

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



TIỂU LUẬN MÔN PHÁT TRIỂN TRÒ CHƠI

XÂY DỰNG VÀ PHÁT TRIỂN GAME 2D – CATCHING BALL

Giảng viên hướng dẫn: **ThS. Hà Lê Hoài Trung**

Sinh viên thực hiện: **Chiêm Trường An – 51800001**

Nguyễn Ngọc Bảo Trâm – 51800502

Lớp: 18050203

Khóa: 22

Nhóm: 03

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN



TIỂU LUẬN MÔN PHÁT TRIỂN TRÒ CHƠI

XÂY DỰNG VÀ PHÁT TRIỂN GAME 2D – CATCHING BALL

Giảng viên hướng dẫn: **ThS. Hà Lê Hoài Trung**

Sinh viên thực hiện: **Chiêm Trường An – 51800001**

Nguyễn Ngọc Bảo Trâm – 51800502

Lớp: 18050203

Khóa: 22

Nhóm: 03

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021

LỜI CẢM ƠN

Phát triển trò chơi là một môn học rất hay và bổ ích, chúng em xin chân thành cảm ơn thầy ThS. Hà Lê Hoài Trung đã hướng dẫn và tạo điều kiện tốt nhất để chúng em hoàn thành bài tiểu luận môn Phát triển trò chơi với đề tài “Xây dựng và phát triển game 2D tên *Catching Ball*”. Giúp chúng em có thêm kiến thức về những giai đoạn từ lên ý tưởng đến hiện thực game, hiểu biết khái quát hơn về cách sử dụng game engine để xây dựng một game là như thế nào.

Trong quá trình thực hiện bài tập này nhóm vẫn khó tránh khỏi những sai sót không mong muốn, kính mong thầy có thể góp ý và giúp đỡ chúng em. Nhóm xin chân thành cảm ơn thầy.

BÀI TẬP LỚN NÀY ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Chúng tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của thầy Vũ Đình Hồng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong bài tập lớn này còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào chúng tôi xin hoàn toàn chịu trách nhiệm về nội dung bài tập lớn của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 20 tháng 10 năm 2020

Tác giả

(ký tên và ghi rõ họ tên)

Chiêm Trường An

(Trưởng nhóm)

Nguyễn Ngọc Bảo Trâm

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(ký và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(ký và ghi họ tên)

TÓM TẮT

Xã hội ngày càng phát triển kéo theo nhu cầu sử dụng các dịch vụ giải trí của con người ngày càng tăng cao. Cùng với sự phát triển của công nghệ thông tin, trong những năm gần đây ngành công nghệ game nổi lên và đem lại nguồn doanh thu khổng lồ cho các lập trình viên. Có thể nói hiện nay game không chỉ là một phương tiện giải trí cơ bản, nó còn là một lĩnh vực tiềm năng và đáng để theo đuổi.

Về công cụ hỗ trợ phát triển game, nhóm sử dụng Engine Unity – một game engine rất phổ biến hiện nay nhằm thực hiện việc xây dựng và phát triển trò chơi *hứng bóng 2D* mang tên “*Catching Ball!*”. Thiết kế game là một công việc không hề dễ dàng, đòi hỏi rất nhiều thời gian và công sức để hoàn thành, tuy nhiên bản thân sẽ tích lũy được những trải nghiệm thú vị, cũng như qua đó học hỏi thêm những kiến thức mới về nền tảng Unity Framework và ngôn ngữ C#.

Trò chơi *Hứng bóng* là một thể loại game cổ điển đã được thịnh hành từ rất lâu với nhiều phiên bản và thể loại khác nhau. Nay nhóm em sẽ viết lại và phát triển thêm các chi tiết mới mẻ và thú vị cho game thêm sinh động hơn. Luật chơi rất đơn giản nhưng đòi hỏi sự khéo léo và tập trung cao độ của người chơi.

Mục tiêu của bài tập lớn lần này, nhóm mong muốn áp dụng được những kiến thức đã học để xây dựng và phát triển một trò chơi 2D đạt được hiệu quả cao nhất.

