**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**ĐỒ ÁN THỰC TẬP CƠ SỞ**

**Đề tài : Lập trình game Unity 2D sử dụng C#  
Shooting Monster**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn** | **: Bùi Văn Kiên** |
| **Họ và tên sinh viên** | **: Nguyễn Đình Tú** |
| **Mã sinh viên** | **: B22DCCN746** |
| **Lớp** | **: D22CQCN02-B** |

***Hà Nội – 2025***

Mục Lục

**[Mục Lục 1](#_Toc6853)**

**[Danh Mục Hình Ảnh 2](#_Toc21687)**

**[Danh Mục Bảng 3](#_Toc1410)**

**[LỜI MỞ ĐẦU](#_Toc12975)** [4](#_Toc12975)

**[CHƯƠNG I: GIỚI THIỆU ĐỒ ÁN VÀ CÁC CÔNG NGHỆ SỬ DỤNG](#_Toc6696)** [4](#_Toc6696)

**[1.1. Giới thiệu chung về đồ án 5](#_Toc19883)**

**[1.2. Các công nghệ sử dụng trong đồ án 5](#_Toc4235)**

[1.2.1. Unity – Nền tảng phát triển game 5](#_Toc7714)

[1.2.2. C# – Ngôn ngữ lập trình 5](#_Toc26476)

[1.2.3. GitHub – Quản lý mã nguồn 6](#_Toc5968)

[1.2.4. Visual Studio – Môi trường phát triển 6](#_Toc32401)

[1.2.5. Unity Asset Store – Tài nguyên hỗ trợ 6](#_Toc21877)

**[1.3. Mục tiêu của đồ án 6](#_Toc5761)**

**[CHƯƠNG II: THIẾT KẾ CỐT TRUYỆN VÀ LÊN Ý TƯỞNG CHO TRÒ CHƠI](#_Toc32542)** [7](#_Toc32542)

**[2.1. Lên ý tưởng chung. 7](#_Toc2871)**

[2.1.1. Cốt truyện 7](#_Toc31122)

[2.1.2. Thể loại 7](#_Toc2370)

[2.1.3. Số lượng người chơi 8](#_Toc20957)

[2.1.4. Logic cơ bản của trò chơi 8](#_Toc14648)

[2.1.5. Tính năng (Features) 8](#_Toc1814)

**[2.2. Thiết kế và lên ý tưởng cho các đối tượng trong trò chơi 9](#_Toc31584)**

[2.2.1. Bảng người chơi 9](#_Toc1989)

[2.2.2. Bảng quái vật 9](#_Toc24059)

[2.2.3. Bảng vũ khí 10](#_Toc924)

[2.2.4. Màn chơi và chế độ 11](#_Toc2716)

[2.2.5. Lối chơi tổng quan 11](#_Toc31715)

**[CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ TRÒ CHƠI](#_Toc2920)** [11](#_Toc2920)

**[3.1. Phân tích hệ thống 11](#_Toc4158)**

[3.1.1. Mô tả dự án 11](#_Toc3898)

[3.1.2. Mô tả hệ thống bằng UML 12](#_Toc12990)

[3.1.3. Kịch bản chuẩn và ngoại lệ 13](#_Toc19451)

[3.1.4. Usecase tổng quan 15](#_Toc1369)

**[3.2. Thiết kế hệ thống trò chơi 16](#_Toc22706)**

[3.2.1. Sơ đồ lớp thực thể hệ thống 16](#_Toc29755)

[3.2.2. Sơ đồ lớp module tấn công kẻ địch 18](#_Toc25878)

**[Chương IV: Kết luận và hướng phát triển 18](#_Toc31633)**

**[4.1. Kết luận 18](#_Toc8872)**

**[4.2. Hướng phát triển 18](#_Toc15550)**

Danh Mục Hình Ảnh

Hình 1: Usecase tổng quan 15

Hình 2: Biểu đồ lớp thực thể của trò chơi 16

#### Hình 3: Sơ đồ lớp module tấn công kẻ địch 17

Danh Mục Bảng

Bảng 1: Các tính năng 8

Bảng 2: Bảng nhân vật 8

Bảng 3: Bảng quái vật 9

Bảng 4: Bảng vũ khí 9

Bảng 5: Từ khóa 12

Bảng 6: Kịch bản Start Game 13

Bảng 7: Kịch bản tấn công quái vật 13

Bảng 8: Quái tấn công người chơi 14

Bảng 9: Mô tả tổng quan các lớp 16

## **LỜI MỞ ĐẦU**

Trong thời đại công nghệ số hiện nay, ngành công nghiệp game đang phát triển mạnh mẽ, đặc biệt là trên nền tảng di động. Việc phát triển một trò chơi không chỉ giúp nâng cao kỹ năng lập trình mà còn rèn luyện tư duy logic, khả năng thiết kế và quản lý dự án. Đồ án "Shooting Monster" được thực hiện nhằm mục tiêu áp dụng kiến thức đã học vào thực tế, tạo ra một sản phẩm game 2D sử dụng Unity và ngôn ngữ lập trình C#.

