạng về chủ đề, đồ họa, âm thanh, tính năng, gameplay, v.v. Tuy nhiên, sau khi phân tích và đánh giá các sản phẩm này, em nhận thấy vẫn tồn tại một số hạn chế nhất định. Đó là:

Hạn chế về tính tương tác và trải nghiệm người chơi: Các game bắn súng 3D hiện tại chưa đáp ứng được mức độ tương tác và trải nghiệm mà người chơi mong đợi. Việc tương tác giữa các nhân vật, vật phẩm, môi trường chưa thực sự tự nhiên và linh hoạt, dẫn đến sự hạn chế trong khả năng thực hiện các chiến thuật và phản ứng của người chơi.[1]

Hạn chế về độ phức tạp và tính khó vượt qua: Một số game bắn súng 3D quá khó hoặc quá dễ, không đáp ứng được mức độ thách thức của người chơi. Điều này dẫn đến tình trạng người chơi quá nhanh chóng chán và không muốn tiếp tục chơi.[1]

Hạn chế về tính đa dạng và sáng tạo: Các game bắn súng 3D hiện tại chưa đáp ứng được yêu cầu của người chơi về tính đa dạng và sáng tạo. Các chủ đề, tính năng, vũ khí, vật phẩm, v.v. trong game chưa đủ phong phú và đa dạng, dẫn đến sự nhàm chán khi chơi trong thời gian dài.[1]

Dựa trên các hạn chế trên, mục tiêu của đề tài là xây dựng một sản phẩm game bắn súng 3D đáp ứng được mức độ tương tác và trải nghiệm người chơi cao, tính thách thức phù hợp.

## Định hướng giải pháp

Dựa trên phân tích và đánh giá của phần 1.2, Đồ án sẽ phát triển trò chơi bằng công cụ hỗ trợ phát triển trò chơi Unity. Unity là một trong những công cụ phát triển game 3D phổ biến nhất hiện nay, với nhiều ưu điểm vượt trội giúp lập trình viên có thể phát triển game bắn súng 3D một cách dễ dàng và hiệu quả và nó hỗ trợ đa nền tảng, thư viện đa dạng, tích hợp dễ dàng.

Có sử dụng công nghệ Navmesh để tìm đường đi ngắn nhất giúp enemy có thể tìm thấy player một cách thông minh và tránh được các vật cản.

## Bố cục đồ án

Phần còn lại của báo cáo đồ án tốt nghiệp này được tổ chức như sau:

Chương 2 sẽ trình bày về thiết kế của trò chơi, bao gồm có bối cảnh, cốt truyện cho từng map, chi tiết trò chơi bao gồm danh sách kho súng, các Enemy, các vật phẩm phụ trợ trong game.

Chương 3 chương này sẽ có nhiệm vụ giới thiệu và so sánh các thể loại game bắn súng từ đó chọn một thể loại game tiềm năng nhất để phát triển, tiếp theo trình bày về những chức năng chính mà người dùng có thể thực hiện được với game và sẽ phân tích chúng theo các biểu đồ của UML.

Chương 4 trình bày về những công nghệ được sử dụng trong việc phát triển đồ án gồm có C# và Mono Framework, Engine phát triển trò chơi Unity, khái quát, giới thiệu về công nghệ cũng như lý do lựa chọn công nghệ đó, giải thích một số thuật ngữ cần biết trong Unity.

Chương 5 phân tích thiết kế chi tiết cho dự án, trình bày về kiến trúc được sử dụng, thiết kế giao diện. Sau đó là kết quả đã đạt được cùng với những ý kiến đóng góp của người dùng.

Chương 6 trình bày những khó khăn đã gặp phải, giải pháp và đóng góp nổi bật của đồ án.

Chương cuối cùng, kết quả đạt được sau quá trình nghiên cứu và xây dựng, cùng với đó là những gì cần được cải thiện và định hướng phát triển tiếp trong tương lai Sau đây là chương 2, trình bày về thiết kế của trò chơi.

# THIẾT KẾ TRÒ CHƠI

Ở Chương 1, đã giới thiệu về mục tiêu, phạm vi và hướng đi của đề tài. Trong chương này, phần trình bày là về sơ lược bối cảnh và cốt truyện trong trò chơi, phân tích thiết kế lối chơi chính.

## Bối cảnh câu chuyện

Do Nga đã chiếm được Ukraine vào năm 2023 và thành lập một chính phủ bù nhìn thân Nga. Thủ tướng Ukraine cùng với một triệu lính đã trốn sang Mỹ và cầu viện Mỹ. Mặc dù không trực tiếp tham chiến, Mỹ đã cung cấp vũ khí và cho 100 nghìn quân gia nhập vào quân đội Ukraine và quay trở lại chiến đấu. Nga nhận ra mình bị chơi xỏ và bắt giữ một lính Mỹ, ép buộc anh ta khai ra và công khai lên truyền hình. Người dân Mỹ thấy các chiến sĩ của mình bị Nga giết hại và căm phẫn, dẫn đến các cuộc biểu tình ở các bang của Mỹ. Tổng thống Mỹ buộc phải phát động chiến tranh với Nga, các đồng minh trong khối NATO không thể ngồi yên. Từ đó, cuộc chiến tranh thế giới thứ ba đã bùng nổ.[2]

## Cốt truyện chính

Câu chuyện kể về điệp viên hai mang, ông là điệp viên của một cơ quan tình báo của quốc gia Mỹ được gài vào quân Nga để dò thám thông tin quân sự và gửi về cho tổ chức. Những thông tin này có tầm quan trọng trong việc giúp Mỹ đánh bại Nga ở Thế chiến thứ 3. Nhưng sau khi bị phát hiện, ông đã được giao nhiều nhiệm vụ như là tiêu diệt căn cứ của địch, phá hủy thuốc sản sinh zombie, và thoát ra ngoài...

### Map Aranchu

Người chơi là một điệp viên đang đối mặt với một tình huống khẩn cấp. Kẻ địch đã phát hiện được người chơi là nội gián và họ đang tìm cách để giết chết. Nhiệm vụ của người chơi là tiêu diệt hết 6 kẻ thù trong căn cứ để giữ lại an toàn cho mình. Người chơi bước vào căn cứ với súng và boom sẵn sàng. Tuy nhiên, người chơi biết rằng nhiệm vụ này không dễ dàng khi người chơi phải đối đầu với những kẻ địch khắp nơi. Các kẻ thù đã phân bố rải rác trên toàn bộ căn cứ và bảo vệ bằng cảm biến và bẫy.

Người chơi phải tỉnh táo và chủ động, di chuyển từng bước, phản ứng nhanh nhạy để đối phó với các cuộc tấn công của kẻ thù. Nhiệm vụ của người chơi là tiêu diệt hết tất cả kẻ thù kết sống sót. Các kẻ thù xuất hiện từ khắp nơi, và người chơi phải sử dụng súng và lựu đạn của mình để đánh bại chúng. Người chơi phải chủ động tìm kiếm và hạ gục các kẻ thù, đồng thời tránh né các cạm bẫy và tấn công của chúng.

Sau khi hạ gục được kẻ địch, người chơi tiến vào căn cứ và giải quyết những thử thách tiếp theo. Những thử thách tiếp theo sẽ càng khó khăn hơn, vì vậy người chơi cần phải sẵn sàng cho mọi thử thách và cố gắng hết sức để đưa ra những quyết định đúng đắn

### Map Haru

Sau khi thành công số sót ở map 1 gián điệp của chúng ta tiếp tục thực hiện nhiệm vụ tiêu diệt quân địch trên map 2. Anh ta đã được chỉ đường đến một căn hầm bí mật nơi quân địch đang sản xuất zombie và chuẩn bị tiến hành tấn công vào thành phố.

Khi gián điệp của chúng ta tiến vào căn hầm, anh ta phát hiện ra rằng nơi đây đã bị quân địch trang bị sẵn một đội quân bắn súng đầy tinh nhuệ để bảo vệ hoạt động sản xuất của họ. Người chơi sẽ phải sử dụng tất cả kỹ năng bắn súng của mình để vượt qua đội quân bảo vệ và tiêu diệt hết các tay súng này.

Tuy nhiên, không chỉ có đội quân bảo vệ, căn hầm này cũng rất nguy hiểm với những con zombie đang được sản xuất. Người chơi sẽ phải đối mặt với những con zombie hung dữ và sử dụng các vũ khí có sẵn để tiêu diệt chúng.

Trong suốt quá trình di chuyển, người chơi sẽ phát hiện ra những bình máu trên đường đi. Những bình máu này sẽ giúp người chơi hồi phục sức khoẻ để tiếp tục nhiệm vụ của mình. Tuy nhiên, để tiếp cận được những bình máu này, người chơi sẽ phải vượt qua những trở ngại và thử thách khác nhau.

Cuối cùng, khi gián điệp của chúng ta đã tiêu diệt được toàn bộ quân địch và zombie trong căn hầm, anh ta tiếp tục tiến vào map tiếp theo để tiếp tục nhiệm vụ của mình.

### Map Kanu

Bối cảnh của map 3 là khu doanh trại quân đội, nơi đây đang sản xuất và cung cấp các loại thuốc để tạo ra zombie. Nhiệm vụ của người chơi là phát hủy thuốc sản sinh ra zombie.

Trong khu doanh trại, người chơi sẽ gặp phải các lính quân địch đang xếp hàng chờ lệnh chiến đấu. Để tìm và phá hủy được thuốc, người chơi phải tìm đường đến kho lưu trữ thuốc và các thiết bị sản xuất. Trong khi di chuyển, người chơi sẽ gặp phải các lính quân địch đang tuần tra trong khu vực của mình. Để tránh bị phát hiện, người chơi cần phải di chuyển cẩn thận và tránh xa các tay súng địch.

Người chơi sẽ được trang bị vũ khí và đạn dược để tiêu diệt lính quân địch và phá huỷ các thiết bị trong doanh trại. Ngoài ra, người chơi còn được trang bị một con AI tàn hình để dẫn đường đến các mục tiêu cần phá huỷ. Con AI này có khả năng tìm kiếm đường đi ngắn nhất giúp người chơi thuật tiện trong quá trình tìm kiếm.

Tuy nhiên, việc phá huỷ doanh trại không phải là dễ dàng. Người chơi sẽ gặp phải các trở ngại như các bẫy mìn, chướng ngại vật và các tay súng địch tinh nhuệ. Vì vậy, người chơi cần phải sử dụng kỹ năng, sự khéo léo và nhanh nhẹn để hoàn thành nhiệm vụ. [3]

## Thiết kế lối chơi

### Map Aranchu

***Xác định các nhân vật chính:***

Người chơi: Là nhân vật mà người chơi sẽ hóa thân vào trong game, được trang bị súng và bom để tiêu diệt kẻ thù.

Kẻ thù: Là những kẻ thù mà người chơi sẽ phải đối đầu, với số lượng là 6 trong trường map này.

***Xác định nhiệm vụ:***

Mục tiêu của người chơi trong map 1 là phải tiêu diệt hết 6 kẻ thù để qua màn.

***Xác định bối cảnh:***

Bối cảnh của map 1 có thể là một căn cứ quân sự hoặc một khu vực địch kiểm soát. Trong căn cứ hoặc khu vực này, người chơi phải đối đầu với 6 kẻ thù được phân bố khắp nơi.

***Tạo sự căng thẳng:***

Khi người chơi tiến vào map, họ sẽ gặp ngay kẻ thù và phải bắn để tự bảo vệ. Kẻ thù sẽ nằm trong nhiều vị trí khác nhau, không chỉ ở một vị trí nhất định, vì vậy người chơi sẽ phải chủ động tìm kiếm và di chuyển để tìm ra chúng. Ngoài ra, người chơi còn phải đối mặt với những cạm bẫy và tấn công của kẻ thù.

***Kết thúc:***

Khi người chơi tiêu diệt hết 6 kẻ thù, họ sẽ hoàn thành mục tiêu của map 1 và có thể tiếp tục vào map tiếp theo

### Map Kanu

***Xác định các nhân vật chính:***

Người chơi: người sẽ hóa thân vào một gián điệp được giao nhiệm vụ tiêu diệt các zombie được sản xuất trong căn hầm của quân địch.

Kẻ thù: 2 tay súng địch và các zombie

***Xác định nhiệm vụ:***

Người chơi phải tiêu diệt hết các zombie được sản xuất trong căn hầm của quân địch. Người chơi cũng sẽ đối đầu với các tay súng địch được cử đến để bảo vệ căn hầm và ngăn chặn người chơi tiêu diệt zombie. Người chơi sử dụng các loại vũ khí như súng và boom để tiêu diệt zombie và tay súng địch.

