ĐẠI HỌC BÁCH KHOA HÀ NỘI

TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

KHOA HỌC MÁY TÍNH

--------------------o0o------------------



**BÁO CÁO PROJECT**

ĐỀ TÀI

LẬP TRÌNH GAME 2D BẰNG UNITY

Giáo viên hướng dẫn: Nguyễn Hồng Phương

Sinh viên thực hiện: Mã Thiên Lý – 20204582

TP. HÀ NỘI, THÁNG 07 NĂM 2023

**MỤC LỤC**

[LỜI MỞ ĐẦU 4](#_Toc140441808)

[CHƯƠNG 1. Giới thiệu 5](#_Toc140441809)

[**1.1. Giới thiệu** 5](#_Toc140441810)

[**1.2.Mục tiêu.** 5](#_Toc140441811)

[CHƯƠNG 2. Android platform. 7](#_Toc140441812)

[**2.1. Android.** 7](#_Toc140441813)

[**2.2. Lịch sử của Android.** 8](#_Toc140441814)

[CHƯƠNG 3. Unity Engine. 11](#_Toc140441815)

[**3.1. Giới thiệu.** 11](#_Toc140441816)

[**3.2. Sơ lược lịch sử hình thành và phát triển của Unity.** 12](#_Toc140441817)

[***3.2.1. Hình thành.*** 12](#_Toc140441818)

[***3.2.2. Giải thướng*** 12](#_Toc140441819)

[***3.2.3. Khách hàng*** 12](#_Toc140441820)

[**3.3. Tính năng của engine Unity** 13](#_Toc140441821)

[**3.4. Các khái niệm trong Unity** 13](#_Toc140441822)

[***3.4.1. Asset*** 13](#_Toc140441823)

[***3.4.2. Scene*** 14](#_Toc140441824)

[***3.4.3. Game Object*** 14](#_Toc140441825)

[***3.4.4.Component*** 14](#_Toc140441826)

[***3.4.5.Script*** 14](#_Toc140441827)

[***2.4.6.Prefab*** 14](#_Toc140441828)

[**3.5. Tổng quan kiến trúc engine Unity trên Android** 15](#_Toc140441829)

[***3.5.1. Kiến trúc tổng quan*** 15](#_Toc140441830)

[***2.6.2. Chu kỳ sống của thành phần script gắn trên đối tượng game*** 16](#_Toc140441831)

[CHƯƠNG 4. Game Project. 17](#_Toc140441832)

[**4.1.** **Tổng quan.** 17](#_Toc140441833)

[***4.1.1.*** ***Giới thiệu về sản phẩm***. 17](#_Toc140441834)

[***4.1.2. Mô tả.*** 17](#_Toc140441835)

[**4.2.** **Thiết kế của game.** 18](#_Toc140441836)

[***4.2.1.*** ***Biểu đồ use case.*** 18](#_Toc140441837)

[***4.2.2.*** ***Biểu đồ Activity Diagram.*** 18](#_Toc140441838)

[***4.2.3.*** ***Biểu đồ Class Diagram.*** 21](#_Toc140441839)

[**4.3.** **Sản phẩm hoàn thành.** 24](#_Toc140441840)

[***4.3.1.*** ***Giao diện bắt đầu game.*** 24](#_Toc140441841)

[***4.3.2.*** ***Giao diện maps.*** 26](#_Toc140441842)

[***4.2.3. Các màn chơi.*** 28](#_Toc140441843)

[Tổng kết 33](#_Toc140441844)

[Tài liệu tham khảo 34](#_Toc140441845)

# **LỜI MỞ ĐẦU**

Trong thời đại công nghệ hóa, hiện đại hóa, trò chơi số đã không còn quá xa lạ với mỗi người. Trò chơi số không chỉ giúp giải trí mà còn ứng dụng vào trong dạy học, tạo hứng thú học tập đồng thời củng cố kiến thức cho học sinh.

Ở nước ta hiện nay việc xây dựng 1 game không phải quá phức tạp tuy nhiên để tạo ra một trò chơi hay và chất lượng không phải dễ, nhất là một trò chơi mang tính giáo dục lại là một vấn đề nan giải. Để tạo ra một trò chơi phù hợp cho giáo dục, với học sinh mà vẫn phải hấp dẫn và hay thì người làm game cần hiểu rõ mục tiêu hướng tới cũng như ngôn ngữ, công cụ mà trò chơi mình sẽ làm. Vì vậy, chúng ta cần phân tích trò chơi để biết trò chơi đã đạt đủ yêu cầu cần có hay chưa cũng như dựa vào những phân tích hệ thống trò chơi để thiết kế một trò chơi hoàn thiện nhất.

Game của em có tên gọi là “Treasure Hunt”. Đây là một trò chơi có nhập vai phổ biến.

Em đã cố gắng tạo nên một trò chơi cũng như phân tích một trò chơi hoàn thiện nhất, tuy nhiên vẫn không thể bỏ qua những sai sót trong quá trình làm việc. Mong quý thầy cô và các bạn đóng góp và chỉnh sửa để giúp nhóm hoàn thiện hơn nữa. Chúng em xin chân thành cảm ơn!

# **CHƯƠNG 1. Giới thiệu**

## **1.1. Giới thiệu**

Android là một hệ điều hành phát triển dựa trên nhân Linux. Nhân linux là một phần quan trọng của hệ điều hành Ándroid và cung cấp các chức năng cốt lõi như quản lý quyền, giao tiếp mạng, quản lý bộ nhớ và nhiều tính năng khác. Android sử dụng cách nhập dữ liệu bằng cảm ứng màn hình (touch inputs) và hỗ trợ hành động faanf tương ứng với hoạt động trong thế giới thực, chẳng hạn như vuốt ngang (swiping), chạm (tapping), thu phóng (pinching) và giãn ngang (reverse pinching) để tương tác với các đối tượng trên màn hình. Ngoài ra, nó cũng hỗ trợ bàn phím ảo (vỉtual keyboard) cho việc nhập liệu. Mạc dù Android được thiết kế chủ yêu dành cho việc sử dụng màn hình cảm ứng, nhưng nó cũng đã được sử dụng trong các thiết bị khác như máy chơi game, máy ảnh kỹ thuật số, máy tính cá nhân thông thường và các thiết bị điện tử khác. Điều này cho thấy tính linh hoạt của Android và khả năng sử dụng rộng rãi trên nhiều nền tảng khác nhau.