MỤC LỤC

TIỂU LUẬN MÔN PHÁT TRIỂN TRÒ CHƠI.....	1
TIỂU LUẬN MÔN PHÁT TRIỂN TRÒ CHƠI.....	2
LỜI CẢM ƠN	3
BÀI TẬP LỚN NÀY ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG	4
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	5
TÓM TẮT	6
MỤC LỤC	7
CHƯƠNG I: GIỚI THIỆU.....	9
1.1 Giới thiệu đề tài	9
1.2 Phương pháp sử dụng	9
1.2.1 Game engine - Unity Engine.....	9
1.2.2 Ngôn ngữ lập trình – C#, Javascript	11
1.3 Lý do và mục đích chọn đề tài	11
1.4 Một số thông tin về game.....	11
1.4.1 Tên game.....	11
1.4.2 Bản quyền	11
1.4.3 Phiên bản.....	11
1.4.4 Ý tưởng game.....	12
1.4.5 Luật chơi	12
1.4.6 Đối tượng nhắm đến.....	12
1.4.7 Tóm tắt dòng game	12
CHƯƠNG II :XÂY DỰNG GAME	12
2.1 Menu Game.....	13
2.2 Player.	13
2.3 Enemy.	13
2.4 GameController.	14
2.5 UiController	14

CHƯƠNG III: HIỆN THỰC	16
3.1 Menu chính	16
3.2 Setting	16
3.3 Khung cảnh trong game	17
3.4 Giao diện kết thúc game	18
CHƯƠNG IV: KẾT LUẬN	19
TÀI LIỆU THAM KHẢO.....	20

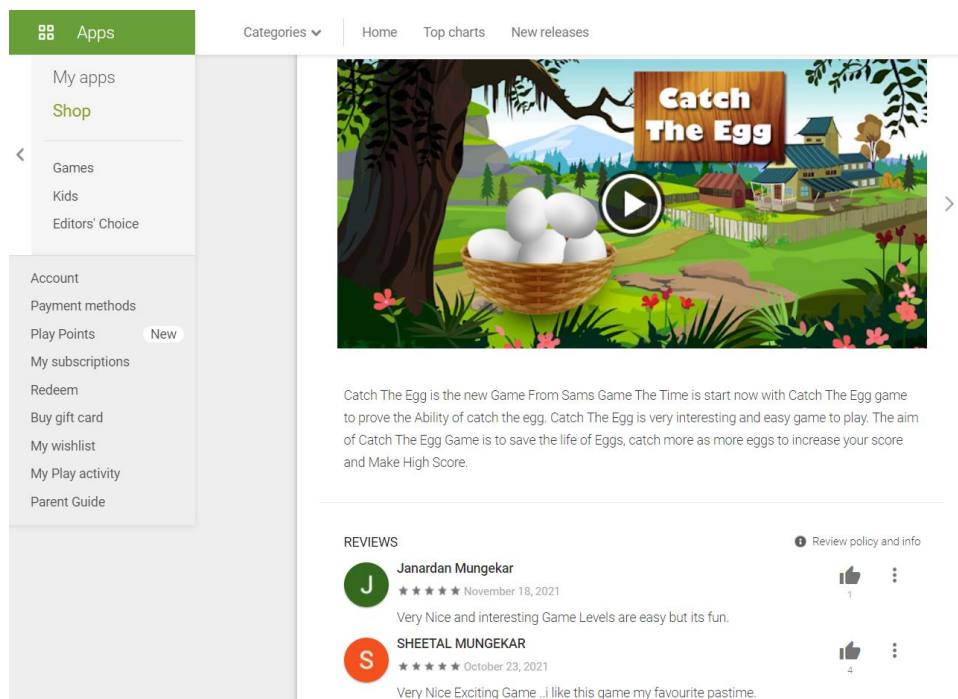
CHƯƠNG I: GIỚI THIỆU

1.1 Giới thiệu đề tài

Trong xã hội ngày càng phát triển, song song với những tựa game 3D nổi tiếng, một tầng lớp các bạn trẻ lại muốn quay về ngày xưa tìm lại những tựa game 2D trước đây gắn liền với tuổi thơ của họ. Từ đó, nhu cầu về game 2D cổ điển tăng cao.

Để đáp ứng được nhu cầu trong bối cảnh thị trường hiện nay, nhóm đã triển khai thực hiện dự án game mini 2D mang tên *Catching Ball!* bằng Unity Engine. Game được triển khai với mục tiêu giúp người chơi xả stress sau những ngày làm việc căng thẳng, tạo cảm giác muốn chinh phục trò chơi, thử thách bản thân,...

