|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HỌC VIỆN KỸ THUẬT MẬT MÃ  **KHOA CÔNG NGHỆ THÔNG TIN**  Logo HvKTMM  BÁO CÁO MÔN HỌC  **PHÁT TRIỂN GAME TRÊN ANDROID**  Đề tài: **Xây dựng game platform**     |  |  | | --- | --- | | Giảng viên hướng dẫn: **Nguyễn Thái Sơn**  Nhóm sinh viên thực hiện**:** Đinh Minh Quang - CT050441  Phạm Văn Dũng - CT050410  Nhóm 2 |  | |  |  | |  |  | |  |  |     Hà Nội, 2025 |

MỤC LỤC

[LỜI CẢM ƠN 6](#_Toc198947330)

[CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN 8](#_Toc198947332)

[**1.1. Giới thiệu tổng quan** 8](#_Toc198947333)

[**1.2. Khảo sát thị trường** 9](#_Toc198947334)

[1.2.1. Phương pháp khảo sát 9](#_Toc198947335)

[1.2.2. Thu thập dữ liệu 9](#_Toc198947336)

[1.2.3. Báo cáo dữ liệu 9](#_Toc198947337)

[**1.3. Chức năng dự kiến** 11](#_Toc198947338)

[1.3.1. Cơ chế di chuyển nhân vật 11](#_Toc198947339)

[1.3.2. Thiết kế cấp độ 11](#_Toc198947340)

[1.3.3. Tương tác với môi trường 11](#_Toc198947341)

[1.3.4. Hệ thống âm thanh và nhạc nền 11](#_Toc198947342)

[**1.4. Yêu cầu phi chức năng** 12](#_Toc198947343)

[1.4.1. Hiệu suất (Performance) 12](#_Toc198947344)

[1.4.2. Tính tương thích (Compatibility) 12](#_Toc198947345)

[1.4.3. Khả năng duy trì (Maintainability) 12](#_Toc198947346)

[1.4.4. Tính ổn định (Stability) 12](#_Toc198947347)

[***1.5. Một số game platformer hiện nay*** 12](#_Toc198947348)

[**1.6. Công nghệ sử dụng** 15](#_Toc198947349)

[1.6.1. Unity 15](#_Toc198947350)

[1.6.2. C# 15](#_Toc198947351)

[CHƯƠNG 2. PHÂN TÍCH 19](#_Toc198947352)

[**2.1. Danh sách tác nhân** 19](#_Toc198947353)

[**2.2. Yêu cầu chức năng** 19](#_Toc198947354)

[**2.3. Sơ đồ use case tổng quát** 20](#_Toc198947355)

[**2.4. Đặc tả ca sử dụng** 21](#_Toc198947356)

[2.4.1. Đặc tả use case chọn chế độ chơi 21](#_Toc198947357)

[2.4.2. Đặc tả use case cài đặt 21](#_Toc198947358)

[2.4.3. Đặc tả use case di chuyển nhân vật 22](#_Toc198947359)

[2.4.4. Đặc tả use case thu thập vật phẩm 22](#_Toc198947360)

[2.4.5. Đặc tả use case điểm số 22](#_Toc198947361)

[2.4.6. Đặc tả use case thoát trò chơi 23](#_Toc198947362)

[CHƯƠNG 3. THIẾT KẾ 24](#_Toc198947363)

[**3.1. Biểu đồ tuần tự** 24](#_Toc198947364)

[3.1.1. Biểu đồ tuần tự chức năng chọn chế độ chơi 24](#_Toc198947365)

[3.1.2. Biểu đồ tuần tự chức năng cài đặt 25](#_Toc198947366)

[3.1.3. Biểu đồ tuần tự chức năng di chuyển nhân vật, thu thập vật phẩm và điểm số 26](#_Toc198947367)

[**3.2. Biểu đồ hoạt động** 27](#_Toc198947368)

[3.2.1. Biểu đồ hoạt động use case cài đặt 27](#_Toc198947369)

[3.2.2. Biểu đồ hoạt động use case chọn màn chơi 28](#_Toc198947370)

[3.2.3. Biểu đồ hoạt động use case di chuyển nhân vật, thu thập vật phẩm và hiển thị điểm số 29](#_Toc198947371)

[CHƯƠNG 4: XÂY DỰNG TRÒ CHƠI 30](#_Toc198947372)

[**4.1. Quá trình thiết kế trò chơi** 30](#_Toc198947373)

[4.1.1. Lên Kế Hoạch và Thiết Kế 30](#_Toc198947374)

[4.1.2. Phát Triển Game với Unity 30](#_Toc198947375)

[4.1.3. Kiểm Tra và Tinh Chỉnh 30](#_Toc198947376)

[4.1.4. Triển Khai và Bảo Trì 30](#_Toc198947377)

[**4.2. Giới thiệu giao diện trò chơi** 30](#_Toc198947378)

[4.2.1. Màn hình đăng nhập, đăng ký 30](#_Toc198947379)

[4.2.2. Màn hình trước khi chơi game 31](#_Toc198947380)

[4.2.3. Màn hình chọn cấp độ chơi game 32](#_Toc198947381)

[4.2.4. Màn hình chơi game 33](#_Toc198947382)

[4.2.5. Màn hình kết thúc màn chơi 35](#_Toc198947383)

[KẾT LUẬN 36](#_Toc198947384)

[TÀI LIỆU THAM KHẢO 38](#_Toc198947385)

DANH MỤC HÌNH VẼ

[Hình 1.1. Hình ảnh minh họa thể loại game đi cảnh 8](#_Toc199007047)

[Hình 1.2. Biểu đồ khảo sát thời gian chơi game 10](#_Toc199007048)

[Hình 1.3. Biểu đồ khảo sát thiết bị chơi game 10](#_Toc199007049)

[Hình 1.4. Unity 15](#_Toc199007050)

[Bảng 2.1. Danh sách Actor 19](#_Toc199007051)

[Bảng 2.2. Danh sách Use case 19](#_Toc199007052)

[Hình 2.1. Sơ đồ use case tổng quát 20](#_Toc199007053)

