**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA: CÔNG NGHỆ THÔNG TIN**

-----------------

BÁO CÁO THỰC NGHIỆM THUỘC HỌC PHẦN: PHÁT TRIỂN ỨNG DỤNG GAME

**XÂY DỰNG GAME 2D “The Lost Fox”**

**GVHD: Ths. Vũ Đức Huy**

**Nhóm: 6**

**Sinh viên: Lưu Hải Long**

**Nguyễn Văn Hợp**

**Nguyễn Hoàng Giang**

**Lớp:2024IT6028003 Khóa: K16**

**Hà Nội - 2022**

**MỤC LỤC**

**DANH MỤC TỪ VIẾT TẮT**

**DANH MỤC HÌNH ẢNH**

**DANH MỤC BẢNG BIỂU**

# **MỞ ĐẦU**

**1.Giới thiệu về đề tài**

Trong bối cảnh sự bùng nổ của ngành công nghiệp game và sự phát triển mạnh mẽ của công nghệ kỹ thuật số, việc xây dựng và phát triển các ứng dụng game không chỉ là một lĩnh vực hấp dẫn mà còn là cơ hội để thử thách và rèn luyện kỹ năng học và làm việc nhóm. Vì vậy để bổ sung kiến thức cũng như vận dụng thực tiễn chúng em chọn đề tài *“Xây dựng game 2D The Lost Fox bằng Unity”* để làm đề tài cho thực nghiệm môn học Phát triển ứng dụng Game. Từ đây thông qua những kĩ thuật thiết kế cũng như xây dựng kịch bản và trải nghiệm nhập vai nhân vật

## **2.Nội dung nghiên cứu**

### **2.1. Giới thiệu về unity**

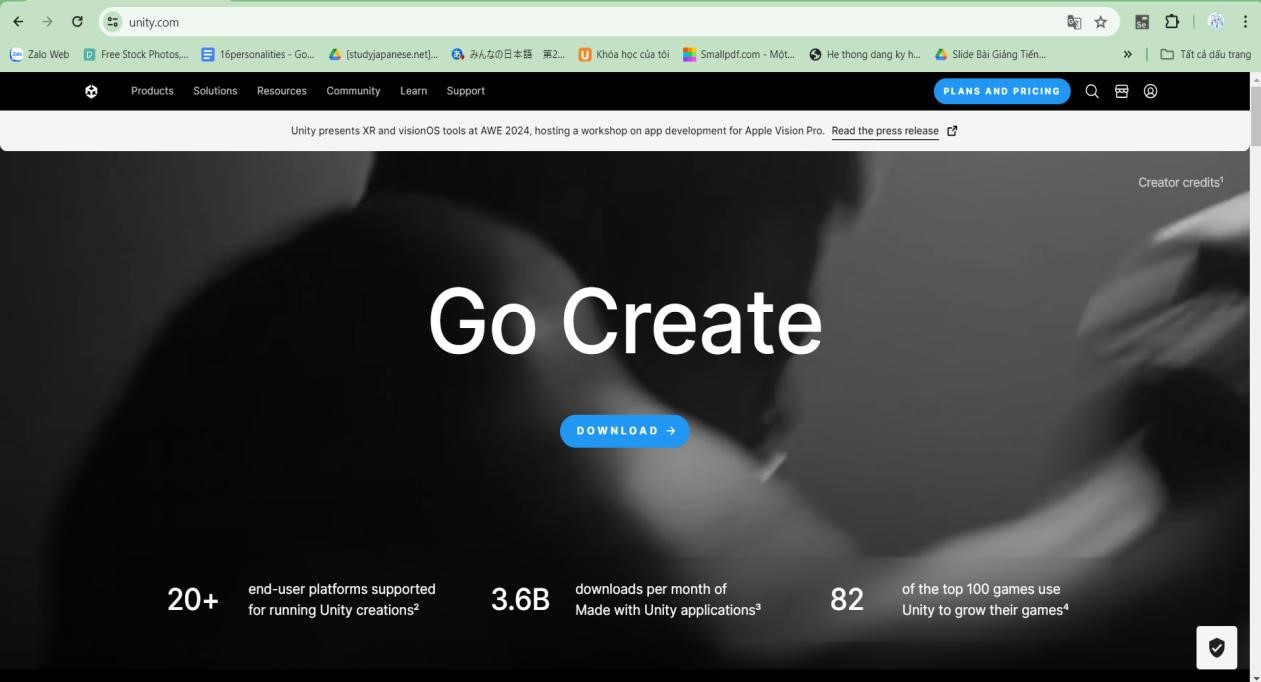


*Hình 1: Giao diện Unity*

Unity được dùng để làm video game, hoặc những nội dung có tính tương tác như thể hiện kiến trúc, hoạt hình 2D, 3D thời gian thực. Unity có thể tạo ra nhiều loại game 2D, 3D đa dạng, dễ sử dụng với người làm game chưa chuyên nghiệp, chất lượng cao, chạy hầu hết trên các hệ điều hành. Trong Unity chúng ta có thể import trực tiếp các file mô hình đa dạng thiết kế và sự thiết kế hoàn thiện tiếp nếu chưa xong trong khi đó công việc import chỉ diễn ra một lần. Không như việc phải dùng các công cụ khác để thực hiện viết game chúng ta sẽ phải xuất chúng ra một dạng nào đó và mỗi lần sửa lại phần mô hình chúng ta lại phải import lại, và như thế là quá mất thời gian trong việc tạo và chỉnh sửa các mô hình theo ý muốn. Ngoài ta Unity còn cho chúng ta trực tiếp tạo các mô hình nếu muốn, việc đặt các thuộc tính vật lý trong Unity cũng cực kỳ dễ dàng và hỗ trợ sẵn nhiều chức năng.