***Xác định bối cảnh:***

Người chơi sẽ phải đối mặt với các con zombie liên tục xuất hiện và tấn công. Đồng thời, trong căn hầm sản xuất zombie này còn có các tay súng giỏi của quân địch, sẵn sàng giết chết bất kỳ ai đặt chân vào khu vực của họ. Người chơi sẽ phải sử dụng các kỹ năng bắn súng và chiến đấu của mình để vượt qua được các thử thách và nguy hiểm này. Các bình máu sẵn có trên đường đi cũng giúp người chơi tái tạo sức khỏe và tiếp tục hành trình tiêu diệt zombie và đối đầu với quân địch

***Tạo sự căng thẳng:***

kết thúc sẽ được đạt khi người chơi tiêu diệt hết tất cả các zombie và đạt được mục tiêu là phá hủy cơ sở sản xuất thuốc của quân địch. Sau khi hoàn thành mục tiêu, người chơi sẽ được chuyển đến map tiếp theo hoặc kết thúc game. Trong quá trình chơi, người chơi cũng có thể gặp phải các tình huống nguy hiểm và cần phải tìm cách vượt qua để đạt được mục tiêu, tạo ra thêm sự căng thẳng và hứng thú cho trò chơi.

***Kết thúc:***

Khi người chơi tiêu diệt hết số Enemy và Zombie, có thể tiếp tục vào map tiếp theo.

### Map Muda

***Xác định các nhân vật chính:***

Người chơi: là một chiến sĩ trong đội quân đặc nhiệm được giao nhiệm vụ tiêu diệt căn cứ sản xuất thuốc zombie của kẻ thù.

Các tay súng: là những tay súng giỏi của đối thủ, sẽ xuất hiện khắp nơi để ngăn cản nhiệm vụ của người chơi.

***Xác định nhiệm vụ:***

Nhiệm vụ của người chơi là tiêu diệt kẻ địch và phá hủy thuốc zombie. Trong quá trình di chuyển, người chơi cần phá hủy các cấu trúc, vật phẩm liên quan đến sản xuất thuốc zombie để giảm sức mạnh của kẻ thù.

***Xác định bối cảnh:***

Map 3 diễn ra trong một khu doanh trại quân đội bí mật, nơi sản xuất thuốc zombie. Khu doanh trại được chia thành nhiều khu vực, trong đó có trại lính, khu sản xuất, kho lưu trữ, trạm phóng tên lửa... Người chơi sẽ phải đi qua các khu vực này để đạt được mục tiêu.

***Tạo sự căng thẳng:***

Khu doanh trại quân đội rất bí mật và được bảo vệ chặt chẽ bởi các tay súng giỏi của kẻ thù, các các thay súng còn có khả năng phát hiện và báo cáo cho đồng đội mình biết để tấn công người chơi do đó người chơi sẽ phải tiếp cận mục tiêu thông qua nhiều thủ đoạn và mưu mẹo. Các tay súng của đối thủ cũng được trang bị vũ khí tối tân và sẵn sàng tiêu diệt người chơi, tạo ra sự căng thẳng và khó khăn trong việc tiến hành nhiệm vụ.

***Kết thúc:***

Sau khi tiêu diệt toàn bộ căn cứ sản xuất thuốc zombie và phá hủy các cấu trúc liên quan, người chơi sẽ thành công trong nhiệm vụ của mình. Tuy nhiên, sự thành công này không đến dễ dàng, người chơi sẽ phải đối mặt với nhiều thử thách và nguy hiểm trong quá trình di chuyển và tiêu diệt kẻ thù. [4]

## Chi tiết về game

### Súng có thể sử dụng trong game

Trong bảng 2.1 Giới thiệu tất cả các súng có trong game. Ban đầu bạn sẽ được tặng miễn phí 2 súng khi vào game, sau đó thì bạn phải dùng tiền của mình để mua các loại súng khác.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên | Số đạn | Tốc độ đạn | Tính sát thương | Hình ảnh |
| AK47 | 30 | 600 m/s | 47 |  |
| AWP | 10 | 1000 m/s | 11 |  |
| M4A1 | 30 | 600 m/s | 33 |  |
| Scar20 | 20 | 800 m/s | 80 |  |
| UMP45 | 40 | 700 m/s | 50 |  |

Bảng 2.1. Bảng giới thiệu súng

Súng có thể được mua bằng vàng hoặc bằng kim cương của người chơi, mỗi súng có thể nâng cấp lên cấp cao hơn bằng kim cương và nó sẽ tăng được, sát thương, số lượng đạn ..

### Các loại vật phẩm hỗ trợ

Trong bảng 2.2 minh họa bảng giới thiệu vật phẩm phụ trợ trong game bao gồm công dụng và hình ảnh của chúng.

|  |  |  |
| --- | --- | --- |
| Tên | Công dụng | Hình ảnh |
| Bình máu | Giúp hồi phục một lượng máu nhất định |  |
| Lựu đạn | Tấn công nhiều kẻ địch, với sát thương cao |  |
| Con dao | Tất công kẻ định nếu ở phạm vi gần |  |
| Ống nhòm | giúp nhìn rõ hơn mục tiêu |  |
| Băng đạn | Thay bắng đạn cho súng |  |

Bảng 2.2. Bảng giới thiệu vật phẩm phụ trợ

***Bình máu:*** Bình máu được sử dụng để hồi phục sức khỏe của người chơi. Trong các trận đấu căng thẳng, người chơi thường bị gây thương tích và cần phải nhanh chóng hồi phục để tiếp tục chiến đấu. Do đó, bình máu là một vật phẩm cần thiết để đảm bảo sức khỏe của người chơi.

***Lựu đạn:*** Lựu đạn là một vật phẩm để nâng cao khả năng chiến đấu của người chơi. Khi sử dụng lựu đạn, người chơi có thể nhanh chóng nạp lại đạn cho vũ khí của mình mà không phải tốn nhiều thời gian để đổi súng hoặc đi tìm đạn. Điều này giúp người chơi tiết kiệm thời gian và tập trung hơn vào việc chiến đấu.

***Con dao:*** Con dao được sử dụng cho các trường hợp gần gũi, khi không thể sử dụng vũ khí súng được. Nó giúp người chơi tấn công một cách nhanh chóng và chính xác hơn so với việc sử dụng vũ khí súng trong khoảng cách gần.

***Ống nhòm:*** Ống nhòm là một thiết bị giúp người chơi nhìn xa hơn và xác định vị trí của kẻ địch.

### Thiết kế phần thưởng khi qua màn

Bảng 2.3 thể hiện số kim cương và số lượng chìa khóa bạn có thể nhận được khi qua một màn.

|  |  |  |
| --- | --- | --- |
| Map | Số kim cương | Số lượng key |
| Map 1 | 150 | 2 |
| Map 2 | 200 | 1 |
| Map 3 | 150 | 3 |
| Map 4 | 200 | 2 |
| Map 5 | 200 | 3 |

Bảng 2.3.Bảng phần thưởng khi qua màn

Thu thập thìa khóa bạn sẽ có thể mở rương bí ẩn để có vàng, và bạn có thể dùng số vàng của mình để đi mua súng.

### Các loại kẻ địch trong game

Trong bảng 2.4 đã liệt kế các con enemy bao gồm tên, mô tả, hình ảnh của nó.

|  |  |  |
| --- | --- | --- |
| Tên kẻ địch | Mô tả | Hình ảnh |
| Shadow Assassins | Tấn công tầm xa, tìm player, tuần tra |  |
| Dark Reapers | Tuần tra, báo cáo khi phát hiện player, tấn công player |  |
| Grim Reapers | Tấn công tầm gần, tìm player, tuần tra |  |

Bảng 2.4.Các loại Enemy trong game

***Dark Reapers*** là một con enemy xuất hiện ở cả map 1 và map 3. Nó có khả năng tuần tra, tìm kiếm player và tấn công từ xa. Với sức mạnh và sự linh hoạt của mình, Dark Reapers có thể gây ra nhiều rắc rối cho người chơi. Tuy nhiên, điều đáng sợ hơn cả là khả năng báo cáo vị trí của người chơi cho đồng đội khi đang ở map 3. Điều này khiến cho việc tiêu diệt Dark Reapers trở nên khó khăn hơn bao giờ hết.

***Grim Reapers*** mặc dù chỉ có khả năng tấn công ở tầm gần nhưng sát thương của nó rất mạnh và máu của nó cũng nhiều hơn so với các loại enemy khác. Điều này khiến cho người chơi phải cực kỳ cẩn thận khi tiếp cận với nó. Một lần bị tấn công bởi Grim Reapers, người chơi sẽ nhanh chóng thất bại.

Trong chương này đã trình bày về thiết kế bối cảnh, cốt truyện và lối chơi. Chương 3 sẽ đi vào phân tích, thiết kế và đưa ra những chức năng và yêu cầu cần thiết cho trò chơi.

# KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

Ở Chương 2, đã trình bày về thiết kế của trò chơi. Chương 3 sẽ có nhiệm vụ giới thiệu và so sánh các thể loại game bắn súng từ đó chọn một thể loại game tiềm năng nhất để phát triển, tiếp theo trình bày về những chức năng chính mà người dùng có thể thực hiện được với game và phân tích chúng theo các biều đồ của UML.

## Khảo sát hiện trạng

### Các thể loại game bắn súng phổ biến nhất hiện nay

Game bắn súng hiện nay được phân loại thành nhiều thể loại khác nhau, tùy vào cách chơi, phong cách, đồ họa, âm thanh, tính năng và mục đích chơi của từng loại game. Tuy nhiên có 3 thể loại game bắn súng phổ biến nhất.

1.First Person Shooter (FPS): Đây là thể loại game bắn súng được thiết kế với góc nhìn người thứ nhất. Người chơi sẽ nhìn thấy toàn bộ môi trường trò chơi thông qua góc nhìn của nhân vật trong game. Ví dụ: Call of Duty, Battlefield, Doom, Counter-Strike.[5]

2.Third Person Shooter (TPS): Thể loại game bắn súng này thiết kế với góc nhìn từ phía sau nhân vật, cho phép người chơi quan sát được toàn bộ nhân vật và môi trường xung quanh. Ví dụ: Gears of War, Max Payne, Resident Evil.[5]

3.Battle Royale: Thể loại game bắn súng có yếu tố sinh tồn, trong đó người chơi phải chiến đấu với nhau để sống sót. Game thường có hàng trăm người chơi và chỉ một người chiến thắng. Ví dụ: Fortnite, PlayerUnknown’s Battlegrounds (PUBG), Apex Legends.[6]

### Ưu và nhược điểm của các thể loại game bắn súng

1.Third Person Shooter (TPS)

Ưu điểm: Góc nhìn từ xa giúp người chơi có cái nhìn tổng thể và tốt hơn trong việc di chuyển và tấn công. Đồ họa được chăm chút, chi tiết hơn so với FPS. Thường có nhiều nhân vật với tính cách và cốt truyện khác nhau, giúp người chơi dễ dàng đồng cảm và tham gia vào câu chuyện. Tính chiến thuật cao hơn, người chơi phải suy nghĩ và tính toán kỹ trước khi tấn công hoặc di chuyển.[7]

Nhược điểm: Tốc độ gameplay chậm hơn so với FPS, không thực sự phù hợp với những người thích hành động nhanh. Khó khăn trong việc ẩn nấp và tránh đạn, đặc biệt khi phải đối đầu với nhiều kẻ địch. Đôi khi góc nhìn không được tối ưu, gây khó chịu và khó chơi.

2.First Person Shooter (FPS)

Ưu điểm: Tạo cảm giác chân thực, sống động nhờ góc nhìn trực tiếp từ con mắt của nhân vật. Tốc độ gameplay nhanh, hành động liên tục đòi hỏi sự tập trung và phản xạ nhanh. Dễ dàng tạo ra các pha đánh lừa và bất ngờ đối với người chơi khác. Thường có chế độ chơi đơn và đa người chơi (multiplayer), thu hút đông đảo game thủ trên toàn thế giới.[8]

3.Battle Royale

Ưu điểm: Cung cấp trải nghiệm chơi game đầy kịch tính và căng thẳng khi người chơi cần phải giành chiến thắng trên một hòn đảo hoặc trong một khu vực thu nhỏ. Có tính tương tác cao giữa người chơi, đặc biệt là trong chế độ chơi đơn hoặc đội hình. Thường được cập nhật thường xuyên với nhiều tính năng mới, giữ cho game luôn hấp dẫn và thú vị.[6]

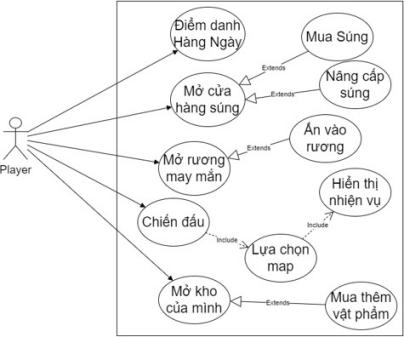
Nhược điểm: Có thể trở nên đơn điệu và nhàm chán nếu chơi quá nhiều lần hoặc không có động lực để giành chiến thắng. Những vấn đề liên quan đến kết nối mạng và độ trễ có thể ảnh hưởng đến trải nghiệm chơi game của người chơi.

Dựa trên những phân tích trên, tuỳ thuộc vào sở thích và yêu cầu của người chơi. Tuy nhiên, Dòng game bắn súng First Person Shooter (FPS) được cho là tiềm năng nhất vì trong FPS, người chơi có thể trải nghiệm một cách tương đối chân thực về việc sử dụng vũ khí và chiến đấu. FPS có tính tương tác cao giữa người chơi, đặc biệt là trong các chế độ chơi đối kháng.