[Bảng 2.3. Đặc tả use case chọn chế độ chơi 21](#_Toc199007054)

[Bảng 2.4. Đặc tả use case cài đặt 21](#_Toc199007055)

[Bảng 2.5. Đặc tả use case di chuyển nhân vật 22](#_Toc199007056)

[Bảng 2.6. Đặc tả use case thu thập vật phẩm 22](#_Toc199007057)

[Bảng 2.7. Đặc tả use case điểm số 23](#_Toc199007058)

[Bảng 2.8. Đặc tả use case thoát trò chơi 23](#_Toc199007059)

[Hình 3.1. Biểu đồ tuần tự chức năng chọn chế độ chơi 24](#_Toc199007060)

[Hình 3.2. Biểu đồ tuần tự chức năng cài đặt 25](#_Toc199007061)

[Hình 3.3. Biểu đồ tuần tự chức năng di chuyển, thu thập vật phẩm và cập nhật điểm số 26](#_Toc199007062)

[Hình 3.4. Biểu đồ hoạt động chức năng cài đặt 27](#_Toc199007063)

[Hình 3.5. Biểu đồ hoạt động chức năng chọn màn chơi 28](#_Toc199007064)

[Hình 3.6. Biểu đồ hoạt động use case di chuyển nhân vật, thu thập vật phẩm và hiển thị điểm số 29](#_Toc199007065)

[Hình 4.1. Giao diện game 31](#_Toc199007066)

[Hình 4.2. Giao diện chính trong game 32](#_Toc199007067)

[Hình 4.3. Giao diện chọn cấp độ chơi game 33](#_Toc199007068)

# LỜI CẢM ƠN

Trong quá trình thực hiện báo cáo này, nhóm chúng em đã nhận được sự giúp đỡ và hướng dẫn tận tình của thầy. Những ý kiến đóng góp của thầy đã giúp nhóm em cải thiện bài báo cáo của mình tốt hơn để đáp ứng yêu cầu môn học.

Vì thời gian gấp rút cũng như kiến thức còn hạn hẹp nên không thể tránh được thiếu sót trong quá trình báo cáo. Nhóm em mong nhận được sự chỉ bảo, góp ý của thầy và các bạn trong lớp để chúng em có điều kiện bổ sung, tối ưu sản phẩm.

Nhóm em xin chân thành cảm ơn.

# LỜI NÓI ĐẦU

Trong thế giới game, thể loại platformer đã từ lâu chiếm được tình cảm của nhiều người chơi bởi lối chơi thú vị và trải nghiệm hấp dẫn. Dự án thiết kế game platformer mà chúng em thực hiện không chỉ là một thử thách sáng tạo mà còn là một cơ hội để khám phá những yếu tố cốt lõi làm nên thành công của thể loại này.

Mục tiêu của dự án là tạo ra một trò chơi không chỉ cuốn hút về mặt hình ảnh mà còn phong phú về cơ chế gameplay. Người chơi sẽ được đưa vào một hành trình đầy màu sắc, nơi họ có cơ hội vượt qua các chướng ngại vật, giải quyết các câu đố và khám phá những bí mật ẩn giấu. Qua đó, chúng em mong muốn mang lại cho người chơi một trải nghiệm vừa giải trí vừa kích thích tư duy.

Thông qua báo cáo này, chúng em sẽ trình bày chi tiết về quy trình thiết kế, từ ý tưởng ban đầu, phát triển nhân vật, cho đến việc xây dựng môi trường và các cơ chế gameplay. Chúng em hy vọng rằng sản phẩm cuối cùng không chỉ đáp ứng được những tiêu chí thiết kế mà còn mang lại niềm vui và sự hứng thú cho người chơi ở mọi lứa tuổi.

# CHƯƠNG 1. GIỚI THIỆU TỔNG QUAN

## **1.1. Giới thiệu tổng quan**

Trong thời đại công nghệ số phát triển mạnh mẽ, các trò chơi điện tử đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của nhiều người. Trong số đó, game platform – một thể loại trò chơi truyền thống với những bước nhảy vượt chướng ngại vật và khám phá thế giới – vẫn luôn giữ được vị trí quan trọng nhờ tính giải trí cao và lối chơi đơn giản nhưng cuốn hút.

Platform game hay còn gọi là game đi cảnh. Đây là một thể loại game điện tử và cũng là một phần của game hành động. Mục đích cốt lõi của tất cả game đi cảnh là người chơi sẽ điều khiển nhân vật trong game vượt qua các chướng ngại vật trong game bằng cách leo trèo, nhảy xa, đu dây.

Game đi cảnh thường được thấy nhiều trên hệ máy console và đã đạt được thời kỳ đỉnh cao của game khi chiếm đến hơn 30% tổng số lượng game console lúc bấy giờ. Nhưng thời đại hoàng kim của platform game đã qua đi kể từ khi game bắn súng ra đời và thị trường game đi cảnh có phần giảm sút nhưng vẫn luôn là một thể loại game có lượng fan trung thành cao.

Điểm cơ bản nhất của game platform là lối chơi điều khiển nhân vật trong môi trường với địa hình khó khăn. Đồng thời, người chơi còn phải vượt qua rất nhiều đối thủ trong game và tránh né tất cả các chướng ngại vật mới có thể chiến thắng. Lối chơi theo phong cách năng động, thách thức khả năng phản xạ và sự khéo léo của người chơi.



Hình 1.1. Hình ảnh minh họa thể loại game đi cảnh

## **1.2. Khảo sát thị trường**

### 1.2.1. Phương pháp khảo sát

* Sử dụng Google Form để thu thập thông tin người dùng về độ tuổi, nền tảng người chơi yêu thích, điều mọi người thích ở trò chơi.
* Thực hiện phỏng vấn cá nhân và nhóm nhỏ để thu thập ý kiến chi tiết hơn về trải nghiệm chơi game, lý do chọn game và ý kiến về game.