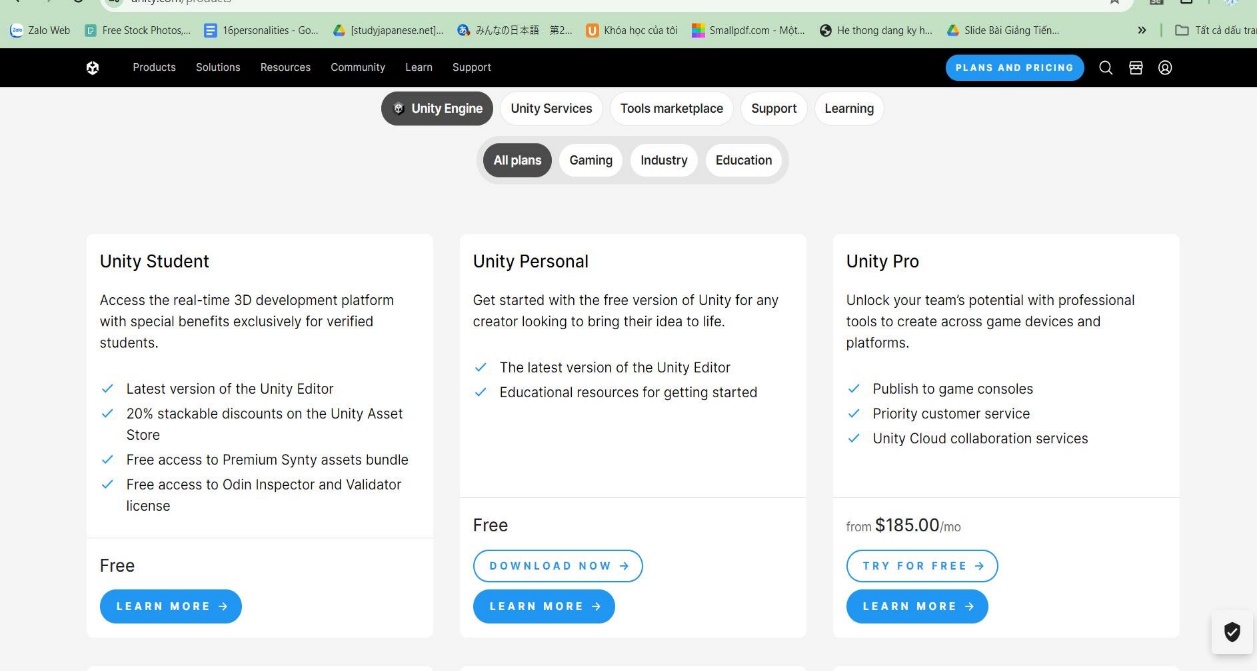
### **2.2. Cài đặt Unity**

Bước 1: Truy cập vào một trong các trình duyệt hiện nay và gõ từ khóa “Unity” và thành tìm kiếm, rồi sau đấy truy cập vào trang [https://www.unity.com](https://www.unity.com/)



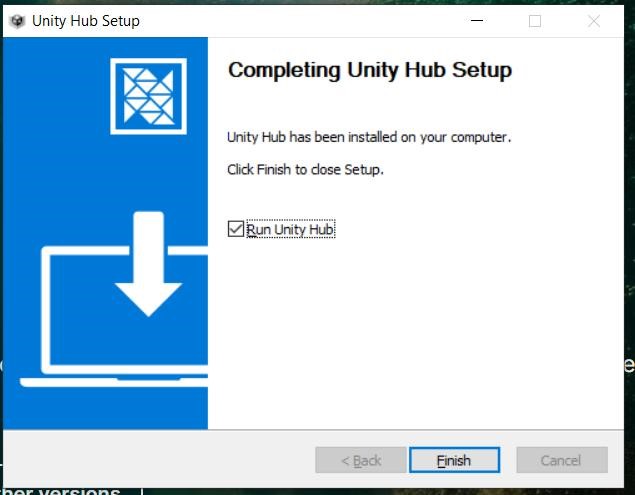
*Hình 2: Giao diện trang web* [*https://www.unity.com*](https://www.unity.com/)

Bước 2: Nhấn vào “Get started” trên thanh công cụ, cuộn chuột xuống và chọn mục Unity Personal.