## Tổng quan chức năng

### Biểu đồ use case tổng quát

Tác nhân Player vài trò là người chơi, người mà vào hệ thống và có thể thực hiện được các chức năng ở trên như: điểm danh, chiến đấu, mở cửa hàng súng…



Hình 3.1: Biểu đồ Use Case tổng quát.

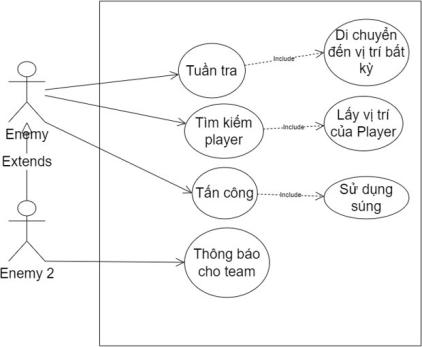
|  |  |
| --- | --- |
| Tên chức năng | Mô tả chức năng |
| Điểm danh hàng ngày | Mỗi một ngày khi người chơi vào game  thì hệ thống sẽ tiến hành điểm danh( mỗi ngày một lần) và sẽ có nhưng phần thưởng ví dụ như: vàng, kim cương, bình máu, hoặc súng... những vật phẩm này sẽ được cộng trực tiếp vào khó của người dùng. |
| Mở cửa hàng súng | Ngoài 2 súng mà người chơi đã được  cung cấp lúc mới đầu thì họ có thể vào cửa hàng để mua thêm súng, và có thể nâng cấp như khẩu súng mà mình đã sửa hữu. |
| Mở rương may mắn | Muốn kiếm được vàng để mua súng hoặc  mua các vật phẩm phụ trợ thì người chơi phải chơi game và vượt qua các map chơi để thu thập thìa khóa và từ thìa khóa sẽ mở đường rương bía ẩn, số vàng là ngẫu nhiên tùy thuộc vào độ may mắn của người chơi. |
| Chiến đầu | Người chơi sẽ vừa qua các map có các  nhiệm vụ khác nhau như: giết hết quân địch, phá hủy thuốc zombie... |
| Kho lưu chữ | Người cho sẽ vào đây để có thể xem số  lượng các vật phầm phụ trợ và có thể mua thêm ví dụ như bình hp, lựu đạn, đạn... |

Bảng 3.1.Mô tả chức năng trong Use case tổng quát

### Biểu đồ use case phân rã vào cửa hàng súng

Hình 3.2: Biểu đồ Use case phân rã vào cửa hàng súng

|  |
| --- |
|  |
| Tên chức năng | Mô tả chức năng |
| Hiển thị thông tin súng | Khi người dùng vào cửa hàng súng sẽ chọn súng và hệ thống sẽ hiển thị thông tin súng bao gồm giá tiền, độ sát thương, độ chính xác... |
| Mua súng | Khi bạn chọn súng nếu súng đó bạn chưa sở hữu bạn có thể mua nó bằng vàng hoặc bằng kim cương |
| Nâng cấp súng | Nếu súng bạn chọn đã thuộc quyền sở hữu của bạn thì có thể nâng cấp nó để lên level cao hơn tương ứng vào các chỉ số như độ chính xác, tinh sát thương sẽ cao hơn |

Bảng 3.2.Mô tả chức năng trong use case phân rã chiến đấu của Player

Hình 3.3: Biểu đồ Use Case phân rã chiến đấu của Enemy

Có 2 tác nhân chính ở trong Use case này là Enemy 1 Và Enemy 2

Enemy 1: Có các chức năng chính là tuần tra, tìm kiếm player, tấn công player Enemy 2: Có tất cả các chức năng giống Enemy 1 ngoài ra còn có thêm chức năng thông báo cho team biết khi phát hiện có player.

|  |  |
| --- | --- |
| Tên chức năng | Mô tả chức năng |
| Tuần tra | Enemy tự động di chuyển đến một vị trí  bất kỳ trong bản đồ để tìm Player |
| Tìm kiếm Player | Enemy có thể tự tìm kiếm player và chạy  đến để tấn công |
| Tân công | Enemy có thể sử dụng súng để tấn công  Player |

Bảng 3.3.Mô tả chức năng trong use case phân rã chiến đấu của Enemy

### Biểu đồ use case phân rã mở kho Player

Hình 3.4: Biểu đồ use case phân rã mở kho Player

|  |  |
| --- | --- |
| Tên chức năng | Mô tả chức năng |
| Mở kho | Player có thể mở kho đồ của mình và xem vật phẩm phụ trợ ( bình máu, lựu đạn, đạn.. ) họ cũng có thể mua thêm để phụ vụ cho các trận đấu |

Bảng 3.4.Mô tả chức năng trong use case phân rã mở kho Player

## Đặc tả chức năng

### Đặc tả use case mua súng

|  |  |
| --- | --- |
| Mã UseCase | 001 |
| Tên UseCase | Mua súng |
| Tác nhân | Người chơi |
| Tiền điều kiện | Người chơi chưa sở hữu khẩu súng này. |
| Luồn sự kiện chính | 1. Người chơi: Ấn vào icon cửa hàng súng. 2. Hệ thống: Hiển thị danh sách súng mà cửa hàng đó đang có. 3. Người chơi: Chọn một khẩu súng mà mình yêu thích. 4. Hệ thống: Hiển thị Model 3D khẩu súng đó, cũng như các thông tin khác như giá khẩu súng khi mua bằng vàng hoặc bằng kim cương, độ chính xác, tính sát thương... 5. Người chơi: Ấn chọn mua bằng vàng hoặc kim cương nếu đủ hiển thị đã mua thành công và cập nhật lại trang thái cho khẩu súng. Nếu không có đủ tiền thì chuyển sang luồng rẽ nhánh A1 6. UseCase kết thúc. |

Bảng 3.5. Bảng đặc tả Use Case mua súng

### Đặc tả use case nâng cấp súng

|  |  |
| --- | --- |
| Mã UseCase | 002 |
| Tên UseCase | Nâng cấp súng |
| Tác nhân | Người chơi |
| Tiền điều kiện | Khẩu súng của người chơi đạng sở hữu chưa đạt  Level Max. |
| Luồn sự kiện chính | 1. Người chơi: Ấn vào icon cửa hàng súng. 2. Hệ thống: Hiển thị danh sách súng mà cửa hàng đó đang có. 3. Người chơi: Chọn một khẩu súng mà mình đã sở hữu. 4. Hệ thống: Hiển thị Model 3D khẩu súng đó, cũng như các thông tin khác level hiện tại của khẩu súng và các chỉ số của level đó như độ chính xác, tính sát thương... 5. Người chơi: Ấn chọn nâng cấp súng. 6. Hệ thống: Hiển thị thông tin súng level tiếp theo. 7. Người chơi: Xác nhận nâng cấp súng bằng kim cương nếu thành công cập nhật lại trang thái của súng ở level tiếp theo, nếu thất bạn thì chuyển sang luồng rẽ nhanh A1 8. Use Case kết thúc. |
| Luồng sự kiện rẽ nhánh | Luồng rẽ nhánh A1: Người chơi không có đủ kim  cương để nâng cấp súng.   1. Hệ thống: thống báo cho người chơi biết là không có đủ kim cương. 2. Người chơi: Ấn thoát khỏi cửa hàng để chiến đầu thắng qua các màn kiếm kim cương để mua súng |
| Hậu điều kiện | Người chơi đã nâng cấp được súng |

Bảng 3.6.Bảng đặc tả Use Case nâng cấp súng

### Đặc tả usecase chiến đấu

|  |  |
| --- | --- |
| Tên Use Case | Chiến đấu |
| Tác nhân | Người chơi |
| Tiền điều kiện | Người chơi chưa chơi Map này |
| Luồn sự kiện chính | 1. Người chơi: Ấn vào icon chiến đấu. 2. Hệ thống: Hiển thị danh sách danh sách các map chơi cho player chọn. 3. Người chơi: Chọn một map chơi. 4. Hệ thống: Hiển thị thông tin bao gồm: nhiệm vụ của map chơi đó và phần thưởng nhận được khi hoàn thành, đạt được càng nhiều sao thì phần thưởng càng lớn. 5. Người chơi: Ấn chọn vào map và chiến đấu. 6. Hệ thống: Chuyển sang một scence mới có chưa map đó, các con enemy sẽ tự động tuần tra, tìm kiếm và tấn công player. phụ thuộc vào từng nhiệm vụ của từng map. 7. Người chơi: Điều khiển nhân vật của mình, tiêu diệt các kẻ thù để hoàn thành nhiệm vụ. Nếu người chơi bị chết khi chưa hoàn thành được một chút nhiệm vụ nào hệ thống sẽ chuyển sang luồng rẽ nhanh A1. 8. Hệ thống: Hiển thị màn hình thắng cuộc. 9. Hệ thống sẽ tính số sao của map đó cho player và phần thưởng sẽ được cộng và lưu vào hệ thống. 10. Use Case kết thúc. |
| Luồng sự kiện rẽ nhánh | Luồng rẽ nhánh A1: Người chơi bị chết khi chưa  hoàn thành nhiệm vụ   1. Hệ thống: Hiển thị màn hình thua cuộc. 2. Người chơi: Ấn thoát gọi map quay lại màn hình chính và quay lại bước 1 của luồng chính |
| Hậu điều kiện | Người chơi qua được map chơi, hiển thị số sao đạt  được |

Bảng 3.7.Bảng đặc tả Use Case chiến đấu

## Yêu cầu phi chức năng

### Yêu cầu về kỹ thuật

1. Hệ thống điều khiển của game cần phải được thiết kế tốt để đảm bảo sự chính xác và nhanh nhạy trong việc bắn súng và di chuyển. Cần có nút tấn công, nút di chuyển, nút nhảy, nút ngồi, nút đổi vũ khí, nút thay đổi tầm nhìn, nút ẩn và các phím tắt khác.

2. Đồ họa: Đồ họa trong game là yếu tố rất quan trọng, ảnh hưởng đến trải nghiệm chơi game của người chơi. Cần chú ý đến việc sử dụng công nghệ đồ họa hiện đại để tạo ra những hình ảnh sống động, chân thực, đẹp mắt và tối ưu hóa hiệu suất để tránh giật lag.

3. Âm thanh: Âm thanh cũng là một yếu tố quan trọng, giúp tăng thêm cảm giác thực tế khi chơi game. Cần sử dụng những hiệu ứng âm thanh phù hợp để đưa người chơi vào không khí của trò chơi.

4. Tiêu hao tài nguyên: Game cần được thiết kế để sử dụng tài nguyên máy tính hiệu quả và tiết kiệm.

5. Tính bảo mật: Game cần được bảo mật để tranh người chơi hack vàng hoặc kim cương dẫn đến phá đảo game dễ hơn rất nhiều.

### Yêu cầu về thiết kế

1. Thiết kế gameplay:

Thiết kế gameplay hấp dẫn, đa dạng và phong phú để thu hút sự chú ý của người chơi.

Tạo ra các bản đồ chơi đa dạng, có tính chiến thuật cao để người chơi có thể thỏa mãn sự sáng tạo và trí tưởng tượng của mình.

Đảm bảo tính cân bằng giữa các vũ khí để tạo ra một trải nghiệm chơi game công bằng.

Chương này đã trình bày kết quả khảo sát và so sánh những thể loại game bắn súng phổ biến nhất hiện nay. Từ những ưu điểm và nhược điểm của những thể loại này ta thấy dòng game bắn súng First Person Shooter (FPS) được cho là tiềm năng nhất vì trong FPS, người chơi có thể trải nghiệm một cách tương đối chân thực về việc sử dụng vũ khí và chiến đấu. Chương này cũng trình bày cho ta thấy các Use Case chính trong hệ thống và đặc tả các Use Case này. Chương tiếp theo sẽ trình bày về những công nghệ được sử dụng để phát triển trò chơi này.

# CÔNG NGHỆ SỬ DỤNG

Chương 3 đã trình bày hiện trạng và phân tích thiết kế hệ thống. Chương 4 sẽ trình bày các công nghệ sử dụng để xây dựng trò chơi này. Các công nghệ được sử dụng trong đồ án này bao gồm, ngôn ngữ lập trình C#, công cụ hỗ trợ phát triển trò chơi Unity và công nghệ Navmesh.

## Ngôn ngữ lập trình C# và Mono frameWork

### Giới thiệu

C# là một ngôn ngữ lập trình đối tượng được phát triển bởi Microsoft. Nó là một trong những ngôn ngữ được sử dụng rộng rãi trong việc phát triển game trên Unity. C# có cú pháp đơn giản và dễ hiểu, hỗ trợ đa nền tảng và được tích hợp trong Visual Studio IDE, là một trong những công cụ lập trình phổ biến nhất trên thị trường.[9]

Mono Framework là một nền tảng ứng dụng đa nền tảng, cung cấp môi trường chạy ứng dụng .NET trên các hệ điều hành khác nhau, bao gồm cả hệ điều hành di động như iOS và Android. Mono cung cấp một số thành phần chính để phát triển game trên Unity, bao gồm Mono Runtime, Mono C# Compiler và Mono Class Libraries.[10]

Trên Unity, C# được sử dụng để phát triển các script điều khiển hành động của các đối tượng trong game, xử lý sự kiện và tương tác với các thành phần khác của game. Mono Framework cung cấp môi trường chạy các script C# trong Unity, cho phép phát triển game đa nền tảng và tích hợp các tính năng bảo mật và xử lý nhanh.