### 1.2.2. Thu thập dữ liệu

* **Nhóm tuổi 10-17:** Nhóm tuổi nhỏ thích các trò chơi với đồ họa hấp dẫn, dễ chơi, và có yếu tố giải trí cao.
* **Nhóm tuổi 18-25:** Đây là độ tuổi ưa chuộng những game có thử thách và yêu cầu kỹ năng cao, thường chơi game platform trên nhiều nền tảng (PC, console, mobile)
* **Nhóm tuổi 26-35:** Đây là nhóm người chơi có kinh nghiệm, thường đánh giá cao nội dung và cốt truyện game, cũng như chất lượng kỹ thuật của trò chơi.
* **Nhóm trên 35 tuổi:** Dù ít hơn, nhưng họ vẫn quan tâm đến game platform, thường tìm kiếm những trò chơi có lối chơi truyền thống, không quá phức tạp.

### 1.2.3. Báo cáo dữ liệu

* Sau khi tiến hành thu thập dữ liệu thì nhóm thu được kết quả
* 69% người chơi ở nhóm tuổi 10-17
* 27% người chơi ở nhóm tuổi 18-25
* 3% người chơi ở nhóm tuổi 26-35
* 1% người chơi ở nhóm tuổi trên 35
* Khảo sát thời gian chơi game:

A graph of green rectangular bars

Description automatically generated with medium confidence

Hình 1.2. Biểu đồ khảo sát thời gian chơi game

- Khảo sát thiết bị chơi game:

A graph with green bars

Description automatically generated

Hình 1.3. Biểu đồ khảo sát thiết bị chơi game

* Một số chia sẻ của những người tham gia thu thập thông tin:
* Tôi thích nhân vật có tốc độ di chuyển trung bình, không quá nhanh nhưng cũng không quá chậm.
* Kết hợp thêm các kỹ năng nhảy khác như trượt và leo tường là một điểm thú vị.
* Tôi thích đồ họa pixel , nó tạo cảm giác hoài cổ và nghệ thuật cho game.
* Màu sắc trong game nên tươi sáng và sống động, tạo nên một không gian vui nhộn và thú vị.
* Yếu tố phá hủy trong môi trường (phá tường, cây cối) cũng là một điều tôi rất thích, vì nó làm tăng khả năng tương tác của người chơi với thế giới trong game.
* Tôi muốn các yếu tố này có độ phức tạp vừa phải, không quá khó nhưng cũng đủ thử thách để người chơi cần suy nghĩ.

## **1.3. Chức năng dự kiến**

### 1.3.1. Cơ chế di chuyển nhân vật

* Điều khiển cơ bản: di chuyển trái, phải, nhảy, cúi, và chạy. Người chơi có thể điều khiển nhân vật qua các phím hoặc cần điều khiển.
* Cơ chế nhảy: Nhân vật có thể nhảy đơn, nhảy kép (double jump) hoặc nhảy trên các bề mặt khác nhau.
* Tương tác với môi trường: Nhân vật có thể leo tường, trượt trên mặt đất, hoặc vượt qua chướng ngại vật.

### 1.3.2. Thiết kế cấp độ

* Cấp độ có độ phức tạp: Cấp độ thiết kế với nhiều tầng lớp, lối đi ẩn và các chướng ngại vật đa dạng.
* Cấp độ tuyến tính và phi tuyến: Người chơi có thể lựa chọn lối đi trong cấp độ, có thể là tuyến tính hoặc phi tuyến để khám phá nhiều lựa chọn.
* Cấp độ đa dạng: Các cấp độ có sự thay đổi về môi trường (như rừng, núi lửa, thành phố, không gian) và độ khó tăng dần.
* Checkpoint và hồi sinh: Người chơi có thể lưu tiến độ ở một số điểm (checkpoint) và hồi sinh tại điểm đó khi thất bại.

### 1.3.3. Tương tác với môi trường

* Tương tác với vật thể: Nhân vật có thể đẩy, kéo, hoặc phá hủy các vật thể trong game.
* Cơ chế vật lý: Vật lý ảnh hưởng đến hành vi của vật thể, ví dụ như hộp rơi, nước chảy, gió thổi.
* Chướng ngại vật: Các chướng ngại vật như bẫy, lửa, gai, và các yếu tố nguy hiểm khác trong môi trường.

### 1.3.4. Hệ thống âm thanh và nhạc nền

* Hiệu ứng âm thanh chi tiết: Âm thanh khi nhân vật di chuyển, nhảy, đánh, và tương tác với môi trường.
* Nhạc nền thay đổi theo tình huống: Nhạc nền thay đổi tùy vào mức độ căng thẳng hoặc sự kiện trong game, ví dụ: nhạc sôi động trong trận đấu boss và nhạc nhẹ nhàng khi khám phá.
* Tùy chỉnh âm thanh: Người chơi có thể tùy chỉnh âm lượng của nhạc nền và hiệu ứng âm thanh.

## **1.4. Yêu cầu phi chức năng**

### 1.4.1. Hiệu suất (Performance)

* Tốc độ tải nhanh: Game phải có thời gian tải nhanh, đặc biệt là khi chuyển đổi giữa các cấp độ hoặc màn hình menu.
* Khung hình ổn định (Frame Rate): Đảm bảo game chạy mượt mà với tốc độ khung hình ổn định trên các thiết bị được hỗ trợ, tránh tình trạng giật, lag.
* Tối ưu hóa tài nguyên: Tối ưu hóa bộ nhớ và hiệu suất để game không tiêu tốn quá nhiều tài nguyên hệ thống, đặc biệt là trên các nền tảng có cấu hình thấp như mobile.

### 1.4.2. Tính tương thích (Compatibility)

* Tương thích nhiều thiết bị: Game phải chạy mượt trên các thiết bị với các hệ điều hành và cấu hình phần cứng khác nhau (PC, console, mobile).
* Hỗ trợ nhiều độ phân giải: Game cần tương thích với nhiều độ phân giải màn hình khác nhau, từ HD, Full HD đến 4K, đảm bảo hiển thị tốt trên tất cả các màn hình.

### 1.4.3. Khả năng duy trì (Maintainability)

* Dễ dàng sửa lỗi và cập nhật: Game cần có cấu trúc mã nguồn rõ ràng và dễ bảo trì, giúp các nhà phát triển dễ dàng sửa lỗi hoặc cập nhật nội dung mới.
* Khả năng mở rộng tính năng: Dễ dàng thêm các tính năng hoặc yếu tố mới mà không cần phải thay đổi toàn bộ hệ thống của game.