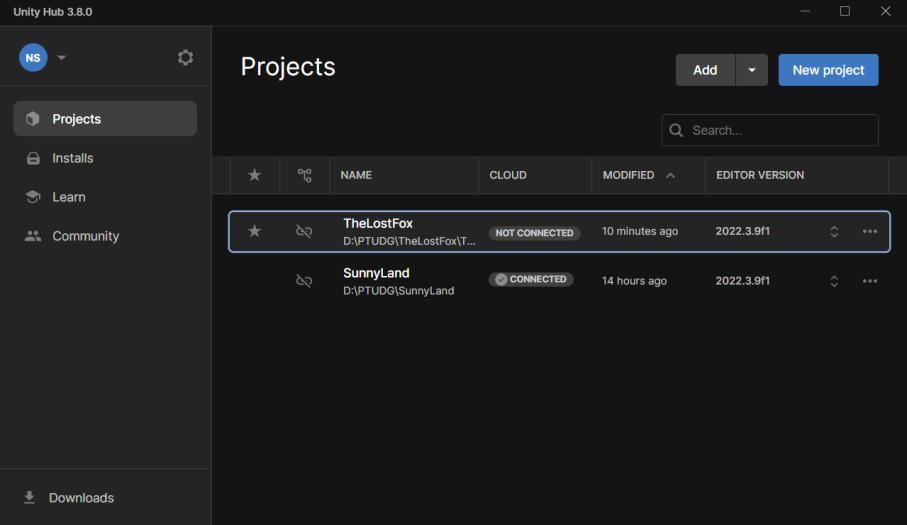
*Hình 3: Unity store*

Bước 3: Sau khi download thành công bạn hãy nhấn file .exe bên dưới để tiến hành cài đặt.



*Hình 4: Hoàn thành cài đặt Unity*

Bước 4: Sau khi cài đặt xong, màn hình Unity Hub hiện lên, Lúc này bạn cần tạo tài khoản bằng cách chọn “Create account”. Nếu đã có tài khoản unity thì bạn chọn “Sign in”. Sau khi đăng nhập xong, bạn cần cài đặt phiên bản Unity editor để sử dụng Unity, bạn chọn thư mục chứa rồi sau đó nhấn vào “Install Unity Editor”.



*Hình 5: Cài đặt thành công*

### **2.3. Các thành phần trong Unity**

* Assets: tài nguyên xây dựng nên một dự án trên Unity. Những tài nguyên có thể là hình ảnh, âm thanh, mô hình 2D, 3D, chất liệu

(material), texture,.. hoặc cả một project hoàn chỉnh.

* Scenes: Trong Unity, một cảnh chơi (hoặc một phân đoạn) là những màn chơi riêng biệc, một khu vực trong game hoặc thành phần có trong nội dung của trò chơi (các menu). Các thành phần này được gọi là Scene. Bằng các tạo ra nhiều Scenes, chúng ta có thể phân phối thời gian và tối ưu tài nguyên, kiểm tra các phân đoạn trong game một cách độc lập.
* Game Object: Khi Asset được sử dụng trong các Scene, Unity định nghĩa đó là Game Object. Là một thuật ngữ thông dụng, đặc biệt trong mảng lập trình. Tất cả các Game Object đều chứa ít nhất một thành phần cơ bản là Transform, lưu trữ thông tin về vị trí, góc xoay và tỉ lệ của Game Object. Thành phần Transform có thể được tùy biến và chỉnh sửa trong quá trình lập trình.
* Components: là các thành phần trong game, bổ sung tính năng cho các Game Object. Mỗi Component có chức năng riêng biệt. Đa phần các Component phụ thuộc vào Transform, vì nó lưu trữ các thông số cơ bản của Game Object. Bản chất của Game Object là không có gì cả, các đặc tính và khả năng của Game Object nằm hoàn toàn trong các Component. Do đó chúng ta có thể xây dựng nên bất kỳ Game Object nào trong game mà chúng ta có thể tưởng tượng được.
* Scripts: là thành phần thiết yếu trong quá trình phát triển game. Unity cung cấp cho lập trình viên khả năng viết Script bằng ngôn ngữ:

JavaScript, C#. Ở đây chúng ta viết C# với Visual Studio 2022.

* Prefabs: thực chất là Game Object được lưu trữ lại để tái sử dụng. Các Game Object được nhân bản từ một prefab sẽ giống nhau hoàn toàn, ngoại trừ thành phần Transform để phân biệt và quản lý được tốt hơn. Để tạo prefabs ta đơn giản chỉ cần kéo một Game Object vào cửa sổ Project.
* Collider: Hệ thống xử lý va chạm bao gồm 2D và 3D. Được chia ra các va chạm của các hình cơ bản:

+ Box collider: Va chạm cho các vật hình hộp chữ nhật đối với 3D và hình chữ nhật đối với 2D. Với Is Trigger: dạng true/ false, cho phép va chạm có đi xuyên qua không?

* Rigidbody: Hệ thống mô phỏng vật lý trong game.

Use Gravity: Nếu được kích hoạt, các đối tượng bị ảnh hưởng bởi lực hấp dẫn.

Is Kinematic: Nếu được kích hoạt, các đối tượng sẽ không được thúc đẩy bởi động cơ vật lý.

Collision detection: Được sử dụng để ngăn chặn các đối tượng chuyển động nhanh qua các đối tượng khác mà không phát hiện va chạm.

* Sprite: là một hình ảnh 2D của game, có thể là hình ảnh đầy đủ hoặc cũng có thể là một bộ phận nào đó. Unity cho phép tùy chỉnh màu sắc, kích thước, độ phân giải của một hình ảnh 2D.
* Animator: là tập một hình ảnh động dựa trên sự thay đổi liên tục của nhiều sprite khách nhau.
* Audio Source: nơi chứa các tệp âm thanh của game.
* Camera: là một game object đặc biệt trong scene, dùng để xác định tầm nhìn, quan sát các đối tượng khác trong game.
* Transform: quản lý object trong không gian ba chiều, theo ba thông số:

+ Position: quản lý vị trí hiện tại của Object

+ Rotation: quản lý các thông số quay của object theo các trục x, y, z + Scale: quản lý các thông số phóng to, thu nhỏ theo các trục x, y, z.

* Renderer: Sorting Layer, Order in Layer, Color, Material, Sprite.

### **2.4. Giới thiệu về Visual Studio 2022**



*Hình 6: Giao diện visual studio 2022*

Visual studio là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Năm 1997, phần mềm lập trình nay có tên mã Project Boston. Nhưng sau đó, Microsoft đã kết hợp các công cụ phát triển, đóng gói thành sản phẩm duy nhất.

Visual Studio là hệ thống tập hợp tất cả những gì liên quan tới phát triển ứng dụng, bao gồm trình chỉnh sửa mã, trình thiết kế, gỡ lỗi. Tức là, bạn có thể viết code, sửa lỗi, chỉnh sửa thiết kế ứng dụng dễ dàng chỉ với 1 phần mềm Visual Studio mà thôi. Không dừng lại ở đó, người dùng còn có thể thiết kế giao diện, trải nghiệm trong Visual Studio như khi phát triển ứng dụng Xamarin, UWP bằng XAML hay Blend vậy.

### **2.3. Tính năng**

Tính đến nay, Visual Studio vẫn được coi là phần mềm lập trình hệ thống hàng đầu, chưa có phần mềm nào có thể thay thế được nó. Được đánh giá cao như vậy bởi Visual Studio sở hữu nhiều tính năng cực kỳ hấp dẫn. Cụ thể:

* Đa nền tảng: Phần mềm lập trình Visual Studio của Microsoft hỗ trợ sử dụng trên nhiều nền tảng khác nhau. Không giống như các trình viết code khác, Visual Studio sử dụng được trên cả Windows, Linux và Mac

Systems. Điều này cực kỳ tiện lợi cho lập trình viên trong quá trình ứng dụng.

* Đa ngôn ngữ lập trình: Visual Studio cũng cho phép sử dụng nhiều ngôn ngữ lập trình khác nhau từ C#, F#, C/C++, HTML, CSS, Visual Basic, JavaScript,… Bởi vậy, VisualStudio có thể dễ dàng phát hiện và thông báo cho bạn khi các chương trình có lỗi.
* Hỗ trợ website: Visual Studio code cũng hỗ trợ website, đặc biệt trong công việc soạn thảo và thiết kế web.
* Kho tiện ích mở rộng phong phú: Mặc dù Visual Studio có hệ thống các ngôn ngữ hỗ trợ lập trình khá đa dạng. Nhưng nếu lập trình viên muốn sử dụng một ngôn ngữ khác, bạn có thể dễ dàng tải xuống các tiện ích mở rộng. Tính năng hấp dẫn này được hoạt động như một phần chương trình độc lập nên không lo làm giảm hiệu năng của phần mềm.
* Lưu trữ phân cấp: Phần lớn các tệp dữ liệu đoạn mã của Visual Studio đều được đặt trong các thư mục tương tự nhau. Đồng thời, Visual Studio cũng cung cấp một số thư một cho các tệp đặc biệt để bạn lưu trữ an toàn, dễ tìm, dễ sử dụng hơn.
* Kho lưu trữ an toàn: Với Visual Studio, bạn có thể hoàn toàn yên tâm về tính lưu trữ, bởi phần mềm đã được kết nối GIT và một số kho lưu trữ an toàn được sử dụng phổ biến hiện nay.
* Màn hình đa nhiệm: Visual Studio sở hữu tính năng màn hình đa nhiệm, cho phép người dùng mở cùng lúc nhiều tập tin, thư mục dù chúng có thể không liên quan tới nhau.
* Hỗ trợ viết code: Khi sử dụng code vào trong lập trình, với Visual Studio, công cụ này có thể đề xuất tới các lập trình viên một số tùy chọn thay thế nhằm điều chỉnh đôi chút để đoạn code áp dụng thuận tiện hơn cho người dùng.
* Hỗ trợ thiết bị đầu cuối: Phần mềm Visual Studio cũng tích hợp các loại thiết bị đầu cuối, giúp người dùng không cần chuyển đổi giữa hai màn hình hay trở về thư mục gốc khi thực hiện một thao tác cần thiết nào đó.
* Hỗ trợ Git: Do kết nối với GitHub nên Visual Studio cho phép hỗ trợ sao chép, kéo thả trực tiếp. Các mã code này sau đó cũng có thể thay đổi và lưu lại trên phần mềm.
* Intellisense: Tính năng nhắc Intellisense được sử dụng hầu hết trong các phần mềm lập trình, bao gồm cả Visual Studio. Tuy nhiên, so với các trình viết mã, Visual Studio vẫn được đánh giá cao về tính chuyên nghiệp. Đặc biệt, tính năng này còn có thể phát hiện tất cả các đoạn mã không đầy đủ, nhắc lập trình viên, gợi ý sửa đổi, khai báo biến tự động trong trường hợp lập trình viên quên, giúp bổ sung cú pháp còn thiếu,…
* Tính năng comment: Một tính năng cũng khá hay ho, hỗ trợ cho người lập trình trong trường hợp “nhớ nhớ quên quên” đó là tính năng bình luận. Tính năng này cho phép lập trình viên để lại nhận xét, giúp dễ dàng ghi nhớ công việc cần hoàn thành, không bỏ sót công đoạn nào.