Tổng quan, C# và Mono Framework là những công nghệ quan trọng và được sử dụng rộng rãi trong việc phát triển game trên Unity.

### Lý do lựa chọn

C# là ngôn ngữ lập trình chính được sử dụng trên Unity. C# là ngôn ngữ lập trình mạnh mẽ, linh hoạt và dễ học, đặc biệt là đối với những người đã từng làm việc với các ngôn ngữ lập trình như C++ và Java.

Mono Framework là một bộ công cụ mã nguồn mở hỗ trợ việc phát triển ứng dụng chạy trên nhiều nền tảng, bao gồm cả Windows, MacOS và Linux. Mono Framework cho phép các nhà phát triển tạo ra các ứng dụng đa nền tảng một cách dễ dàng.

C# và Mono Framework có thể tích hợp dễ dàng với Unity. Unity hỗ trợ C# và cho phép các nhà phát triển sử dụng Mono Framework để phát triển các ứng dụng chạy trên nhiều nền tảng.

C# và Mono Framework cung cấp các tính năng và thư viện hỗ trợ mạnh mẽ cho việc phát triển game. Các tính năng này bao gồm đa luồng, lập trình hướng đối tượng, thư viện đồ họa và âm thanh, và nhiều tính năng khác.

## Công cụ hỗ trợ phát triển trò chơi Unity

### Giới thiệu

Unity [11] là một trình giúp phát triển game đa nền tảng được phát triển bởi Unity Technologies. Unity cho phép người dùng phát triển các trò chơi trên nhiều nền tảng như Windows, macOS, Linux, Android, iOS, Xbox, PlayStation, Nintendo Switch và nhiều hơn nữa. Unity được sử dụng phổ biến trong ngành công nghiệp game để phát triển các trò chơi 2D và 3D.

Unity được thiết kế để cung cấp cho các nhà phát triển một môi trường làm việc linh hoạt và dễ sử dụng, bao gồm một bộ công cụ đồ họa, một bộ công cụ lập trình và một bộ công cụ hỗ trợ game đa nền tảng. Unity cũng cung cấp cho các nhà phát triển một thư viện phong phú của các công cụ và tài liệu học tập để giúp họ phát triển các trò chơi đa dạng và phức tạp.

Với Unity, các nhà phát triển có thể tạo ra các trò chơi với đồ họa ấn tượng, hiệu ứng âm thanh, hệ thống vật lý, các tính năng AI và đa dạng các cơ chế gameplay khác nhau. Ngoài ra, Unity cũng hỗ trợ các công nghệ mới như VR và AR, cho phép người dùng tạo ra các trò chơi trải nghiệm thực tế ảo và thực tế tăng cường.

### Tổng quan về các thành phần trong unity

1.Scene: Scene là một không gian làm việc trong Unity, cho phép bạn thêm và sắp xếp các đối tượng trong trò chơi. Mỗi trò chơi sẽ được chia thành nhiều Scene khác nhau.

2.GameObject: GameObject là thành phần cơ bản của Unity, được sử dụng để đại diện cho một đối tượng trong trò chơi. Mỗi GameObject sẽ có một hoặc nhiều Component.

3.Component: Component là các thành phần cơ bản của một GameObject, cho phép bạn thêm các tính năng vào GameObject. Ví dụ như Component Rigidbody để xử lý vật lý, Component Mesh Renderer để hiển thị hình ảnh, hoặc Component Script để thêm các kịch bản logic vào GameObject.

4.Prefab: Prefab là một GameObject được lưu trữ để tái sử dụng trong các Scene khác nhau. Nó cho phép bạn tạo ra một đối tượng chỉnh sửa một lần và sử dụng nhiều lần trong toàn bộ trò chơi.

5.Asset: Asset là bất kỳ tài nguyên nào được sử dụng trong trò chơi, bao gồm các hình ảnh, âm thanh, video, văn bản, v.v.

6.Inspector: Inspector là giao diện đồ họa trong Unity, cho phép bạn chỉnh sửa các thuộc tính của một GameObject hoặc Component.

7.Script: Script là một đoạn mã lập trình được sử dụng để thêm các tính năng vào trò chơi. Nó được viết bằng các ngôn ngữ lập trình như C, JavaScript, hoặc Boo.

8.Animation: Animation là một thành phần trong Unity cho phép bạn tạo ra các hoạt hình và điều khiển chuyển động của các đối tượng trong trò chơi.

### Lý do lựa chọn

Unity là một công cụ phát triển game đa nền tảng và được sử dụng rộng rãi trong ngành công nghiệp game. Nó có khả năng hỗ trợ phát triển game bắn súng 3D với các tính năng như đồ họa tuyệt vời, hệ thống vật lý và âm thanh chất lượng cao, hỗ trợ nhiều nền tảng và cộng đồng đông đảo. Ngoài ra, Unity cũng có thể tích hợp dễ dàng với các công cụ khác để phát triển game như Blender, Maya và Photoshop. Tất cả những lý do trên đã khiến Unity trở thành một lựa chọn tốt cho phát triển game bắn súng 3D.[11]

## Công nghệ Navmesh

### Giới thiệu

Navmesh là một công nghệ trong Unity được sử dụng để tạo đường đi tự động cho các đối tượng di chuyển trong một không gian 3D. Điều này giúp cho các đối tượng trong trò chơi có thể di chuyển một cách linh hoạt, tránh được các vật cản và không gây ra va chạm không đáng có. Navmesh sử dụng một bản đồ mô phỏng các khu vực có thể đi qua trong không gian 3D và áp dụng thuật toán để tìm ra đường đi tối ưu cho các đối tượng. Navmesh được sử dụng rộng rãi trong các trò chơi như bắn súng, đua xe, game nhập vai và nhiều thể loại game khác.[12]

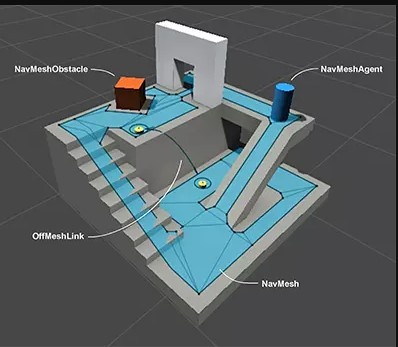
### Một số khái niệm cơ bản

1.NavMesh: Cấu trúc data cho việc tìm đường đi cho đối tượng.

2.NavMesh Agent: Component giúp bạn tạo di chuyển cho đối tượng kết hợp với NavMesh.

3.Off-Mesh Link: Component cho phép di chuyển tắt qua nhau mà không theo quy luật Navigation bình thường là phải qua từng điểm một.

4.NavMesh Obstacle: Component mô tả đối tượng nào bị tránh khi trên đường đi.



Hình 4.1: Hình ảnh minh họa Navmesh [13]

### Lý do lựa chọn

Trong game bắn súng 3D, việc sử dụng Navmesh giúp cho các enemy di chuyển một cách thông minh, đáp ứng được các tình huống đa dạng và phức tạp, giúp cho trải nghiệm chơi game của người chơi trở nên thú vị và chân thật hơn. Do đó, Navmesh là lựa chọn tốt để giải quyết vấn đề di chuyển enemy trong game bắn súng 3D.

Chương này đã liệt kê và giải thích sơ bộ các công nghệ được sử dụng để xây dựng đồ án. Chương tiếp theo sẽ trình bày về chi tiết thiết kế và kết quả sau khi hoàn thành đồ án.

# THỰC NGHIỆM VÀ ĐÁNG GIÁ

Chương 4 đã trình bày về những công nghệ được sử dụng để xây dựng đồ án. Trong chương 5 này sẽ trình bày về thiết kiến trúc, thiết kế lớp, thiết kế dữ liệu, giao diện, kết quả đạt được cùng những ý kiến đóng góp.

## Thiết kế kiến trúc

Kiến trúc OOP (Object-Oriented Programming) là một phương pháp lập trình hướng đối tượng. Nó tập trung vào việc mô hình hóa các đối tượng trong thế giới thực thành các đối tượng trong chương trình. Trong OOP, các đối tượng được xem như là các thực thể, mỗi thực thể có các thuộc tính và phương thức riêng biệt.[14]

### Object

Đối tượng trong OOP bao gồm 2 thành phần chính:

+ Thuộc tính (Attribute): là những thông tin, đặc điểm của đối tượng.

+ Phương thức (Method): là những hành vi mà đối tượng có thể thực hiện.

Ví dụ thực tế về đối tượng là smartphone. Đối tượng này sẽ có:

+ Thuộc tính: màu sắc, bộ nhớ, hệ điều hành. . .

+ Phương thức: gọi điện, chụp ảnh, nhắn tin, ghi âm. . .

### Class

Lớp là sự trừu tượng hóa của đối tượng. Những đối tượng có những đặc tính tương tự nhau sẽ được tập hợp thành một lớp. Lớp cũng sẽ bao gồm 2 thông tin là thuộc tính và phương thức.

Một đối tượng sẽ được xem là một thực thể của lớp..

### Đặc tính cơ bản của OOP

***Tính đóng gói ( Encapsulation)***

Tính đóng gói cho phép che giấu thông tin và những tính chất xử lý bên trong của đối tượng. Các đối tượng khác không thể tác động trực tiếp đến dữ liệu bên trong và làm thay đổi trạng thái của đối tượng mà bắt buộc phải thông qua các phương thức công khai do đối tượng đó cung cấp.

Tính chất này giúp tăng tính bảo mật cho đối tượng và tránh tình trạng dữ liệu bị hư hỏng ngoài ý muốn.

***Tính kế thừa (Inheritance)***

Đây là tính chất được sử dụng khá nhiều. Tính kế thừa cho phép xây dựng một lớp mới (lớp Con), kế thừa và tái sử dụng các thuộc tính, phương thức dựa trên lớp cũ (lớp Cha) đã có trước đó.

Các lớp Con kế thừa toàn bộ thành phần của lớp Cha và không cần phải định nghĩa lại. Lớp Con có thể mở rộng các thành phần kế thừa hoặc bổ sung những thành phần mới.

Ví dụ:

+ Lớp Cha là smartphone, có các thuộc tính: màu sắc, bộ nhớ, hệ điều hành. . .

+ Các lớp Con là iPhone, Samsung, Oppo cũng có các thuộc tính: màu sắc, bộ nhớ, hệ điều hành. . .

***Tính đa hình (Polymorphism)***

Tính đa hình trong lập trình OOP cho phép các đối tượng khác nhau thực thi chức năng giống nhau theo những cách khác nhau.

***Tính trừu tượng (Abstraction)***

Tính trừu tượng được thực hiện bằng cách sử dụng các lớp trừu tượng và phương thức trừu tượng. Một lớp trừu tượng là một lớp mà không thể tạo ra đối tượng của nó trực tiếp, nhưng chỉ có thể sử dụng nó để tạo các lớp con. Một phương thức trừu tượng là một phương thức mà không có thân hàm, chỉ có khai báo, và được kế thừa bởi các lớp con để triển khai.

Tính trừu tượng giúp giảm sự phức tạp của một chương trình bằng cách tập trung vào các khái niệm trừu tượng chính thay vì chi tiết bên trong của đối tượng. Nó cũng giúp cải thiện tính tái sử dụng mã, vì các đối tượng có thể được tạo ra dựa trên các lớp trừu tượng và các phương thức trừu tượng có thể được sử dụng để triển khai các lớp con khác nhau mà không cần phải viết lại mã.

## Thiết kế chi tiết

### Thiết kế giao diện

|  |  |
| --- | --- |
| Thông tin | Thông số |
| Độ phân giải màn hình | HD (720p).  Full HD (1080p). QHD (1440p). |
| Kích thước màn hình | Từ 1366 x 768 trở lên.  Tối ưu với 1920 x 1080. |
| Chế độ màn hình | Toàn màn hình. |
| Font chữ | Tiêu đề và từ khóa: Arial.  Các nút bấm, số lượng coin... : SansBold |

Bảng 5.1.Thông tin thiết kế giao diện

***Thiết kế màn hình***

Phần thiết kế màn hình được dựng bằng phần mềm Photoshop, sử dụng một số hình ảnh đơn giản hoặc trên các nguồn miễn phí trên Internet và các ảnh chụp cảnh từ Unity Editor. Phần thiết kế hình ảnh này không phải là giao diện cuối cùng của trò chơi.

***Thiết kế giao diện chính của trò chơi:***

Thanh thông tin ở dưới màn hình: Có các nút thể hiện cho các chức năng như mở cửa hang súng để mua súng hoặc nâng cấp súng, mở hộp may mắn có thể kiếm được vàng, kiểm tra các phẩn thưởng khi điểm danh, cuối cùng là nút play sẽ vào chọn map chiến đấu và hoàn thành nhiện vụ ở map đó.

Thanh thông tin ở bên trái màn hình: có các nút như myBag sẽ hiển thị ra tất cả các vật phẩm phụ trợ của người chơi đang có và người chơi có thể mua thêm, và nút Ranking sẽ giúp người chơi biết được họ đang ở rank nào (nếu sau này hệ thống phát triển thành game Online).

Các thông tin ở trên cùng: dùng để hiển thị thông tin như tên người chơi, level người chơi, số vàng hay số kim cương mà người chơi sở hữu. Ở giữa sẽ hiển thì model 3d của nhân vật 2 cái nút 2 bên có thể ấn để chuyển trạng thái của nhân vật. Hình 5.1 ở dưới là để mình họa thiết kế giao diện màn hình chính của trò chơi.



Hình 5.1: Thiết kế giao diện màn hình chính

***Thiết kế giao diện điểm danh hàng tuần:***

Gồm có 2 phần chính là tiêu đề và các phần thưởng mà người chơi có thể đạt được khi họ chơi game trong mỗi hình ảnh hiển thị sản phẩm sẽ được chia làm 3 phần: ngày nhận phần thường, số lượng của phần thưởng, cuối cùng là hình ảnh minh họa về phần thưởng.

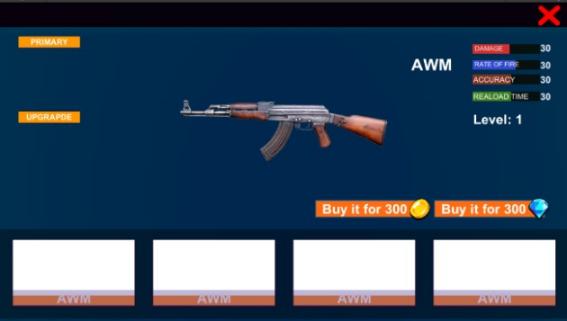
Hình 5.2 là hình minh họa về thiết kế giao diện điểm danh.

Hình 5.2: Thiết kế giao diện điểm danh hàng ngày

***Thiết kế giao diện cửa hàng súng:***

Ở dưới cùng hiển thị tất các súng có trong shop nếu người chơi đã sở hữu súng này thì nó sẽ không còn hình nền đen che đi nữa. Trong phần hiển thị súng nó sẽ chia làm 2 phần tên súng và hình ảnh của súng.

Ở bên trái màn hình sẽ hiển thị các thông tin của súng như: Sức mạnh của súng, độ chính xác và thời gian nạp đạn, level, tên súng. Ở dưới thì hiển thị giá tiền và có nút để người chơi có thể mua súng bằng vàng hoặc bằng kim cương.

Ở bên phải màn hình sẽ hiển thị nút nâng cấp súng.

Hình 5.3: Thiết kế giao diện cửa hàng súng

***Thiết kế giao diện trong màn hình chiến đấu***

Thiết kế giao diện Game play của game sẽ bao gồm 3 phần chính:

Phần trên cùng hiển thị số lượng kẻ địch còn sống, hình ảnh màu vàng bên phải là thể hiện mini map được chiếu từ trên xuống dưới với góc 90 độ.   
 Ở giữa có tâm súng

Hình 5.4: Thiết kế giao diện trong màn hình chiến đấu

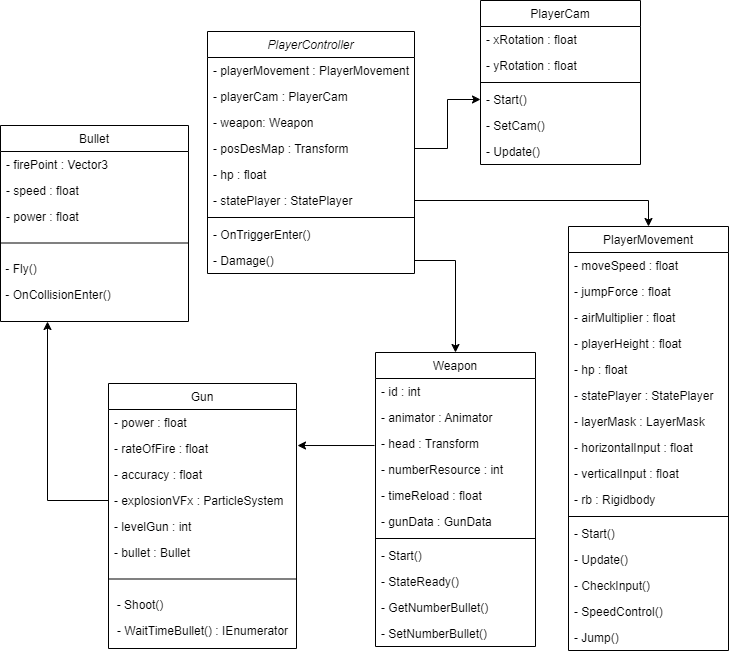
***Thông tin, thiết kế chung khác :***

Các nút cơ bản có 3 trạng thái là bình thường, dưới con trỏ và bị click, dưới đây chỉ đưa ra thiết kế cho nút bấm, tùy vào chức năng khác nhau mà kích thước cũng sẽ khác nhau.

**Hình 5.5:** Ví dụ về nút Lucky Box

Thông báo hiện lên sau khi thực hiện một số hành động ví dụ như mua súng, nâng cấp súng,... nếu thành công hoặc thất bại sẽ có thanh thông báo phản hồi cho người dùng biết.

### Thiết kế lớp

Trong phần này sẽ trình bày về các lớp liên quan đến chức năng điều khiển player trong game bắn súng. Do đấy là một trong nhưng chắc năng rất quan trọng thể hiện độ mượt của game và là phần then chốt giúp một game bắn súng thành công và cũng là lý do khiến rất nhiều người chơi xóa game.  


Hình 5.5: Biểu đồ lớp liên quan đến các chức năng mà player có thể thực hiện khi chiến đấu

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tên | Kiểu dữ liệu | Vai trò |
| Thuộc tính | playerMovement | PlayerMovement | Điều khiển nhân vật di chuyển |
| playerCam | PlayerCam | Điều khiển góc nhìn của nhân vật |
| weapon | Weapon | Vũ khí của nhân vật |
| posDesMap | Transfrom | Lưu lại vị trí của con AI chỉ đường nếu nhân vật đang ở Level 4 |
| hp | float | Lượng máu của nhân vật |
| statePlayer | StatePlayer | Trạng thái của nhân vật |
| Phương thức | OntriggerEnter() | void | Chịu trách nhiệm phát hiện va chạm giữa nhân vật và bình máu |
| Damage() | void | Trừ máu người chơi |

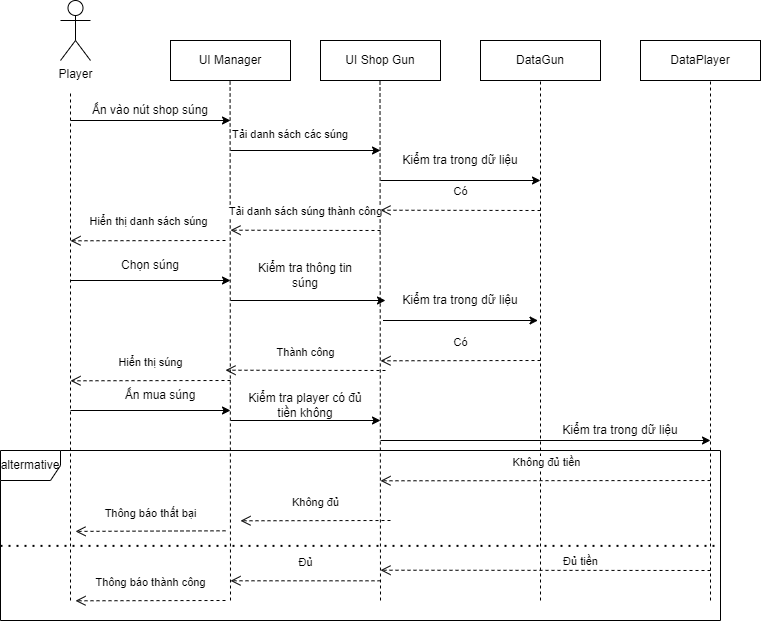
Bảng 5.2.Đặc tả lớp PlayerController

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tên | Kiểu dữ liệu | Vai trò |
| Thuộc tính | moveSpeed | float | Thuộc tính thể hiện tốc độ di chuyển của Player |
| jumpForce | float | lực nhảy của nhân vật |
| airMultiplier | float | Lực cản của không khí khi nhân vật vừa nhảy vừa di chuyển |
| playerHeight | float | Chiều cao của nhân vật dùng để bắn  raycast để biết nhân vật có đang chạm đất hay không |
| layerMask | LayerMask | Mỗi một nhóm Object sẽ có một layer xác định, thuộc tính này sẽ dùng để hạn chế layer cần bắn vào khi sử dụng Raycast giúp tối ưu hóa hiệu năng của game |
| horizontalInput | float | thuộc tính này sẽ nhận giá trị giúp nhân vật khi nó di chuyển theo chiều ngang |
| verticalInput | float | thuộc tính này sẽ nhận giá trị giúp nhân vật khi nó di chuyển theo chiều dọc |
| rb | Rigibody | Rigidbody là một thành phần của đối tượng được sử dụng để mô phỏng hành vi vật lý của đối tượng đó. Rigidbody cho phép đối tượng di chuyển và tương tác với các vật thể khác trong môi trường 3D của trò chơi |
| Phương thức | CheckInput() | void | hàm này sẽ lấy được các giá trị như verticalInput hay horizontalInput |
| SpeedControl() | void | Điều chỉnh tốc độ của nhân vật nếu nó vượt quá tốc độ cho phép sẽ tính toán lại |
| Jump() | void | Hàm sẽ sẽ giúp nhân vật nhảy |
| Update() | void | Hàm này dùng Raycast để kiểm tra xem player có đang ở dưới mặt đất hay không |

Bảng 5.3.Bảng đặc tả PlayerMovement

Bảng 5.4.Đặc tả lớp Weapon

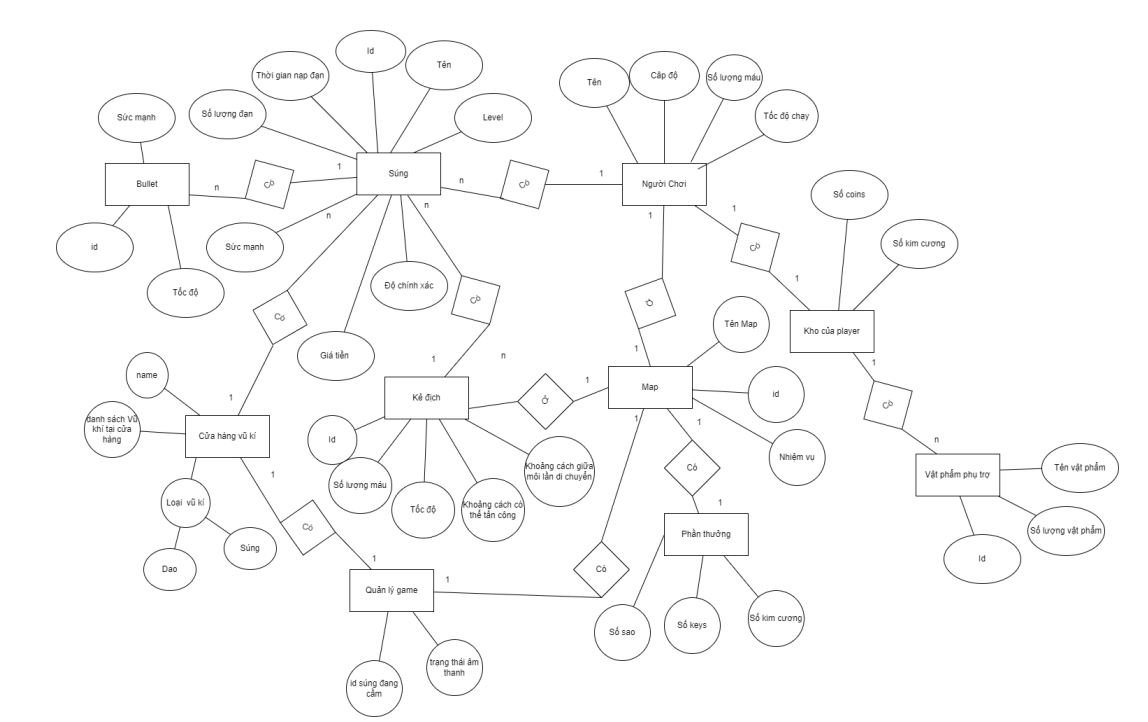
|  |  |
| --- | --- |
|  | |
|  | Tên | | Kiểu dữ liệu | Vai trò |
| Thuộc tính | animator | | Animator | Quản lý các anima-  tion của player |
| headerGun | | Transform | Gắn position đầu  súng phục vụ cho việc nhân vật sẽ bắn ra viên đạn ở đó |
| timeReload | | float | Thời gian nạp đạn  của súng |
| gunData | | GunData | Sẽ lưu lại thông tin  của súng như: Thời gian nạp đạn, sức mạnh của súng, thời gian nạp đạn. |
| id | | int | Dùng để định danh  súng |
| Phương thức | StateReady() | | void | Hàm này kiểm tra  xem súng có đang nạp đạn hay không. |
| GetNumberBullet() | | void | Lấy số lượng đạn. |
| Shoot() | | void | Dùng để bắn ra viên  đạn. |
| Update() | | void | Lấy được vi trí chính  giữa màn hình |

Hình 5.6: Biểu đồ thể hiện trình tự mua súng

Thiết kế cơ sở dữ liệu

Hình 5.12 dưới là biểu đồ thực thể liên kết.

Vì mô hình thiết kế của trò chơi không sử dụng cơ sở dữ liệu mà lưu trữ theo dạng File gọi là ScriptableObjects hay lưu trữ trực tiếp vào đối tượng trên Scene hoặc bằng PlayerPrefs nên biểu đồ thực thể liên kết sẽ có một số điểm khác biệt so với biểu đồ cho hệ thống sử dụng cơ sở dữ liệu.



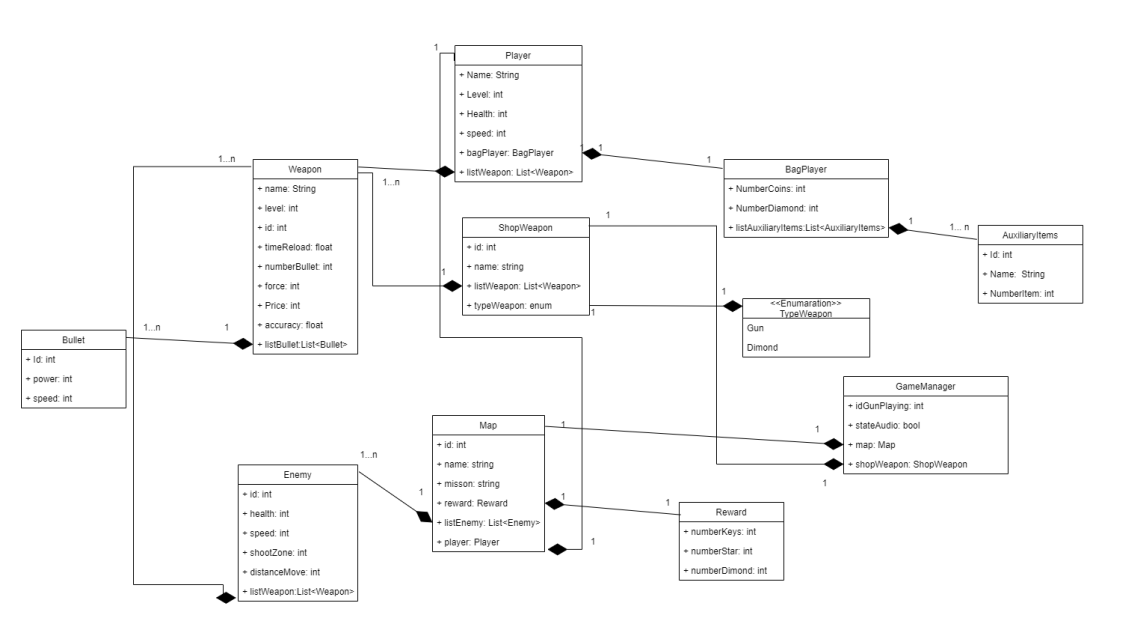
Hình 5.7: Biểu đồ thực thể liên kết

Các thực thể cùng một kiểu được lưu thành các object riêng và có địa chỉ lưu trữ khác nhau, các quan hệ giữa các thực thể sẽ trỏ luôn đến thực thể còn lại thay vì lưu một giá trị của thực thể kia. Vì vậy các thực thể trên đây sẽ không cần sử dụng đến thuộc tính khóa.

Giải thích sơ đồ thực thể liên kết:

* Người chơi: có các thuộc tính như tên, cấp độ, số lượng máu, tốc độ chạy.
* Kho của player: có số lượng coin, số lượng kim cương, vật phẩm phụ trợ.
* Vật phẩm phụ trợ: có id vật phẩm,tên vật phẩm, số lượng vật phẩm.
* Map: id, tên map, nhiệm vụ, người chơi,phần thưởng, kẻ địch, quản lý game.
* Phần thưởng: có số sao, số thìa khóa, số kim cương nhận được mỗi khi người chơi chơi thắng một màn.
* Kẻ địch: có id, số lượng máu, tốc độ, khoảng cách có thể tấn công, khoảng cách có thể di chuyển, súng.
* Súng: có id, tên súng, cấp độ của súng, thời gian nạp đạn, sức mạnh của súng, giá tiền, độ chính xác, đạn.
* Viên Đạn: có id, sức mạnh, tốc độ của đạn.
* Cửa hàng vũ khí: có tên của hàng, danh sách vũ khí, lọa vũ khí.
* Loại vũ khí: Dao hoặc súng.
* Quản lý game: có id súng đang cầm, trạng thái âm thanh.

Từ biểu đồ thực thể liên kết và phân tích phía trên để thiết kế ra các lớp lưu trữ dữ liệu như hình [5.13](#_bookmark92) dưới đây



Hình 5.8: Biểu đồ các lớp dữ liệu

Trong Game Offline sử dụng Unity có 3 hình thức lưu trữ dữ liệu chính:

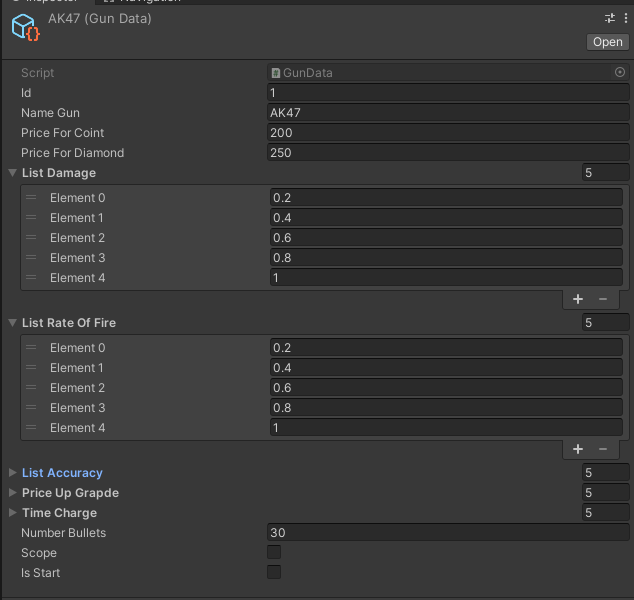
**Sử dụng ScriptableObject:** là một lớp đặc biệt trong Unity được sử dụng để lưu trữ dữ liệu tĩnh giữa các phiên chơi và các tài nguyên trong game. Điều này có nghĩa là dữ liệu của ScriptableObject được lưu trữ trong tệp và có thể được tải lên bất cứ lúc nào trong suốt quá trình chơi game.

**Sử dụng PlayerPrefs:** PlayerPrefs là một lớp trong Unity được sử dụng để lưu trữ các giá trị dữ liệu đơn giản, chẳng hạn như các cài đặt của người dùng, mức độ hoàn thành của người chơi, hoặc bất kỳ loại dữ liệu nào có thể được lưu trữ dưới dạng chuỗi.

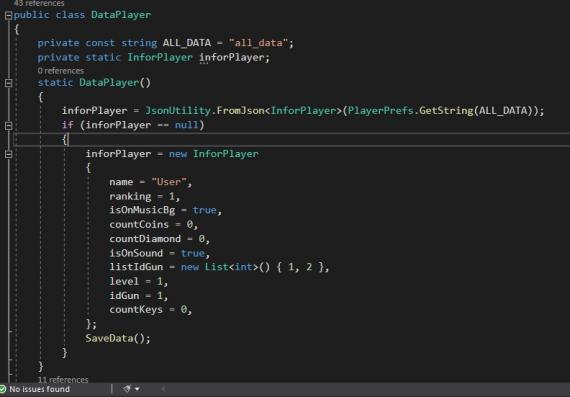
Các giá trị được lưu trữ bởi PlayerPrefs sẽ được lưu trữ trong bộ nhớ của thiết bị và được giữ nguyên giá trị cho đến khi nó được ghi đè hoặc bị xóa.

**Lưu dữ trực tiếp vào object trên Scene:** là dữ liệu động có thể thay đổi trong qua trình chơi game như sẽ bị reset lại về giá trị mặc định khi người chơi thoát ra.

Dưới đây là một số ví du minh họa trong việc lưu trữ liệu trong game sử dụng Unity.

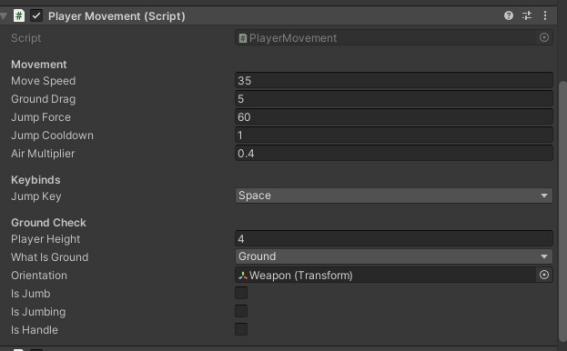
Hình 5.9: Ví dụ về lưu trữ dữ liệu sử dụng ScriptTableObject

Ở hình [5.14](#_bookmark93) trên có thể thấy Data của khẩu súng AK47 đã được lưu trữ dưới dạng ScriptTable, các giá trị này là giá trị tĩnh.



Hình 5.10: Ví dụ về lưu trữ dữ liệu sử dụng PlayerPrefs

Ở trên hình [5.15](#_bookmark94) Các giá trị như level của người chơi, số lượng coin, và số lượng kim cương của người chơi, hay danh sách id của súng của người chơi... các giá trị này đều là giá trị động có thể thay đổi nó chỉ được mất đi khi người chơi xóa game.



Hình 5.11: Ví dụ về lưu trữ dữ liệu vào trực tiếp đối tượng trên Sence

Ta thấy hình [5.16](#_bookmark95) Các giá trị như tốc độ, lực nhảy... của nhân vật sẽ có thể thay đổi trong quá trình chơi game, và nó sẽ bị quay về giá trị mặc định khi người chơi qua màn khác hoặc thoát game vào lại.

## Xây dựng ứng dụng

### Thư viện và công cụ sử dụng

Để phát triển trò chơi này, những công cụ cần có bao gồm:

**IDE lập trình:** đây là công cụ khiên quyết để phát triển không chỉ trò chơi, mà còn cả ứng dụng, web, hay các hệ thống khác. Công cụ này để xây dựng những phần mã nguồn để quản lý, để xử lý các logic vận hành của trò chơi.

**Engine:** là một phần mềm hoặc công cụ lập trình được sử dụng để tạo ra các trò chơi hoặc ứng dụng đa phương tiện. Nó cung cấp một nền tảng để xây dựng và quản lý các tài nguyên, tính năng, hình ảnh, âm thanh và các phần khác của trò chơi. Nó cũng bao gồm các công cụ để thiết kế môi trường trò chơi, tạo ra các chuyển động, tạo ra các kịch bản và kiểm tra các tính năng của trò chơi.

**Thiết kế hình ảnh:** những trò chơi từ tầm trung trở lên sẽ cần nhiều hình ảnh mà những packet đồ họa cũng có giá khá cao và đôi khi chúng cũng không phù hợp ghép với nhau. Vì vậy người phát triển cũng cần tự thiết kế một số hình ảnh riêng cho dự án.

**Thiết kế Model 3D:** cũng giống như lý do phía trên nhưng trong dự án này hầu hết chỉ là chỉnh sửa chứ không thiết kế mô hình mới.

Những công cụ thể được sử dụng sẽ được liệt kê ở bảng [5.5](#_bookmark98) dưới đây, bao gồm có mục đích sử dụng, phiên bản sử dụng và địa chỉ cung cấp công cụ.

|  |  |  |
| --- | --- | --- |
| **Mục đích** | **Công cụ** | **Địa chỉ URL** |
| IDE lập trình | Visual Studio Code | code.visualstudio.com |
| Engine làm game | Unity Editor 2020.3.38f1 | unity.com |
| Thiết kế hình ảnh | Adobe Photoshop CC 2020 | [www.adobe.com](http://www.adobe.com/) |
| Chỉnh sửa 3d | Blender 3.1 | [www.blender.org](http://www.blender.org/) |

Bảng 5.5.Danh sách thư viện và công cụ sử dụng

### Kết quả đạt được

Sau 4 tháng phát triển và tìm hiểu, trò chơi đã được hoàn thành phiên bản đầu tiên, phiên bản này tập trung triển khai những yêu cầu và những chức năng như đã phân tích ở Chương 2. Phiên bản đầu này đã đầy đủ các chức năng chính, những chức năng cơ sở để có thể phát triển cho những phiên bản tiếp theo.

Hình [5.6](#_bookmark100) dưới đây là thông kê về thông tin của project

|  |  |
| --- | --- |
| Thông tin | Thống kê |
| Tổng dung lượng dự án | 3.08 GB |
| Tổng dung lượng mã nguồn | 600KB |
| Tổng số dòng mã nguồn | 8216 dòng |
| Số lớp quản lý | 11 |
| Số lớp UI | 23 |
| Số lớp đối tượng trong trò chơi | 5 |
| Số lớp đối tượng | 5 |
| Số lớp dữ liệu | 5 |
| Dung lượng sản phẩm | 378 MB |

Bảng 5.6.Thông tin project

### Minh họa các chức năng chính

Hình [5.17](#_bookmark102) dưới đây là hình ảnh của một số màn hình chính và có vai trò quan trọng.

Màn hình chính của game



Hình 5.12: Màn hình chính của game

Màn hình menu chính của game: như ở phần thiết kế đã chình bày màn hình này sẽ gồm:

Mở cửa hang súng để mua súng hoặc nâng cấp súng, Mở hộp may mắn có thể kiếm được vàng, Kiểm tra các phẩn thưởng khi điểm danh, cuối cùng là nút play sẽ vào chọn map và hoàn thành nhiện vụ ở map đó.

Thanh thông tin ở bên trái màn hình: có các nút như myBag sẽ hiển thị ra tất cả các vật phẩm phụ trợ của người chơi đang có và người chơi có thể mua thêm, và nút Ranking sẽ giúp người chơi biết được họ đang ở rank nào (nếu sau này hệ thống phát triển thành game Online).

Các thông tin ở trên cùng: dùng để hiển thị thông tin như: tên người chơi, level người chơi, số vàng hay số kim cương mà người chơi sở hữu. Ở giữa sẽ hiển thì model 3d của nhân vật 2 cái nút 2 bên có thể ấn để chuyển trạng thái của nhân vật.

Màn hình chiến đầu của map 1

Hình 5.13: Màn hình chiến đầu của map 1

Hình [5.18](#_bookmark103) là màn hình chiến đầu của map 1 gồm các phần sau:

Ở trên góc phải màn hình hiển thị MiniMap giúp player quan sát được map tốt hơn. bên trái hiển hị số lượng kẻ định còn sống.

Ở giữa hiển thị vũ kí của nhân vật có thể là súng, lừu đạn, tâm của súng...

Dưới cùng bên phải hiển khẩu súng mà nhân vật đang cầm: Bao gồm cả số lượng đạn, tên súng.

Ở các viền của màn hình hiển thị màu đỏ: thể hiện người chơi đang bị tấn công. Phía xa xa có bình máu người chơi có thể chạm vào nó sẽ giúp hồi 500 máu.

Màn hình nâng cấp súng

Hình 5.14: Màn hình nâng cấp súng

Hình [5.19](#_bookmark104) trên là màn hình hiển thị chức năng nâng cấp súng:

Khi ấn UPGRAPDE màn hình sẽ hiển thị một cửa sổ xác nhận ở trên hiển thị level hiện tại của súng. Ở giữa hiển thị level tiếp theo và các thông số khi nâng cấp lên. Cuối cùng ở dưới hiển thị số kim cương mà người chơi phải trả cho lần nâng cấp này.

Màn hình quản lý kho súng của người chơi

Hình 5.15: Màn hình quản lý kho súng của người chơi

Hình [5.20](#_bookmark105) ở trên là màn hình quản lý khó súng của người chơi:

Bên trái hiển thị danh sách súng mà người chơi đang sở hữu, người chơi có thể chọn để thay đổi khẩu súng hiện tại của mình.

Bên phải hiển thị số lượng vật phẩm phụ trợ người chơi có thể dùng.

Màn hình người chơi chọn map chơi

Hình 5.16: Màn hình người chơi chọn map chơi

Hình [5.21](#_bookmark106) là màn hình người chơi chọn map chơi, trong màn hình này sẽ hiển thị 2 thống tin chính là:

Mission: Có nghĩa là nhiệm vụ của người chơi phải hoàn thành trong map này.

Reward: Phần thường mà người chơi có nhận được khi hoàn thành map bao gồm số kim cương và số thìa khóa, phụ thuộc và số sao người chơi có thể nhận được càng nhiều sao thì số phần thưởng sẽ nhận được càng cao.

## Kiểm thử

Kiểm thử về độ nguy hiểm của các kẻ địch có trong hệ thống.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Trường hợp | Quy trình kiểm thử | Kết quả mong đợi | Kết quả |
| 1 | Tư động tuần tra | Chọn map chơi có địa  hình phức tạp nhiều nhà cửa, có nhiều vật cản | Player tranh được các  vật cản, không bị đơ trong qua trình di chuyển | Đạt |
| 2 | Tư động tìm kiếm  người chơi | Điều khiển người  chơi chạy chốn khỏi enemy xem nó có biết tự tìm đến không | Tìm thấy người chơi | Đạt |
| 3 | Tư động tìm tấn công  người chơi | Điều khiển người  chơi đến vùng mà enemy có thể tấn công được | Tấn công người chơi | Đạt |
| 4 | Khi ở map 3 sẽ có các  con enemy tuần tra khi phát hiện người chơi sẽ thông báo cho các con eneny khác | Ngươi chơi bắn hoặc  đến vùng tuần tra của con enemy đó | Thông báo cho các  con enemy khác để bắn player | Đạt |

Bảng 5.7.Kiểm thử về độ nguy hiểm của các kẻ địch có trong hệ thống

Kiểm thử về sức mạnh của từng loại vũ khí trong game

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Trường hợp | Quy trình kiểm thử | Kết quả mong đợi | Kết quả |
| 1 | AK47 có thể bắn với  sức mạnh 200 | Tìm kiếm kẻ thủ với  máu 1000 | Hi vọng bắn 5 phát  chết con enemy | Đạt |
| 2 | AK47 khi nâng cấp  lên cấp 2 với sức mạnh 400 | Tìm kiếm kẻ thủ với  máu 1000 | Hi vọng bắn 3 phát  chết con enemy | Đạt |
| 3 | AMP có ống nhắm có  thể nhìn rõ kẻ địch dù ở khoảng cách xa | Điều khiển người  chơi chạy ra xa con nhân vật và sử dụng súng AMP bật ông ngắm lên | Kì vọng có thể nhìn  thấy rõ kẻ địch và có thể bắn trúng được nó | Đạt |
| 4 | Lừu đạn | Điều khiển player ra  một chỗ có nhiều kẻ định và sử dụng lừu đạn để tấn công | Kỳ vòng là có thể tiêu  diệt 3,4 còn enemy nếu nó đứng gần vùng ném lừu đạn | Đạt |

Bảng 5.8.Kiểm thử về sức mạnh của từng loại vũ khí trong game

Kiểm thử về các chức năng của player và con AI chỉ đường ở map 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Trường hợp | Quy trình kiểm thử | Kết quả mong đợi | Kết quả |
| 1 | Player di chuyển | Ấn các phím mũi tên hoặc các nút A,W,S,D để di chuyển player | Hi vọng có thể di chuyển mượt mà | Đạt |
| 2 | Vừa xoay Player vừa di chuyển | Ấn chuột để xoay nhân vật và ấn các phím để di chuyển | Kết quả mong đợi nhật vật có thể di chuyển đúng hướng | Đạt |
| 3 | AI chỉ đường | Người chơi ấn phím f  nếu ở map 4 | Kết quả mong đợi AI  chỉ đường có thể tìm đúng đường đi đến chỗ vật phẩm cần phá hủy | Đạt |

Bảng 5.9.Kiểm thử về các chức năng của Player

## Triển khai

Trò chơi được xây dựng theo dạng trò chơi cho hệ điều hành Windows, và phát hành trên cửa hàng itch.io theo dạng tải miễn phí.

**Đường dẫn tới trò chơi**: https://luanngo1807.itch.io/falconsurvivor

Cài đặt trò chơi, vào đường dẫn tới địa chỉ tải trò chơi trên cửa hàng itch.io, download một phiên bản. Sau khi tải về sẽ được một file nén dạng rar, giải nén file ra folder, chạy file FalconSurvivor.exe để chạy trò chơi.

Yêu cầu cấu hình hệ thống tối thiểu

|  |  |
| --- | --- |
| Linh kiện | Cấu hình |
| Hệ điều hành | Windows 8 / Windows 10 64 bit |
| Vi xử lý | 2.5 GHz Dual-Core 64-bit CPU |
| Dung lượng RAM | 4GB RAM |
| Đồ họa | 2GB VRAM |
| Bộ nhớ | 400MB dung lượng trống |

Bảng 5.10.Cấu hình tối thiểu

Sau khi phát hành trò chơi và thu thập ý kiến phản hồi, có thể tổng hợp lại thành các ý chính như sau. Về ưu điểm, giao diện trò chơi được thiết kế tốt, dễ sử dụng, màu sắc môi trường ổn, có cốt chuyện rõ ràng . Ngoài những phản hồi về ưu điểm thì trò chơi cũng thu được một số góp ý về những vấn đề như sau:

Kẻ địch bắn tầm xa di chuyển chưa linh hoạt, ở level 3,4 ánh sáng chưa đủ dẫn đến nhìn các kẻ địch khó, lối chơi kém phong phú, thiếu hấp dẫn.

Chương này đã trình bày về thiết kiến trúc, thiết kế lớp, thiết kế dữ liệu, giao diện cùng với đó là kết quả đạt được và phản hồi của người chơi. Chương sau sẽ về những khó khăn gặp phải khi xây dựng đồ án.

# CÁC GIẢI PHÁP VÀ ĐÓNG GÓP NỔI BẬT

Trong chương trước, ta đã thiết kế các thành phần, thiết kế giao diện và kết quả đạt được qua quá trình phát triển. Trong chương này, sẽ trình bày về những khó khăn gặp phải và cách giải quyết những khó khăn đó.

## Hiệu năng của trò chơi

### Vấn đề gặp phải

Trong quá trình phát triển game gặp phải vấn đề hiệu năng phần lớn sẽ liên quan đến hiệu ứng đồ họa và số lượng các thực thể tồn tại trên một cảnh.

Đối với cảnh menu trong cảnh này sẽ có một chức năng được sử dụng nhiều lần đó là mở rương may mắn, khi người chơi muốn kiếm được vàng, thì họ phải chiến đấu ở các màn chơi để lấy thìa rồi mở rương vàng.

Trong Hình [6.1](#_bookmark116) minh họa khi mở rương có quá nhiều vàng.

Hình 6.1: Hình ảnh mở rương sinh quá nhiều vàng

Vấn đề gặp phải ở đây là mỗi lần mở rương đó sẽ sinh ra nhiều đối tượng vàng giống nhau và chúng được di chuyển đến đích sau đó sẽ bị xóa đi, rất nhiều lần như vậy sẽ gặp phải vấn đề phân mảnh bộ nhớ và làm giảm hiệu suất của game



Hình 6.2: Hình ảnh hiệu ứng thủng tường, hiệu ứng bắn súng

Ở trên Hình [6.2](#_bookmark117) ta thấy các nhân vật khi bắn nhau sẽ phải sinh ra rất nhiều viên đạn, và các hiệu ứng như thủng tường, hiệu ứng máu bắn ra, ... Chúng có đặc điểm chung là giống nhau và khi sinh ra sẽ bị xóa đi. Tuy nhiên, điều này có thể dẫn tới phân mảnh bộ nhớ và tăng đáng kể hiệu năng. Việc quản lý số lượng đối tượng cần được cân nhắc kỹ để tránh tình trạng bộ nhớ bị phân mảnh và giảm hiệu năng của game. Ngoài ra, các hiệu ứng như cháy nổ và hiệu ứng ánh sáng cũng là những yếu tố tốn nhiều hiệu năng, cần được tối ưu để đảm bảo game hoạt động mượt mà và trải nghiệm chơi game tốt hơn cho người chơi.

### Quá trình giải quyết vấn đề

Để giải quyết vấn đề về hiệu năng bộ nhớ bị phân mảnh khi sinh ra các đối tượng giống nhau trong game bắn súng 3D, tôi sử dụng kỹ thuật pooling object.

Pooling object là kỹ thuật cho phép tái sử dụng các đối tượng đã được tạo ra trước đó thay vì tạo ra các đối tượng mới khi cần. Điều này giúp giảm thiểu việc tạo và xóa các đối tượng liên tục trong game, giảm thiểu tình trạng bộ nhớ bị phân mảnh và tăng hiệu năng của game.

Với vấn đề trong game bắn súng 3D ,tôi sử dụng pooling object để quản lý số lượng đạn, hiệu ứng thủng tường, hiệu ứng máu, và các đối tượng giống nhau khác. Khi các đối tượng này cần được sử dụng, tôi sẽ lấy chúng từ pool đối tượng đã được tạo sẵn, thay vì tạo mới. Khi không còn sử dụng, chúng tôi sẽ trả lại các đối tượng này vào pool để sử dụng lại trong tương lai. Hiểu đơn giản là thay vì xóa hẳn object thì kỹ thuật nay sử dụng hàng đợi để tắt bật [object.[15]](#_bookmark144)

### Kết quả đạt được

Nhờ sử dụng kỹ thuật pooling object đã giảm thiểu việc tạo mới đối tượng và tăng hiệu suất của game. Việc giảm thiểu số lượng đối tượng tạo ra cũng giúp giảm thiểu tình trạng bộ nhớ bị phân mảnh, giúp game chạy mượt mà hơn.