Catching ball! được xây dựng dựa theo tựa game *hứng trứng* huyền thoại nổi tiếng trước đây. Do nhu cầu ngày càng tăng, tựa game này phát sinh ra nhiều phiên bản và biến thể khác nhau, từ đó chúng ta sẽ dễ dàng tìm thấy và cài đặt được những tựa game liên quan trên các chợ Ứng dụng như Google Play, CH Play, App Store, Microsoft Store.



Hình 1 Game hứng trứng trên Google Play

1.2 Phương pháp sử dụng

Nhóm sử dụng nền tảng ngôn ngữ C# và Unity Engine để hiện thực game.

1.2.1 Game engine - Unity Engine

Unity là một 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 trên websites. Unity tạo ra nhiều loại hình game đa dạng, hỗ trợ nhập vào vào rất nhiều mô hình định dạng khác nhau, hỗ trợ mô hình trực tiếp.

Lượng tài liệu hướng dẫn nhiều, có một cộng đồng lớn và diễn đàn riêng. Unity có hai phiên bản là Unity Pro có tính phí và Unity Free để người dùng dễ dàng lựa chọn, do đó Unity không chỉ dành cho một công ty lớn mà cả giới làm game nghiệp dư cũng có thể sử dụng.

1.2.2 Ngôn ngữ lập trình – C#.

Unity3D 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 nên cũng khá dễ sử dụng. Ngôn ngữ lập trình chính của Unity là C#, ngoài ra còn có hỗ trợ cho Javascript.

1.3 Lý do và mục đích chọn đề tài

Thứ nhất, chúng em chọn đề tài này là vì muốn có thêm kiến thức về việc tìm hiểu thêm các game engine khác trên thị trường (cụ thể là unity engine), cách thức hoạt động của game engine và cấu trúc ngôn ngữ của nó (C#).

Thứ hai, loại hình game 2D cổ điển này đang ngày càng phát triển và có chỗ đứng trong xã hội. Khi mà nhu cầu đòi hỏi về những game mang tính chinh phục, thú vị, lôi cuốn ngày càng tăng.

1.4 Một số thông tin về game

1.4.1 Tên game

Catching Ball! – Hứng bóng

1.4.2 Bản quyền

Bản quyền thuộc về TA Software Team

1.4.3 Phiên bản

Phiên bản 1.0 - Ban hành ngày 01/12/2021

Tác giả: TA Software Team

1.4.4 Ý tưởng game

Trong game này, những quả bóng từ trên cao rơi xuống một cách ngẫu nhiên, người chơi điều khiển thanh chắn di chuyển qua trái qua phải để hứng bóng, nếu hứng được một quả bóng người chơi sẽ được cộng thêm điểm, nếu làm rơi sẽ kết thúc game. Nếu hứng phải boom cũng kết thúc game. Và nếu hứng phải hộp bí ẩn thì sẽ nhận ngẫu nhiên các hiệu quả khác nhau. Trong quá trình chơi, người chơi phải né tránh boom và ăn điểm liên tục, tương ứng với số điểm càng cao thì độ khó càng tăng lên theo các cột mốc 40 và 60. Mỗi lần đổi mức chơi thì số lượng banh và boom sẽ tăng lên càng nhiều.

1.4.5 Luật chơi

Người chơi sẽ điều khiển một thanh trượt ở phía dưới cửa sổ di chuyển sang trái hoặc sang phải để hứng bóng. Nhiệm vụ của người chơi là phải hứng thật nhiều bóng để thu thập được thật nhiều điểm.

Sẽ có ba loại bóng:

- Bóng có điểm: Người chơi cần cố gắng thu thập thật nhiều bóng này để gia tăng số điểm của player.
- Hộp bí ẩn: Trong quá trình hứng bóng, nếu người chơi hứng trúng hộp bí ẩn thì có thể nhận được các hiệu ứng như tăng/giảm tốc độ người chơi, thu nhỏ người chơi, đảo chiều di chuyển và có thể bị trừ hết số điểm hiện tại.
- Bóng boom: Nếu người chơi hứng trúng bóng boom thì người chơi thua cuộc và kết thúc trò chơi.