### 1.4.4. Tính ổn định (Stability)

* Game không bị crash: Đảm bảo rằng game hoạt động ổn định, không bị đứng, giật hoặc crash trong quá trình chơi.
* Xử lý lỗi tốt: Nếu có lỗi xảy ra, game cần có cơ chế để khôi phục lại tiến trình hoặc ít nhất là không làm mất toàn bộ dữ liệu của người chơi.

## ***1.5. Một số game platformer hiện nay***

Thể loại game platformer đã phát triển mạnh mẽ qua nhiều thập kỷ và hiện nay vẫn giữ được sự phổ biến với những tựa game sáng tạo và độc đáo. Dưới đây là một số trò chơi platformer nổi bật trong thời gian gần đây:

* **Celeste:** Celeste là một trò chơi platformer 2D nổi bật với lối chơi thử thách cao và câu chuyện cảm động. Người chơi sẽ điều khiển nhân vật Madeline leo lên ngọn núi Celeste, phải đối mặt với các chướng ngại vật phức tạp đòi hỏi kỹ năng điều khiển chính xác. Điểm đặc biệt của Celeste là khả năng tạo ra sự cân bằng giữa độ khó và tính nhân văn trong cốt truyện.
* **Hollow Knight:** Hollow Knight là một tựa game platformer kết hợp yếu tố hành động và khám phá (Metroidvania). Người chơi sẽ điều khiển một hiệp sĩ nhỏ bé khám phá thế giới dưới lòng đất Hallownest đầy rẫy kẻ thù và bí ẩn. Đồ họa tay vẽ đẹp mắt, cùng với hệ thống chiến đấu và thiết kế bản đồ tinh vi, đã giúp Hollow Knight trở thành một trong những tựa game platformer được yêu thích nhất.
* **Ori and the Will of the Wisps:** Ori and the Will of the Wisps là phần tiếp theo của Ori and the Blind Forest, một trò chơi platformer lấy bối cảnh trong một thế giới kỳ diệu. Game gây ấn tượng với người chơi bởi đồ họa nghệ thuật tinh xảo, âm nhạc cảm xúc, và lối chơi kết hợp giữa platformer và giải đố. Các cấp độ được thiết kế mượt mà và thử thách người chơi trong việc điều khiển nhân vật qua các cảnh quan đầy mê hoặc.
* **Super Mario Odyssey:** Là một phần của dòng game Mario nổi tiếng, Super Mario Odyssey mang đến trải nghiệm platformer 3D với lối chơi sáng tạo và thế giới mở rộng lớn. Người chơi sẽ đồng hành cùng Mario trong chuyến phiêu lưu khám phá các vương quốc mới, sử dụng chiếc mũ đặc biệt của mình để nhảy và tương tác với môi trường theo cách đầy thú vị. Đây là một trong những trò chơi được đánh giá cao nhờ sự sáng tạo và độ giải trí cao.
* **Cuphead:** Cuphead là một trò chơi platformer hành động có phong cách đồ họa hoạt hình cổ điển, lấy cảm hứng từ những bộ phim hoạt hình Mỹ thập niên 1930. Trò chơi nổi bật với độ khó cao, yêu cầu người chơi vừa vượt qua các màn platform phức tạp vừa chiến đấu với những con trùm (boss) mạnh mẽ. Sự độc đáo trong đồ họa và gameplay của Cuphead đã tạo ra dấu ấn riêng biệt trong thể loại platformer.

Những tựa game trên đại diện cho sự đa dạng và sáng tạo của thể loại platformer trong thời đại hiện nay, từ việc kết hợp lối chơi cổ điển với cốt truyện sâu sắc, cho đến việc khai thác đồ họa độc đáo để tạo ra những trải nghiệm đáng nhớ cho người chơi.

## **1.6. Công nghệ sử dụng**

### 1.6.1. Unity

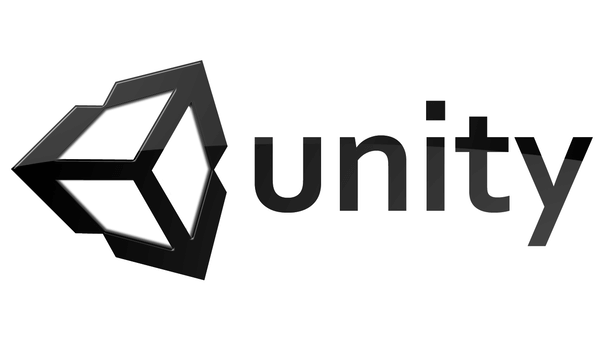
Unity là một công cụ trò chơi đa nền tảng được phát triển bởi Unity Technologies, chủ yếu được sử dụng để phát triển trò chơi điện tử và mô phỏng cho máy tính, bảng điều khiển và thiết bị di động.

Với các tính năng đa nền tảng, Unity phổ biến với cả các nhà phát triển game tự do và trong các studio game. Nó được sử dụng để tạo các trò chơi như Pokemon Go, Heathstone, Rimworld, Cuphead và nhiều trò chơi khác.

Lập trình Unity 2D và 3D được lập trình dựa vào 3 ngôn ngữ chính là C#, Boo và UnityScript. Thông thường, ngôn ngữ chính mà lập trình viên Unity sử dụng phổ biến nhất hiện nay là C#.

Bởi vì sự có mặt lâu từ năm 2005, Unity đã phát triển một lượng lớn người dùng và một thư viện tài nguyên khổng lồ. Unity không chỉ có tài liệu tuyệt vời, mà còn có vô số video và hướng dẫn trực tuyến đáng kinh ngạc cho người dùng.

Chính vì vậy, Unity là một sự lựa chọn thông minh cho người mới bắt đầu tiếp cận với các công cụ game. Nằm trong danh sách các công cụ trò chơi điện tử , Unity đóng vai trò như một cổng thông tin kiến ​​thức và tài nguyên được xây dựng chỉ dựa trên cộng đồng rộng lớn của họ.