# **CHƯƠNG 1. THIẾT KẾ Ý TƯỞNG GAME**

* 1. **Giới thiệu**

The Lost Fox là một trò chơi 2D thuộc thể loại nhập vai, phiêu lưu. Đồ họa được thiết kế đơn gian với đồ họa pixel, tạo cảm giác cổ điển nhưng thân thuộc với người chơi. Được phát triển cho nền tảng thiết bị PC hệ điều hành Window.

* 1. **Thể loại game**

Đây là một tựa game 2D thuộc thể loại nhập vai, phiêu lưu. Trong đó, người chơi tương tác với nhân vật thông qua góc nhìn ngang, điều khiển nhân vật thông qua các thao tác di chuyển trái, phải, nhảy và tấn công.

Yếu tố game mang lại: Tăng khả năng quan sát và sự nhanh nhẹn, cốt truyện hấp dẫn, sự bí ẩn của các màn chơi và đồng thời cũng mang tính giải trí.

* 1. **Tóm tắt game**

Hành trình của Cáo kéo dài trong 5 màn chơi, tương ứng với mỗi màn là một khu rừng với những khó khăn, thử thách, bài học, bạn đồng hành, kẻ thù khác nhau. Trò chơi kết thúc với việc Cáo tìm được đường về nhà.

* 1. **Khách hàng mục tiêu**

Trò chơi được phát triển cho những độ tuổi trẻ để giải trí, thư giãn sau những phút học tập, làm việc mệt mỏi. Với lối chơi đơn giản bằng các thao tác di chuyển và luật chơi dễ hiểu. Độ tuổi tối thiểu để có thể thao tác game phù hợp là từ trên 14 tuổi.

* 1. **Điểm mạnh của game**

Tựa game kết hợp giữa yếu tố phiêu lưu và giải đố, tạo nên một trải nghiệm chơi game phong phú và đa dạng, Người chơi phải sử dụng trí thông minh và sự khéo léo để vượt qua các màn chơi, khiến họ không thể rời mắt khỏi màn hình.

* 1. **Phong cách nghệ thuật game**

Với phong cách đồ họa pixel 2D, “The Lost Fox” mang đến một thế giới đầy màu sắc và chi tiết, mỗi màn chơi được thiết kế tỉ mỉ, từ những khu rừng rậm rạp đến những hang động tối tăm tạo nên một không gian sống động và hấp dẫn.

* 1. **Thiết bị trải nghiệm game**

Game được xây dựng bằng Unity nên có khả năng phát triển tốt trên các nền tảng lớn. Nhưng hiện tại game tập trung chủ yếu vào nền nảng Window(Laptop và PC)

# **CHƯƠNG 2. THIẾT KẾ PHÁT TRIỂN GAME**

# **2.1. Thiết kế kịch bản game**

### **2.1.1. Cách chơi chính**

Trong tựa game “The Lost Fox”, người chơi sẽ thực hiệt một loạt các hành động như tìm kiếm chìa khóa, vượt qua chướng ngại vật, chiến đấu với yêu quái, thu thập đạn để chiến đấu với yêu quái, giải mã những câu đố để tìm ra được chìa khóa, việc khám phá bản đồ để tìm ra lối đi đến đích.

**2.1.2. Cốt chuyện của game**

The Lost Fox sử dụng nghệ thuật kể chuyện 3 hồi, lấy cảm hứng từ điện ảnh “Con cáo đi lạc” giúp người chơi dễ dạng nắm bắt được nội dung cốt truyện game. Gồm 3 quá trình là các giai đoạn mở đầu, phát triển và kết thúc.

Hồi 1: Mở đầu và thiết lập hoàn cảnh, giới thiệu nhân vật: (Level 1)

Người chơi được giới thiệu về cáo qua các hoạt ảnh ngắn cho thấy cáo là sinh vật ham chơi thích theo đuổi những điều kỳ lạ. Câu chuyện bắt đầu khi Cáo bắt đầu đuổi theo 1 con bướm có màu sắc kì lạ đến một nơi xa. Chỉ khi Cáo nhận ra mình bị lạc cậu mới ý thức được mình cần tìm đường trở lại khu rừng. Hồi 1 của trò chơi được gói gọn trong màn chơi đầu tiên.

Hồi 2: Phát triển nhân vật

Trong cơn hoảng loạn và xen lẫn nỗi nhớ nhà, Cáo vội vã tìm đường quay về nhưng đáng lo là cậu đã đi nhầm theo một con đường khác nhưng chưa từng đi qua trước đó. Đầu tiên là khu rừng âm u, nơi đầy rẫy các quái vật ong, ốc sên mang độc. Chúng đều gây hại cho cậu khi va phải. Cáo thoạt đầu có vẻ sợ sệt nhưng bằng tài lanh và mưu trí cáo đã tiêu diệt được một vài kẻ thù và vượt qua nơi này một cách an toàn. Nhưng cáo sớm nhận ra bản thân vẫn chưa an toàn khi cậu vẫn bị mắc kẹt trong khu rừng khác các quái vật ong và ốc sên mang độc xuất hiệt càng lúc càng dày đặc. Dù cho đã có kinh nghiệm ở lần đối đầu trước nhưng với số lượng lớn kẻ địch như vậy cáo cũng rơi vào một tình thế khó khăn. Thật may thay bằng bản lĩnh và sự khôn ngoan cáo lại một lần nữa vượt qua được vòng vây của kẻ thù và thực sự thoát khỏi khu rừng âm u.

Sau cuộc chiến vừa rồi Cáo đặt chân đến thung lũng vực thẳm. Nơi đây không chỉ nhiều quái vật mà bên dưới chân cáo là cả một vực thẳm, một khi xểnh chân, Cáo sẽ rơi xuống và vĩnh viễn không thể trở về. Trên hành trình vượt qua thử thách lần này Cáo gặp được một người bạn đồng hành là một chú thỏ, vốn thông thạo địa hình này. Thỏ đã giúp Cáo định hình đường đi và băng qua cánh rừng, tiếp tục hành trình.

Cáo chợt nhận ra trời đã tối, khu rừng lại càng thêm âm u, đáng sợ hơn. Lúc này Cáo đã trưởng thành hơn được phần nào nên việc vượt qua nơi này đối với Cáo cũng không quá khó khăn. Nhanh chóng tiêu diệt địch.

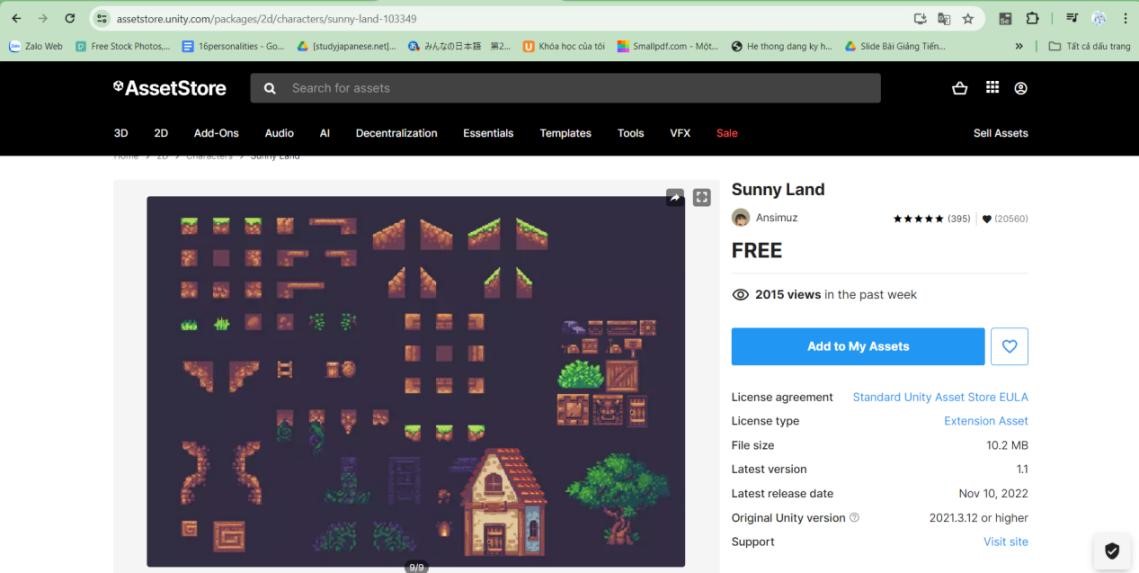
Hồi 3: Kết thúc

Khi tất cả kết thúc, ánh sáng đầu tiên của bình minh hiện lên, Cáo chợt nhận ra cảnh quen thuộc. Sau cuộc hành trình dài, Cáo cũng đã trở về được nơi mà cậu vốn thuộc về. Cà rốt, lương thực có ở khắp mọi nơi. Cáo tung tăng chạy nhảy, gặp lại bầy đàn, Cáo bắt đầu kể lại câu chuyện về hành trình mà bản thân cậu đã trải qua. Trò chơi kết thúc trong sự vỡ òa của Cáo. Đây cũng được gói gọn trong màn chơi cuối của game.

### 2.1.3. Các phần tử của game (Game Elements)

Phần lớn, các tài nguyên được sử dụng để làm game là lấy từ asset store và Open game Art: <http://assetstore.unity.com/> và <https://opengameart.org/>

Phần giao diện bản đồ map:



*Hình 7: Page asset store*

Sunny Land Forest Assets:



*Hình 8: Tài nguyên*



*Hình 9: Tài nguyên*

Sunny Land Artwork:



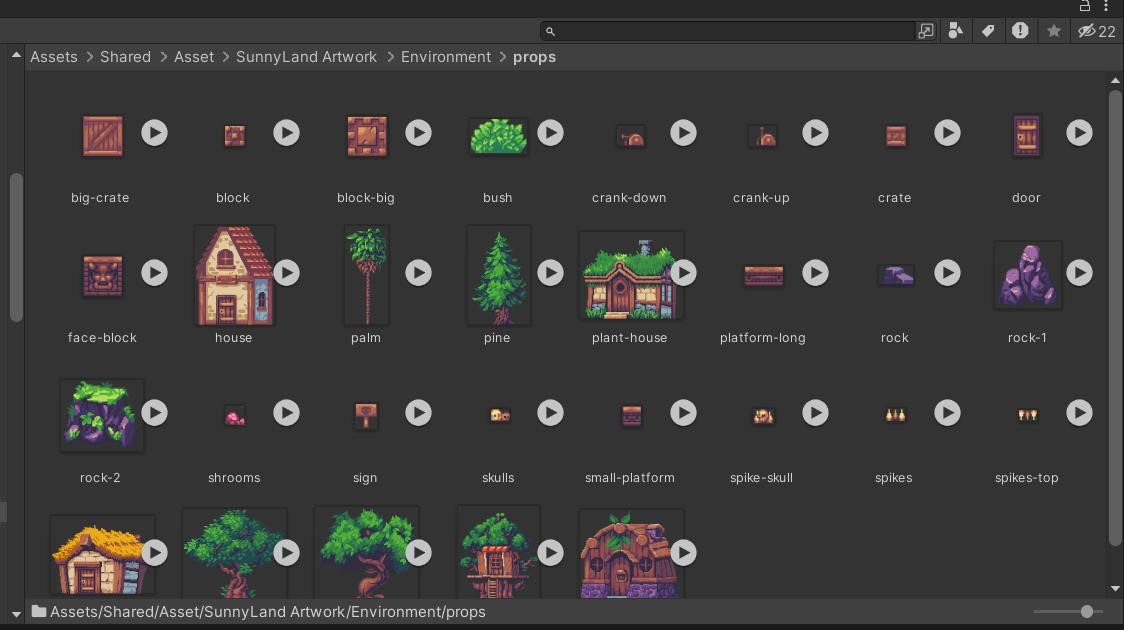
*Hình 10: Tài nguyên*

Giao diện cho màn hình Game Over:

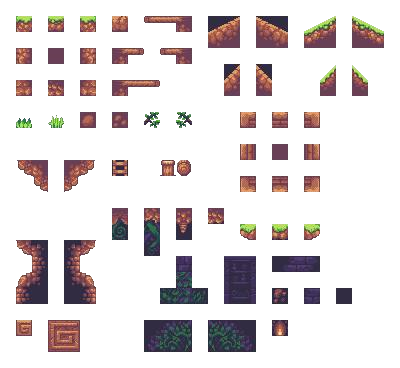


*Hình 11: Tài nguyên game over*

Sunny Land Expansion Trees:

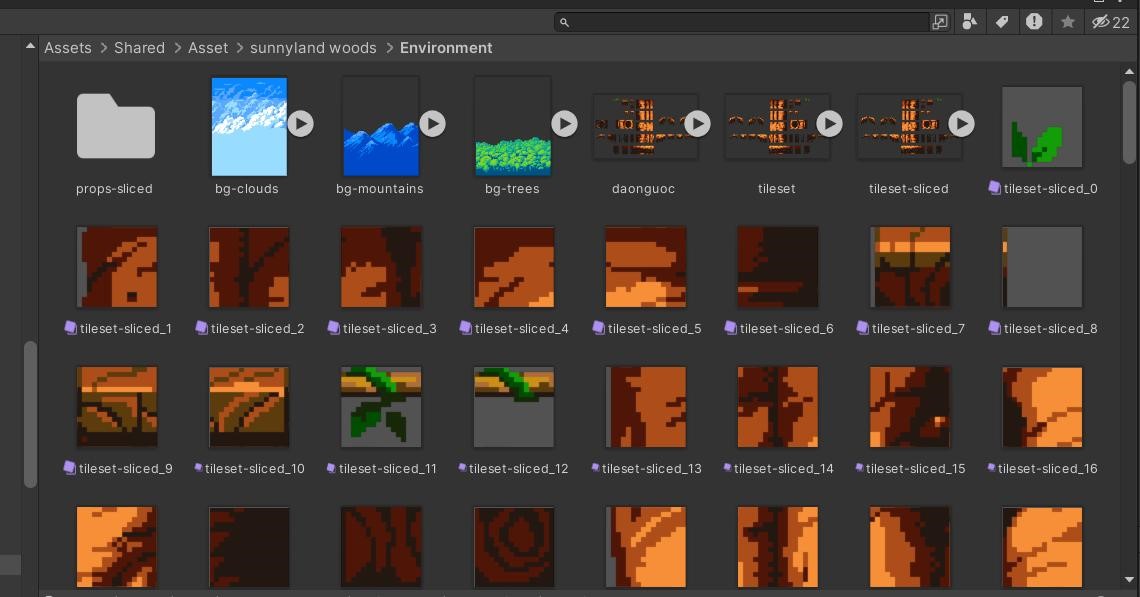


*Hình 12: Tài nguyên màn 3*

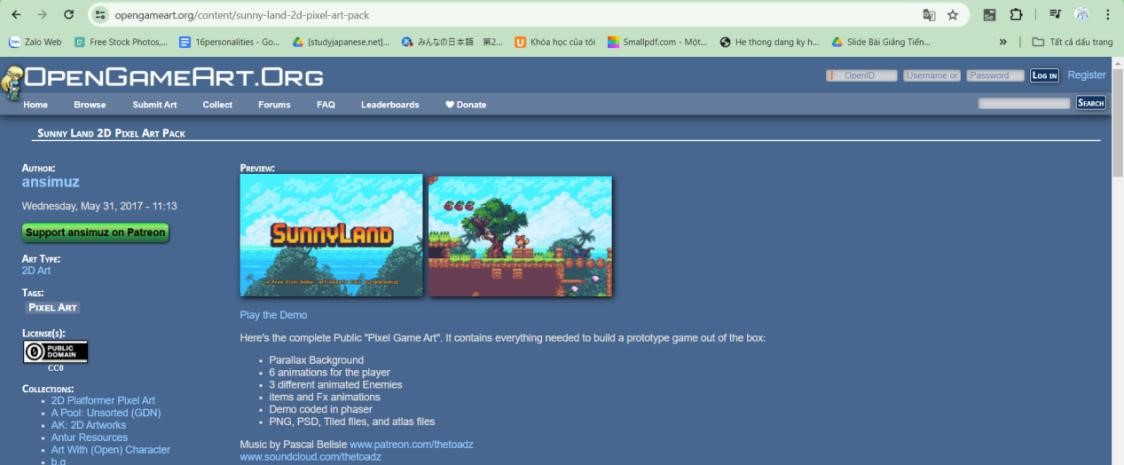


*Hình 13: Tài nguyên màn 3*

Sunny Land woods: <https://ansimuz.itch.io/sunnyland-forest>



*Hình 14: Tài nguyên màn 3*



*Hình 15: Page Open Game Art*

Các phần tử trong game:

|  |  |  |
| --- | --- | --- |
| **Phần tử trong game** | **Vai trò** | **Cách tương tác với môi trường** |
| Cáo | Là nhân vật chính, do  người chơi điều khiển | Di chuyển lên, xuống,  Trái (A), Phải(D), tấn công các thực thể khác |
| Gấu | Là người canh gác cổng ở màn 2. Lượng máu là 20 | Khả năng: phun lửa (hỏa thuật), rung chấn, đều gây 2 sát thương cho Cáo |
| Ốc sên | Là kẻ địch của người chơi | Di chuyển qua lại, gây 1 sát thương khi va phải |
| Ong | Là kẻ địch của người chơi | Là thực thể bay, di chuyển qua lại, gây 1 sát thương khi va chạm |
| Gai độc | Là chướng ngại vật cản  đường người chơi | Vị trí tùy chỉnh, không di chuyển, gây 1 sát thương khi  va chạm |

|  |  |  |
| --- | --- | --- |
| Tiên thỏ | Là nhân vật NPC, hướng dẫn người chơi | Thực hiện các cuộc thoại với cáo khi va chạm |
| Cà rốt | Là vật phẩm có thể nhặt, giúp cáo hồi lại lượng máu đã mất | Giúp hồi 1 máu (tim, mạng) khi chạm |
| Sao | Là vật phẩm tích lũy năng lượng cho người chơi | Được nhặt khi va chạm |
| Chìa khóa | Là vật phẩm nhặt, giúp mở khóa các cánh cổng  (màn 2) | Có thể nhặt khi va chạm |
| Cổng | Là chướng ngại vật, cản người chơi | Mở khi thu thập được chìa khóa và thực hiện va chạm |
| Cầu treo cũ | Xuất hiện ở màn khu vực vực thẳm | Trở lên rung lắc sau khi người chơi đứng lên, sau 4s thì cầu vỡ |
| Quả cầu  hủy diệt | Được tạo ra bằng việc người chơi tấn công, khi có sao | Gây 1 sát thương cho địch |
| Hỏa cầu | Được phóng ra bởi gấu, khi đụng độ người chơi | Gây 2 sát thương cho người  chơi |
| Rung trấn | Được tạo ra khi Cáo ở gần gấu 1 khoảng | Gây 2 sát thương cho người chơi khi va chạm |
| Thỏ | Vô hại, được tạo theo cốt truyện | Có thể bị tiêu diệt bởi người chơi (Cáo) |

2.1.4. Các cơ chế của game (Game Mechanic – [1] MECHANICS, )

*- Game Mechanic là các phương thức kết hợp các yếu tố trong game (Game Element) để định nghĩa và bổ trợ cho Gameplay. Game Mechanic xác định trò chơi sẽ diễn ra như thế nào.*

*- VD: “Khi avatar đi vào đầm lầy, những con cóc đen bắt đầu thải ra một loại khí độc mà người chơi có thể nhìn thấy tràn ngập màn hình. Nó bắt đầu ở dưới cùng và tăng lên với tốc độ 1 inch cứ sau 3 giây; hết 3 phút, khí cao ngang mặt avatar, nếu lúc đó avatar vẫn ở trong đầm lầy thì avatar đó chết. Nếu avatar quay trở lại đầm lầy sau đó, khí sẽ hết nhưng quá trình sẽ bắt đầu lại từ đầu.”*

*- Tham khảo thêm cách viết của tài liệu [1]*

2.2. Thiết kế giao diện *(Tham khảo tài liệu [2])*

2.2.1. Biểu đồ - Flowchart

2.2.2. Mô tả

2.2.3. Giao diện các màn hình

*- Hình ảnh màn hình thiết kế*

*- Mô tả hoạt động của các đối tượng trên mỗi màn hình: các button, các thông số, …*

2.3. Thiết kế âm thanh

*- Sự kiện, tình huống A cần âm thanh A1🡪 tài nguyên cho âm thanh đó.*

CHƯƠNG 3. CÀI ĐẶT CHƯƠNG TRÌNH VÀ KẾT QUẢ

*Trình bày các kỹ thuật chính cài đặt chương trình và kết quả tương ứng.*

KẾT LUẬN

3.1. Kết quả đạt được

3.2. Hướng phát triển

TÀI LIỆU THAM KHẢO

[1] Tác giả 1, Tác giả 2, Tác giả 3 (2019), *Tên sách,*  Nhà xuất bản ABCDEFGH, Hà Nội.

[2] Tác giả 1 Tác giả 2 Tác giả 3 (2019), *Tên giáo trình,*  Nhà xuất bản ABCDEFGH, Hà Nội.

…

PHỤ LỤC