**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA CÔNG NGHỆ PHẦN MỀM**



**ĐỒ ÁN NHẬP MÔN ỨNG DỤNG DI ĐỘNG**

**TÌM HIỂU REACT NATIVE , UNITY VÀ**

**XÂY DỰNG ỨNG DỤNG GAME BALL**

**Mã lớp :** SE114.J12

**Giảng viên hướng dẫn :** Ths. Huỳnh Tuấn Anh

**Sinh viên thực hiện :** 13520990 Võ Duy Tuấn

13521007 Phan Văn Tùng

15520778 Cao Du Thái

**LỜI CẢM ƠN**

Để hoàn thành được đồ án này, trước tiên, nhóm chúng em xin gửi lời cảm ơn chân thành đến toàn thể giảng viên Trường Đại học Công nghệ thông tin – ĐHQG TP.HCM nói chung, các thầy cô trong Khoa Công nghệ Phần mềm nói riêng đã tận tâm truyền đạt kiến thức, kinh nghiệm, chỉ dạy tận tình cho nhóm chúng em từ những ngày đầu bước chân vào trường, vào chuyên ngành.

Đặc biệt, lời cảm ơn sâu sắc nhóm chúng em xin gửi tới thầy Huỳnh Tuấn Anh, người đã đồng hành cùng nhóm chúng em trong suốt học kỳ qua. Được sự hướng dẫn trực tiếp của thầy cùng với sự tận tâm chỉ bảo, luôn theo sát và hỗ trợ hết lòng, gắn bó trong từng chặng đường nhỏ trong khoảng thời gian này để giúp cho nhóm chúng em hoàn thành tốt đồ án lần này.

Cuối cùng, nhóm chúng em cũng gửi lời cảm ơn tới các anh chị, bạn bè trong Khoa Công nghệ Phần mềm đã nhiệt tình hướng dẫn, góp ý về những thiếu sót cũng như chia sẻ những kinh nghiệm, tài liệu trong quá trình làm đồ án này cũng như trong suốt thời gian em học tập tại trường.

Mặc dù nhóm chúng em đã cố gắng, nỗ lực rất nhiều để học hỏi, tìm hiểu, nghiên cứu và thu thập được nhiều thông tin, dữ liệu cùng với sự giúp đỡ của mọi người, nhưng do kiến thức còn nhiều hạn chế nên đồ án này không tránh khỏi những thiếu sót. Hơn ai hết, nhóm chúng em mong nhận được sự góp ý từ thầy cô để đồ án của nhóm chúng em được hoàn thiện hơn.

Xin chân thành cảm ơn !

**MỤC LỤC**

[1. Giới thiệu 4](#_Toc516051525)

[1.1. Đặt vấn đề 4](#_Toc516051526)

[1.2. Đối tượng người dùng 5](#_Toc516051527)

[2. Về đồ án 5](#_Toc516051528)

[2.1.1 Nền tảng React Native 5](#_Toc516051529)

[2.1.2 Công cụ lập trình 11](#_Toc516051530)

[2.1.3 Cài đặt React Native 12](#_Toc516051531)

[2.1.4 Một số Components trong React Native 16](#_Toc516051532)

[2.2 Giới thiệu về nền tảng Unity 26](#_Toc516051536)

[2.2.1 Giới thiệu chung 26](#_Toc516051537)

[2.2.2 Tính năng của Unity 27](#_Toc516051538)

2.2.3 Các khái niệm cơ bản của Unity.....................................................................................................27

2.2.4 Làm quen với môi trường Unity.....................................................................................................30

2.2.5 Tìm hiểu về khởi tạo các đối tượng................................................................................................30

[3. Chọn đề tài đồ án và thực hiện 33](#_Toc516051540)

4.1 Sơ đồ phân rã chức năng......................................................................................................................33

[5. Cơ sở dữ liệu 35](#_Toc516051541)

[6. Giao diện thiết kế 35](#_Toc516051542)

6.1 Danh sách màn hình...........................................................................................................................36

6.2Chi tiết thiết kế....................................................................................................................................36

[7. Tài liệu tham khảo 37](#_Toc516051543)

[8. Phân công nhiệm vụ 37](#_Toc516051544)

[9. Tổng kết – Đánh giá 37](#_Toc516051545)

**1. Giới thiệu**

**1.1. Đặt vấn đề**

Với xu hướng công nghệ hiện nay thì ngành công nghệ thông tin đã có những bước phát triển vượt bậc, điển hình là ở Việt Nam. Dân số Việt Nam hơn 90 triệu người thì có đến 39,8 triệu người sử dụng Internet (chiếm 44%), và tỉ lệ người sử dụng smartphone ngày càng gia tăng nhanh chóng, chiếm 84% so với người dùng điện thoại phổ thông, tăng 6% so với năm 2016.

Cuộc sống ngày càng hiện đại, nhu cầu giải trí càng cao. Sau những giờ mệt mỏi căng thẳng về công việc hay cuộc sống thì những giờ giải lao là vô cùng quý báu và cần thiết để giúp cho cơ thể được thư giãn, đầu óc giải tỏa căng thẳng mệt mỏi, thêm vào đó tinh thần được thoải mái hơn và năng động hơn... Ngoài nhưng giờ phút có thể giảo lao hay vui chơi được như đá banh, hoặc thể thao, nhưng cũng có nhưng lúc cơ thể cảm thấy mệt mỏi và muốn nằm nghỉ nhưng lại muốn được giải lao ngắn về tình thần, với thời xưa thì cần có truyền hình hay radio để có thể mang lại những cảm giác khoan khói và dễ chịu đối với cuộc sống, nhưng do ngày nay đã quá hiện đại và mức sống người dân đã nâng cao, việc sở hữu một chiếc điện thoại thông minh là điều mà gần như bất kỳ ai cũng có và cần có, từ những bạn học sinh tiểu học tới nhưng cô chú bác lớn tuổi, vậy bắt kịp với xu thể và sự tiến bộ đó điện thoại thông minh cũng đã ngày càng phát triển và hỗ trợ tối ưu đối với người dùng, qua bộ môn nhập môn di động này, nhóm chúng em cũng đã học được về những kiến thức cơ bản về lạp trình di động và muốn mang tới cho người sử dụng được sự thoải mái bằng một game nhỏ viết trên nền tảng Unity và được chạy trên hệ điệu hành Android hoặc ios rất dễ chơi và đơn giản

Với nhưng điều đơn giản như vậy là là nhưng điều rất cần thiết đối với cuộc sống như hiện nay, chỉ cần về nằm nghỉ và bật điện thoại lên và chơi một game nhẹ nhàng, không quá phức tạp, dễ hiểu và hình ảnh đẹp là có thể thấy thoải mái sau nhưng giờ làm việc căng thẳng mệt mỏi,

Với nhưng game hay ứng dụng nặng và phức tạp thì đòi hỏi phải thời gian và công sức suy nghĩ và “cày kéo” nhưng đối với những game nhẹ và đơn giản thì đã là quá đủ để giải trí và thư dãn rồi chìm vào giấc ngủ sâu

* Điều học được từ bộ môn Nhập môn di động này là nhóm e đã biết được cách viết những app hay game để chạy trên nền tảng hệ điều hành và cài vào di động của mình để sử dụng và trải nghiệm

.