1.4.6 Đối tượng nhắm đến

Độ tuổi: Mọi lứa tuổi

1.4.7 Tóm tắt dòng game

Mở đầu game người chơi sẽ loading vào màn hình hiển thị main menu của game với các tính năng “Start Game”, “Exit Game”, “Volume”. Đối với nút “Start Game” người chơi sẽ bắt đầu game bằng thanh hứng bóng di chuyển trái phải để hứng những quả banh đồng thời né các quả boom trong quá trình đó sẽ có tỉ lệ xuất hiện những hộp bí ẩn đi kèm với các hiệu ứng khác nhau có thể là hiệu ứng bắt lời hoặc hiệu ứng có lợi. Đối với nút “Exit Game”, người chơi sẽ thoát khỏi ứng dụng. “Volume” sẽ giúp người chơi thay đổi âm lượng cho phù hợp với sở thích.

CHƯƠNG II :XÂY DỰNG GAME

2.1 Menu Game.

- Tạo background chọn UI -> Image. Chọn ảnh dạng sprite làm background . Ở component Rect Transform chọn stretch full màn hình.
- Tạo TitleText UI -> Text , size là 110 và căn giữa màn hình.
- Tạo các button chọn UI -> Button. Trong button tạo thêm 1 icon nhấp chuột phải vào object button chọn Ui -> Image và chọn image đại diện cho chức năng button. Lần lượt là Start, Setting và Quit. Ở các button cho thêm các animation để thay đổi khi di chuyển chuột vào . Tạo animation Window -> Animation . Bấm record trong tag Animation để tạo hình ảnh trạng thái
 - o Normal : với các thông số như sau : Image.Transparent của Button là 0 , Text Size là 68, Image.Transparent của Icon là 0(để ẩn icon)
 - o Highlight : Image.Transparent của Button và Icon là 1, Text Size là 48, Position.Text di chuyển sang phải.
- Tạo Setting Volume Ui -> Slider. Để lưu giá trị volume toàn cục ta gọi PlayerPrefs để lưu giá trị volumeSlider.value. Tiếp theo ta cần khởi tạo hàm ChangeVolume trong đó thực hiện việc thay đổi giá trị AudioListener.volume = volumeSlider.value. Và hàm này được gọi trong eventOnChange của Slider.
- LoadBackground khi click vào nút Start . Khi nhấn vào nút Start sẽ chạy animation Load FadeBackground , transparent của FadeBackground chạy từ 0 đến 1 và khi chạy đến cuối frame sẽ kích hoạt event SceneMenu2Game. Hàm SceneMenu2Game sẽ thực hiện load Scene thứ 2 bằng script SceneManager.LoadScene(“Game”).

2.2 Player.

- Cách di chuyển cho Player, sẽ nhận xDir = Input.GetAxis(“Horizontal”) . Để làm cho Player di chuyển Rigidbody.velocity = xDir.speed.
- Xét giới hạn di chuyển Player , giới hạn di chuyển trong khung nhìn của người chơi bằng method position của Rigidbody.
- Xét va chạm của Player và xử lý va chạm qua OnTriggerEnter, Player sẽ có 3 dạng va chạm là Ball, Boom và Present.
 - o Ball : Kiểm tra other.gameObject.CompareTag(Ball) sẽ gọi hàm cộng điểm và Destroy(other.gameObject).
 - o Boom : Kiểm tra other.gameObject.CompareTag(Boom) sẽ Destroy(other.gameObject) và gọi hàm setGameOver.
 - o Present : Kiểm tra other.gameObject.CompareTag(Present) và tương ứng index từ 0 – 4 thì sẽ nhận được các hiệu ứng khác nhau.
 - 0 và 1 sẽ gọi hàm tăng/giảm tốc độ của Player.
 - 2 thì gọi hàm thay đổi kích thước của Player và có hiệu quả trong 3s.
 - 3 thì gọi hàm thay đổi chiều mà Player đang chuyển động.
 - 4 thì gọi hàm trừ hết tất cả số điểm của Player đang sở hữu.