Hình 1.4. Unity

### 1.6.2. C#

**C#** (đọc là "C thăng" hoặc "C sharp") là một ngôn ngữ lập trình hiện đại, hướng đối tượng, và được phát triển bởi Microsoft. Nó được giới thiệu lần đầu vào năm 2000 như một phần của nền tảng .NET Framework. C# được thiết kế để xây dựng nhiều loại ứng dụng khác nhau, từ ứng dụng desktop, web, di động, game cho đến các dịch vụ đám mây và ứng dụng doanh nghiệp.

**Lịch sử phát triển**

* **Ra đời:** C# được Anders Hejlsberg, một kiến trúc sư phần mềm nổi tiếng (người đã tạo ra Turbo Pascal và Delphi), dẫn đầu nhóm phát triển tại Microsoft.
* **Ảnh hưởng:** Ngôn ngữ này chịu ảnh hưởng mạnh mẽ từ các ngôn ngữ lập trình phổ biến khác như C++, Java, và ở một mức độ nào đó là Delphi và Visual Basic.
* **Phiên bản:** C# đã trải qua nhiều phiên bản, với mỗi phiên bản đều được bổ sung các tính năng mới và cải tiến hiệu suất. Hiện tại, C# đang ở phiên bản 12 (tính đến thời điểm tháng 5 năm 2024), đi kèm với .NET 8.

**Đặc điểm nổi bật**

* **Hướng đối tượng (OOP):** C# hỗ trợ đầy đủ các nguyên tắc của lập trình hướng đối tượng như đóng gói (encapsulation), kế thừa (inheritance), đa hình (polymorphism), và trừu tượng (abstraction).
* **An toàn kiểu (Type-safe):** C# là một ngôn ngữ an toàn kiểu, nghĩa là trình biên dịch sẽ kiểm tra các lỗi liên quan đến kiểu dữ liệu trong quá trình biên dịch, giúp giảm thiểu lỗi runtime.
* **Quản lý bộ nhớ tự động (Automatic Memory Management):** Thông qua Garbage Collector (GC) của .NET, C# tự động quản lý việc cấp phát và giải phóng bộ nhớ, giúp lập trình viên không cần lo lắng về các vấn đề như rò rỉ bộ nhớ.
* **Tương thích đa nền tảng (Cross-platform):** Với sự ra đời của .NET Core (và sau này là .NET 5 trở đi), các ứng dụng C# có thể được phát triển và chạy trên nhiều hệ điều hành khác nhau như Windows, macOS, và Linux.
* **Ngôn ngữ hiện đại:** C# liên tục được cập nhật với các tính năng ngôn ngữ hiện đại như LINQ (Language Integrated Query), async/await để lập trình bất đồng bộ, biểu thức lambda, pattern matching, và nhiều hơn nữa.
* **Hệ sinh thái mạnh mẽ:** C# được hỗ trợ bởi một hệ sinh thái phong phú từ Microsoft và cộng đồng, bao gồm .NET Framework, .NET (Core), các công cụ phát triển mạnh mẽ như Visual Studio và Visual Studio Code, cùng với vô số thư viện và framework.
* **Linh hoạt:** C# có thể được sử dụng để xây dựng nhiều loại ứng dụng khác nhau.
* **Dễ học (tương đối):** Mặc dù là một ngôn ngữ mạnh mẽ, C# có cú pháp khá rõ ràng và dễ tiếp cận, đặc biệt đối với những người đã có kinh nghiệm với C++, Java.

**Ứng dụng phổ biến**

C# được sử dụng rộng rãi trong nhiều lĩnh vực:

* **Phát triển ứng dụng Web:** Với ASP.NET Core, Blazor.
* **Phát triển ứng dụng Desktop:** Với Windows Forms, WPF (Windows Presentation Foundation), UWP (Universal Windows Platform), MAUI.
* **Phát triển Game:** Đặc biệt mạnh mẽ với game engine Unity.
* **Phát triển ứng dụng Di động:** Với Xamarin (nay là một phần của .NET MAUI).
* **Điện toán đám mây (Cloud Computing):** Xây dựng các dịch vụ và ứng dụng trên Microsoft Azure.
* **Trí tuệ nhân tạo (AI) và Máy học (Machine Learning):** Với các thư viện như ML.NET.
* **Internet of Things (IoT):** Phát triển ứng dụng cho các thiết bị IoT.
* **Ứng dụng Doanh nghiệp (Enterprise Applications):** Xây dựng các hệ thống lớn, phức tạp cho doanh nghiệp.

**Ưu điểm và Nhược điểm**

**Ưu điểm:**

* **Hiệu suất tốt:** Mã C# được biên dịch thành Intermediate Language (IL) và sau đó được Just-In-Time (JIT) compiler biên dịch thành mã máy gốc, mang lại hiệu suất tốt.
* **Năng suất cao:** Nhờ cú pháp rõ ràng, các tính năng ngôn ngữ hiện đại, và sự hỗ trợ mạnh mẽ từ IDE (như Visual Studio), lập trình viên có thể phát triển ứng dụng nhanh chóng.
* **Cộng đồng lớn và tài liệu phong phú:** Dễ dàng tìm kiếm sự hỗ trợ, thư viện và tài liệu học tập.
* **Sự hậu thuẫn từ Microsoft:** Đảm bảo sự phát triển và hỗ trợ lâu dài cho ngôn ngữ.
* **Khả năng tương tác (Interoperability):** Có thể dễ dàng tương tác với mã được viết bằng các ngôn ngữ khác, ví dụ như C++.

**Nhược điểm:**

* **Phụ thuộc vào .NET:** Mặc dù đã có .NET đa nền tảng, C# vẫn gắn liền với hệ sinh thái .NET.
* **Đường cong học tập ban đầu:** Đối với người mới hoàn toàn với lập trình, việc nắm bắt các khái niệm hướng đối tượng và hệ sinh thái .NET có thể cần thời gian.
* **Kích thước ứng dụng:** Đôi khi ứng dụng .NET có thể có kích thước lớn hơn so với các ứng dụng được viết bằng một số ngôn ngữ khác.