**1.2. Đối tượng người dùng**

Từ độ tuổi 5 tuổi đến 60 tuổi , chỉ cần biết sử dụng điện thoại là có thể tải trò chơi và chơi game được

# **2. Về đồ án**

## **2.1.1 Giới Thiệu Về Nền tảng React Native**

Để hiểu rõ hơn về React Native là gì, trước tiên chúng ta cần phân biệt sự khác nhau giữa ứng dụng Hybrid và ứng dụng Native

* 1. *Ứng dụng Hybrid*

Là chương trình phần mềm trên điện thoại di động được viết dựa trên nền tảng Web (HTML5, CSS3, JavaScript), bản chất hoàn toàn là ứng dụng Web nhưng có thêm được các tính năng thao tác phần hệ điều hành như tập tin, truy cập máy ảnh,… Toàn bộ những thứ này đều được bao bọc bởi một lớp ứng dụng Native mà nổi bật là Phonegap/Cordova. Phần Web được xử lý hiển thị bởi Webview, phần tính năng truy cập hệ thống được cung cấp bởi các hàm API, gọi hàm bằng JavaScript thông qua API thì chương trình bao bọc sẽ gọi trực tiếp Native xuống hệ điều hành. Bằng cách này, ứng dụng Web có thêm những tính năng cao cấp của ứng dụng Native, và do được tính hợp mã nguồn sẵn nên tốc độ của ứng dụng Hybrid nhanh hơn ứng dụng Web.

* 1. *Ứng dụng Native (iOS, Android)*

Là ứng dụng được phát triển trực tiếp bằng ngôn ngữ của hệ điều hành đó cung cấp. Ví dụ với iOS là Objective-C, Swift và Android là Java. Các ứng dụng viết bằng ngôn ngữ này được biên dịch ra ngôn ngữ máy trên điện thoại và có toàn bộ tính năng mà hệ điều hành đó cung cấp. Do là ngôn ngữ trực tiếp cũng như không phải thông qua ứng dụng nào khác nên tốc độ là nhanh nhất.

Điểm hạn chế khi viết ứng dụng Native đó là tốn thời gian re-compile. Mỗi lần sửa đổi một chức năng bạn cần re-build. Với iOS khi sử dụng Swift cho Codebase lớn, bạn sẽ thấy tốc độ re-complie khá là chậm.

Nhưng tại sao ứng dụng Native vẫn được ưu chuộng và phát triển hơn ứng dụng Web hay ứng dụng Hybrid. Lý do rất đơn giản, các ứng dụng Native mang lại trải nghiệm tốt hơn cho người dùng.

c. React Native là gì ?

React Native là một framework giúp chúng ta viết các ứng dụng Native chỉ bằng JavaScript do Facebook phát hành. Nó sử dụng cùng thiết kế như React. Hiểu đơn giản bạn viết code JavaScript để có thể sử dụng cho cả thiết bị iOS và Android. Do đó React Native sẽ rất dễ dàng với người đã nắm vững JavaScript.

Sự ra đời của React Native để khắc phục các điểm yếu của ứng dụng Web và Hybrid. Với React Native, bạn không phải đang xây dựng một “Mobile Web App”, một “HTML5 App”, hay một “Hybrid App”. Bạn đang build một “Real Mobile App” cho tất cả các hệ điều hành mobile nhưng không phân biệt bạn đang dùng Objective-C, Swift hay Java. React Native xây dựng giao diện cơ bản giống như các ứng dụng iOS và Android thông thường. Bạn chỉ cần đặt các Component lại với nhau bằng JavaScript và React.

Sử dụng React Native sẽ giảm thời gian re-compiling, cho bạn cảm giác như real-time compiling. Thay vì phải mất thời gian re-compiling, React Native hỗ trợ Hot Reloading, đơn giản sử dụng tổ hợp phím Command⌘ + R trên iOS simulator.

React Native hỗ trợ “drop down to native code” (Objective-C, Swift hay Java) nếu bạn cần tùy chỉnh hoặc tối ưu một số trường hợp đặc biệt cho ứng dụng của bạn. Bạn sẽ dễ dàng xây dựng một phần/chức năng trong ứng dụng của bạn bằng React Native, và ngược lại. Đó cũng là cách mà các ứng dụng của Facebook đang sử dụng.

d. Ưu điểm & nhược điểm của React Native

* **Ưu điểm** **:**
* Mang lại trải nghiệm Native
* Lập trình bằng một ngôn ngữ JavaScript duy nhất
* Tốc độ build nhanh
* Đa nền tảng : Phát triển được trên cả Android & iOS
* Chức năng “Hot reloading”, không cần phải buil lại tất cả mã nguồn
* Phù hợp với nhóm phát triển nhỏ
* **Nhược điểm** **:**
* Vẫn còn thiếu các View Components cho Android như Map, Modal, Slider, Spinner hoặc các module như Camera Roll, Media, PushNotificationIOS.
* Không hỗ trợ Window Phone : Đây sẽ là điểm trừ lớn nếu so với ứng dụng Hybrid. Tuy nhiên, nếu bạn chỉ tập trung cho iOS và Android thì tất nhiên đây không phải là vấn đề lớn.
* Không build được ứng dụng iOS trên Window và Linux : Tất nhiên đây không phải là điểm yếu kém của bản thân React Native, mà đó là do yêu cầu từ Apple.
* React Native không thể build được ứng dụng quá phức tạp nếu bạn không biết Swift/Objecive-C, Java. Tính phức tạp ở đây là ứng dụng của bạn cần phải chỉnh sửa các Components. Bạn nên nhớ là để viết được một ứng dụng Native bằng JavaScript luôn luôn có sẵn các Components đã được viết từ Swift/Objective-C (iOS) và Java (Android) với tính chất 1 – 1. Trường hợp bạn muốn chỉnh sửa một Components nào đó như thay đổi thành phần hoặc thêm API thì bạn phải tự viết bằng chính ngôn ngữ tương ứng của iOS hoặc Android. Tin vui cho bạn là cũng nhiều lập trình viên khác đã viết nhiều Components cần thiết cho hầu hết ứng dụng (đây cũng là lý do vì sao Facebook biến React Native thành mã nguồn mở).
* Không nên dùng để viết Game có tính đồ họa và cách chơi phức tạp.
* Dùng ES2015 / ES6 : Đây là cấu trúc mới cho JavaScript từ 2015, vì khá là mới nên những cấu trúc của nó có thể bạn chưa quen, dẫn tới việc khó khăn trong việc tiếp cận.
* Bạn có thể phải mất nhiều thời gian để theo hết hệ sinh thái của React, tuy nhiên tất cả là để làm cho đời lập trình viên fullstack đẹp đẽ và những sản phẩm chất lượng hơn, bao gồm: ReactJS, React Native, Flux, Relay, GraphQL. Bạn không cần phải quan tâm những thứ này chỉ để viết ứng dụng với React native