2.3 Enemy.

- Có 3 loại enemy : Ball, Boom và Present. Tương ứng mỗi loại có các chức năng khác nhau.
 - o Ball sẽ có ngẫu nhiên 5 quả bóng và có số điểm tương ứng từ 1 đến 5.
 - o Boom sẽ làm cho gameover.
 - o Present sẽ cho ngẫu nhiên một chức năng có lời/hại cho người chơi.
- Loại bỏ việc rơi theo trọng lực của các enemy. Rigidbody unchecked Use Gravity và Is Kinematic. Vì đã bỏ việc rơi theo trọng lực nên cần điều khiển nó bằng script. Tạo script tên Mover , trong hàm Update ta xét `transform.position += Vector3.down * speed * Time.deltaTime`. Sau đó gắn script nào vào các enemy.
- Để cho hình ảnh của các enemy được sống động ta cần làm enemy vừa rơi vừa xoay tròn. Tạo script tên Rotation, để làm thay đổi rotation ta dùng `Rigidbody.angularVelocity = Random.insideUnitSphere`. Sau đó gắn script vào các enemy.

2.4 GameController.

- Hàm tạo các enemy. Đặt tên hàm là SpawnObject tham số đầu vào là GameObject. Trong hàm tạo vị trí để xuất hiện Object ngẫu nhiên bằng `spawnPos = new Vector2(Random.range(-9.8f,10f),7.5f)` là ngẫu nhiên vị trí x trong khoảng -9.8 đến 10. Sau đó tạo Object , `Instantiate(obj, spawnPos, Quaternion.identity)`.
- Để tạo Ball theo thời gian . Đầu tiên ta đặt biến `m_spawnTimeBall` là khoảng cách thời gian mỗi lần tạo. Kiểm tra nếu `m_spawnTimeBall < 0` và `! IsGameOver` thì sẽ random 1 quả banh trong 5 với số điểm nhận được tương ứng của mỗi quả banh. Sau đó gọi hàm SpawnObject để tạo quả banh tương ứng.
- Để tạo Boom cũng tương tự tạo banh.
- Để tạo Present thì có khác biệt là thêm yếu tố tỉ lệ xuất hiện của present để tránh việc cứ đúng thời gian thì nó luôn luôn tạo 1 present. Kiểm tra nếu `m_spawnTimePresent < 0` và `! IsGameOver` thì tạo tiếp một biến tỉ lệ là `randomPercent` kiểm tra nếu `randomPercent < 30` thì sẽ gọi hàm SpawnObject còn không thì nó sẽ đợi tới khoảng thời gian tạo Present lần sau và tiếp tục kiểm tra `randomPercent`.
- Kiểm tra GameOver. Nếu `IsGameOver` bằng true sẽ cho `Time.timeScale = 0`. Đồng thời gọi hàm `showPanelGV` bên UIControler. Tiếp theo gọi `bestscore` được lưu trong PlayerPref và kiểm tra nếu score hiện tại `> bestscore` thì sẽ lưu lại `PlayerPref.SetInt("best_score",score)`.
- Tạo hàm `rePlay` để chơi lại.
`SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex)`. Nút PlayAgain ở panelGameOver bên UI thêm event replay thì khi nhấn sẽ chạy hàm.
- Hàm exit nếu người chơi nhấn thì sẽ chạy animation FadeBackground và ở cuối frame sẽ kích hoạt event `SceneManager.LoadScene("MainMenu")` để trở về màn hình menu.