## **CHƯƠNG I: GIỚI THIỆU ĐỒ ÁN VÀ CÁC CÔNG NGHỆ SỬ DỤNG**

### Giới thiệu chung về đồ án

Trong thời đại công nghệ số phát triển mạnh mẽ, ngành công nghiệp trò chơi điện tử đang trở thành một lĩnh vực đầy tiềm năng và hấp dẫn, không chỉ về mặt giải trí mà còn mở ra nhiều cơ hội nghề nghiệp cho sinh viên ngành Công nghệ Thông tin. Với mong muốn làm quen và tiếp cận quy trình phát triển game thực tế, đồ án "Shooting Monster" đã được thực hiện như một bài tập lớn mang tính ứng dụng cao.

*Shooting Monster* là một game 2D hành động bắn súng góc nhìn ngang (side-scrolling shooter), được phát triển dưới dạng chơi đơn (single-player). Trong game, người chơi sẽ điều khiển một nhân vật, sử dụng súng để tiêu diệt các quái vật đang tràn đến từ nhiều hướng khác nhau. Trò chơi mang phong cách hành động nhanh, có cơ chế điều khiển đơn giản nhưng đòi hỏi phản xạ tốt, kết hợp với hiệu ứng âm thanh và đồ họa sinh động.

Thông qua dự án này, sinh viên được thực hành toàn bộ quy trình phát triển phần mềm từ thiết kế, lập trình, kiểm thử đến tối ưu hóa và hoàn thiện sản phẩm. Đồng thời, đồ án cũng là cơ hội để áp dụng kiến thức lập trình C#, sử dụng Unity và các công cụ phát triển hiện đại trong môi trường làm việc thực tế.

### Các công nghệ sử dụng trong đồ án

Trò chơi Shooting Monster đã sử dụng các công cụ và công nghệ sau:

#### 1.2.1. Unity – Nền tảng phát triển game

Unity là một trong những nền tảng phát triển game phổ biến nhất hiện nay, đặc biệt phù hợp với các dự án game 2D hoặc 3D quy mô nhỏ đến trung bình. Với Unity, người phát triển có thể xây dựng trò chơi một cách trực quan nhờ hệ thống kéo-thả, đồng thời có thể kiểm soát logic hoạt động thông qua mã nguồn C#.

Trong *Shooting Monster*, Unity được sử dụng để:

- Thiết kế bản đồ, tạo nhân vật, hiệu ứng kỹ năng và quái vật.

- Xử lý va chạm (collision), vật lý đơn giản và di chuyển.

- Quản lý màn chơi (level), hệ thống điểm số, giao diện người dùng (UI).

- Cài đặt hiệu ứng âm thanh, nhạc nền và hiệu ứng hình ảnh (particles).

#### 1.2.2. C# – Ngôn ngữ lập trình

C# là ngôn ngữ chính được sử dụng trong Unity để phát triển logic cho game. Đây là ngôn ngữ lập trình hướng đối tượng hiện đại, dễ học và rất phù hợp với các ứng dụng tương tác như game.

Trong trò chơi, C# được sử dụng để lập trình:

- Di chuyển và điều khiển nhân vật.

- Tạo và điều khiển hành vi của quái vật (AI đơn giản).

- Hệ thống bắn súng và xử lý đạn.

- Tính toán điểm số, quản lý máu và xử lý khi nhân vật hoặc quái vật bị tiêu diệt.

- Quản lý các màn chơi, điều kiện chiến thắng hoặc thua cuộc.

#### 1.2.3. GitHub – Quản lý mã nguồn

GitHub là một nền tảng quản lý mã nguồn thông qua hệ thống Git, giúp theo dõi lịch sử phát triển, kiểm soát phiên bản và dễ dàng làm việc nhóm (nếu cần). Việc sử dụng GitHub trong đồ án giúp:

- Sao lưu mã nguồn một cách an toàn.

- Dễ dàng quay lại phiên bản ổn định khi phát sinh lỗi.

- Ghi lại lịch sử thay đổi và tiến trình thực hiện đồ án.

Link repository:  <https://github.com/tsunoyaa/ShootingMonter.git>

#### 1.2.4. Visual Studio – Môi trường phát triển

Visual Studio là môi trường phát triển tích hợp (IDE) hỗ trợ tốt cho C# và Unity. Đây là công cụ chính để viết và quản lý mã nguồn trong quá trình phát triển game.

Tính năng nổi bật:

- Gợi ý cú pháp thông minh (IntelliSense).

- Gỡ lỗi (debug) trực tiếp trong khi chạy game.

- Tích hợp quản lý project Unity liền mạch.

#### 1.2.5. Unity Asset Store – Tài nguyên hỗ trợ

Unity Asset Store là kho tài nguyên phong phú, cung cấp sẵn các asset như hình ảnh nhân vật, hiệu ứng, nhạc nền, UI, prefab... giúp rút ngắn thời gian phát triển.

Trong đồ án này sử dụng các tài nguyên:

- Sprite 2D cho nhân vật và quái vật.

- Âm thanh nền và hiệu ứng tiếng súng.

- Các hiệu ứng ánh sáng, vụ nổ, kỹ năng (nếu có).

- Giao diện người dùng (UI) như bảng điểm, menu, thanh máu...

### Mục tiêu của đồ án

Xây dựng một game 2D hành động dạng chơi đơn có tính hoàn chỉnh và khả năng mở rộng.

Hiểu và vận dụng quy trình phát triển game bằng Unity và ngôn ngữ C#.

Tạo ra hệ thống nhân vật, quái vật, kỹ năng, điểm số, giao diện tương tác và kết thúc màn chơi.

Làm quen với kỹ thuật xử lý va chạm, điều khiển nhân vật, và tạo AI đơn giản.

Áp dụng kỹ năng lập trình và tư duy logic để giải quyết các bài toán thực tế trong phát triển game.

## **CHƯƠNG II: THIẾT KẾ CỐT TRUYỆN VÀ LÊN Ý TƯỞNG CHO TRÒ CHƠI**

Trong chương này sẽ giới thiệu và lên ý tưởng cho trò chơi, xác định rõ ràng các logic, chức năng của trò chơi và thiết kế các kỹ năng và sức mạnh cho các nhân vật hiện diện trong trò chơi

### 2.1. Lên ý tưởng chung.

#### 2.1.1. Cốt truyện

Trong một thế giới hậu tận thế, các chiều không gian va chạm khiến hàng loạt cánh cổng không gian mở ra trên Trái Đất. Qua các cánh cổng đó, vô số sinh vật dị dạng và quái vật tràn vào nhân giới. Thành phố từng yên bình giờ đây trở thành vùng đất hoang tàn.

Người chơi vào vai một trong bốn người sống sót cuối cùng: **Jack**, **Peter**, **Lisa** hoặc **Rose**. Không có quân đội, không có viện trợ, họ phải tự mình chiến đấu để sinh tồn. Với mỗi con quái vật bị tiêu diệt, nhân vật sẽ tích lũy kinh nghiệm, nâng cấp bản thân và trang bị thêm vũ khí nhằm kéo dài sự sống và tìm kiếm hi vọng cuối cùng để đóng lại các cánh cổng không gian.

#### 2.1.2. Thể loại

Trò chơi thuộc thể loại bắn súng 2D góc nhìn từ trên xuống (Top-down shooter) kết hợp yếu tố sinh tồn (Survival) và hành động (Action).

#### 2.1.3. Số lượng người chơi

Chế độ chơi: Chơi đơn (Single-player).

Người chơi điều khiển 1 nhân vật trong suốt màn chơi để tiêu diệt quái vật và sống sót lâu nhất có thể.

Trong tương lai, trò chơi có thể mở rộng thêm các chế độ co-op hoặc PvE nhiều người.

#### 2.1.4. Logic cơ bản của trò chơi

**Khởi đầu:**

Người chơi chọn 1 trong 4 nhân vật (Jack, Peter, Lisa, Rose).

Bắt đầu ở một vị trí cố định trên bản đồ.

**Giao chiến:**

Quái vật xuất hiện ngẫu nhiên trên bản đồ với tốc độ cố định.

Người chơi sử dụng súng để tiêu diệt chúng.

Sau mỗi lần tiêu diệt, người chơi tích lũy điểm và kinh nghiệm.

**Lên cấp và trang bị:**

Khi đạt đủ điểm kinh nghiệm, người chơi **lên cấp**.

Mỗi cấp cho phép chọn thêm một vũ khí mới trong danh sách, tối đa mang theo 15 súng.

**Kết thúc:**

Trò chơi kết thúc khi nhân vật bị hạ gục (HP = 0).

Điểm số được lưu lại dựa trên thời gian sống sót và số quái vật tiêu diệt được.