Tính đến tháng 7 năm 2013, cửa hàng Google Play đã có hơn một triệu ứng dụng Android được xuất bản và hơn 50 tỷ ứng dụng được tải xuống [\*]. Một cuộc khảo sát từ tháng 4 đến tháng 5 năm 2013 của những nhà phát triển ứng dụng di động cho biết 71% trong số họ tạo ứng dụng cho Android. Ngoài ra, theo cuộc khảo sát năm 2015 khác, cho thấy rằng với những nhà phát triển chuyên nghiệp coi nền tảng Android là nền tảng “ưu tiên” hàng đầu. Tỷ lệ này cao hơn iOS(37%) và các nền tảng khác, cho thấy sự hấp dẫn và sự lựa chọn Android làm nền tảng chính cho phát triển ứng dụng.

Android là một hệ điều hành mã nguồn mở, được phát hành bởi Google dưới giấy phép mã nguồn mở. Điều này đã tạo điều kiện thuân lợi cho một cộng đồng lớn các nhà phát triển và người đam mê sử dụng mã nguồn mở của Android để thực hiện các dự án cộng đồng. Điều này có thể là thêm các tính năng mới cho người dùng nâng cao hoặc đư Android lên các thiết bị chạy hệ điều hành khác.

Android là hệ điều hành phổ biến nhất trên thế giới cho các thiết bị di động và máy tính bảng. Điều này tạo nên một cuộc cách mạng trong lĩnh vực công nghệ. Hầu hết các ứng dụng Android phổ biến đều hữu ích và giải trí cho người dùng. Vì vậy, tôi muốn tạo ra một game phát triển trên hệ điều hành Android.

## **1.2.Mục tiêu.**

Mục tiêu chính là phát triển một trò chơi có thể chạy trên bất kỳ thiết bị Android nào.

Thiết kế và phát triển các ứng dụng Android hữu ích với giao diện người dùng hấp dẫn, phát triển một trò chơi thân thiện với người dùng để giải trí cho người dùng.

Tìm hiểu về Unity để có thêm kiến thức chuẩn bị cho các project tiếp theo.

# **CHƯƠNG 2. Android platform.**

## **2.1. Android.**

Android là một trong những hệ điều hành phổ biến và được sử dụng rộng rãi nhất. Nó là một bộ công cụ phần mềm cho các thiết bị di động bao gồm hệ điều hành, middleware và các ứng dụng chính:

- Hệ điều hành (Operating System): Android sử dụng phiên bản đặc biệt của hệ điều hành Linux. Nó cung cấp các chức năng cơ bản như quản lý tài nguyên hệ thống, quản lý quá trình, giao tiếp với phần cứng và nhiều chức năng khác.

- Middleware: một tàng trung gian trong Android giữa hệ điều hành và các ứng dụng. Nó cung cấp các dịch vụ và giao diện cho các ứng dụng để tiếp cận các chức năng cao cấp như xử lý đa phương tiện, quản lý cơ sở dữ liệu, của lý sự kiện và giao tiếp mạng.

- Các ứng dụng chính (Key Applications): Android đi kèm với một loạt các ứng dụng chính, bao gồm trình duyết web, ứng dụng gọi điện thoại, ứng dụng tin nhắn, trình đa phương tiện và nhiều ứng dụng khác. Các ứng dụng này cung cấp các chức năng cơ bản và quan trọng cho người dùng.

Android là hệ điều hành di động nguồn mở, được chạy trên kernel của Linux. Nó cung cấp một môi trường phát triển linh hoạt cho việc xây dựng ứng dụng di động trên nền tảng Android. Người phát triển sử dụng Android SDK để tạo các ứng dụng Android, sử dụng JAVA để viết code và sử dụng API cung cấp bởi Android SDK để truy cập vào các tính năng của hệ điều hành Android. Nó sử dụng một máy áo Java không chuẩn được gọi là Dalvik , được đặc biệt hóa để xử lý các quy trình tên các thiết bị di động. Ngôn ngữ lập trình chính được sử dụng trong Android là Java, với nhiều thư viện từ J2SE và các dự án mã nguồn mở của bên thứ ba, chẳng hạn như Apache Commons, SQLite, Webkit, ... Google đã mua lại công ty phát triển ban đầu của phần mêm Android, Android Inc., vào năm 2005. Ngày 5 tháng 11 năm 2007, việc ra mắt phiên bản phân phối Android được công bố cùng với việc thành lập Ơpn Handset Alliance, một liên minh gồm 84 công ty sản xuất phần cứng, phần mềm và viễn thông, cam kết thúc đẩy các tiêu chuẩn mở cho các thiết bị di động. Google đã phát hành phần lớn mã nguồn dưới giấy phép Apache License, một giấy phép phần mềm miễn phí. Dự án mã nguồn mở Android được giao trách nhiệm với việc bảo trì và phát triển tiếp Android. Android bao gồm một nhân dựa trên nhân Linux, đi kèm với các middleware, thư viện và API được viết bằng ngôn ngữ C và phần mềm ứng dụng chạy trên một khung ứng dụng bao gồm các thư viện tương thích với Java dựa trên Apache Harmony. Các nhà phát triển chủ yêu viết mã bằng một phiên bản tùy chính của Java. Vào cuối 2010, Android trở thành nền tảng điện thoại thông minh hàng đầu thế giới. Trong quý đầu tiên năm 2012, Android chiếm 59% thị phần điện thoại thông minh trên toàn cầu. Đến giữa năm 2012, đã có 400 triều thiết bị được kích hoạt và 1 triệu kích hoạt hàng ngày. Các nhà phân tích chỉ ưu điểm của Android là việc nó là một hệ điều hành đa kênh, đa nhà mạng. Vào đầu năm 2013, Android chiếm 70% thị phần điện thoại thông minh trên toàn cầu.

## **2.2. Lịch sử của Android.**

Android, Inc. được thành lập tại Palo Alto, California, Hoa Kỳ vào tháng 10 năm 2003 bởi Andy Rubin, Rich Miner, Nick Sears và Chris Ưhite. Mục tiêu là phát triển các thiết bị di động thông minh hơn có khả năng theo dõi vị trí và sở thích của người dùng.

Ngày 17 tháng 8 năm 2005, Google đã mua lại Android Inc. biến Andrid Inc. thành một công ty con hoàn toàn thuộc sở hữu của Google. Các thành viên chủ chốt của Android Inc. đã ở lại công ty sau thương vụ mua lại này. Sau khi được mua lại bởi Google, đội ngũ do Rubin dẫn đầu đã phát triển một nền tảng thiết bị di động dựa trên hạt nhân Linux. Google đã tiếp thị nền tảng này và tạo ra mối quan hệ với các nhà sản xuất và nhà mạng. Vào tháng 9 năm 2007, InformationWeek đăng tin về một nghiên cứu từ Evacuee serve cho biết Google đã nộp đơn đăng ký bằng sáng chế trong lĩnh vực viễn thông di động.

Ngày 5 tháng 11 năm 2007, Open Handset Alliance ra đời với mục tiêu phát triển các tiêu chuẩn mở cho các thiết bị di động. Họ đã công bố sản phẩm đầu tiên của mình – nền tảng Android dựa trên hạt nhân Linux và thu hút được nhiều thành viên mới sau đó.

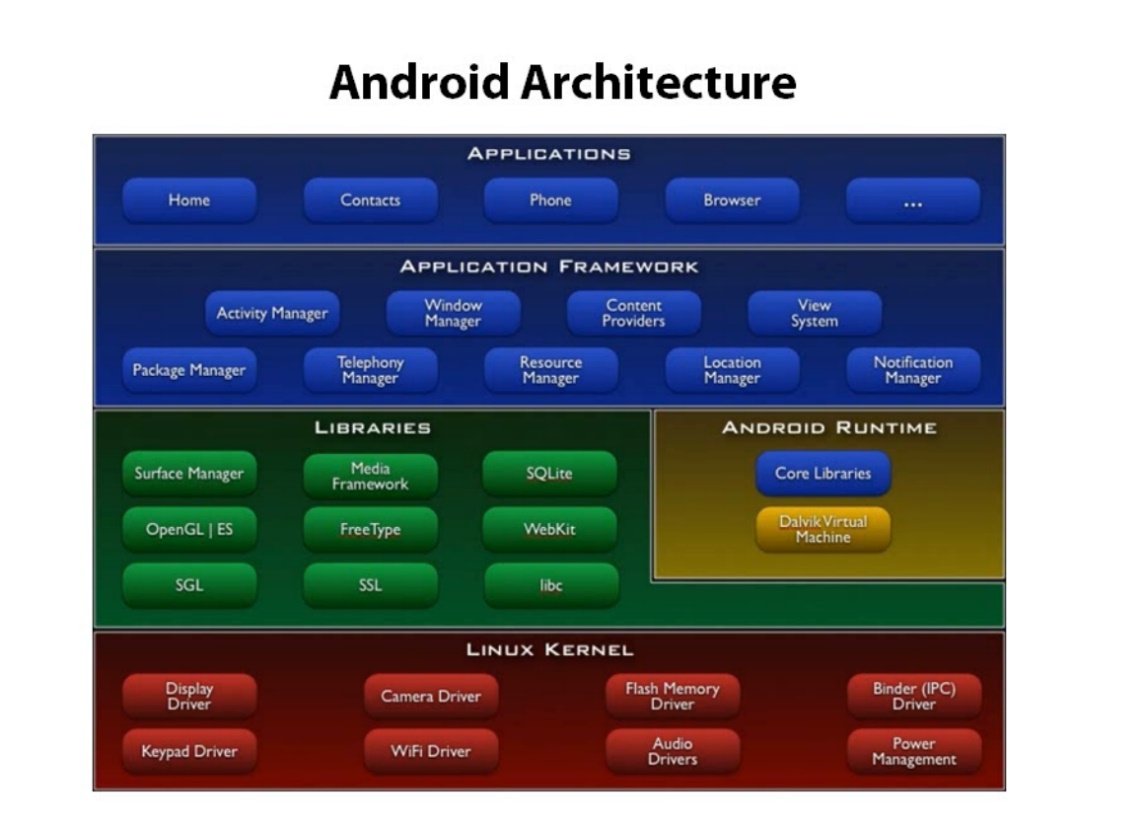
Dự án mã nguồn mở Android (AOSP dưới sự chỉ đạo của Google, chịu trách nhiệm bảo trì và phát triển Android. Mục tiêu chính của AOSP là tạo ra một sản phẩm thục tế thành công để cải thiện trải nghiệm di động cho người dùng cuối. AOSP cũng quản lý Chương trình tương thích Android, định nghĩa thiết bị “tương thích với Android” là thiết bị có thể chạy bất kỳ ứng dụng nào được phát triển bởi bên thứ ba sử dụng Android SDK và NDK. Chương trình này giúp ngăn chặn các phiên bản Android không tương thích.

Android được cập nhật thường xuyên kể từ phiên bản ban đầu là “Astro”, trong đó mỗi bản cập nhật sửa lỗi và thêm tính năng mới. Mỗi phiên bản được đặt tên theo thứ tự bản chữ cai, với 1.5 “Cupcake” là phiên bản đầu tiên được đặt theo tên một loại món tráng miệng, và mọi cập nhật sau này cũng tuân theo quy ước đặt tên này (đặt tên theo thứ tự bảng chữ cái và theo quy ước đặt tên từ một loại món tráng miệng). Một số phiên bản Android:

1. Cupcake
2. Donut
3. Eclair
4. Froyo
5. Gingerbread
6. Honeycomb
7. Ice Cream Sandwich
8. Android 4.2 Jelly Bean (API level 17)
9. Android 4.3 Jelly Bean (API level 18)
10. Android 4.4 Kit Kat (API level 19)
11. Android 5 Lollipop (API level 21)

Từ phiên bản Android 10, Google đã thay đổi cách đặt tên để trách nhầm lẫn và cải thiện tính nhận diện của phiên bản Android. Phiên bản mới nhất là Android 12 công bố tháng 10 năm 2021, cung cấp tính năng mới gồm giao diện người dùng được thiết kế lại, cải thiện bảo mật và quyền riêng tư, tích hợp công nghệ hình ảnh điều khiển tự nhiên, chế độ Hảo nguồn, tối ưu hiệu suất và nhiều tính năng tiện ích khác.

**2.3. Kiến trúc của Android.**



Kiến trúc của Android là một mô hình tổ chức và cấu trúc hệ thống của hệ điều hành Android. Kiến trúc này bao gồm các thành phần chính su đây:

- Lớp ứng dụng (Application Layer): đây là lớp mà người dùng cuối tương tác trực tiếp. Nó chứa các ứng dụng và giao diện người dùng (UI) mà người dùng có thể sử dụng trên thiết bị Android.

- Framework (khung chương trình): Lớp này chứa các API và dịch vụ hỗ trợ để phát triển ứng dụng Android. Nó bao gồm các thành phần như Activity Manager, Content Providers, Réource Manager, Location Manager, và các miền khác cung cấp các chức năng như quản lý vòng đời ứng dụng, quản lý dữ liệu, quản lý tài nguyên và vị trí.

- Lớp thư viện Android (Android Libraries): là một bộ các thư viện Java có sẵn cho các nhà phát triển, giúp xây dựng các ứng dụng Android. Điển hình là thư viện điều khiển giao diện (UI), thư viện đồ họa (Graphics), thư viện đa phương tiến (Media), và thư viện kết nối mạng (Networking).

- Lớp Runtime (Thời gian chạy): Android sử dụng môi trường thực thi Dalvik hoặc ART (Android Runtime) để chạy các ứng dụng. Nhờ sự tối ưu hóa, Dalvik và ART cho phép các ứng dụng Android chạy một cách hiệu quả trên các thiết bị có tài nguyên hạn chế. Máy ảo Dalvik thực thi các tệp trong định dạng Dalvik Executable (.dex), tối ưu hóa cho tổn thất bộ nhớ tối thiểu. Máy ảo được dựa trên đăng ký và chạy các lớp được biên dịch bởi trình biên dịch ngôn ngữ Java đã được chuyển đổi thành định dạng .dex bằng công cụ “dx” đi kèm. Máy ảo Dalvik phụ thuộc vào hạt nhân Linux để cung cấp các chức năng cơ bản như luồng và quản lý bộ nhớ cấp thấp.

- Hạt nhân Linux: Andoird sử dụng hạt nhân Linux làm lớp giao tiếp phần cứng và phần mềm. Hạt nhân Linux cung cấp các công cụ và các driver cần thiết cho việc quản lý giao tiếp và hoạt động của phần cứng trên thiết bị Android.

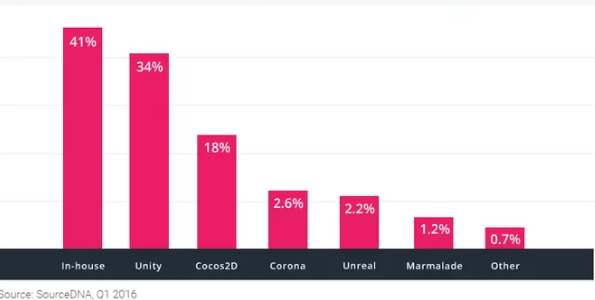
# **CHƯƠNG 3. Unity Engine.**

## **3.1. Giới thiệu.**

Unity là một “cross – flatform game engine” tạm hiểu là công cụ phát triển game đa nền tảng được phát triển bởi Unity Technologies. Game engine này được sử dụng để phát triển game trên PC, consoles, thiết bị di động và tên websites.

Ra mắt đầu tiên vào năm 2005 tại sự kiện Apple’s Worldwide Developer Cònerence bởi nhà sáng lập David Helgason, trải qua hơn 12 năm phát triển, nay Unity đã có version 5.5 hoàn thiện hơn về rất nhiều mặt. Tháng 5 năm 2012 theo cuộc khảo sát Game Developer Megzine được công nhân là Game engine tốt nhất cho mobile. Năm 2014 Unity thắng giải “Best Engine” tại giải UK’s annual Develop Industry Exellence.

Trong bốn năm kể từ khi công cụ của Unity ra mắt, quy mô của thị trường trò chơi toàn cầu đã bùng nổ từ 27 tỷ đô la lên 135 tỷ đô la, do sự gia tăng của trò chơi trên thiết bị di động, hiện chiếm phần lớn thị trường. Tính đến quý 3 năm 20116 đã có 5 tỷ downlad game và ứng dụng được phát triển bởi Unity. Trong 2,4 tỷ thiết bị di động đã từng tải ít nhất 1 ứng dụng bởi Unity. Trong top 1000 game Mobiles miễn phí thì số lượng game tạo ra bởi Unity chiếm tới 34%.



* Ưu điểm của Unity:
* Chức năng cốt lõi đa dạng bao gồm: cung cấp công cụ dựng hình (kết xuất đồ họa) cho các hình ảnh 2D hoặc 3D, công cụ vật lý (tính toán và phát triển va chạm), âm thanh, mã nguồn, hình ảnh động, trí tuệ nhân tạo, phân luồng, tạo dò nguồn dữ liệu xử lý, quản lý bộ nhớ, dựng ảnh đồ thị và kết nối mạng. Nhờ có các engine mà công việc làm game trở nên ít tốn kém và đơn giản hơn.
* Hỗ trợ đa nền tảng: một trong các thế mạng của Unity3D chính là khả năng hỗ trợ gần như toàn bộ các nền tảng hiện có bao gồm: PlayStation 2, Xbox 360, WiiU, iOS, Android, Windows, Blackberry 10, OS X, Linux, trình duyệt Web và cả Flash.
* Dễ sử dụng: Unity3D được xây dựng trong môi trường phát triển tích hợp, cung cấp một hệ thống toàn diện cho các lập trình viên, từ soạn thảo mã nguồn, xây dựng công cụ tự động hóa đến trình sửa lỗi. Do được hướng đến đồng thời cả lập trình viên không chuyên và studio chuyên nghiệp, nên Unity3D khá dễ sử dụng.