## Cách di chuyển của Enemy

### Vần đề gặp phải

Giai đoạn đầu đồ án đã gặp vấn đề liên quan đến di chuyển của con enemy trong môi trường có nhiều vật cản. Cụ thể, các con enemy di chuyển không chính xác và có thể xuyên qua các vật thể trong game như tường hay đồ vật. Điều này dẫn đến tình trạng game không chính xác và giảm trải nghiệm của người chơi.

Hình 6.3: Ví dụ về cách di chuyển cũ của enemy

Trong ví dụ trên hình [6.3,](#_bookmark122) đối tượng Enemy được di chuyển từ vị trí hiện tại của nó đến vị trí của đối tượng mục tiêu (target) với tốc độ nhất định. Ta sử dụng hàm MoveTowards được sử dụng để cập nhật vị trí của đối tượng theo thời gian, được gọi trong hàm Update() của EnemeyController. Tuy nhiên khi di chuyển kiểu này enemy sẽ không thể tránh được các vật cản ở đường hay các tòa nhà.

### Quá trình giải quyết vấn đề

Để giải quyết vấn đề này,tôi đã sử dụng Navmesh trong Unity. Navmesh là một công cụ mạnh mẽ trong Unity để xử lý vấn đề di chuyển của các đối tượng trong trò chơi. Navmesh giúp tạo ra một mạng lưới bề mặt trên màn hình, gồm các vùng di chuyển được xác định rõ ràng, để các đối tượng có thể di chuyển trên đó một cách thông minh và tự [động.[12]](#_bookmark141)

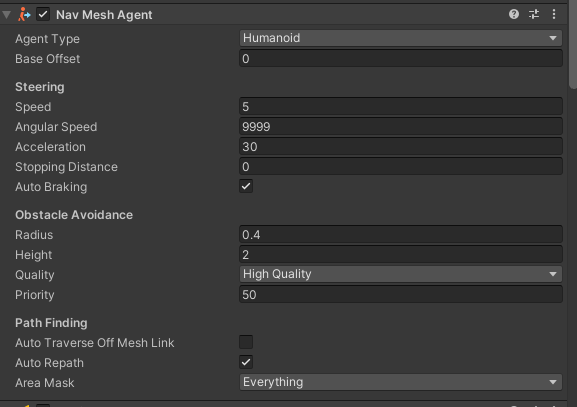


Hình 6.4: Hình minh họa map 1 đã sử dụng Navmesh

Nhìn hình [6.4](#_bookmark124) trên ta thấy có 2 vùng phân biệt là màu trắng và màu xanh dương. Vùng màu trắng gồm các object không thể di chuyển được ví dụ như vật cản, tòa nhà, ô tô hỏng. Vùng màu xanh gồm là vùng mà có thể di chuyển được.

Tiếp theo để enemy có thể di chuyển được ta gắn NavmeshAgent vào vào nó, cùng với các thông số điều chỉnh như tốc độ di chuyển, khoảng cách tối thiểu đến các vật thể khác, và hướng đi của đối tượng.

Như trong hình [6.5](#_bookmark125) minh họa gắn NavmeshAgent vào đối tượng.



Hình 6.5: Hình minh họa gắn NavmeshAgent vào cho đối tượng

Tiếp theo chúng ta có thể sử dụng code để di chuyển nhận vật để đến nơi mình muốn và nó sẽ tìm được đường đi ngắn nhất.

### Kết quả đạt được

Kết quả sau khi áp dụng Navmesh cho con enemy là các con enemy di chuyển thông minh và không xuyên qua các vật thể trong game. Việc sử dụng Navmesh đã cải thiện trải nghiệm chơi game của người chơi và là một trong những đóng góp nổi bật của tôi trong quá trình phát triển game.

Trên đây là phần trình bày về những khó khăn đã gặp phải trong quá trình xây dựng đồ án và quá trình nghiên cứu để tìm ra giải pháp và kết quả đạt được sau cùng. Trong chương tiếp theo, cũng là chương cuối cùng sẽ là lời kết luận cho quá trình xây dựng đồ án.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong chương trước đã trình bày về những vấn đề gặp phải khi xây dựng đồ án và quá trình giải quyết vấn đề. Chương cuối này sẽ là lời kết luận cho đồ án, những gì đã làm được và định hướng.

## Kết luận

Trong suốt quá trình thực hiện đồ án game bắn súng 3D. Nhờ vào những định hướng và chỉ dẫn của Ths. Nguyễn Tuấn Tú, đồ án đã đạt được những thành tựu đáng kể. Em đã thiết kế và phát triển một game bắn súng 3D với đồ họa và hiệu ứng âm thanh sống động, cùng với lối chơi hấp dẫn và đa dạng.

Đầu tiên, em đã học được cách tạo một sản phẩm game đầy đủ với đầy đủ các tính năng. Qua đó, tôi đã hiểu được quá trình phát triển game bao gồm từ khâu lên ý tưởng, thiết kế đến lập trình và kiểm thử.

Thứ hai, em đã học được cách giải quyết vấn đề trong quá trình phát triển game. Em đã đối mặt với nhiều thách thức như vấn đề về chuyển động của con enemy, tuy nhiên, em đã áp dụng kỹ thuật Navmesh để giải quyết vấn đề này.

Ngoài những thành tựu kể trên đồ án vẫn còn một số vấn đề chưa thể giải quyết hoàn toàn được, đặc biệt là về khía cạnh tối ưu hóa hiệu suất của game.

Để tối ưu hóa hiệu suất, một số phương pháp đã được áp dụng như việc tối ưu hóa hệ thống vật lý và cải thiện cách xử lý các sự kiện trong game. Tuy nhiên, kết quả vẫn còn chưa thực sự tối ưu và cần phải tiếp tục nghiên cứu và cải thiện trong tương lai.

So sánh với các sản phẩm tương tự, đồ án của em đạt được một số kết quả đáng kể. Trước tiên, trò chơi Falcon:Survivor mà em phát triển có đồ họa đẹp mắt và âm thanh sống động. Bên cạnh đó, nó cũng có tính năng đa dạng với nhiều bản đồ và vũ khí khác nhau. Đặc biệt, em đã giải quyết được vấn đề về chuyển động không chính xác của con enemy bằng cách sử dụng Navmesh trong Unity.

## Hướng phát triển

Để hoàn thiện sản phẩm game bắn súng 3D của mình, em cần tiếp tục phát triển các chức năng và nhiệm vụ còn thiếu. Đầu tiên, em cần tăng cường tính năng game online để người chơi có thể chơi cùng với nhau thông qua mạng Internet. Điều này sẽ giúp tăng tính thú vị của game và thu hút nhiều người chơi hơn.

Ngoài ra, em cũng muốn tăng tính đồng đội trong game bằng cách tạo ra các chế độ chơi đội hình. Điều này cho phép người chơi có thể tạo ra đội hình của riêng mình và chơi cùng với bạn bè hoặc người chơi khác. Đây là một tính năng hấp dẫn cho những người chơi yêu thích sự cạnh tranh và muốn thể hiện khả năng chiến đấu của mình trong trận đấu.

Ngoài ra, em cũng sẽ phát triển các tính năng mới để nâng cao trải nghiệm người chơi. Ví dụ, em có thể tạo ra các bản đồ mới, các chế độ chơi mới, hoặc tăng cường hệ thống vũ khí, phương tiện di chuyển, hệ thống phòng thủ... để đưa trò chơi của mình lên một tầm cao mới và hấp dẫn hơn với người chơi.

Tóm lại, để phát triển sản phẩm game bắn súng 3D của mình, em sẽ tập trung vào việc phát triển tính năng game online, tính đồng đội và các tính năng mới khác để tăng cường trải nghiệm người chơi. Đây sẽ là những điểm mạnh và giúp sản phẩm của tôi trở nên nổi bật và thu hút được nhiều người chơi hơn.

Trên đây là phần trình bày về kết luận những gì làm được, chưa làm được, những điều đã học hỏi được qua quá trình xây dựng đồ án. Cùng với đó là những hướng phát triển cho đồ án trong tương lai. Đây cũng là chương cuối cùng và là sự tổng kết cho báo cáo và toàn bộ đồ án.

# TÀI LIỆU THAM KHẢO

1. Kaavya Karthikeyan, *The evolution of first person shooter (fps) games*. [On- line]. Available: [https : / / www. gameopedia . com / evolution -](https://www.gameopedia.com/evolution-of-first-person-shooter-fps-games/) [of-first-person-shooter-fps-games/](https://www.gameopedia.com/evolution-of-first-person-shooter-fps-games/) .
2. Colin Campbell, *How to write a video game story*. [Online]. Available: [https://www.polygon.com/features/2019/1/10/18165611/](https://www.polygon.com/features/2019/1/10/18165611/how-to-write-a-video-game-story-narrative-building-tips) [how-to-write-a-video-game-story-narrative-building-](https://www.polygon.com/features/2019/1/10/18165611/how-to-write-a-video-game-story-narrative-building-tips) [tips](https://www.polygon.com/features/2019/1/10/18165611/how-to-write-a-video-game-story-narrative-building-tips) .
3. MasterClass, *How to write compelling character backstories*. [Online]. Avail-

able: [https://www.masterclass.com/articles/how- to-](https://www.masterclass.com/articles/how-to-write-compelling-character-backstories) [write-compelling-character-backstories](https://www.masterclass.com/articles/how-to-write-compelling-character-backstories) .

1. S. Rogers, *The Guide to Great Video Game Design*. Elsevier, 2007.
2. Michel Sabbagh, *The important differences between first-person and third- person games*. [Online]. Available: [https://www.gamedeveloper.](https://www.gamedeveloper.com/design/the-important-differences-between-first-person-and-third-person-games) [com/design/the-important-differences-between-first-](https://www.gamedeveloper.com/design/the-important-differences-between-first-person-and-third-person-games) [person-and-third-person-games](https://www.gamedeveloper.com/design/the-important-differences-between-first-person-and-third-person-games) .
3. OOP, *The best battle royale games*. [Online]. Available: [https://www.](https://www.pcgamer.com/battle-royale-games/) [pcgamer.com/battle-royale-games/](https://www.pcgamer.com/battle-royale-games/) .
4. TPS, *Third-person shooters*. [Online]. Available: [https://news.xbox.](https://news.xbox.com/en-us/2015/10/09/games-know-your-genres-third-person-shooters/) [com/en-us/2015/10/09/games-know-your-genres-third-](https://news.xbox.com/en-us/2015/10/09/games-know-your-genres-third-person-shooters/) [person-shooters/](https://news.xbox.com/en-us/2015/10/09/games-know-your-genres-third-person-shooters/) .
5. Colin Campbell, *The complete guide to video game genres: From scrollers, shooters, to sports*. [Online]. Available: [https://www.gamedesigning.](https://www.gamedesigning.org/gaming/video-game-genres) [org/gaming/video-game-genres](https://www.gamedesigning.org/gaming/video-game-genres)
6. Microsoft, *C documentation*. [Online]. Available: [https://docs.micr](https://docs.microsoft.com/en-us/dotnet/csharp/)osoft. [com/en-us/dotnet/csharp/](https://docs.microsoft.com/en-us/dotnet/csharp/) .
7. Mono, *Mono documentation*. [Online]. Available: [http://docs.go-](http://docs.go-mono.com/) [mono.com/](http://docs.go-mono.com/) .
8. Unity, *Unity documentation*. [Online]. Available: [https://docs.unity3d.](https://docs.unity3d.com/Manual/index.html) [com/Manual/index.html](https://docs.unity3d.com/Manual/index.html) .
9. NavMesh, *Navmesh*. [Online]. Available: [https://docs.unity3d.](https://docs.unity3d.com/Manual/nav-BuildingNavMesh.html) [com/Manual/nav-BuildingNavMesh.html](https://docs.unity3d.com/Manual/nav-BuildingNavMesh.html)
10. Navigation, *Navigation*. [Online]. Available: [https://docs.unity3d.](https://docs.unity3d.com/Manual/Navigation.html) [com/Manual/Navigation.html](https://docs.unity3d.com/Manual/Navigation.html)
11. OOP, *Object-oriented programming (c)*. [Online]. Available: [https://](https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop) [learn.microsoft.com/en-us/dotnet/csharp/fundament](https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop)als/ [tutorials/oop](https://learn.microsoft.com/en-us/dotnet/csharp/fundamentals/tutorials/oop) .
12. OOP, *Introduction to object pooling*. [Online]. Available: [https://lear](https://learn.unity.com/tutorial/introduction-to-object-pooling)n. [unity.com/tutorial/introduction- to- object- pooling](https://learn.unity.com/tutorial/introduction-to-object-pooling).