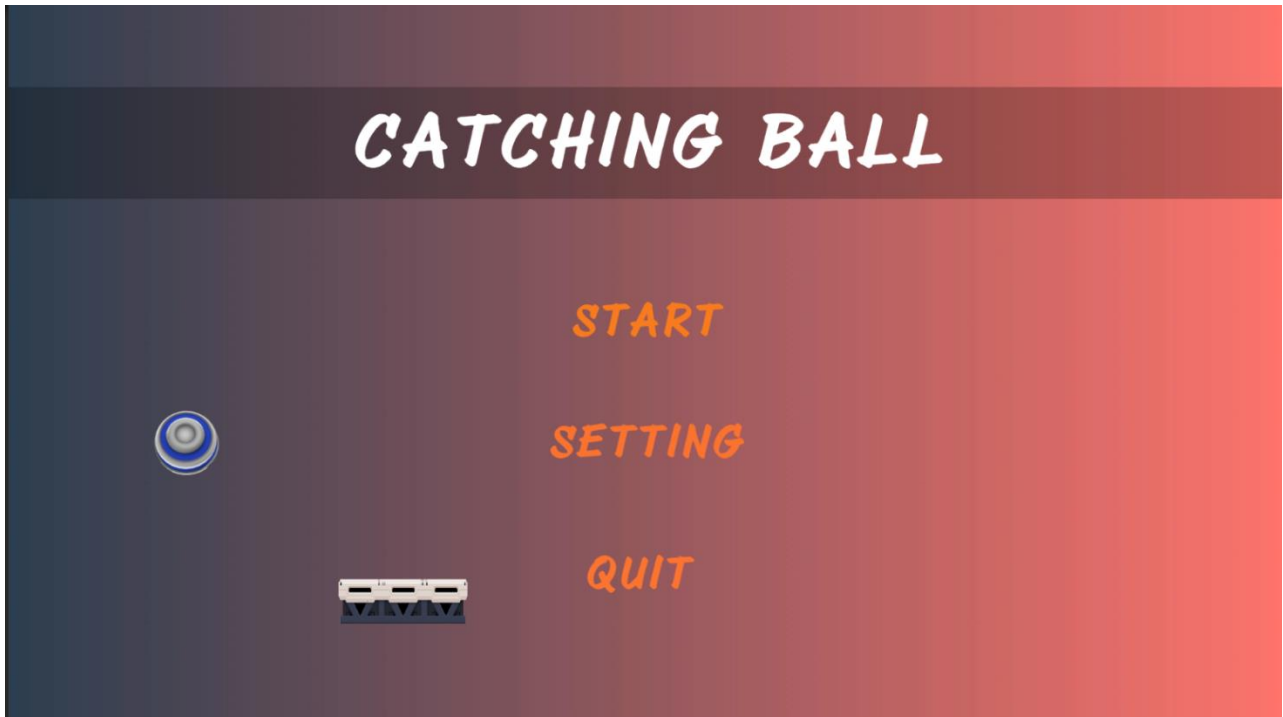
2.5 UiController

- Tạo hàm `setScore` để mỗi lần người chơi ăn ball sẽ được cập nhập điểm lên UI có thể xem được. `textScore.text = string txt`. Và các hàm `setBestScore`, `setYourScore` và `setNotify` cũng tương tự.
- Tạo hàm `isShowPanelGV` để mỗi khi người chơi ăn boom sẽ game over và hiện panel lên. Biến đầu vào là boolean `isShow` , `panelGV.SetActive(isShow)` để người dùng có thể hiện và ẩn tùy mục đích.

- Mỗi khi Player nhận được Object Present cần thông báo cho Player là nhận được chức năng gì. Ta cần tạo một hàm IEnumerator FadeOutCR() để Fade dòng thông báo trong hàm cần khai báo 2 biến currentTime là thời gian hiện tại và duration là khoảng thời gian Fade. Đưa vào vòng lặp while (currentTime < duration) thay đổi biến alpha = Mathf.Lerp(1f, 0f, currentTime / duration) và thay đổi textNotify.color = new Color(textNotify.color.r, textNotify.color.g, textNotify.color.b, alpha). Sau đó tăng biến currentTime. Tiếp theo tạo một hàm public FadeOut() và gọi StartCoroutine(FadeOutCR()) . Hàm FadeOut được gọi bên Player mỗi khi chạm vào Present.