#### 2.1.5. Tính năng (Features)

**Bảng 1: Các tính năng**

|  |  |
| --- | --- |
| **Tính năng** | **Mô tả** |
| **Hệ thống lên cấp** | Người chơi nhận điểm kinh nghiệm và lên cấp khi tiêu diệt đủ quái. |
| **Chọn vũ khí khi lên cấp** | Khi lên cấp, người chơi được chọn vũ khí mới để mang theo. |
| **Hiệu ứng bắn** | Hiệu ứng ánh sáng và âm thanh khi bắn tạo cảm giác chân thật. |
| **Chế độ chơi** | Gồm 2 chế độ: Normal (demo) và Endless (chưa hoàn thiện). |
| **Giao diện HUD trực quan** | Hiển thị HP, điểm số, số lượng súng, cấp độ… |
| **Random Spawn Quái vật** | Vị trí sinh ra quái vật ngẫu nhiên, tạo sự bất ngờ và thử thách. |
| **Hệ thống tính điểm** | Người chơi nhận điểm kinh nghiệm và lên cấp khi tiêu diệt đủ quái. |

### 2.2. Thiết kế và lên ý tưởng cho các đối tượng trong trò chơi

#### 2.2.1. Bảng người chơi

Trò chơi cung cấp 4 nhân vật chính để người chơi lựa chọn. Mỗi nhân vật có chỉ số máu khác nhau, nhưng sát thương gây ra là như nhau, không có kỹ năng đặc biệt. Tốc độ di chuyển đồng đều.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên nhân vật** | **Máu (HP)** | **Sát thương cơ bản** | **Số lượng súng mang tối đa** | **Tốc độ di chuyển** |
| Jack | 20 | 5 | 15 | 5 |
| Peter | 22 | 5 | 15 | 5 |
| Lisa | 24 | 5 | 15 | 5 |
| Rose | 26 | 5 | 15 | 5 |

#### 2.2.2. Bảng quái vật

Có 4 loại quái vật được thiết kế, mỗi loại có ngoại hình và hành vi khác nhau nhưng cùng chỉ số về máu và sát thương. Chúng xuất hiện ngẫu nhiên trên bản đồ.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên quái vật** | **Máu (HP)** | **Sát thương (Min-Max)** | **Tốc độ di chuyển** | **Ghi chú** |
| Quỷ xanh | 40 | 2 - 3 | 2 | Mang  nhiều độc tố |
| Quỷ đỏ | 40 | 2 - 3 | 2 | Hung hãn |
| Quỷ tím | 40 | 2 - 3 | 2 | Ranh mãnh |
| Quỷ vàng | 40 | 2 - 3 | 2 | Nhanh nhẹn |

#### 2.2.3. Bảng vũ khí

Người chơi có thể mang theo tối đa 15 khẩu súng. Mỗi khẩu súng có thông số khác nhau như tốc độ bắn, sát thương, và thời gian hồi. Dưới đây là ba loại súng tiêu biểu:

Mỗi khi người chơi lên cấp, họ sẽ được chọn thêm vũ khí mới từ danh sách có sẵn, tối đa 15 khẩu.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tên súng** | **Start Time Btw Shots** | **Bullet Force** | **Min Damage** | **Max Damage** | **Số đạn ra từ một lần bắn** | **Ghi chú** |
| SMG-1 | 0.25 | 9 | 6 | 16 | 1 | Tốc độ bắn nhanh, sát thương thấp, thích hợp tầm trung |
| Shotgun | 0.65 | 9 | 16 | 32 | 3 | Tốc độ chậm nhưng sát thương cao, thích hợp tầm gần |
| Pistol-Z | 0.45 | 9 | 28 | 40 | 1 | Tốc độ vừa phải, sát thương cao, thích hợp tầm xa |

#### 2.2.4. Màn chơi và chế độ

Hiện tại trò chơi có một bản đồ duy nhất, và hai chế độ chơi chính:

**a, Normal Mode** (Chế độ chơi mặc định):

Là bản **demo** thử nghiệm.

Quái vật xuất hiện ngẫu nhiên, số lượng có giới hạn theo thời gian.

Mục tiêu: Tiêu diệt toàn bộ quái vật và sống sót đến cuối.

**b, Endless Mode** (Dự kiến):

Quái vật xuất hiện không giới hạn cho đến khi người chơi chết.

Người chơi càng sống lâu, số điểm tích lũy càng cao.

Tăng sát thương và máu quái vật, số kinh nghiệm cần để lên cấp

Thêm vũ khí mạnh mẽ có thể nhận được khi lên cấp

#### 2.2.5. Lối chơi tổng quan

Người chơi chọn một trong bốn nhân vật (sẽ mở được nhân vật mới khi đạt thành tựu nhất định).

Bắt đầu màn chơi, nhân vật sẽ chiến đấu với các đợt quái vật xuất hiện ngẫu nhiên.

Mỗi khi tiêu diệt đủ số lượng quái vật hoặc đạt mốc điểm kinh nghiệm, người chơi sẽ được tăng cấp, chọn thêm súng mới.

Trò chơi kết thúc khi người chơi hết máu.

Điểm số được tính dựa trên sốquái vật tiêu diệt và thời gian sống sót.

## **CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ TRÒ CHƠI**

Trong chương này, đồ án sẽ đi sâu vào việc phân tích và thiết kế chi tiết cho một số chức năng chính của trò chơi.

### **3.1. Phân tích hệ thống**

#### **3.1.1. Mô tả dự án**

**a. Xác định phạm vi dự án**

#### Shooting Monster là một trò chơi hành động bắn quái vật, phát triển trên nền tảng Unity 2D, sử dụng ngôn ngữ lập trình C#. Trò chơi hướng đến người chơi đơn, phát hành trên máy tính. Người chơi sẽ điều khiển một nhân vật với các loại súng để tiêu diệt các quái vật, nâng cao cấp độ, thu thập thêm súng sống sót đến cuối cùng đạt được số điểm cao nhất có thể.

**b. Tính năng trò chơi chi tiết**

Tính năng Main Menu: Giao diện chính cho phép người chơi bắt đầu trò chơi, điều chỉnh các tùy chọn cài đặt, truy cập hướng dẫn và thoát khỏi trò chơi. Đây là tính năng cơ bản giúp người chơi tương tác với các chức năng khác của game một cách trực quan và thuận tiện.

Chức năng MainMenu:

Bắt đầu trò chơi: Giao diện MainMenu -> chọn Play -> chọn độ khó -> chọn nhân vật -> bắt đầu chơi

Chức năng cài đặt: Setting: Cài đặt trò chơi: độ khó, âm lượng, điều khiển, đồ họa video, Key Bindings, return để quay lại MainMenu

Extras: xem tên tác giả

Exit: để thoát trò chơi

Chức năng GamePlay:

Chức năng thanh máu: Hiện thị số máu hiện tại của người chơi.

Chức năng đếm mạng hạ gục: số quái bị tiêu diệt.

Chức năng level: Hiện level hiện tại và số lượng kinh nghiệm cho cấp tiếp theo.

Chức năng di chuyển và tấn công: Các nút mặc định A,S,W,D để di chuyển trái, xuống, lên, phải, space để lướt nhanh, chuột trái để bắn, chuột trái chọn súng khi lên cấp.

Chức năng bắn đạn và hiệu ứng: Súng bắn đạn và có hiệu ứng tia lửa.

Chức năng hiển thị sát thương: Hiển thị sát thương gây lên quái, và quái gây lên người chơi.

Chức năng thời gian: còn bao nhiêu thời gian để sống sót.

Chức năng điểm số: Hiện điểm số sau khi chết

Chức năng chơi lại: Hiển thị nút chơi lại: R

#### **3.1.2. Mô tả hệ thống bằng UML**

Các từ khóa được dùng trong trò chơi

**Bảng 5: Từ khóa**

|  |  |  |
| --- | --- | --- |
| STT | Tên | Mô tả |
| 1 | Play | Chọn mức độ chơi, chọn nhân vật |
| 2 | Settings | Các cài đặt trò chơi |
| 3 | Extras | Hiển thị tên tác giả |
| 4 | Exit | Thoát trò chơi |
| 5 | Start Game | Bắt đầu trò chơi |
| 6 | Player | Người chơi điều khiển nhân vật trong game |
| 7 | Enemy | Quái vật phải tiêu diệt, có thể tấn công người chơi |
| 8 | Weapon | Vũ khí: các loại súng |
| 9 | Spawners | Các điểm xuất hiện quái vật |
| 10 | Dame | Sát thương từ súng của người chơi gây lên quái, sát thương từ quái lên người chơi |
| 11 | Health Bar | Thanh máu hiển thị Hp của người chơi |
| 12 | Exp Bar | Hiển thị Level và tiến trình lên cấp |
| 13 | Killed | Số quái vật đã hạ gục |
| 14 | Timer | Bộ đếm thời gian chơi, thời gian còn lại để sống sót |
| 15 | Level Up | Hiển thị màn hình level up để người chơi chọn súng khi lên cấp, khi chọn súng thời gian sẽ ngưng đọng |
| 16 | Bullet | Hình ảnh đạn bay tới quái, chạm vào sẽ gây sát thương và biến mất |
| 17 | Lose Game | Hiển thị lên màn hình rằng đã thua, nhấn R để chơi lại, hiển thị điểm số đã đạt được |

#### **3.1.3. Kịch bản chuẩn và ngoại lệ**

**a, Kịch bản Start Game**

**Bảng 6: Kịch bản Start Game**

|  |  |
| --- | --- |
| Kịch bản | Start Game |
| Actor | Người chơi |
| Mục đích | Vào chơi game |
| Main Event | 1.Nhấn nút Play  2. Chọn độ khó  3. Chọn nhân vật  4. Nhấn Start Game  5. Vào giao diện chơi |
| Ngoại lệ | Không có ngoại lệ |

**b, Kịch bản tấn công quái vật**

**Bảng 7: Kịch bản tấn công quái vật**

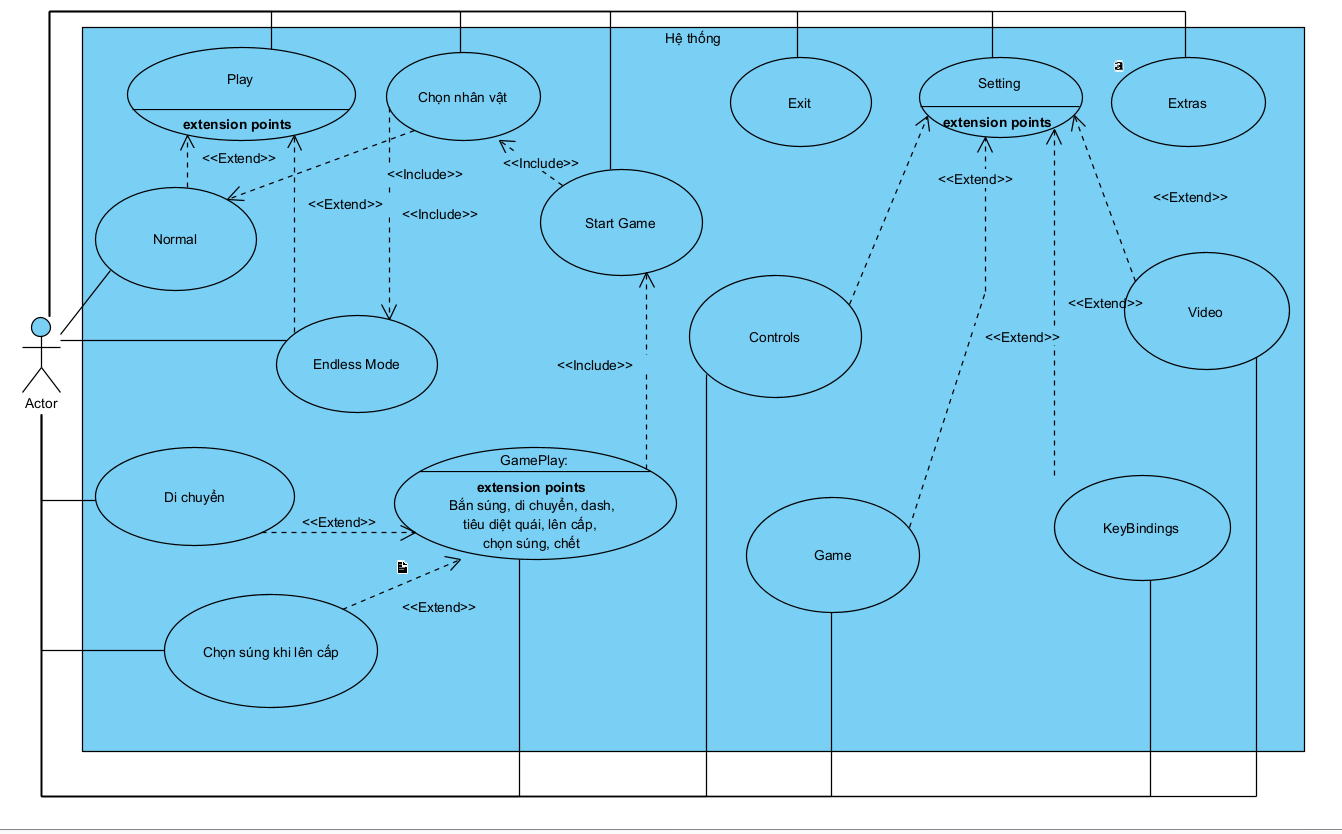
|  |  |
| --- | --- |
| Kịch bản | Tấn công quái vật |
| Actor | Người chơi |
| Tiền Điều Kiện | Start Game |
| Mục đích | Tiêu diệt quái vật |
| Main Event | 1. Sau khi vào giao diện chơi, người chơi di chuyển nhân vật và tấn công quái bằng súng 2. Súng tự động bắn quái 3. Đạn bay khỏi súng va chạm với quái, gây sát thương tùy theo loại súng 4. Quái mất máu, chết nếu hết máu 5. Người chơi nhận được thông tin quái chết, tăng mạng hạ gục, tăng kinh nghiệm |
| Ngoại lệ | Đạn bay hụt khỏi quái, ngoài phạm vi rồi biến mất |

**c, Kịch bản quái tấn công người chơi**

**Bảng 8: Quái tấn công người chơi**

|  |  |
| --- | --- |
| Kịch bản | Quái tấn công người chơi |
| Actor | Quái |
| Tiền Điều Kiện | Start Game |
| Mục đích | Gây sát thương lên người chơi |
| Main Event | 1. Sau khi vào giao diện trò chơi, quái tìm kiếm người chơi theo thuật toán A\*   2. Quái tiếp cận và tấn công người chơi bằng va chạm  3. Người chơi mất máu theo dame của quái |
| Ngoại lệ | Sau khi bị gây sát thương bởi một quái khác, người chơi sẽ miễn nhiễm sát thương trong một khoảng thời gian ngắn và không nhận sát thương từ quái đó |

#### 3.1.4. Usecase tổng quan



Hình 1: Usecase tổng quan

Mô tả Usecase tổng quan:

Chức năng MainMenu:

Play: chọn độ khó để bắt đầu chơi.

Normal, Endless Mode: độ khó khi chơi, chọn để chọn nhân vật.

Chọn nhân vật: Chọn nhân vật để bắt đầu chơi.

Start Game: Bắt đầu vào GamePlay

Setting: Tùy chỉnh cài đặt phù hợp lối chơi

Controls: điều khiển chuột, độ mượt chuyển động

Game: Cài đặt độ khó, âm lượng

Video: cài đặt hình ảnh, đồ họa, cấu hình

Keybindings: Cài đặt nút phù hợp lối chơi

Extras: Hiển thị tên tác giả

Exit: Thoát game

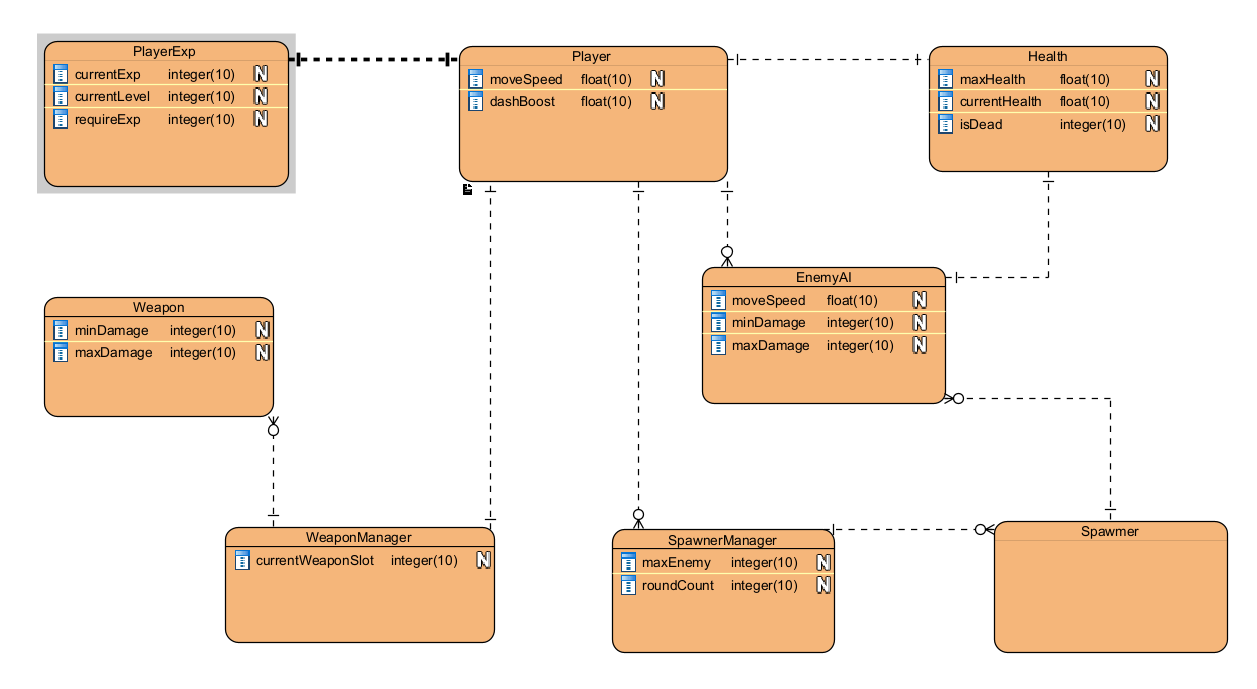
Chức năng GamePlay:

Di chuyển: Nhân vật di chuyển, né, lướt, tránh quái.

Chọn súng khi lên cấp.

### **3.2. Thiết kế hệ thống trò chơi**

#### **3.2.1. Sơ đồ lớp thực thể hệ thống**

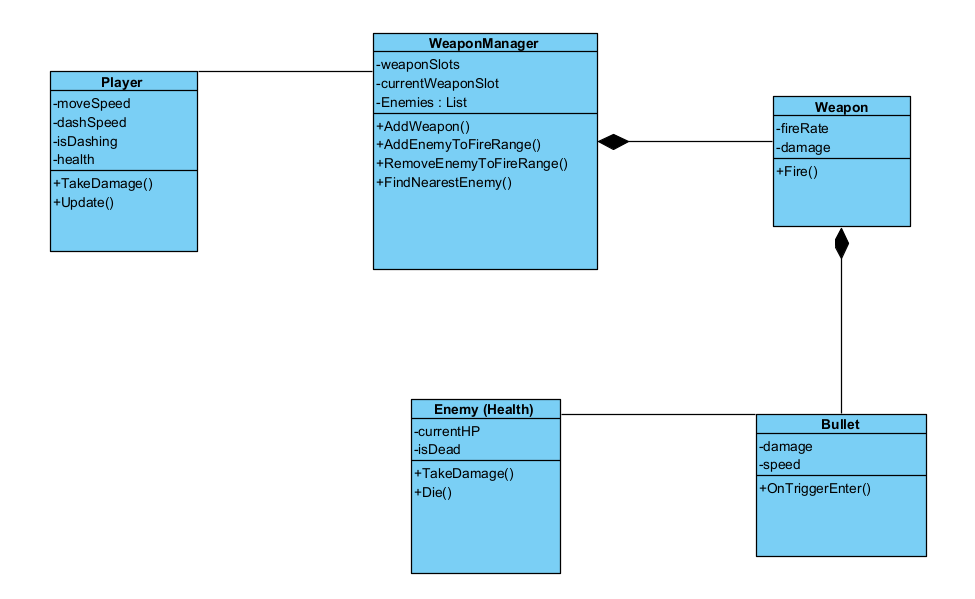
****

Hình 2: Biểu đồ lớp thực thể của trò chơi

**Bảng 9: Mô tả tổng quan các lớp**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên lớp | Mô tả chức năng chính | Quan hệ với lớp khác |
|  | Player | Điều khiển nhân vật người chơi, xử lý di chuyển, dash và nhận sát thương | Gọi Health, LosePanel, Animator, Rigidbody2D, TextMeshProUGUI, PlayerExp |
|  | Health | Quản lý máu, xử lý khi nhận sát thương và chết | Tương tác với HealthBar,  WeaponManager, Killed |
|  | HealthBar | Cập nhật giao diện thanh máu | Được gọi bởi Health |
|  | PlayerExp | Quản lý điểm kinh nghiệm, lên cấp và mở giao diện nâng cấp | Gọi  LevelUpPanel |
|  | LevelUpPanel | Giao diện nâng cấp kỹ năng/kỹ năng mới khi lên cấp | Kích hoạt từ PlayerExp |
|  | EnemyAI | EnemyAI | Gọi Health, Player, WeaponManager, EnemyController, Seeker (Pathfinding) |
|  | EnemyController | Hiển thị hiệu ứng khi quái bị sát thương và gây hiệu ứng đặc biệt | Gọi Health, EnemyAI, ColoredFlash |
|  | Bullet | Xử lý va chạm giữa đạn và quái vật, gây sát thương | Gọi Health, EnemyController |
|  | WeaponManager | Quản lý vũ khí của người chơi và danh sách kẻ địch trong phạm vi tấn công | Được gọi bởi EnemyAI, Health, SpawnerManager |
|  | SpawnerManager | Điều khiển quy trình tạo quái vật theo từng đợt (wave) và vòng chơi | Gọi Spawner, WeaponManager, Player |
|  | Spawner | Sinh quái vật tại một vị trí cố định | Được gọi bởi SpawnerManager |

#### **3.2.2. Sơ đồ lớp module tấn công kẻ địch**



**Chương IV: Kết luận và hướng phát triển**

### **4.1. Kết luận**

Đồ án "Shooting Monster" đã giúp em áp dụng kiến thức về lập trình C#, Unity và phát triển game vào thực tế. Qua quá trình thực hiện, em đã hiểu rõ hơn về quy trình phát triển game, từ thiết kế đến triển khai.

### **4.2. Hướng phát triển**

- Thêm nhiều loại quái vật và kỹ năng mới.

- Phát triển hệ thống nhiệm vụ và phần thưởng.

- Tối ưu hóa hiệu suất và trải nghiệm người dùng.

- Mở rộng hỗ trợ trên các nền tảng khác như iOS, WebGL.