## **3.2. Sơ lược lịch sử hình thành và phát triển của Unity.**

### ***3.2.1. Hình thành.***

Phần lõi của Unity ban đầu được viết bởi Joachim Ante vào nằm 2001. Sau đó công ty được hình thành vào năm 2005 và bắt đầu với phiên bản 1.0. Đến nắm 2007, Unity được nâng lên phiên biển 2.0. Unity bắt đầu hỗ trọ iPhone vào năm 2008. Vào tháng 6/2010, Unity chính thức hỗ trọ Android và cho ra đời phiên bản 3.0 có hỗ trợ Android vào tháng 9/2010. Có theer thấy tốc độ phát triển của Unity khá nhanh.

### ***3.2.2. Giải thướng***

Unity đã đoạt được nhiều giải lớn với những giải thícch sau:

* Nămm 2006, Unity đạt “Best Use of Mac OS X Graphics” tại Apple’s WWDC. Đây là lần đầu tiên một công cụ phát triển game đạt được chất lượng do giải thưởng uy tín này đưa ra.
* Năm 2009, Unity Technolohies có tên trong “Top 5 công ty game của năm” do Gamasutra tổ chức.
* Nắm 2010,, Unity đoạt giải Best Engine Finalist do Develop Magazine bình chọn, giải Technology Innovation Award của Wall Street Journal ở thể loại phần mềm.

### ***3.2.3. Khách hàng***

Unity được trên 250.000 người đăng kí sử dụng gồm Bigpoint, Cartoon Network, Coca-Cola, Disney, Electronic, Arts, LEGO, Microsoft, NASA, Ubisoft, Warner Bros, các hãng phim lớn nhỏ, các chuyên gia độc lập, sinh viên và những người đam mê.

## **3.3. Tính năng của engine Unity**

Môi trường phát triển của tích hợp với tính năng kế thừa, khả năng chính sửa đồ họa, chức năng kiểm tra chi tiết, và đặc biệt tính năng xem trước game ngay trong lúc xây dựng (live game preview).

Triển khai được trên nhiều nền tảng:

* Chương trình độc lập trên Windows và MacOs.
* Trên web, thông qua Unity Web Player plugin cho Internet Explorer, Firefox, Safari, Opera, Chrome, cho cả Windows và Mac OS.
* Trên Mac OS Dashboard widget.
* Cho Nintendo Wii (cần mua license thêm)
* Cho iphone, iPad application (cần mua license thêm)
* Cho Google Android (cần mua license thêm)
* Cho Microsoft Xbox 360 (cần mua license thêm)
* Cho Sony PlayStation 3 (cần mua license thêm)

Tài nguyên (model, âm thanh, hình ảnh, .. ) được tải vào trong Unity và tự động cập nhật nếu tài nguyên có sự thay đổi. Unity hỗ trợ kiểu định dạng từ 3DS Max, Maya, Blender, Cinema 4D và Cheetah3D.

Graphics engine sử dụng Direct3D (Windows), OpenGL (Mac, Windows), OpenGL ES (iPhone OS), và các API khác trên Wii.

Hỗ trợ bump mapping, reflection mapping, parallax mapping, Screen Space Ambient Occlusion, ....

Unity Asset Server: đây là một tính năng khá mới của Unity, theo đó Unity sẽ cung cấp một hệ thống quản lý theo dạng phiên bản cho tất cả asset và cả script.

Đây là một kho chứa các tài nguyên cần thiết cho việt làm game. Khi import cũng như sửa chữa, trạng thái của asset ngay lập tức được cập nhật. Server chạy trên database opensource PostgreSQL và có thể truy cập trên cả Mac lẫn Windows, Linux. Asset Server đòi lỏi một khoản chi phụ trội là $499 cho mỗi bản copy Unity, và một license Unity Pro.

**3.4. Các khái niệm trong Unity**.

### ***3.4.1. Asset***

Đây là kho tài nguyên cho việc xây dựng game trong một project của Unity. Các tài nguyên này có thể là hình ảnh, âm thanh, hoặc mô hình 3D có sẵn. Unity sẽ tham chiến đến các tập tin chúng ta sẽ sử dụng để tạo ra các tài nguyên cho trò chơi. Đây là lý do tại sao trong bất kỳ thư mục chứa project sử dụng Unity thì tất cả các tập tin tài nguyên phải được lưu trữ trong một thư mục con tên là Assets.

### ***3.4.2. Scene***

Trong Unity, chúng ta có thể xem Scense là các màn chơi, cấp độ chơi riêng lẻ, hoặc các vùng của nội dung trò chơi. Ví dụ như Main menu, Options, About, ...

Bằng cách xây dựng trò chơi với nhiều cảnh, chúng ta sẽ có thể phân phối thời gian tải và thử nghiệm các phần khác nhau của trò chơi riêng lẻ một cách nhanh chonhs và chính xác.

### ***3.4.3. Game Object***

Khi một tài nguyên được sử dụng trong một scene, khi đó chúng ta có thể coi tài  
nguyên này là một “Game Object” mới. Mỗi GameObject phải chứa ít nhất một  
thành phần, đó là thành phần “Transform”. Transform chứa các phép để biến đổi  
góc quay, tỷ lệ hay tịnh tiến của đối tượng. Từ đây trong báo cáo này chúng em sẽ  
gọi GameObject trong cửa sổ Hierarchy là đối tượng game.

***3.4.4.Component***  
Component là các thành phần trong một Game Object của Unity. Bằng cách đính  
kèm các thành phần vào cho một đối tượng, chúng ta có thể áp dụng ngay các phần  
mới của game engine vào đối tượng. Thông thường các thành phần này được Unity  
xây dựng sẵn như ánh sáng, camera, particle, hiệu ứng vật lý..

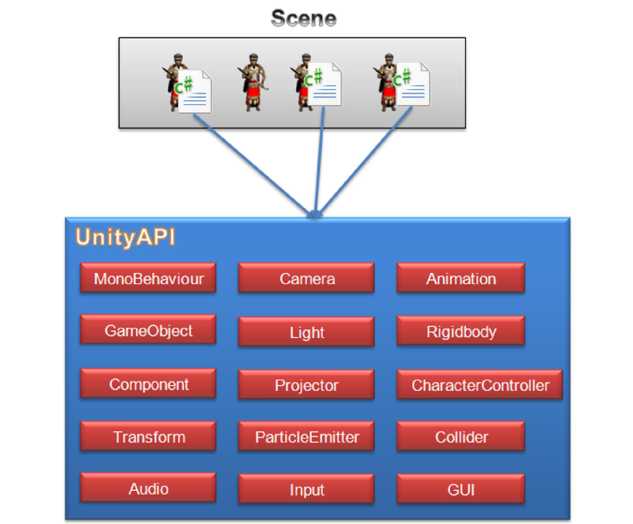
***3.4.5.Script***  
Script là thành phần quan trọng nhất trong Unity, có thể xem scripts như là linh hồn của game. Chúng ta có thể viết kịch bản cho game bằng C#, Java Scripts, hoặc Boo (một dẫn xuất của ngôn ngữ Python). Theo nhiều người đã sử dụng Unity thì code bằng C# sẽ giúp game chạy nhanh hơn và giúp kiểm soát code tốt hơn do tất cả các biến phải được khai báo rõ ràng. Mặt khác ngôn ngữ C# rất tiện dụng để lập trình, nên trong luận văn này, chúng em dùng ngôn ngữ C# để viết kịch bản cho game.Mỗi file script C# là một class bắt buộc kế thừa từ lớp MonoBehaviour, có tên class phải trùng với tên file script.

***2.4.6.Prefab***  
Hãy tưởng tượng Prefab là một cái thùng rỗng, mà bên trong nó chúng ta có thể  
chứa đựng các thành phần hay đối tượng khác nhau, chúng ta có thể viết kịch bản  
cho hành động của Prefab (khởi tạo, di chuyển, hay hủy đối tượng). Chúng ta có thể sử dụng đối tượng này nhiều lần trong trò chơi, và cũng có thể sử dụng lại cho  
project khác. Prefab cho phép chúng ta lưu trữ các đối tượng, toàn bộ thành phần  
bên trong và cấu hình hiện tại.

## **3.5. Tổng quan kiến trúc engine Unity trên Android**

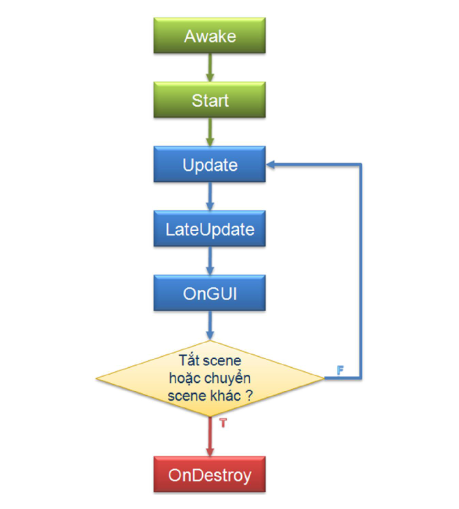
### ***3.5.1. Kiến trúc tổng quan***

Engine Unity hỗ trợ cho chúng ta UnityAPI để viết script game. UnityAPI là API  
lập trình game trong Unity rất mạnh. UnityAPI chứa các đối tượng và phương thức  
hỗ trợ hầu hết các đối tượng và các loại thành phần trong Unity. Trong một scene thường có nhiều đối tượng game. Mỗi đối tượng này có thể có  
hoặc không có đoạn script nào gắn lên đó. Nếu muốn gắn script vào đối tượng, ta  
bắt buộc phải kế thừa class đó từ lớp MonoBehaviour của UnityAPI và tên class  
phải trùng với tên file script. Mỗi script khi gắn lên đối tượng game đều được đối  
tượng game xem như một thành phần bên trong và được cấp phát vùng nhớ khi chạy game.



Bên trong UnityAPI chứa rất nhiều lớp hỗ trợ lập trình game, trong đó có một số  
lớp quan trọng như :  
o MonoBehaviour: tất cả các script muốn gắn vào một đối tượng game bắt buộc  
phải kế thừa từ lớp này.  
o GameObject: lớp cha của tất cả các thực thể trong scene.  
o Component: lớp cha của tất cả các thành phần có thể gắn vào đối tượng.  
o Transform: giúp thay đổi vị trí, xoay, biến đổi tỉ lệ mô hình.  
o Input: hỗ trợ lập trình với chuột, cảm ứng đa điểm, cảm biến gia tốc.  
o Camera: giúp lập trình camera.  
o Light: giúp tạo ánh sáng trong game.  
o Projector: giúp chiếu texture lên bề mặt vật thể.  
o ParticleEmitter: hỗ trợ tạo các hiệu ứng particle đẹp mắt.  
o Audio: hỗ trợ lập trình với âm thanh.  
o Animation: chạy chuyển động của mô hình nhân vật.  
o Rigidbody: giúp tạo hiệu ứng vật lý liên quan đến trọng lực như bóng nẩy, lăn,..  
o CharacterController: giúp điều khiển nhân vật di chuyển theo độ cao địa hình.  
o Collider: hỗ trợ lập trình va chạm giữa các vật thể.  
o GUI: giúp lập trình giao diện người dùng trên Unity.

***2.6.2. Chu kỳ sống của thành phần script gắn trên đối tượng game***.  
Chu kỳ sống của một thành phần script được tính kể từ khi scene được chạy cho đến lúc scene bị tắt hoặc chuyển sang scene khác.



**Hình 2.14 Chu kỳ sống của thành phần script**

Tùy theo trạng thái của scene mà sự kiện tương ứng sẽ được gọi.

o Awake: được gọi khi script được load xong  
o Start: được gọi khi script được load xong nếu thành phần script đó không bị  
disable.  
o Update: đây là sự kiện thường sử dụng nhất và được gọi liên tục từng frame để  
vẽ lại màn hình.  
o LateUpdate: sự kiện này chạy sau sự kiện Update và được gọi liên tục từng  
frame.  
o OnGUI: sự kiện này dùng để vẽ GUI và được gọi liên tục từng frame, chỉ trong  
sự kiện này ta mới có thể sử dụng các lớp hỗ trợ tạo giao diện người dùng của  
UnityAPI.  
o OnDestroy: được gọi khi thành phần script bị hủy khỏi bộ nhớ.

# **CHƯƠNG 4. Game Project.**

## **Tổng quan.**

### ***Giới thiệu về sản phẩm***.

* Tên trò chơi: Treasure Hunt.
* Thể loại: 2d nhập vai.
* Đối tượng: 12+
* Nền tảng: Android.

### ***4.1.2. Mô tả.***

Người chơi nhập vai là một nhà thám hiểm tìm kiến kho báu bị mất tích từ hàng trăm năm trước. Trong trò chơi, nhiệm vụ của người chơi là tìm thấy kho báu. Có kho báu lớn sẽ cần có mật khẩu để mở khóa. Để có được mật khẩu người chơi phải trao đồi vật phẩm với dân làng. Vật phẩm này có được khi tiêu diệt được các quái vật. Còn các kho báu nhỏ được mở ra sau khi có chìa khóa tương ứng với kho báu.

Khi bắt đầu, hiện ra màn hình khởi đầu của trò chơi có một đoạn animation và hai button “play” và “help”. Ấn vào button “play” hiện ra các bản đồ (tổng công hai bản đồ), mỗi bản đồ gồm nhiều màn chơi. Mỗi màn chơi được kí hiệu bằng kí hiệu khác nhau trên bản đồ. Khi bấm vào kí hiệu hệ thống sẽ kiểm tra xem màn chơi được chọn có cùng hòn đảo với màn chơi cũ (điểm bắt đầu) không. Nếu không hệ thống hiện câu hỏi tính toán để người chơi trả lời. Nếu điểm đánh dấu cùng hòn đảo hoặc người chơi trả lời được, chuyển sang màn hình chơi. Người chơi điều khiển nhân vật chơi để tiêu diệt quái vật để thu thập vật phẩm với mục đích cuối cùng là mở kho báu lớn ở màn chơi mountain.

Mỗi khi bắt đầu một màn chơi hệ thống sẽ tạo tilemap ngẫu nhiên, sắp xếp lại vị trí của các quái vật. Khi người chơi mở được kho báu lớn của mỗi map, hệ thống sẽ thay đổi mật khẩu của kho báu đó.

### ***Sản phẩm.***

Source:

Chơi thử:

## **Phân tích thiết kế.**

### ***Biểu đồ use case.***

**A diagram of a diagram

Description automatically generated**

### ***Biểu đồ Activity Diagram.***

**A diagram with text and black text

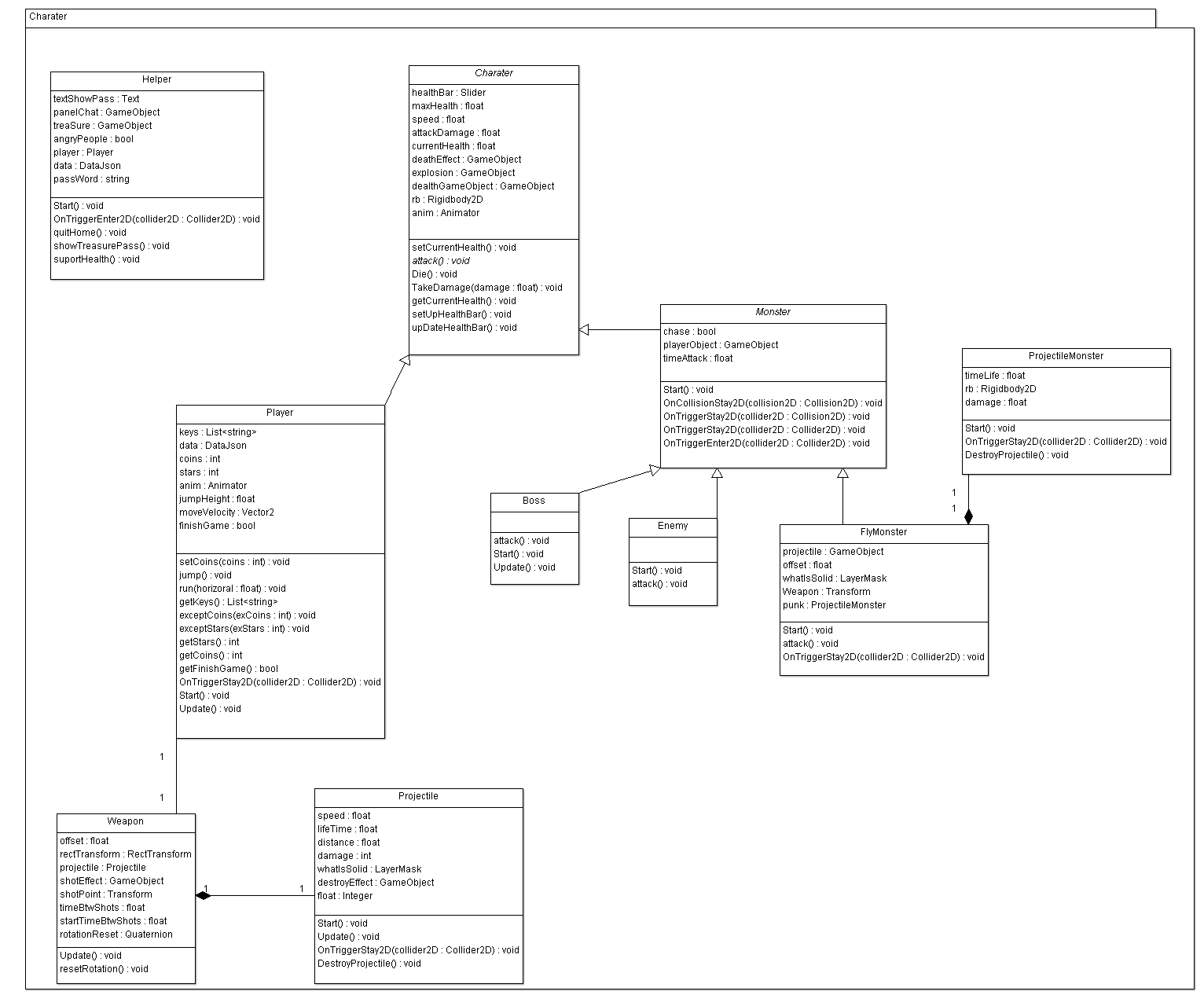
Description automatically generated**

**A diagram of a diagram

Description automatically generated**

### ***Biểu đồ Class Diagram.***

a, Biểu đồ class Diagram của gói Charater.

b, Biểu đồ class diagram cho gói Dungeon

A screenshot of a computer

Description automatically generated

c, Biểu đồ class diagram của gói Control.

A screenshot of a computer screen

Description automatically generated

d, Biểu đồ class diagram của gói Item.

A screenshot of a computer

Description automatically generated

e, Biểu đồ class diagram của gói Data.

A screenshot of a computer

Description automatically generated

## **Sản phẩm hoàn thành.**

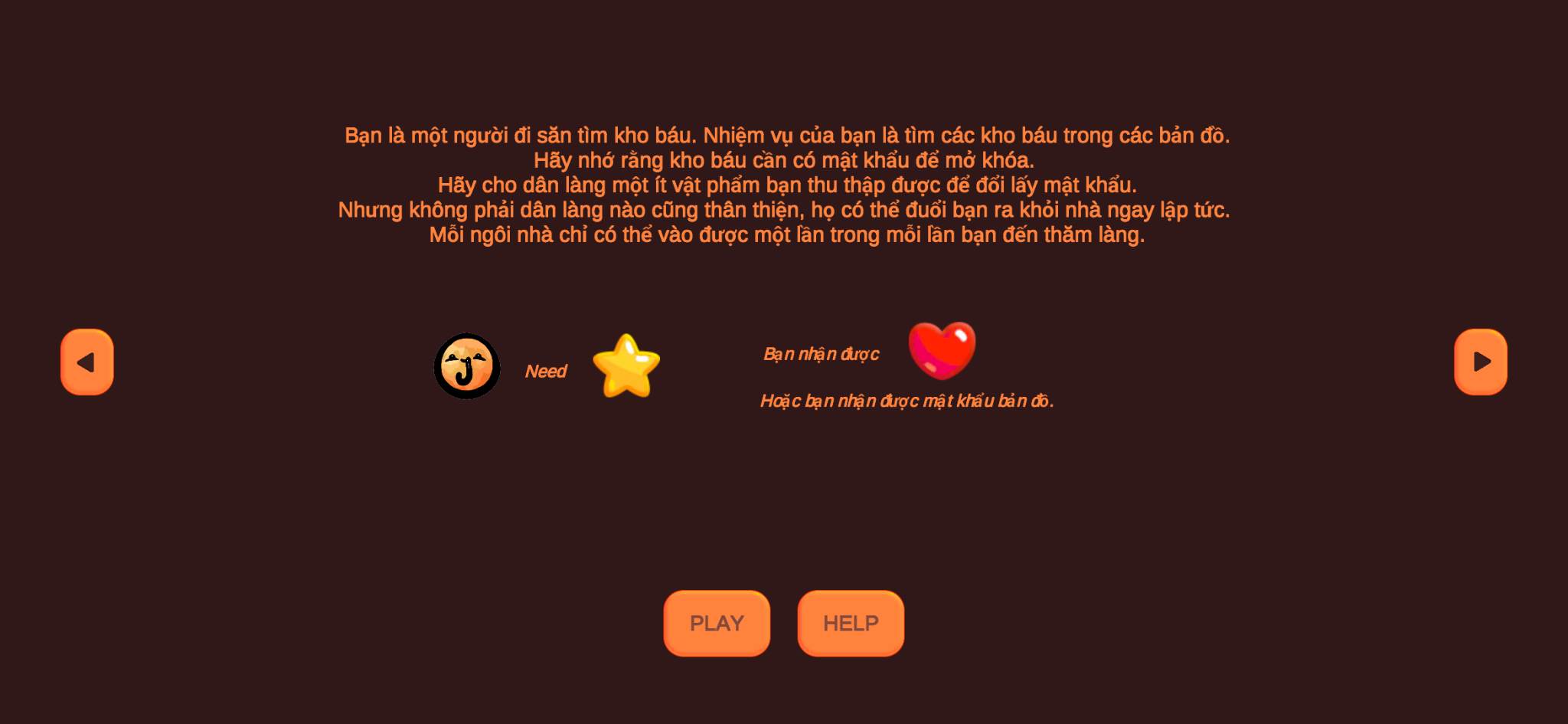
### ***Giao diện bắt đầu game.***

a,



Màn hình mở đầu của trò chơi có hai button “play” và “help”.

Khi bấm vào button “help” sẽ hiện ra các luật của trò chơi, hướng dẫn người chơi.



A screenshot of a computer

Description automatically generated

A screenshot of a video game

Description automatically generated

### ***Giao diện maps.***

Sau khi bấm vào button “play”, màn hình hiện ra bản đồ chơi.

A map of a small island with boats and a tower

Description automatically generated

**Bản đồ 1.**

A video game screen with boats and a river

Description automatically generated

**Bản đồ 2.**

Khi kiểm tra màn chơi không cùng hòn đảo với màn chơi cũ sẽ hiện lên màn hình câu hỏi.

A rectangular object with buttons

Description automatically generated

**Màn hình câu hỏi.**

### ***4.2.3. Các màn chơi.***



**Màn chơi “River”.**

A screenshot of a video game

Description automatically generated

Màn hình trò chơi khi ấn dừng.

A screenshot of a video game

Description automatically generated

A video game with cartoon characters

Description automatically generated

**Màn chơi “Tower”.**

A video game screen with a cartoon character and a house

Description automatically generated

**Màn chơi “Village”.**

A video game screen with a green screen and a green background

Description automatically generated

Khi người chơi di chuyển vào ngồi nhà sẽ có thể gặp được người hỗ trợ.

A screenshot of a video game

Description automatically generated

Khi người chơi đến gần người hỗ trợ màn hình giao dịch hiện lên. Nếu người chơi ấn vào bản đồ, mật khẩu kho báu lớn sẽ hiện lên đồng thời trừ 100 Stars.

A screen shot of a computer screen

Description automatically generated

A video game with trees and a cartoon character

Description automatically generated

**Màn chơi “Mountain”.**

A keyboard with orange squares

Description automatically generated

Sau khi hoàn thành màn chơi “Mountain” (tiêu diệt boss của màn chơi) chuyển sang màn hình kho báu lớn.

# **Tổng kết**

# **Tài liệu tham khảo**