# CHƯƠNG 2. PHÂN TÍCH

## **2.1. Danh sách tác nhân**

|  |  |  |
| --- | --- | --- |
| **STT** | **Actor** | **Mô tả** |
| 1 | Người chơi | Là người sử dụng trò chơi. |

Bảng 2.1. Danh sách Actor

## **2.2. Yêu cầu chức năng**

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả** |
| 1 | Chọn chế độ chơi | Chọn 1 trong 3 chế độ chơi |
| 2 | Cài đặt | Tùy chỉnh màn chơi |
| 3 | Thoát trò chơi | Thoát khỏi trò chơi |
| 4 | Di chuyển nhân vật | Nhân vật có thể di chuyển trái, phải, nhảy |
| 5 | Thu thập vật phẩm | Thu thập vật phẩm để tăng điểm số |
| 6 | Điểm số | Hiển thị số điểm |

Bảng 2.2. Danh sách Use case

## **2.3. Sơ đồ use case tổng quát**

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.1. Sơ đồ use case tổng quát

## **2.4. Đặc tả ca sử dụng**

### 2.4.1. Đặc tả use case chọn chế độ chơi

|  |  |
| --- | --- |
| **Use Case** | Chọn chế độ chơi |
| **Mô tả** | Người dùng chọn độ khó của màn chơi. |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động trò chơi |
| **Dòng hành động chính** | Chức năng bắt đầu khi người dùng bấm chọn chế độ  - Người dùng chọn 1 trong 3 chế độ với độ khó tương ứng. |
| **Dòng hành động thay thế** | - Người dùng chọn nút back để thoát khỏi trò chơi. |

Bảng 2.3. Đặc tả use case chọn chế độ chơi

### 2.4.2. Đặc tả use case cài đặt

|  |  |
| --- | --- |
| **Use Case** | Cài đặt |
| **Mô tả** | Người dùng thiết lập cài đặt của trò chơi. |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động và vào game thành công |
| **Dòng hành động chính** | Chức năng bắt đầu khi người dùng bấm chọn cài đặt |
| **Dòng hành động thay thế** |  |

Bảng 2.4. Đặc tả use case cài đặt

### 2.4.3. Đặc tả use case di chuyển nhân vật

|  |  |
| --- | --- |
| **Use Case** | Di chuyển nhân vật |
| **Mô tả** | Người dùng di chuyển nhân vật trong game |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động game và vào game thành công |
| **Dòng hành động chính** | Người dùng thực hiện nhấn phím di chuyển  - Cập nhật vị trí của người chơi.  - Di chuyển nhân vật.  - Kiểm tra va chạm.  - Kết thúc di chuyển khi người dùng ngừng tương tác với phím di chuyển. |
| **Dòng hành động thay thế** |  |

Bảng 2.5. Đặc tả use case di chuyển nhân vật

### 2.4.4. Đặc tả use case thu thập vật phẩm

|  |  |
| --- | --- |
| **Use Case** | Thu thập vật phẩm |
| **Mô tả** | Người dùng thu thập vật phẩm trong trò chơi |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động game và vào game thành công |
| **Dòng hành động chính** | Người dùng thực hiện di chuyển nhân vật và thu thập vật phẩm |
| **Dòng hành động thay thế** |  |
| **Hậu điều kiện** | Tăng điểm số của màn chơi |

Bảng 2.6. Đặc tả use case thu thập vật phẩm

### 2.4.5. Đặc tả use case điểm số

|  |  |
| --- | --- |
| **Use Case** | Điểm số |
| **Mô tả** | Hiện điểm số của người chơi |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động game và vào game thành công |
| **Dòng hành động chính** | - Hiện điểm số của màn chơi khi người chơi hoàn thành trò chơi. |
| **Dòng hành động thay thế** |  |

Bảng 2.7. Đặc tả use case điểm số

### 2.4.6. Đặc tả use case thoát trò chơi

|  |  |
| --- | --- |
| **Use Case** | Thoát trò chơi |
| **Mô tả** | Người chơi thoát trò chơi |
| **Tác nhân** | Người dùng. |
| **Tiền điều kiện** | Người dùng khởi động game và vào game thành công |
| **Dòng hành động chính** | Người chơi bấm nút thoát |
| **Dòng hành động thay thế** |  |

Bảng 2.8. Đặc tả use case thoát trò chơi

# CHƯƠNG 3. THIẾT KẾ

## **3.1. Biểu đồ tuần tự**

### 3.1.1. Biểu đồ tuần tự chức năng chọn chế độ chơi

A diagram of a diagram

Description automatically generated



Hình 3.1. Biểu đồ tuần tự chức năng chọn chế độ chơi

### 3.1.2. Biểu đồ tuần tự chức năng cài đặt

A diagram of a person with a diagram

AI-generated content may be incorrect.



Hình 3.2. Biểu đồ tuần tự chức năng cài đặt

### 3.1.3. Biểu đồ tuần tự chức năng di chuyển nhân vật, thu thập vật phẩm và điểm số

A diagram of a person with a stick figure

Description automatically generated



Hình 3.3. Biểu đồ tuần tự chức năng di chuyển, thu thập vật phẩm và cập nhật điểm số

## **3.2. Biểu đồ hoạt động**

3.2.1. Biểu đồ hoạt động use case cài đặt   
A diagram with black circles and red dots

AI-generated content may be incorrect.

Hình 3.4. Biểu đồ hoạt động chức năng cài đặt

### 3.2.2. Biểu đồ hoạt động use case chọn màn chơi

A diagram of a diagram

Description automatically generated

Hình 3.5. Biểu đồ hoạt động chức năng chọn màn chơi

### 3.2.3. Biểu đồ hoạt động use case di chuyển nhân vật, thu thập vật phẩm và hiển thị điểm số

A screenshot of a computer screen

AI-generated content may be incorrect.

Hình 3.6. Biểu đồ hoạt động use case di chuyển nhân vật, thu thập vật phẩm và hiển thị điểm số

# CHƯƠNG 4: XÂY DỰNG TRÒ CHƠI

## **4.1. Quá trình thiết kế trò chơi**

### 4.1.1. Lên Kế Hoạch và Thiết Kế

Xác định ý tưởng trò chơi, xây dựng kịch bản và thiết kế nhân vật, môi trường, và cơ chế gameplay.

### 4.1.2. Phát Triển Game với Unity

Sử dụng Unity để tạo các cấp độ, lập trình gameplay bằng C#. Tạo hình ảnh và hiệu ứng âm thanh để nâng cao trải nghiệm người chơi.

### 4.1.3. Kiểm Tra và Tinh Chỉnh

Thực hiện kiểm tra toàn diện để phát hiện lỗi và tối ưu hóa hiệu suất. Nhận phản hồi từ người chơi thử nghiệm để cải thiện gameplay.

### 4.1.4. Triển Khai và Bảo Trì

Triển khai trò chơi lên các nền tảng mong muốn. Thực hiện bảo trì và cập nhật thường xuyên để nâng cao trải nghiệm người chơi và giải quyết các vấn đề phát sinh.

## **4.2. Giới thiệu giao diện trò chơi**

### 4.2.1. Màn hình đăng nhập, đăng ký

Màn hình đăng nhập: cung cấp cho người chơi khả năng truy cập vào tài khoản của mình bằng cách nhập email và mật khẩu, gồm các đặc điểm chính sau:

* **Thiết kế nền**: Phần nền có phong cách platform đi cảnh với các yếu tố như cây cối, bầu trời trong xanh, và những đám mây.
* **Phong cách thiết kế**: Phong cách đồ họa pixel art kết hợp với các yếu tố thiên nhiên mang lại cảm giác gần gũi và vui nhộn. Những yếu tố này đặc trưng cho các trò chơi platform cổ điển, giúp người chơi dễ dàng nhận diện giao diện quen thuộc.

A screenshot of a video game

AI-generated content may be incorrect.

Hình 4.1. Giao diện game

* **Thiết kế nền**: Bối cảnh nền vẫn giữ phong cách đi cảnh với cảnh quan thiên nhiên gồm cây cối, bầu trời xanh và đám mây, cùng với các viên gạch đỏ phía dưới. Đây là một phong cách quen thuộc, gợi nhớ đến các trò chơi platform cổ điển như *Mario*.
* **Biểu tượng trong trò chơi**: Ở phía bên trái, có hai khối hộp với dấu hỏi, giống như các biểu tượng trong các trò chơi *Mario*, cho thấy yếu tố tương tác trong trò chơi.
* **Thiết kế**: Phong cách đồ họa pixel cổ điển, thân thiện và vui nhộn, tiếp tục mang lại cảm giác hoài niệm, dễ nhận biết và dễ sử dụng cho người chơi.

### 4.2.2. Màn hình trước khi chơi game

Màn hình trước khi chơi game có các đặc điểm chính sau:

* Thiết kế nền: Cảnh nền có màu sắc tươi sáng, theo phong cách pixel art cổ điển, gồm những nền tảng bằng đá và cỏ xanh, thác nước và bầu trời trong xanh.
* Các nút menu: Có ba nút ở trung tâm màn hình, được gắn nhãn “START,” “LEVEL” và “EXIT.” Đây là các lựa chọn điều hướng chính trong trò chơi.
* Thông tin người chơi: Ở góc trên bên phải, có dòng chữ “Người chơi: Khanh” hiển thị tên của người chơi. Bên cạnh đó là biểu tượng trái tim, có thể đại diện cho sức khỏe hoặc mạng sống của người chơi.
* Phong cách: Thiết kế tổng thể đơn giản nhưng hiệu quả, với bảng màu sáng, bắt mắt, phù hợp với phong cách retro. Các nút bấm lớn và dễ đọc, giúp việc tương tác trở nên dễ dàng hơn.

A video game screen with a pixelated character

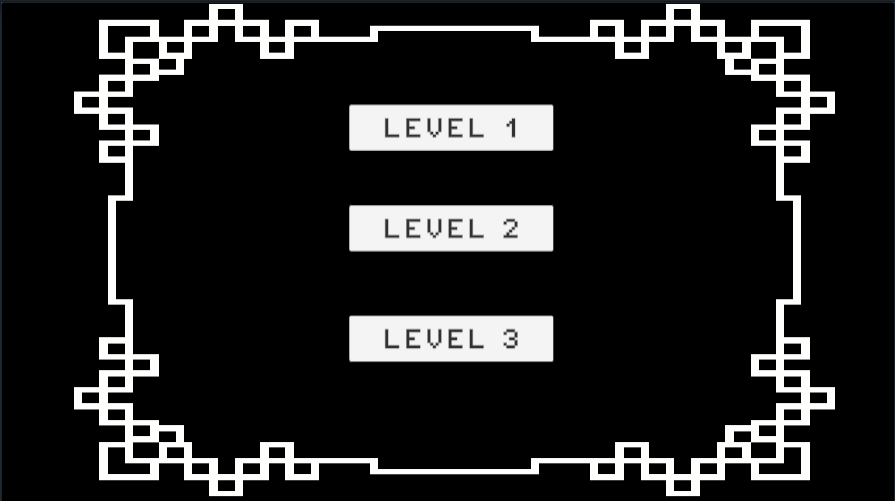
AI-generated content may be incorrect.

Hình 4.2. Giao diện chính trong game

### 4.2.3. Màn hình chọn cấp độ chơi game

Màn hình chọn màn chơi (level) của trò chơi với các đặc điểm chính sau:

* Thiết kế nền: Nền của giao diện là một cảnh quan nhiệt đới với các bụi cây và bầu trời xanh, tạo cảm giác vui vẻ và năng động
* Các lựa chọn cấp độ: Có ba khung hình đại diện cho ba cấp độ khác nhau: Level 1, Level 2, và Level 3.
* Mỗi khung hiển thị hình ảnh nhỏ của màn chơi tương ứng, giúp người chơi hình dung trước thử thách của từng cấp độ.
* Các cấp độ có sự khác nhau về bố cục và vị trí chướng ngại vật hoặc vật phẩm (như gai nhọn, bậc thang, v.v.).
* Nút quay lại: Ở góc trên bên trái là biểu tượng mũi tên quay lại, có thể để quay lại màn hình chính hoặc menu trước đó.
* Thiết kế trực quan: Các cấp độ được sắp xếp rõ ràng, dễ nhìn và dễ chọn. Mỗi level được đánh số và có hình ảnh thu nhỏ, giúp người chơi dễ dàng chọn cấp độ mình muốn chơi.



Hình 4.3. Giao diện chọn cấp độ chơi game

### 4.2.4. Màn hình chơi game

Màn hình chơi game có các đặc điểm chính sau:

* **Thiết kế bối cảnh**: khung cảnh là một khu vực núi non, với các cây xanh làm nền, tạo cảm giác dễ chịu và phù hợp với một trò chơi platform.
* **Nhân vật**: Nhân vật chính đang di chuyển qua một chướng ngại vật (có vẻ là những bẫy gai) và nhặt vật phẩm (có thể là quả dâu đỏ). Điều này gợi ý rằng người chơi có thể thu thập các vật phẩm trong quá trình vượt qua các thử thách trong màn chơi.

A video game screen with pixelated characters

AI-generated content may be incorrect.

Hình 4.4. Giao diện chơi game

### 4.2.5. Màn hình kết thúc màn chơi

Màn hình kết thúc màn chơi trong ảnh có các đặc điểm sau:

**- Thiết kế tổng thể**: Giao diện đơn giản với nền núi, cây cối và các chi tiết pixel thể hiện khung cảnh tự nhiên. Các nút chức năng được bố trí trên một khung chữ nhật màu vàng nhạt ở trung tâm.

**- Các nút chức năng**:

* **Resume (Tiếp tục)**: Cho phép người chơi quay trở lại chơi ngay tại vị trí dừng.
* **Play Again (Chơi lại)**: Để người chơi bắt đầu lại từ đầu màn chơi hiện tại.
* **Exit (Thoát)**: Thoát khỏi trò chơi hoặc về màn hình chính.

**- Mức độ dễ tiếp cận**: Các nút có kích thước lớn, dễ nhấn và được căn giữa trong khung, đảm bảo người chơi dễ dàng thao tác.

**- Biểu tượng**: Ở góc trên có hiển thị level (cấp độ), biểu tượng tạm dừng bên góc trái để quản lý trạng thái trò chơi, và một số hình quả dâu (có thể là điểm số hoặc vật phẩm thu thập).

A video game screen with a blue sign and black arrows

AI-generated content may be incorrect.

**Hình 4.6. Giao diện tạm dừng màn chơi**

# KẾT LUẬN

* Kết quả đạt được:
* **Thiết kế đồ họa**: Game của bạn đã thành công trong việc sử dụng phong cách pixel art cổ điển, mang lại cảm giác hoài niệm và dễ chịu cho người chơi. Nền cảnh quan tươi sáng, bắt mắt, và dễ nhận diện với những yếu tố đặc trưng của dòng game platformer.
* **Giao diện người dùng trực quan**: Các màn hình chính như menu, đăng nhập, và đăng ký đều được thiết kế rõ ràng, dễ sử dụng. Người chơi có thể dễ dàng thao tác và di chuyển qua các phần khác nhau của trò chơi.
* **Gameplay truyền thống**: Lối chơi đi cảnh, nhảy qua các nền tảng, và khám phá vật phẩm là điểm mạnh, thu hút người chơi yêu thích thể loại platformer. Cảm giác vượt chướng ngại vật và khám phá các hộp bí ẩn mang lại niềm vui và thử thách.
* **Tính ổn định và dễ tiếp cận**: Trò chơi chạy mượt mà, không gặp lỗi lớn, và dễ tiếp cận với đa dạng đối tượng người chơi, đặc biệt là những ai quen thuộc với các trò chơi cổ điển.
* Hạn chế:
* **Gameplay chưa đủ đa dạng**: Dù lối chơi thú vị, nhưng cần thêm các yếu tố mới như nhiều loại chướng ngại vật hoặc kẻ thù khác nhau để tăng thêm sự phong phú cho trò chơi.
* **Thiếu các tính năng nâng cao**: Hiện tại, trò chơi chưa có nhiều tính năng nâng cấp, hệ thống điểm số, hay cốt truyện để tạo động lực cho người chơi tiếp tục khám phá.
* Phát triển trong tương lai:
* **Bổ sung cấp độ và thử thách mới**: Tăng cường độ khó qua nhiều cấp độ khác nhau, với chướng ngại vật và vật phẩm đa dạng hơn, để kéo dài tuổi thọ trò chơi.
* **Cải tiến AI và cơ chế đối thủ**: Phát triển AI thông minh hơn cho các kẻ thù và tạo ra nhiều dạng chướng ngại vật có cơ chế phức tạp, tăng thử thách cho người chơi.
* **Thêm hệ thống điểm số và phần thưởng**: Tích hợp hệ thống điểm và phần thưởng để người chơi cảm thấy có mục tiêu và động lực chơi lại.
* **Mở rộng cốt truyện**: Bổ sung thêm các yếu tố cốt truyện hoặc phát triển nhân vật để làm cho người chơi gắn kết hơn với game.
* **Chế độ chơi nhiều người**: Tích hợp chế độ nhiều người chơi để tăng sự tương tác và cạnh tranh giữa các game thủ.

# TÀI LIỆU THAM KHẢO

*[1]* ***Hướng dẫn làm game Platformer với Unity***

*https://laptrinh.io/browse/tutorials/huong-dan-lam-game-don-gian-voi-unity*

*[2]* ***Tự học lập trình game platformer 2D*** *https://www.hoclaptrinh.vn/course/tu-hoc-lap-trinh-game-2d*

*[3]* ***Hướng dẫn làm game platformer với Godot*** *https://xuanthulab.net/lap-trinh-godot-phan-2-game-don-gian.html*

*[4]* ***Tạo game platformer cơ bản bằng Construct 2*** *https://www.howkteam.vn/course/construct-2-118/tao-game-don-gian-voi-construct-2-683*

* Giới hạn nền tảng: Mặc dù C# đã phát triển và hỗ trợ đa nền tảng thô
* Phát triển ứng dụng Desktop: C# được sử dụng rộng rãi để xây dựng