*e. Cách hoạt động của React Native*

Ứng dụng viết bằng React Native được chia làm 2 phần : Phần hiển thị và phần xử lý.

Phần hiển thị được biên dịch từ JavaScript sẽ map với những Components của hệ thống. Ví dụ : Điều hướng, tab, touch… Phần hiển thị này được lấy cảm hứng từ Virtual DOM của React JS, mọi xử lý hiển thị được thực hiện trên một cây DOM ảo, sau đó được React Native render lại bằng Native View.

Phần xử lý vẫn được thực hiện trực tiếp bằng ngôn ngữ JavaScript. Ví dụ “1 + 1 = 2”, biểu thức này được xử lý dưới bộ core thực thi JavaScript, không cần phải thông dịch qua Java hay Swift/Objective-C rồi mới làm phép tính.

f. Một số ứng dụng nổi tiếng viết bằng React Native

Mạng xã hội cực kỳ mạnh mẽ trong cuộc sống của con người hiện nay và nó trở thành nhu cầu gần như thiết yếu trong cuộc sống mỗi ngày của mỗi người.

Sau đây là 3 trong số những ứng dụng vô cùng tuyệt vời và nổi tiếng ứng dụng React Native :

* **FACEBOOK**

React Native bắt đầu khi dự án hackathon của Facebook phát triển để đáp ứng nhu cầu của công ty. Facebook muốn mang lại tất cả các lợi ích của phát triển web – chẳng hạn như lặp đi lặp lại nhanh và có một đội xây dựng toàn bộ sản phẩm – di động. Đó là cách React Native đã được đưa vào cuộc sống và tận dụng trong phát triển ứng dụng dành cho thiết bị di động cho cả ứng dụng iOS và Android.

A hand holding a cell phone

Description generated with very high confidence

Nhóm lập trình đã chuyển đổi tính năng Events Dashboard trong ứng dụng Facebook cho iOS sang React Native để kiểm tra hiệu suất ứng dụng, chẳng hạn như thời gian khởi động, điều này rất quan trọng trong loại hình ứng dụng này.

Đây là một phần quan trọng trong ấn tượng ban đầu của người dùng về ứng dụng và xác định liệu chúng có tồn tại hay ra đi không. Những gì họ đạt được là cắt giảm thời gian tìm hiểu thị trường một nửa.

* **SKYPE**

Phiên bản mới của Skype hoàn toàn được cải tiến bắt đầu từ các icon cho toàn bộ bố cục, thêm một vài tính năng gọn gàng nữa. Như Microsoft cũng đã thông báo, Skype Preview cho iOS đã có sẵn với số lượng giới hạn thông qua TestFlight của Apple dành cho Skype Insiders.



Không chỉ Skype cho iOS mà còn là một phiên bản dành cho máy tính để bàn, đây sẽ là dự án React Native lớn đầu tiên của Windows. Đây là một dấu hiệu tốt cho nền tảng này. Chúng ta sẽ có thể thấy React Native hoạt động khá tốt trên Windows.

* **INSTAGRAM**

Instagram đã vượt qua thử thách để tích hợp React Native vào ứng dụng gốc hiện tại của họ, bắt đầu từ chế độ xem đơn giản nhất mà bạn có thể tưởng tượng : chế độ Push Notifications đã được triển khai thực hiện dưới dạng WebView. Nó không yêu cầu xây dựng cơ sở hạ tầng navigation, vì UI khá đơn giản.

A close up of a device

Description generated with high confidence

Đội ngũ lập trình tại Instagram đã phải đối mặt với một số vấn đề trong quá trình, nhưng họ đã cải thiện đáng kể tốc độ phát triển. 85% đến 99% code được chia sẻ giữa các ứng dụng Android và iOS, tùy thuộc vào sản phẩm, do đó nhóm đã có thể cung cấp ứng dụng nhanh hơn nhiều so với các giải pháp thông thường.

* Trên đây là 3 trong số rất nhiều ứng dụng viết bằng React Native để phát triển và đạt được thành công vô cùng rực rỡ, tiện ích trong cuộc sống của con người và sử dụng rộng rãi trong xã hội.
* Với tốc độ phát triển vượt bậc của công nghệ thông tin và mạng xã hội cũng như smarphone hiện nay thì tương lai của React Native vô cùng tươi sáng và định hướng sẽ tiếp tục phát triển lâu dài.

## **2.1.2 Công cụ lập trình**

Mặc dù chưa có IDE độc quyền nhưng ứng dụng React Native vẫn có thể được phát triển thông qua các trình biên tập văn bản như Sublime Text, Atom, Notepad++ hay Visual Studio Code, Xcode, Expo,…

## **2.1.3 Cài đặt React Native**

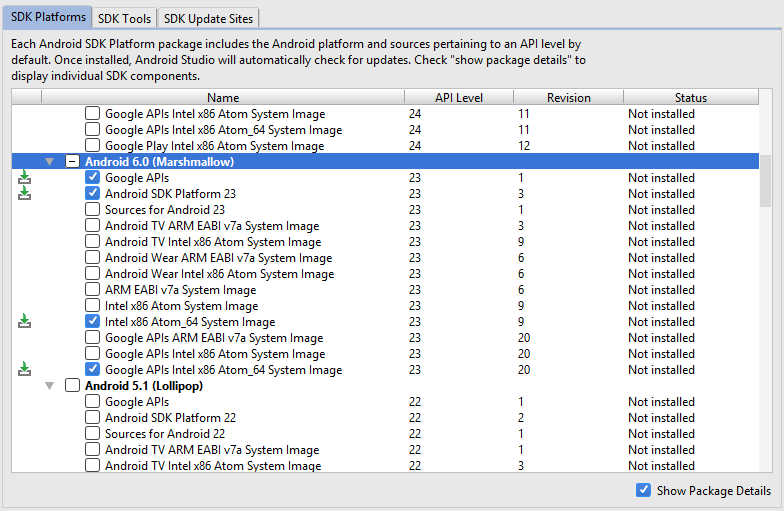
Sau đây là hướng dẫn cài đặt React Native để lập trình ứng dụng Android trên môi trường Windows. Với các hệ điều hành khác, bạn có thể tham khảo hướng dẫn cài đặt tại : [facebook.github.io/react-native/docs/getting-started.html](https://facebook.github.io/react-native/docs/getting-started.html)

* **Bước 1** : Cài đặt **Chocolatey** tại chocolatey.org/install. Mở cmd.exe với quyền quản trị và chạy đoạn lệnh sau :

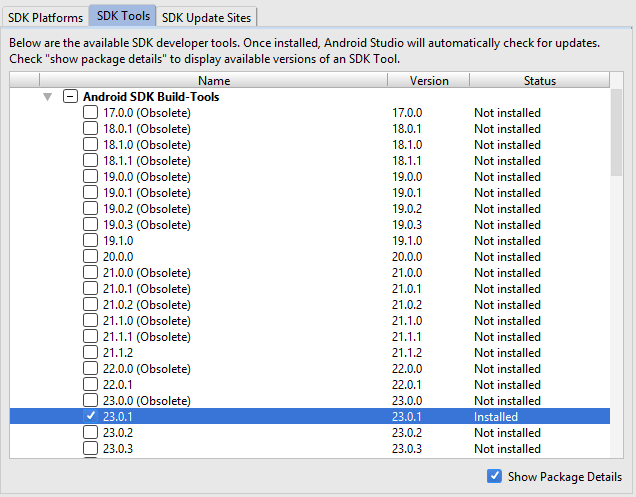
Ảnh có chứa ảnh chụp màn hình

Mô tả được tạo với mức tin cậy rất cao

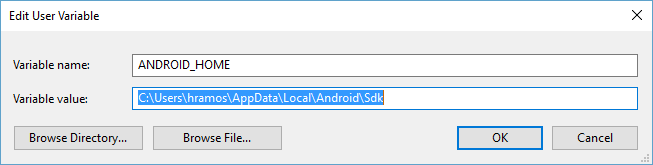
* **Bước 2** : Cài đặt **Node**, **Python 2**, **JDK**. Sau khi cài đặt **Chocolatey**, chạy đoạn lệnh sau : choco install -y nodejs.install python2 jdk8
* **Bước 3** : Cài đặt **React Native CLI**. Mở **Command Prompt**, chạy đoạn lệnh sau : npm install -g react-native-cli
* **Bước 4 :** Cài đặt **Android Studio** tại developer.android.com/studio/index.html. Sau khi cài đặt **Android Studio**, mở **SDK Manager**, chọn & cài đặt như hình sau :



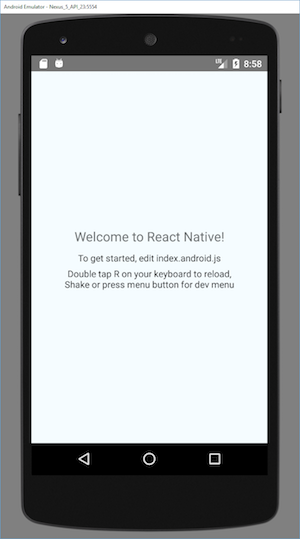
Tiếp tục cài đặt **Android SDK Build-Tools 23.0.1**



Cấu hình biến môi trường **ANDROID\_HOME** : Mở **Control Panel** → **System and Security** → **Change settings**... Chọn tab **Advanced** → **Environment Variables**... → **New**. Tìm đến thư mục cài đặt Android SDK



* **Bước 5** : Tạo ứng dụng đầu tiên
* Mở Command Line tại vị trí muốn khởi tạo project. Gõ lệnh sau : react-native init HelloWorld
* Khởi động AVD hoặc kết nối với điện thoại Android
* Chạy 2 dòng lệnh sau : cd HelloWorld → react-native run-android



# **2.1.4 Một số Components trong React Native**

**2.1.4.1 View**

* Chức năng : Là container chứa các Components khác

A screenshot of a cell phone

Description generated with high confidence

* Thuộc tính :

+ hitSlop

+ pointerEvents

+ removeClippedSubviews

+ collapsable

+ needsOffscreenAlphaCompositing

+ renderToHardwareTextureAndroid

* Phương thức :

+ onLayout()

+ onMagicTap()

+ nativeEvent() : *changedTouches, identifier, locationX, locationY, pageX, pageY, target, timestamp*

**2.1.4.2 ScrollView**

* Chức năng : Là container chứa các Components hiển thị nội dung kiểu danh sách, ví dụ như **FlatList**

A screenshot of a cell phone

Description generated with very high confidence

* Thuộc tính :

+ alwaysBounceVertical

+ pagingEnabled

+ removeClippedSubviews

+ scrollEnabled

+ showsHorizontalScrollIndicator

+ showsVerticalScrollIndicator

+ overScrollMode

* Phương thức :

+ scrollTo()

+ scrollToEnd()

+ scrollWithoutAnimationTo()

+ flashScrollIndicators()

**2.1.4.3 FlatList**

* Chức năng : Hiển thị nội dung kiểu danh sách

A screenshot of a cell phone

Description generated with very high confidence

* Thuộc tính :

+ data

+ horizontal

+ initialNumToRender

+ initialScrollIndex

+ numColumns

+ refreshing

* Phương thức :

+ renderItem()

+ getItemLayout()

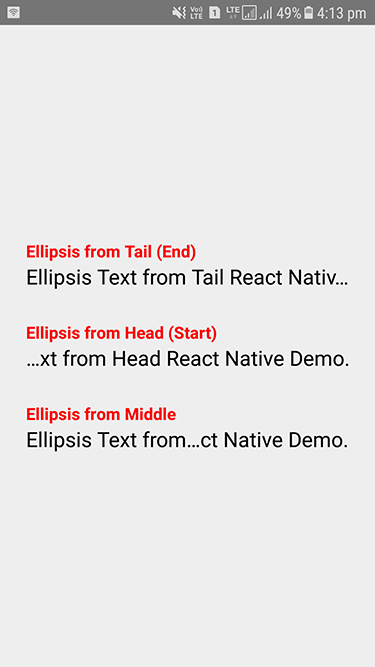
+ keyExtractor()

+ onEndReached()

+ onRefresh()

**2.1.4.4 Text**

* Chức năng : Render nội dung văn bản



* Thuộc tính :

+ selectable

+ ellipsizeMode : *head, middle, tail, clip*

+ numberOfLines

+ allowFontScaling

+ disabled

+ selectionColor

+ adjustsFontSizeToFit

+ minimumFontScale

+ suppressHighlighting

* Phương thức :

+ onLayout()

+ onLongPress()

+ onPress()

**2.1.4.5 TouchableOpacity**

* Chức năng : **TouchableOpacity** tương tự như **Button** nhưng khi chạm vào thì độ trong suốt giảm dần

A screenshot of a cell phone

Description generated with high confidence

* Thuộc tính :

+ activeOpacity

+ tvParallaxProperties

+ hasTVPreferredFocus

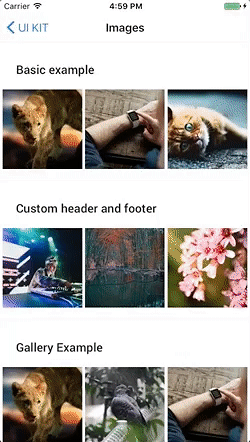
* Phương thức :

+ setOpacityTo()

+ onPress()

**2.1.4.6 Image**

* Chức năng : Dùng để render hình ảnh



* Thuộc tính :

+ blurRadius

+ resizeMode : *cover, contain, stretch, repeat, center*

+ source

+ loadingIndicatorSource

+ resizeMethod : *auto, resize, scale*

* Phương thức :

+ onLayout()

+ onLoad()

+ onLoadEnd()

+ onLoadStart()

+ onProgress()

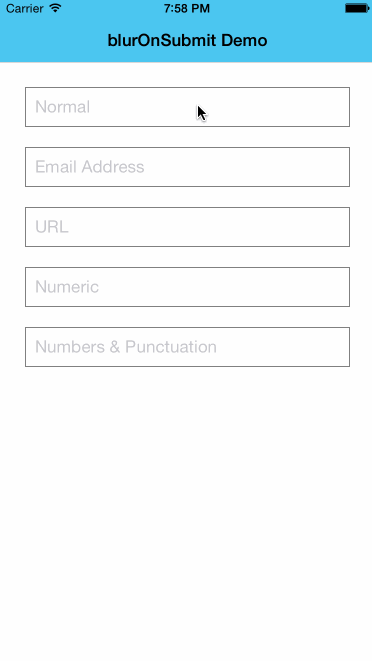
+ getSize()

+ prefetch()

+ queryCache()

**2.1.4.7 TextInput**

* Chức năng : Dùng để nhập dữ liệu dạng văn bản



* Thuộc tính :

+ allowFontScaling

+ autoCapitalize : none, sentences, words, characters

+ autoCorrect

+ autoFocus

+ blurOnSubmit

+ caretHidden

+ clearButtonMode : *never, while-editing, unless-editing, always*

+ clearTextOnFocus

+ contextMenuHidden

+ keyboardType : *default, numeric, email-address, phone-pad*

+ maxLength

+ multiline

+ numberOfLines

+ placeholder

+ value

* Phương thức :

+ onBlur()

+ onChange()

+ onChangeText()

+ onContentSizeChange()

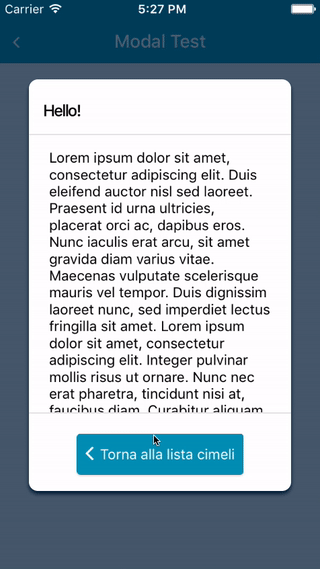
+ onEndEditing()

+ onFocus()

+ onKeyPress()

**2.1.4 8 Modal**

* Chức năng : Dùng để hiển thị nội dung nổi bật lên trên cùng so với các Components khác



* Thuộc tính :

+ visible

+ supportedOrientations : *portrait, portrait-upside-down, landscape, landscape-left, landscape-right*

+ transparent

+ animationType : *none, slide, fade*

+ presentationStyle : *fullScreen, pageSheet, formSheet, overFullScreen*

* Phương thức :

+ onRequestClose()

+ onShow()

+ onDismiss()

+ onOrientationChange()

**2.1.4.9 Picker**

* Chức năng : Dùng để tùy chọn nội dung, tương tự ComboBox trong C#

A screenshot of a cell phone

Description generated with very high confidence

* Thuộc tính :

+ selectedValue

+ enabled

+ mode : *dialog, dropdown*

+ prompt

+ itemStyle

* Phương thức :

+ onValueChange()

**2.2 Giới Thiệu Về Nền Tảng Unity**

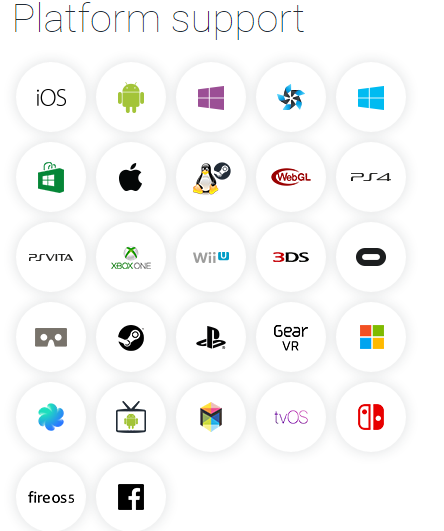
**2.2.1 Giới thiệu chung**

- Unity3D là môi trường phát triển tích hợp, mạnh mẽ, hỗ trợ thao tác kéo thả, tùy biến giao diện nhanh chóng , trực quan.

- Cung cấp các giao diện xử lý đồ họa, tích hợp sắn thư viện vật lý, tính toán va chạm

- Hỗ trợ phát triển cả game 3D và 2D.

- Hỗ trợ nhiều nền tảng thông dụng như OSX, LiNUX, Web, Window, android, Phone 8, Ps3, BB



- Cộng đồng người dùng, hỗ trợ rộng lớn

- Có phiên bản miễn phí có thể chạy được trên cả Android và MAC OSX

**2.2.2 Tính năng của Unity**

* Unity3D hướng tới người sử dụng chuyên nghiệp và cả nghiệp dư nên khá dễ sử dụng. Với ngôn ngữ lập trình bằng C#, Javascrip, hoặc BOO.
* Ngày nay rất nhiều nhà phát triển lựa chọn Unity3D để phát triển bởi khả năng hỗ trợ đa nền tẳng, và sự phát triển tiện dụng của Unity3D
* Đến với Unity3D các bạn sẽ không phải băn khoăn về khẳng năng xử lý các khái niệm đồ họa phức tạp... Tất cả đều trở nên nhanh chóng và dễ dàng với Unity3D

**2.2.3 Các khái niệm cơ bản trong Unity**

**a. GameObject**

- Một đối tượng cụ thể trong Game gọi là GameOject, có thể là nhân vật đồ vật nào đó

- Ví dụ: Cây cối, xe cộ , nhà cửa, người...

**b.** **Component**

- Một GameObject sẽ có nhiều thành phần cấu tạo nên nó như là hình ảnh(spite render), tập hợp các hành động(animation), thành phần xử lý va chạm(Collision), tính toán vật lý(physical), mã điều khiển (scrip), các thành phần khác... Mỗi thứ như vậy gọi là một Component của GameObject.

**c. Sprite**

**-** Là một hình ảnh 2D của một game object có thể là hình ảnh đầy đủ, hoặc có thể là một bộ phần nào đó



1. **Animation**

**-** Là một hình ảnh động dựa trên sự thay đổi liên tục của nhiều Sprite khác nhau

**e. Key Frame**

- Key Frame hay Frame là một trạng thái của Animation. Có thể được tạo nên từ 1 hay nhiều sprite khác nhau

**f. Prefabs**

- Là một khái niệm trong Unity, dùng để sử dụng các đôi tượng giống nhau trong game mà chỉ cần khởi tạo lại các giá trị vị trí, tỷ lệ biếng dạng và góc quay từ một đối tượng ban đầu.

- Ví dụ: Các đối tượng là đông tiền trong game Mario đều có xử lý giống nhau, nên ta chỉ việc tao một đối tượng ban đầu, còn các đối tượng sau đó ta chi việc sử dụng Prefabs, hoặc khi ta lát gạch cho một nền nhà thì một viên gạch cũng được coi là một Prefabs

**g. Sounds**

**-** Là âm thanh trong game

**h. Script**

**-** Script là tập tin có chứa các đoạn mã nguồn, dùng để khởi tạo và sử dụng các giá trị trong game

- Trong Unity có thể dùng ngôn ngữ lập trình C#, Javascript, BOO để lập trình Scrip

- Unity có hỗ trợ sẵn phần mềm MonoDeveloper dùng để lập trình

**i. Scenes**

**-** Quản lý tất cả các đối tượng trong một màn chơi của game



Ảnh minh họa Scenes

**k. Assets**

**-**  Là một game object đặc biệt trong scene, dùng để xác định tầm nhìn, quan sát và các đối tượng khác trong game.

**l. Camera**

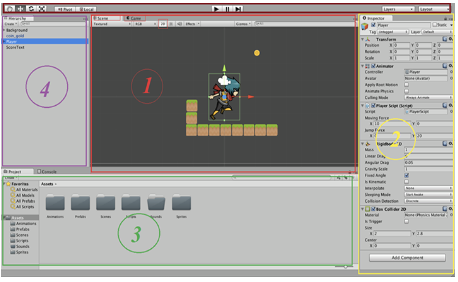
**-** Là một game bject đặc biệt trong scene, dùng để xác định tầm nhìn, quan sát các đối tượng khác trong game

**m. Transform**

**-**  Là 3 phép biến đổi tịnh tiến, quay theo các trục và giúp phóng to thu nhỏ các đỗi tượng

**2.3 Làm Quen Với Mối Trường Unity**

**2.3.1 Tìm hiểu và khởi tạo các đối tượng cơ bản trong unity**

****

Minh họa bố trí của game Unity

1. **Các thành phần và bố trí**
2. **Scene**

* Phần màu đỏ số 1
* Phần này hiển thị các đối tượng trong Scenes một cách trực quan, có thể lựa chọn các đối tượng, kéo thả, phóng to, thu nhỏ, xoay các đối tượng....
* Phần này có thể thiết lập một số thông số như hiển thị ánh sáng, âm thanh, các nhìn 2D hay 3D

1. **Game2**

* Phần màu đỏ số 1
* Phần này cho phép ta xem trước hiển thị từ scene ra game, bên cạnh tab scene

1. **Inspector**

* Phần màu vàng số 2
* Phần này hiển thị các Component của một Game object và các thông số của các component

1. **Project**

* Phần màu xanh lá cây số 3
* Phần này hiển thị thư mục Assets, chứa tất cả các tài nguyên của dự án game.
* Ở phần này, bên cạnh các tab khác, có phần Console để hiển thị các log trong quá trình debug

1. **Hierarchy**

* Phần đóng khung màu tím
* Phần này quản lý tất cả các đối tượng trong scenes, có thể chọn lựa, đổi tên, xóa các đối tượng ra khỏi game

1. **Top bar**

**-**  Phần đóng khung màu đà

**-** Phần này chứa các nút chuyên dụng

**+** Bốn nút bên trái: (1) tùy chọn cho phép dùng chuột kéo toàn bộ scenes, phóng to, thu nhỏ, lựa chọn các đối tượng trong scenes...(2) cho phép dùng chuột di chuyển các đối tượng trong sceness,(3) cho phép dùng chuột quay các đối tượng trong sceness(4) cho phép dùng chuột phóng to thu nhỏ các đối tượng .

**+** ba nút ở giữa: (1) cho phép chạy Demo game. (2)Cho phép dừng game tại một frame nào đó,(3) cho phép chạy từng frame

**+** Hai drop list bên phải: (1) cho phép hiển thị tùy chọn các layer, (2) cho phép chọn và lưu các bố trí do người dùng thiết lập

Ngoài ra còn có cửa sổ khác của Unity, như Sprite Edition, Animation, Animator, consoler... và các bạn có thể tự tuyền biến cho mình một bố cục sao cho thuận tiện với mình nhất

**2.3.2 Xử lý text, hiệu ứng trong game và chuyển màn chơi**

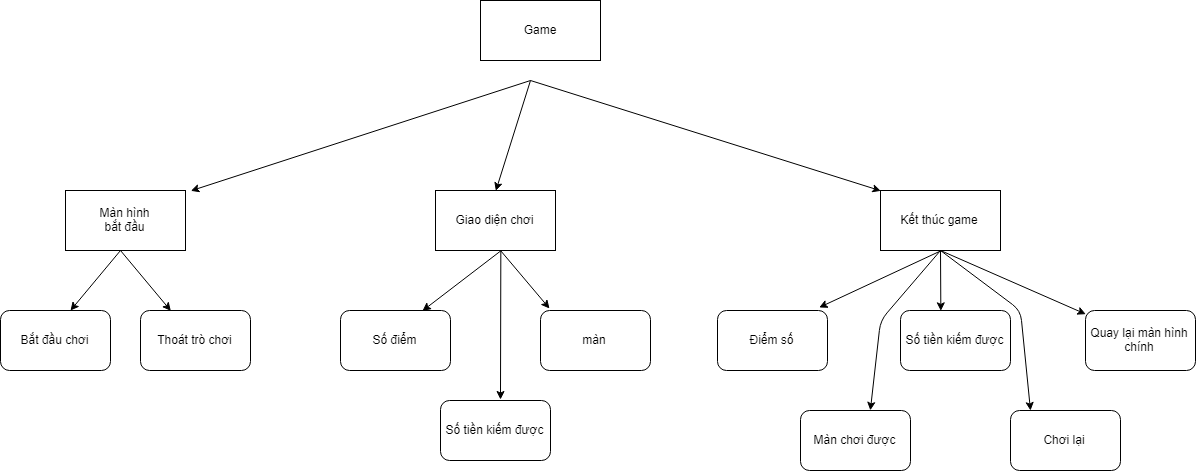
**2.3.3 Một số cơ sở dữ liệu trong game Unity**

**3 Chọn đề tài đồ án và thực hiện**

**3.1 Chọn đề tài**

- Vì thấy được ưu nhược điểm của Unity và cảm thấy làm app game sẽ thú vị hơn trên di động nên nhóm e đã quyết định viết một app game nhỏ để hoàn thành, tận dũng nhưng gì đã được học trong môn học nhập môn di động và áp dụng

**4.1. Sơ đồ phân rã chức năng**



**5. Cơ sở dữ liệu**

Phiên bản đã được cài mặc định các màn chơi game, và nếu có cập nhật thêm thì sẽ thêm màn và thêm hiệu ứng cũng như giao diện đồ họa được đẹp hơn và hoành tráng hơn

**6. Giao diện thiết kế**

**6.1. Danh sách màn hình**

* Màn hình bắt đầu game
* Màn hình vào game
* Màn hình chuyển màn
* Màn hình màn hình kết thúc game

**6.2. Chi tiết thiết kế**

* 1. Màn hình bắt đầu game

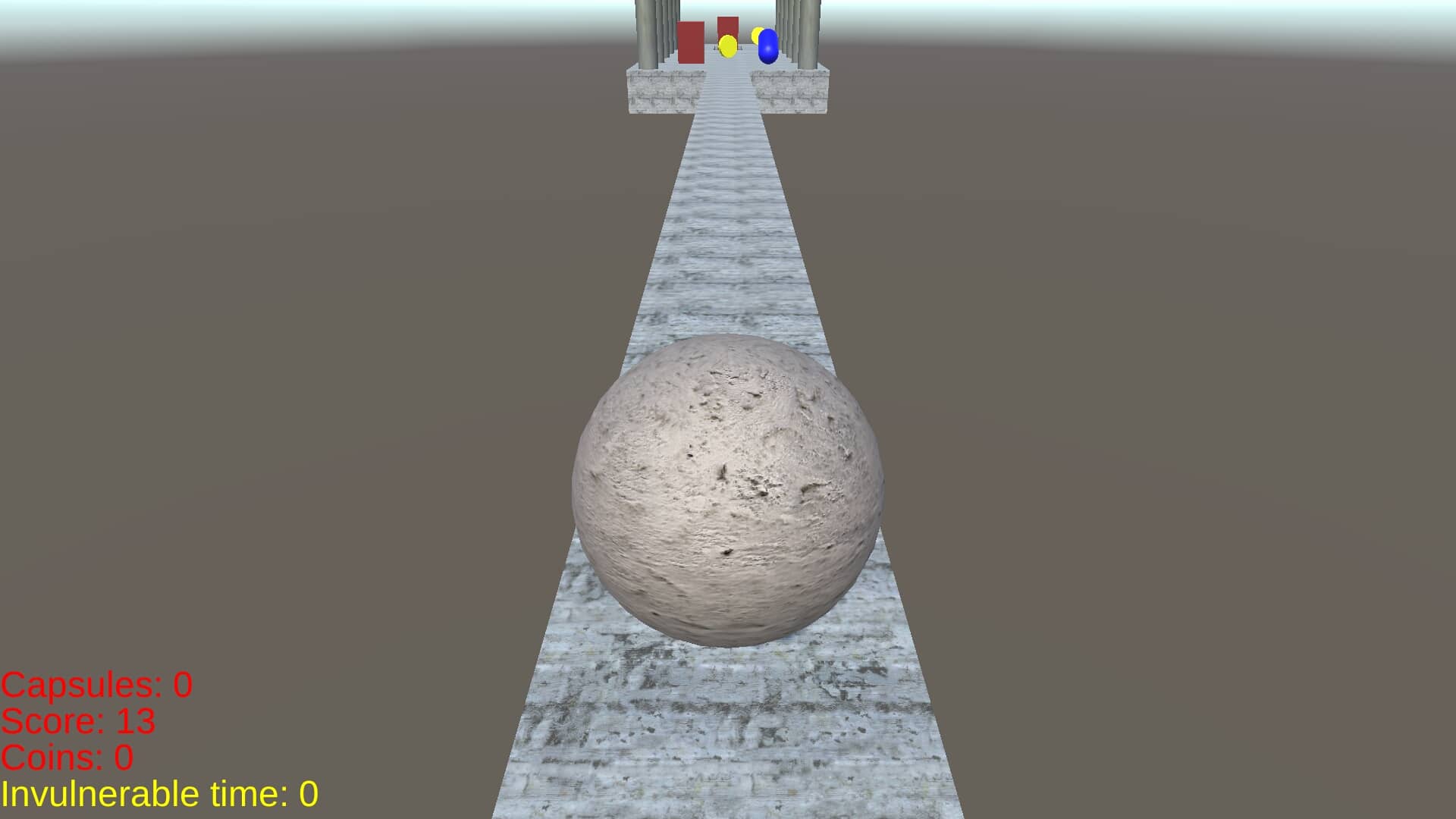


2

1

|  |  |  |
| --- | --- | --- |
| STT | Thành phần | Xử lý |
| 1 | Bắt đầu game | Click vào để bắt đầu chơi game |
| 2 | Thoát game | Click để thoát khỏi trò chơi |

* 1. Màn hình vào game

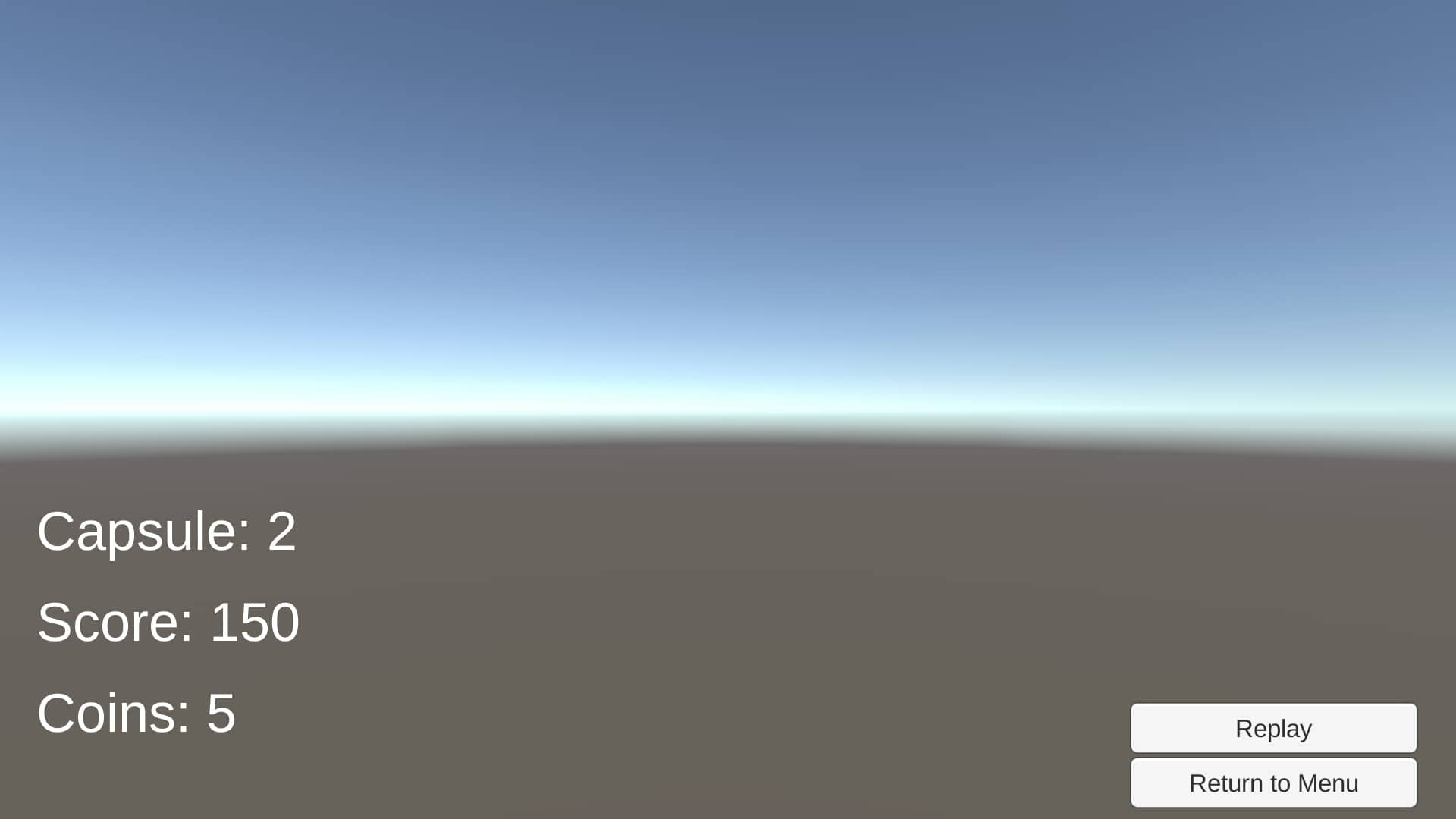


* 1. Màn hình chuyển màn



1

* 1. Màn hình kết thúc game



3

2

1

55

4

|  |  |  |
| --- | --- | --- |
| STT | Thành phần | Xử lý |
| 1 | Hiển thị màn chơi |  |
| 2 | Hiển thị số điểm chơi được |  |
| 3 | Hiển thị số tiền thu thập được |  |
| 4 | Hiển thị chơi lại | Click chuột vào button nếu muốn chơi lại game |
| 5 | Hiển thị quay lại màn hình bắt đầu | Click chuột nếu muốn quay lại màn hình bắt đầu |

**7. Tài liệu tham khảo**

* Trang chủ React Native : facebook.github.io/react-native/docs/tutorial.html
* React Native Navigation : [reactnavigation.org](https://reactnavigation.org)
* React Native Swiper : github.com/leecade/react-native-swiper
* React Native Collapsing Toolbar : github.com/cesardeazevedo/react-native-collapsing-toolbar
* React Native Modal Photo View : github.com/guopeng1994/react-native-modal-photo-view
* React Native Render HTML : github.com/archriss/react-native-render-html
* React Native Open Maps : github.com/brh55/react-native-open-maps
* React Native Vector Icons : github.com/oblador/react-native-vector-icons
* React Native Orientation : github.com/yamill/react-native-orientation
* Unity3D: https://unity3d.com/fr

**8. Phân công nhiệm vụ**

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ tên** | **Nhiệm vụ** |
| 13520990 | Võ Duy Tuấn | * Làm giao diện * Làm bài tập nhóm |
| 15520778 | Cao Du Thái | * Code game logic |
| 13521007 | Phan Văn Tùng | * Viết báo cáo * Làm prefab cho game |

**9. Tổng kết – Đánh giá**

Unity là một nền tảng hay và có sức phát triển cũng như ứng dụng rộng lớn, thêm vào đo nó hoàn toàn miễn phí cho những ai muốn học, đây cũng là những mã nguồn mở và nếu theo học cũng dễ dàng cho người mới bắt đầu,

Chúng em xin cảm ơn thầy Huỳnh Tuấn Anh đã giới thiệu cho chúng em một số kiến thức hữu ích về lập trình ứng dụng di động để hoàn thành tốt đồ án này.

Mặc dù cũng mới tiếp cận và phát triển ứng dụng nên còn khá khó khăn và bỡ ngỡ, rất mong được sự cảm thông và chia sẻ của thầy, và trong tương lai chắc chắn Unity sẽ là công nghệ làm game được nhiều người quan tâm hơn nữa và có bước phát triển rộng mở thêm hơn nữa

**--- HẾT ---**