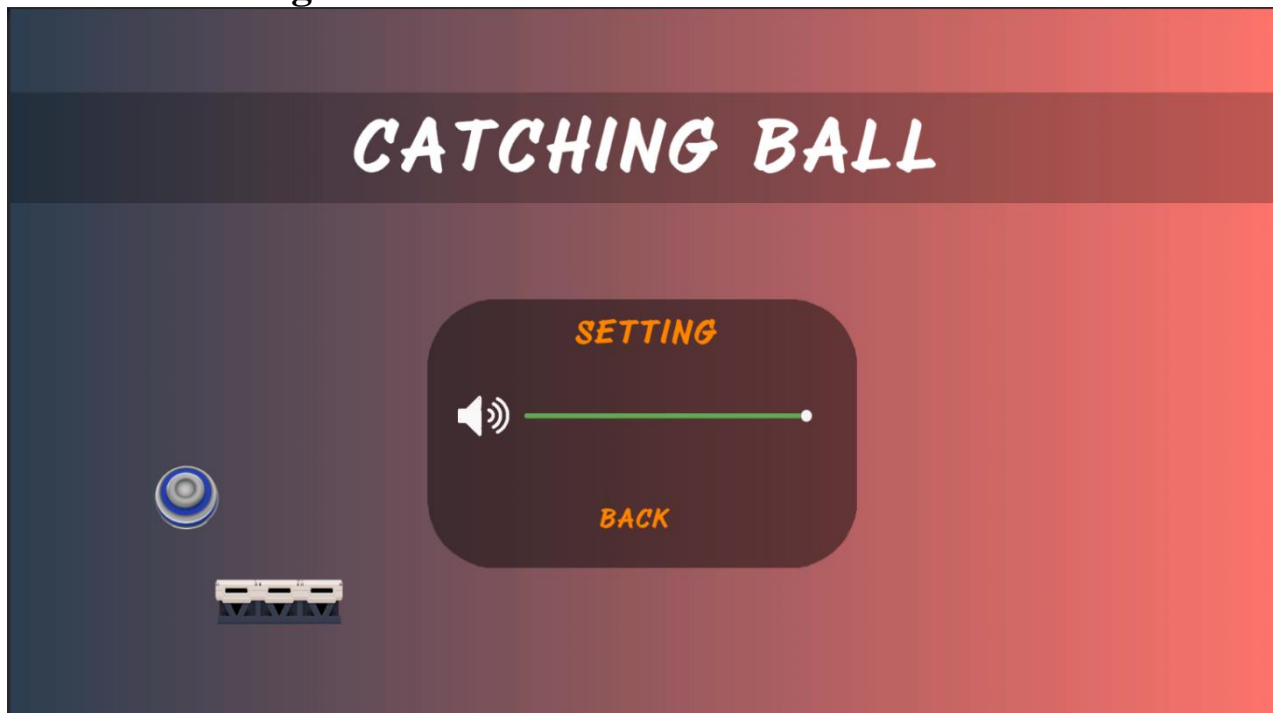
CHƯƠNG III: HIỆN THỰC

3.1 Menu chính



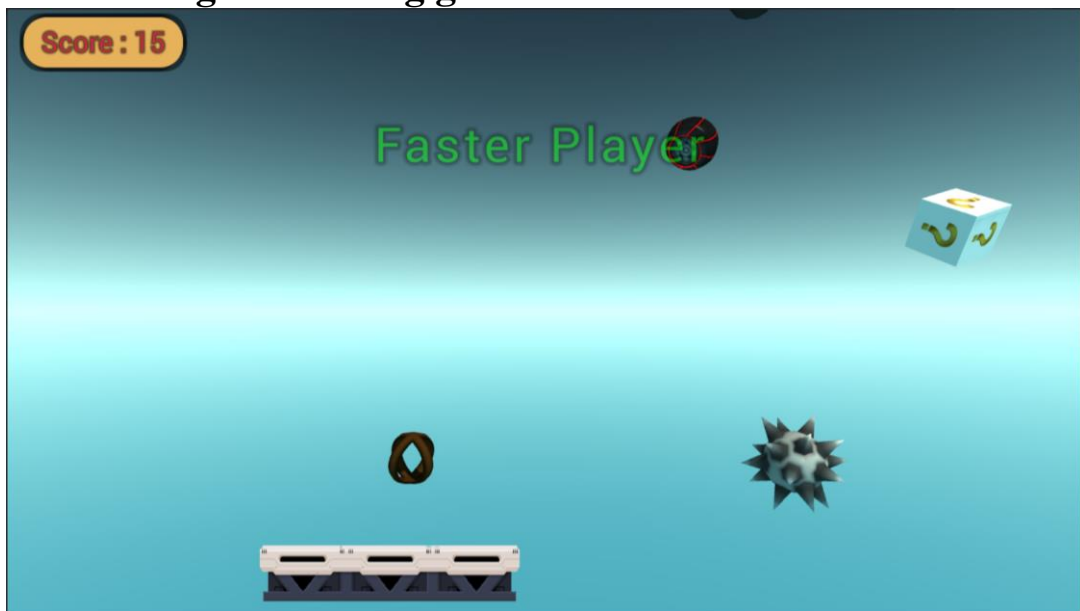
Hình 2. Giao diện menu chính

3.2 Setting



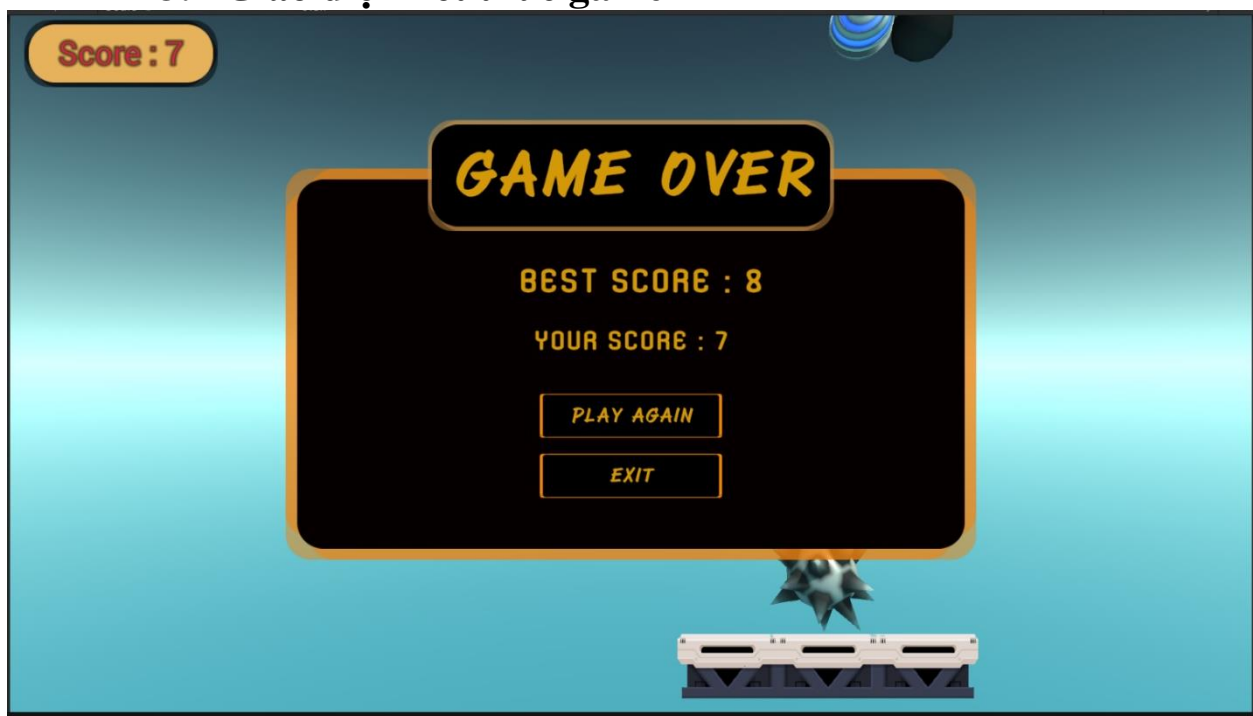
Hình 3. Giao diện hướng dẫn

3.3 Khung cảnh trong game



Hình 4. Giao diện trò chơi

3.4 Giao diện kết thúc game



Hình 5. Giao diện kết thúc trò chơi

CHƯƠNG IV: KẾT LUẬN

Trong tương lai , nhóm sẽ phát triển một số game mini 2D mới lạ, đánh thẳng nhu cầu thị trường hiện nay, tìm hiểu chi tiết hơn về tâm lý người chơi để đưa ra những trải nghiệm tốt nhất cho người chơi.

TÀI LIỆU THAM KHẢO

Tài liệu Tiếng Anh:

- [1] Vũ Đình Hồng, 2019, Slide Bài giảng môn Phát triển Trò chơi, Đại học Tôn Đức Thắng
- [2] Steve Rabin, 2010, Introduction to Game Development, 2nd Edition, Cengage Learning.
- [3] Jesse Schell, 2008, The Art of Game Design - A Book of Lenses, 1st Edition, Morgan Kaufmann.

Tài liệu Internet: