

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM

LÊ ĐỨC SANG
SƠ TUẤN HOÀNG

KHÓA LUẬN TỐT NGHIỆP
KẾT HỢP UNITY3D & VUFORIA
XÂY DỰNG ỨNG DỤNG GAME MINH HỌA

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

TP. HỒ CHÍ MINH, 2016

ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA CÔNG NGHỆ PHẦN MỀM

LÊ ĐỨC SANG – 11520326

SƠ TUẤN HOÀNG – 11520122

KHÓA LUẬN TỐT NGHIỆP
KẾT HỢP UNITY3D & VUFORIA
XÂY DỰNG ỨNG DỤNG GAME MINH HỌA

KỸ SƯ NGÀNH KỸ THUẬT PHẦN MỀM

GIẢNG VIÊN HƯỚNG DẪN
THS. PHẠM THI VƯƠNG

TP. HỒ CHÍ MINH, 2016

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số
ngày của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

1. – Chủ tịch.
2. – Thư ký.
3. – Ủy viên.
4. – Ủy viên.

LỜI CẢM ƠN

Trước hết nhóm xin gửi lời cảm ơn chân thành đến thầy, ThS. Phạm Thi Vương, khoa Công Nghệ Phần Mềm, Trường Đại học Công Nghệ Thông Tin. Trong suốt thời gian thực hiện khóa luận, thầy đã dành rất nhiều thời gian và tâm huyết trong việc hướng dẫn nhóm. Thầy đã luôn có những định hướng, góp ý, sửa chữa những chỗ sai sót giúp nhóm có thể đi được đúng hướng. Khóa luận có được kết quả ngày hôm nay là nhờ sự nhắc nhở, đôn đốc và giúp đỡ nhiệt tình của thầy.

Nhóm cũng xin trân trọng cảm ơn tất cả thầy cô trong khoa Công Nghệ Phần Mềm, cũng như các thầy cô đã giảng dạy, giúp đỡ chúng em trong suốt thời gian học tập ở trường. Những kiến thức nền tảng và chuyên môn mà chúng em học được từ các thầy cô đã giúp chúng em có thể hoàn thành khóa luận này.

Chúng con xin nói lên lòng biết ơn sâu sắc đối với cha mẹ, người đã chăm sóc, nuôi dạy chúng con thành người.

Xin chân thành cảm ơn các anh chị và bạn bè đã ủng hộ, giúp đỡ và động viên chúng em trong thời gian học tập và nghiên cứu ở trường cũng như ngoài xã hội.

Mặc dù đã rất cố gắng để hoàn thành nhưng vẫn không thể tránh khỏi những thiếu sót. Chúng em kính mong nhận được sự góp ý của quý thầy cô.

Nhóm khóa luận

MỤC LỤC

Chương 1.	THỰC TẾ ẢO TĂNG CƯỜNG – AUGMENTED REALITY	3
1.1.	Tổng quan về thực tế ảo tăng cường – Augmented Reality	3
1.2.	Lịch sử phát triển	4
1.2.1.	Giai đoạn khởi đầu	4
1.2.2.	Giai đoạn phát triển	5
1.3.	Đặc trưng	6
1.4.	Kiến trúc hệ thống	6
1.5.	Một số bộ SDK hỗ trợ	7
1.6.	Ứng dụng thực tiễn	10
Chương 2.	VUFORIA AUGMENTED REALITY SDK VÀ UNITY3D	15
2.1.	Vuforia Augmented reality SDK.....	15
2.1.1.	Giới thiệu.....	15
2.1.2.	Giấy phép Vuforia	17
2.1.3.	Các tính năng chính.....	18
2.1.4.	Cấu trúc của vuforia	19
2.1.5.	Trackables.....	21
2.1.6.	Hệ thống quản lý Target.....	30
2.2.	Tìm hiểu về game engine.....	31
2.2.1.	Phân loại game engine.....	31
2.2.2.	Một số game engine trên thị trường	33
2.3.	Unity	34
2.3.1.	Giới thiệu.....	34

2.3.2.	Các tính năng chính	37
2.3.3.	Ngôn ngữ lập trình.....	40
2.3.4.	Kiến trúc Component	42
2.3.5.	Các Component trong Unity.....	44
2.3.6.	Script.....	50
2.3.7.	Physics	53
Chương 3.	ỨNG DỤNG AUGEMENTED REALITY VÀO PHÁT TRIỂN GAME	57
3.1.	Tổng quan	57
3.1.1.	Khảo sát các sản phẩm tương tự.....	57
3.1.2.	Bối cảnh nghiên cứu	60
3.1.3.	Động lực nghiên cứu	62
3.1.4.	Mục tiêu nghiên cứu	62
3.2.	Thiết kế game	63
3.2.1.	Tổng quan về game	63
3.2.2.	Game play và Mechanics	64
3.2.3.	Thiết kế màn chơi.....	69
3.2.4.	Hệ thống giao diện	71
3.2.5.	Môi trường và công cụ phát triển	72
3.2.6.	Sơ đồ thiết kế.....	73
3.2.7.	Thiết kế giao diện	78
3.3.	Thiết kế bộ công cụ hỗ trợ.....	82
3.3.1.	Bộ chữ cái.....	82
3.3.2.	Lăng kính.....	84

3.3.3.	Để giữ thiết bị.....	85
Chương 4.	ỨNG DỤNG THỰC TIỄN	86
4.1.	Cài đặt thực tiễn.....	86
4.2.	Hướng dẫn chơi game.....	86
4.3.	Đánh giá người dùng	90
4.3.1.	Số liệu thống kê	90
4.3.2.	Phân tích số liệu và đưa ra kết luận.....	91
Chương 5.	TỔNG KẾT	92
5.1.	Kết quả đạt được.....	92
5.2.	Khó khăn.....	94
5.3.	Hướng phát triển.....	94
TÀI LIỆU THAM KHẢO.....		95

DANH MỤC BẢNG

Bảng 2.1 Nền tảng phát triển ứng dụng	17
Bảng 2.2 Tỷ lệ sử dụng các ngôn ngữ trong Unity	40
Bảng 2.3 Các thuộc tính của MonoBehaviour	50
Bảng 2.4 Các phương thức của MonoBehaviour	51
Bảng 2.5 Các phương thức cần thừa kế của MonoBehaviour	53
Bảng 2.6 So sánh tốc độ xử lý vật lý của CPU và GPU	53
Bảng 2.7 Các thuộc tính của Rigidbody	55
Bảng 2.8 Các loại Collider	56
Bảng 3.1 Mô tả game Words.....	58
Bảng 3.2 Mô tả Learn English Vocabulary Daily.....	58
Bảng 3.3 Mô tả game Tangram.....	59
Bảng 3.4 Danh sách các use case trong sơ đồ use case của game myWords	74
Bảng 3.5 Các thành phần trong màn hình game start - myWords	78
Bảng 3.6 Các thành phần trong màn hình game play – myWords	79
Bảng 3.7 Các thành phần trong màn hình game pause– myWords	80
Bảng 3.8 Các thành phần trong màn hình world game – myWords	81
Bảng 3.9 Trường hợp kiểm thử.....	82
Bảng 3.10 Thông số góc tới và góc phản xạ	84
Bảng 4.1 Mô tả thành phần màn hình chính	87
Bảng 4.2 Mô tả thành phần màn hình game play.....	88
Bảng 4.3 Mô tả thành phần màn hình pause game	89
Bảng 4.4 Mô tả thành phần màn hình thư viện từ.....	89
Bảng 4.5 Số liệu thống kê đánh giá người dùng về game.....	91

DANH MỤC HÌNH VẼ

Hình 1.1 Minh họa công nghệ AR	3
Hình 1.2 Head- Mounted Display – HMD	5
Hình 1.3 Sản phẩm được thực hiện với Metaio SDK (http://www.metaio.com/)	8
Hình 1.4 Tính năng Smart Terrain của Vuforia (https://www.qualcomm.com)	9
Hình 1.5 Sản phẩm phát triển với Wikitude (http://www.wikitude.com/app/)	9
Hình 1.6 Nhãn hiệu Heinz được nhận diện tăng cường thông tin sử dụng Blippar (https://blippar.com)	10
Hình 1.7 Hệ thống mô phỏng hình học không gian Construct3D	11
Hình 1.8 Mô phỏng hệ thống mô phỏng cấu trúc phân tử hóa học	12
Hình 1.9 Ứng dụng mô phỏng cơ thể con người Anatomy 4D	12
Hình 1.10 Game VitaSoccers (http://arstechnica.com/)	13
Hình 1.11 Thử đồng hồ bằng ứng dụng AR (www.tissot.ch)	13
Hình 1.12 Sửa chữa xe ô tô thông qua các thông số cung cấp từ ứng dụng AR (http://www.metaio.com/)	14
Hình 1.13 Tái tạo hình ảnh của gan ứng dụng công nghệ AR trong phẫu thuật ít xâm lấn (http://medicalaugmentedreality.com/)	14
Hình 2.1 Vuforia SDK	15
Hình 2.2 Chi phí dịch vụ Cloud Recognition	18
Hình 2.3 Cấu trúc của Vuforia SDK	19
Hình 2.4 Minh họa target.	22
Hình 2.5 Hệ trục tọa độ.	23
Hình 2.6 Image Targets	24
Hình 2.7 Multi Target	25
Hình 2.8 Frame Marker	26
Hình 2.9 User-defined Targets	27
Hình 2.10 Text-word Target.	28
Hình 2.11 Cloud Recognition Target	28

Hình 2.12 Virtual Buttons	29
Hình 2.13 Hệ thống Target Manager	30
Hình 2.14 Game 3D đầu tiên tại Việt Nam – Thủy Hử 3D sử dụng engine Unity3D	35
Hình 2.15 Three Kingdoms Online – một game xây dựng bằng engine Unity3D chơi được trên cả Web, iOS và Android	36
Hình 2.16 Republique – một game viết cho iOS, PC và dòng máy Mac có đồ họa chân thực, sắc nét.....	37
Hình 2.17 Thực thể theo kiến trúc component.....	42
Hình 2.18 Các hệ thống xử lý các component	43
Hình 3.1 Sự quan tâm của trẻ em với thiết bị số.....	60
Hình 3.2 Minh họa trẻ em chơi video game.....	61
Hình 3.3 Tổng quan về follow game.....	63
Hình 3.4 Chủ đề trong game myWords	65
Hình 3.5 Hình ảnh trong game myWords	66
Hình 3.6 Từ khóa trong game myWords	66
Hình 3.7 Ký tự trong game myWords.....	67
Hình 3.8 Hiệu ứng đổi hình.....	67
Hình 3.9 Hiệu ứng di chuyển chữ	68
Hình 3.10 Hiệu ứng di chuyển điểm	68
Hình 3.11 Hiệu ứng di chuyển từ.....	69
Hình 3.13 Hệ thống HUD của game	71
Hình 3.14 Sơ đồ use case của game myWords	73
Hình 3.15 Danh sách các Actor trong sơ đồ use case của game myWords	73
Hình 3.16 Sơ đồ tuần tự use case Slect player mode	75
Hình 3.17 Sơ đồ tuần tự use case Configure setting	75
Hình 3.18 Sơ đồ tuần tự use case Select category	76
Hình 3.19 Sơ đồ tuần tự use case Select game mode.....	76
Hình 3.20 Sơ đồ tuần tự use case Select character.....	77

Hình 3.21 Sơ đồ tuần tự use case Show high score	77
Hình 3.22 Màn hình start game myWords	78
Hình 3.23 Màn hình world game myWords	79
Hình 3.24 Màn hình world game myWords	80
Hình 3.25 Màn hình library game myWords	81
Hình 3.26 Một số chữ cái trong bộ 26 chữ cái ABC.....	83
Hình 3.27 Mô tả về bộ 26 chữ cái khi đăng ký trên vuforia	83
Hình 3.28 Định luật phản xạ ánh sáng	84
Hình 3.29 Lăng kính và ứng dụng thực tiễn	85
Hình 3.30 Để giữ thiết bị.....	85
Hình 4.1 Màn hình chính game myWords	86
Hình 4.2 Các thành phần trong màn hình game play.....	87
Hình 4.3 Các thành phần trong màn hình pause game.....	88
Hình 4.4 Màn hình thư viện từ.....	89

DANH MỤC BẢNG DANH MỤC TỪ VIẾT TẮT

❖ Từ viết tắt

STT	Từ viết tắt	Diễn giải
1	API	Application Programming Interface
2	2D, 3D	2-Dimensional, 3-Dimensional
3	PC	Personal Computer
4	App	Application
5	API	Application Programming Interface
6	GUI	Graphical User Interface
7	SDK	Software Development Toolkit
8	UI	User Interface
9	AR	Augmented Reality
10	VR	Virtual Reality

❖ Thuật ngữ

STT	Từ viết tắt	Diễn giải
1	Augemented Reality	Công nghệ thực tại ảo bổ sung thông tin thế giới ảo vào thế giới thực
2	Application	Ứng dụng

3	Cloud	Công nghệ điện toán đám mây
4	Cloud Recognition	Tính năng nhận diện vật thể thông qua điện toán đám mây
5	Computer Vision	Công nghệ thị giác máy tính
6	Developer	Nhà phát triển
7	Device database	Cơ sở dữ liệu thiết bị
8	Engine	Bộ công cụ phát triển ứng dụng
9	Image Target	Ảnh dùng để làm mục tiêu nhận diện
10	Target Manager	Bộ công cụ khởi tạo và quản lý target của vuforia
11	User-Defined Target	Tính năng tạo ảnh mục tiêu do người dùng định nghĩa
12	Video Background Renderer	Bộ kết xuất đồ họa nền
13	Virtual Button	Nút ảo
14	Virtual Reality	Công nghệ thực tại ảo thay thế thế giới thực hoàn toàn bằng thế giới ảo

TÓM TẮT KHÓA LUẬN

Trong thời đại bùng nổ khoa học công nghệ, đặc biệt là công nghệ di động, hầu hết các lĩnh vực trong đời sống đều được ứng dụng kỹ thuật hóa nhằm giảm bớt áp lực cho con người nhưng vẫn đưa đến kết quả khả quan hơn. Từ những quy trình quản lý bán hàng nhỏ lẻ cho đến những hoạt động sản xuất công nghiệp, nghiên cứu khoa học phức tạp đều được áp dụng thành tựu của khoa học công nghệ.

Tuy nhiên, bên cạnh những lợi ích mang lại, sự phát triển của công nghệ, các thiết bị di động cùng với tầm phủ sóng rộng rãi của Internet, việc tiếp xúc với thế giới bên ngoài trở nên dễ dàng hơn. Chính vì vậy, chúng ta dễ bị đưa đẩy vào những mặt xấu mà Internet mang tới như: các tác phẩm văn hóa đồi trụy, trò chơi trực tuyến, web đen... Và đặc biệt đáng lo ngại hơn hết là đối tượng trẻ em, sự nhận thức của các em còn non kém nên khó có thể tránh khỏi được những cám dỗ đó.

Khóa luận này nhằm mục tiêu xây dựng một giải pháp mang tính công nghệ để tạo ra một sân chơi mới, bổ ích cho trẻ em, giúp các em có thể vừa học vừa chơi. Dựa trên sự phát triển không ngừng của nền tảng di động như hiện nay, khóa luận mong muốn đưa ra những sản phẩm có giá trị để phát triển khả năng về tư duy, khơi dậy sự tìm tòi và mong muốn học hỏi của trẻ em.

Nội dung khóa luận được chia làm hai phần. Phần một tập trung vào nghiên cứu tìm hiểu về Unity, công nghệ thực tại ảo Vuforia. Phần hai áp dụng những cơ sở công nghệ vào phát triển game thực tại ảo, không chỉ mang tính năng giải trí mà còn phát triển tư duy của học sinh.

Kết quả cuối cùng khóa luận tạo ra một sản phẩm mang tính giải đố, bổ sung một phần kiến thức cho trẻ đó là:

Game myWords: người chơi có nhiệm vụ tìm câu trả lời đúng với bức hình đã cho, đó là một từ tiếng anh. Trò chơi này giúp học sinh có thể học từ vựng một cách nhanh chóng thông qua hình ảnh và không nhàm chán.

MỞ ĐẦU

Đa phần trẻ em ngày nay không thích đọc sách in. Các em có xu hướng đam mê các thiết bị số như iPad, điện thoại thông minh... Chúng có thể tìm thấy mọi thứ tiện ích khi sử dụng phương tiện này, thậm chí xem những bộ phim hoạt hình yêu thích hàng giờ liền mà không phải chuyển kênh như khi xem tivi. Điều này sẽ gây ra nhiều tác hại nghiêm trọng.

Những thiết bị kỹ thuật số có thể là kho thông tin hữu ích nhưng không mang lại lợi ích gì cho sức khỏe của trẻ. Các em sử dụng hầu hết thời gian dán mắt vào màn hình sẽ lười vận động, không thích đi ra ngoài để hít thở không khí trong lành. Sách điện tử cũng mang đến nhiều thông tin bổ ích cho trẻ tuy nhiên chúng cũng có thể đưa đến sự xao lãng vì bị "cắm đống" bởi những trò chơi điện tử trong đó. Thực tế, một đứa trẻ nghiện game hiếm khi đọc sách. Bên cạnh đó, trẻ em đang dần tách biệt với bạn bè của chúng, những trò chơi dân gian hầu như không còn hứng thú với trẻ, mối quan hệ xã hội của trẻ đang dần bị thu hẹp lại.

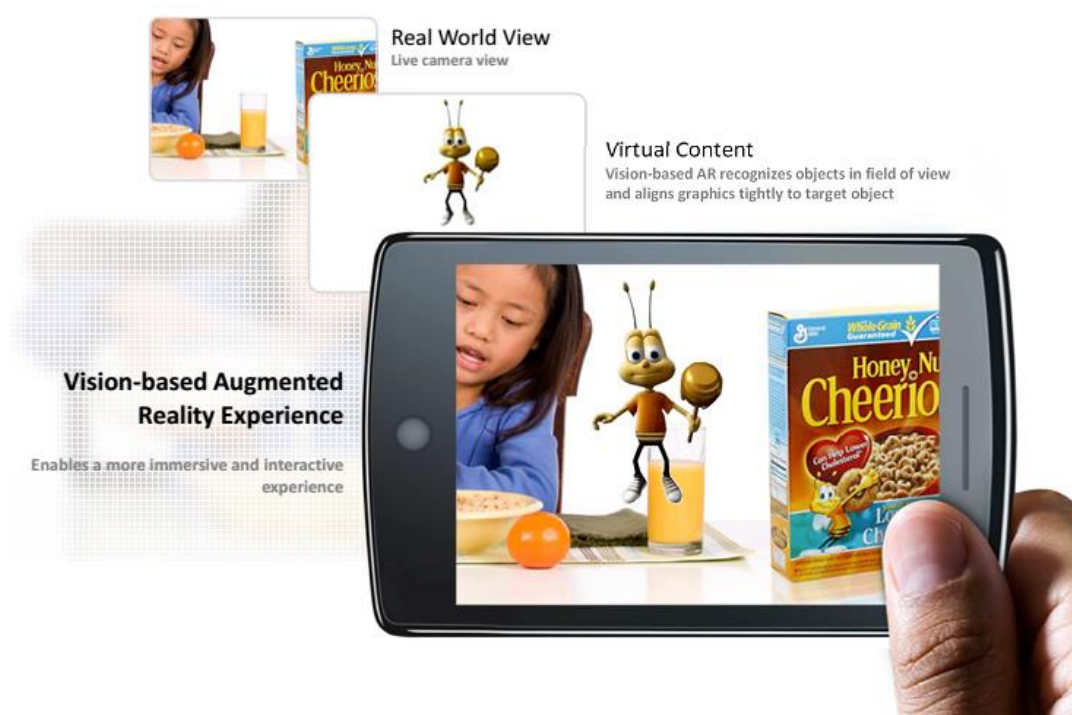
Chính vì vậy, cần có xây dựng một môi trường giáo dục tốt cho trẻ, tận dụng sự phát triển mạnh mẽ của công nghệ và thiết bị kỹ thuật số để phát triển một kho ứng dụng giải trí kết hợp với những kiến thức xã hội mà các em học được từ thầy cô, bạn bè tạo ra một môi trường giải trí lành mạnh, phát huy được tinh thần vừa học vừa chơi. Các em có thể chơi cùng nhau, nâng cao được khả năng làm việc nhóm, tạo được những mối quan hệ tốt, có thể giúp đỡ nhau để cùng phát triển. Đồng thời các em có thể nắm bắt được kiến thức tốt hơn, ghi nhớ lâu hơn, tạo nên một hiệu quả tích cực trong giáo dục.

Với những lý do trên, nhóm đã nghiên cứu, phân tích và mong muốn phát triển nên một kho ứng dụng giải trí cho trẻ, bên cạnh tính năng giải trí, ứng dụng cũng muốn mang đến cho các em kiến thức xã hội như: tiếng anh, toán học, logic học... Chính vì vậy mà đề tài: “Kết hợp Unity3D – Vuforia và xây dựng ứng dụng game minh họa” được chọn.

Chương 1. THỰC TẾ ẢO TĂNG CƯỜNG – AUGMENTED REALITY

1.1. Tổng quan về thực tế ảo tăng cường – Augmented Reality

Thực tế ảo tăng cường – Augmented Reality (AR) là công nghệ giúp cho bạn quan sát những vật trong thế giới thật thông qua màn hình một thiết bị điện tử, không những thế nó còn có thêm các thông tin liên quan đến vật thể đó. Giống như khi bạn lấy điện thoại và hướng camera của nó về trái dừa hầu trước mặt thì trên màn hình ngoài hình ảnh trái dừa còn có thêm thông tin về nguồn gốc cũng như giá trị dinh dưỡng và cách sử dụng. Ngoài việc dựa trên hình ảnh, thông tin còn được tối ưu nhờ vào các dữ liệu đến từ các thành phần khác như GPS, la bàn, gia tốc kế, micro...



Hình 1.1 Minh họa công nghệ AR

Augmented Reality là sự pha trộn giữa thế giới thực và thế giới ảo. Các yếu tố ảo được thêm vào thế giới thực, để tăng cường hoặc thêm các thông tin vào cho nó. Để làm được điều này người ta có thể kết nối camera với máy tính và vẽ hình ảnh ảo vào nguồn dữ liệu video, vì vậy người dùng có thể xem thế giới thực, tăng thêm các đồ họa máy tính trên màn hình.

1.2. Lịch sử phát triển

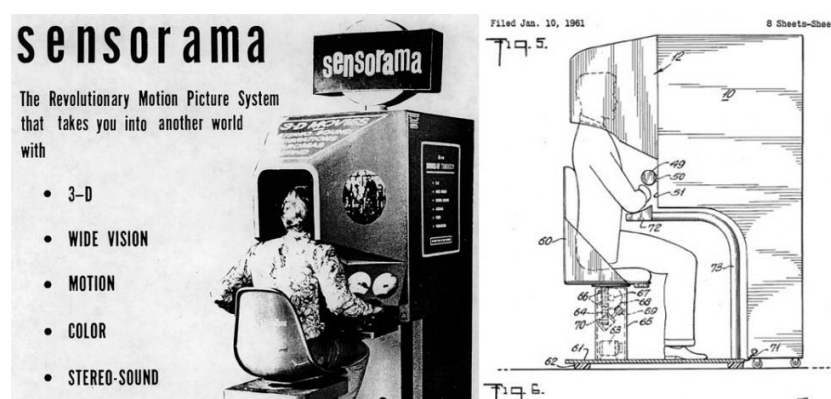
Thuật ngữ Augmented Reality (AR) xuất hiện từ khoảng năm 1990 nhưng điều này không có nghĩa là nó chưa có trước đó. Thời điểm mà con người tạo ra các tiện ích mà có liên quan đến môi trường của họ để cung cấp cho người dùng các thông tin dựa trên đó. AR đã xuất hiện, chỉ là không ai gọi nó bằng cái tên này.

Vì vậy để tìm hiểu về công nghệ AR, ta phải đi sâu vào nghiên cứu lịch sử của nó từ lúc ban đầu cho đến ngày hôm nay.

1.2.1. Giai đoạn khởi đầu

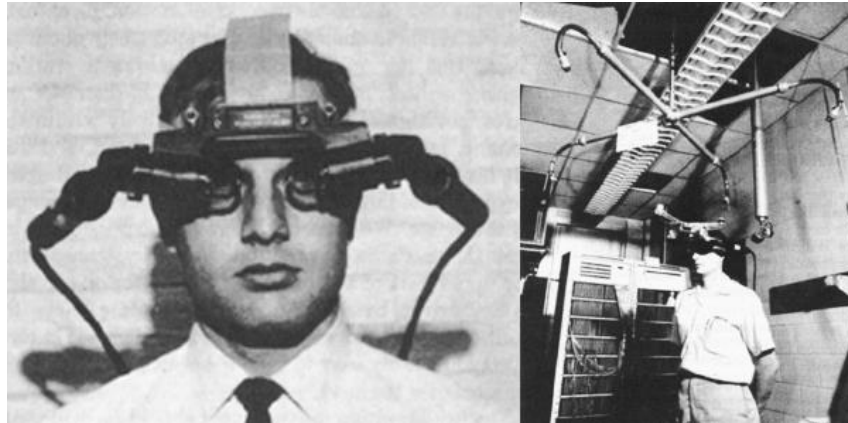
Từ năm 1957, một quý ông được biết đến với tên Morton Helig bắt đầu xây dựng một máy tính có tên là Sensorama. Nó được thiết kế là một trải nghiệm điện ảnh đưa mọi thứ vào giác quan của ta, chứ không phải như máy chơi game Arcade của thập kỷ 80. Nó thổi gió vào ta, rung ghế ta ngồi, chơi nhạc ngay trong mắt ta và chiếu một hình thức lập thể môi trường 3D ở phía trước và hai bên ta.

Nó khá ấn tượng với bộ phim demo “Chu kỳ qua đường của Brookly” nhưng lại chưa bao giờ được bán ra, chi phí làm cũng khá đắt cho những bộ phim dài. Bởi vì nó liên quan đến người cầm máy quay có 3 camera gắn vào mỗi lần, và đây thực sự là một khám phá thực tại ảo, đủ các yếu tố của AR có liên quan đến với cả 2 thiết bị giữa người dùng và môi trường, và thực tế là có môi trường của riêng mình, thế giới thực được xem trong một môi trường thực.



Hình 1.2 Máy tính Sensorama

Vào năm 1966, giáo sư kỹ thuật điện tử Ivan Sutherland tại đại học Harvard đã phát minh ra mô hình đầu tiên của một trong những thiết bị quan trọng nhất được dùng trong cả AR và VR ngày nay – màn hình đội lên đầu (Head- Mounted Display – HMD). Nó là một tác phẩm hoành tráng mặc dù nó khá nặng cho người đội, cho nên phải treo nó lơ lửng trong phòng thí nghiệm. Đó là lý do có nick name là Sword of Damocles.



Hình 1.3 Head- Mounted Display – HMD

1.2.2. Giai đoạn phát triển

Trong khi phải trải qua vài năm một vài hình dạng khác nhau, cụm từ Augmented Reality được cho là do giáo sư Tom Caudell đặt ra khi làm việc trong dự án phát triển và nghiên cứu hệ thống máy tính thích ứng thần kinh của Boeing ở Seattle. Trong khi tìm kiếm cách dễ dàng hơn để giúp sản xuất và xử lý kỹ thuật của công ty hàng không, ông bắt đầu áp dụng công nghệ thực tại ảo và cuối cùng đến với một số phần mềm phức tạp mà có thể che phủ vị trí các cấp nhất định trong quá trình xây dựng.

Đồng thời vào năm 1992, hai nhóm khác đã thực hiện những bước lớn vào thế giới này. LB Resenberg tạo ra thứ mà được công nhận rộng rãi như hệ thống AR đầu tiên cho không quân Hoa Kỳ được gọi là Thiết bị ảo – Virtual Fixtures, thiết bị này là những thứ dùng để mô tả các bước để hướng dẫn người dùng làm các nhiệm vụ riêng của mình.

Nhóm thứ hai gồm 3 thành viên là Steven Feiner, Blair MacIntyre và Doree Seligmann – Tất cả đều là những người dẫn đầu của AR – tạo ra một hệ thống mà họ

gọi là KARMA (Knowledge-based Augmented Reality for Maintenance Assistance). Nhóm từ đại học Columbia xây dựng một hệ thống HMD bằng máy theo dõi của Logitech và gắn nó vào đối tượng mà họ đang nghiên cứu – một máy in. Project cho ra một đồ họa 3D hiển thị cho mọi người biết load và sử dụng mà không cần coi hướng dẫn.

1.3. Đặc trưng

Một hệ thống Augmented Reality có 3 đặc trưng sau:

- Nó phải có sự kết hợp giữa thực và ảo.
- Nó phải tương tác với thời gian thực.
- Nó phải được đăng ký trong không gian 3D.

Yêu cầu đầu tiên chỉ đơn thuần là một hệ thống thực tại tăng cường kết hợp giữa thế giới thực và ảo, có thể được thực hiện bằng cách vẽ đồ họa máy tính trên dòng video. Yêu cầu thứ hai là hệ thống tương tác với thời gian thực, mà thực tế nghĩa là nó phải được cung cấp một dòng video trực tiếp và cập nhật cứ năm giây một lần. Cũng có thể dùng một cách khác để tương tác với thời gian thực.

Yêu cầu thứ ba xác định rằng dữ liệu ảo được đăng ký trong không gian 3D, có nghĩa là nó phải có vị trí tương đối trong không gian 3D và thế giới thực (ngay cả khi các dữ liệu chính được thể hiện trong 2D, như là chữ viết và hình ảnh).

Khác với VR, các thông tin tăng cường trong hệ thống AR liên hệ chặt chẽ với môi trường thực, sự xuất hiện của các thông tin thay đổi theo cách người dùng di chuyển cũng như xem xét các thành phần trong môi trường thực.

1.4. Kiến trúc hệ thống

Một hệ thống AR gồm có 3 phần: Thế giới thực, các thiết bị phần cứng nhận diện các thành phần của môi trường và phần mềm hỗ trợ.

Thiết bị phần cứng cho AR gồm có: bộ vi xử lý, màn hình, cảm biến, các thiết bị đầu vào. Các thiết bị điện toán di động như smartphone, tablet có chứa camera, cảm biến gia tốc, GPS, la bàn...

Các thiết bị AR hiện tại nổi bật có thể kể đến như Google Glass, Vuzix M100 Smart Glasses...

Chìa khóa của các phần mềm AR đó là làm cách nào để tích hợp được các thông tin tăng cường vào thế giới thực. Mỗi phần mềm AR phải lấy được tọa độ thế giới thực từ các hình ảnh camera, tọa độ này không phụ thuộc vào các camera. Quá trình này gọi là đăng ký hình ảnh, sử dụng các phương pháp khác nhau về thị giác máy tính. Quá trình này có thể chia làm 2 bước:

- **Bước 1:** Phát hiện các điểm cần quan tâm, hoặc các đánh dấu chuẩn, các dòng quang học trong các hình ảnh thu được từ camera.
- **Bước 2:** Xây dựng lại thế giới thực dựa trên các thông tin thu thập từ bước 1, có thể tính toán trước 1 số cấu trúc 3D trong trường hợp dựng cảnh quá phức tạp. Sau đó bổ sung thêm các thông tin cần tăng cường.

Các phần mềm này có quá trình xây dựng khá phức tạp, do đó để phát triển nhanh các ứng dụng AR một số hãng phần mềm đã cho ra đời các bộ SDK, tiêu biểu như: Metaio, Vuforia, Mobinet AR, Wikitude, Blippar và Layar.

1.5. Một số bộ SDK hỗ trợ

❖ Metaio

Metaio SDK là một trong những bộ SDK có nhiều tính năng cho phát triển ứng dụng AR. SDK này hỗ trợ xây dựng ứng dụng AR hoàn chỉnh, với các tính năng tracking 2-D, Location-based, Extended 2-D, 3-D.

Đây là bộ SDK có tính phí, bản thương mại với giá từ \$3,490.



Hình 1.4 Sản phẩm được thực hiện với Metaio SDK (<http://www.metaio.com/>)

❖ Vuforia

Vuforia là SDK do hãng Qualcomm phát triển. Điểm mạnh của Vuforia chính là tính năng Recognition: Objects, Images, User-Defined Images, Cylinders, Text, Boxes, Frame Markers. Gần đây, các tính năng mới của Vuforia cũng thu hút rất nhiều sự quan tâm của giới lập trình viên như các tính năng: Smart Terrain, Extended Tracking, Background Effects, Video Playback, Virtual Button, Occlusion Managent.

Bên cạnh đó Vuforia cũng có các tính năng mở rộng tích hợp cho các engine như Unity3D..., phát triển các ứng dụng đa nền tảng.

Vuforia cung cấp bản miễn phí cho lập trình viên và thu phí trên hệ thống cloud lưu trữ hình ảnh với số lượng lớn hơn 1000 bức ảnh.



Hình 1.5 Tính năng Smart Terrain của Vuforia (<https://www.qualcomm.com>)

❖ Wikitude

Wikitude SDK hỗ trợ khá nhiều tính năng cho AR, nổi bật là các tính năng về augmentation và visualization cho các mô hình 3D tĩnh và động. Bên cạnh đó SDK này cũng hỗ trợ phát triển ứng dụng đa nền tảng cùng với hệ thống cloud.

Wikitude phát triển nhiều bản phần mềm thương mại, với mức phí từ 590 € trở lên.



Hình 1.6 Sản phẩm phát triển với Wikitude (<http://www.wikitude.com/app/>)

❖ Blippar

Blippar cung cấp các nền tảng AR để phát triển các ứng dụng quảng cáo và phát hành nội dung. Ưu điểm của Blippar là công nghệ nhận diện hình ảnh.



Hình 1.7 Nhãn hiệu Heinz được nhận diện tăng cường thông tin sử dụng Blippar (<https://blippar.com>)

1.6. Ứng dụng thực tiễn

Thực tại tăng cường – Augmented Reality được ứng dụng trong nhiều lĩnh vực khác nhau, nổi bật là trong các lĩnh vực: trò chơi giải trí, quảng cáo, giáo dục, bảo trì và sửa chữa sản phẩm, y học. Chúng ta sẽ đi qua các ứng dụng minh họa trong một số lĩnh vực để hiểu được tầm quan trọng và tính ứng dụng của AR trong tương lai.

❖ Lĩnh vực giáo dục

Thực tế ảo tăng cường có thể áp dụng bổ sung trong các chương trình giảng dạy tiêu chuẩn. Các giáo trình, đồ họa, đoạn phim và âm thanh có thể chuyển sang môi trường tương tác thời gian thực. Sách giáo khoa, Flashcards và tài liệu giảng dạy khác có thể đính kèm hình ảnh liên kết, khi được quét bởi các thiết bị AR, nó sẽ hiển thị thông tin chi tiết mà hình ảnh đó mô tả. Học sinh có thể tham gia tương tác trực tiếp với thiết bị mô phỏng, những mô hình, hình ảnh thực tế sẽ tạo ra một môi trường học

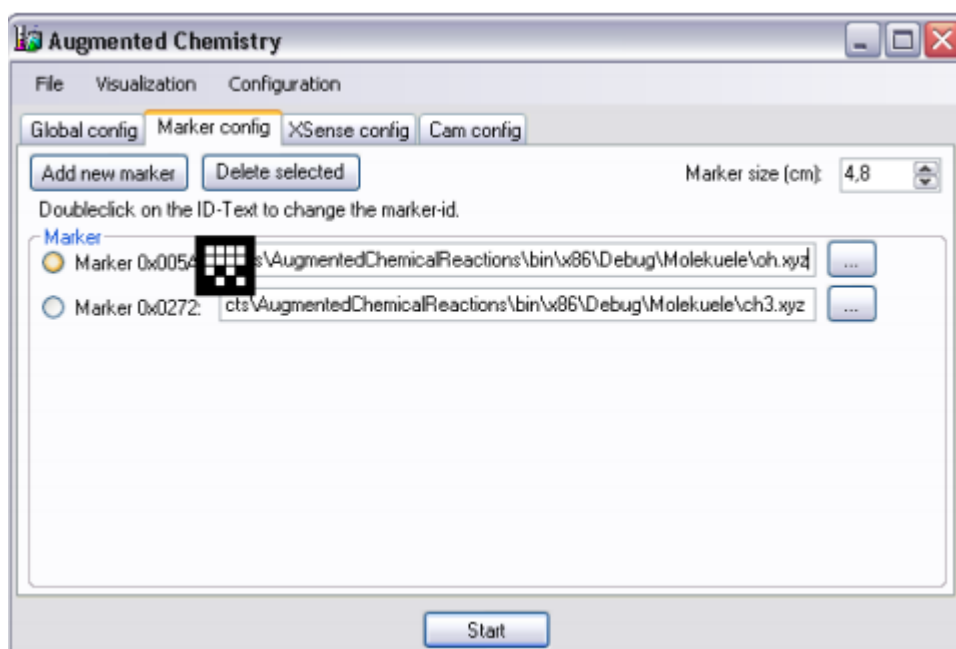
tập năng động hiệu quả. Một số ứng dụng thực tiễn công nghệ AR trong ngành giáo dục:

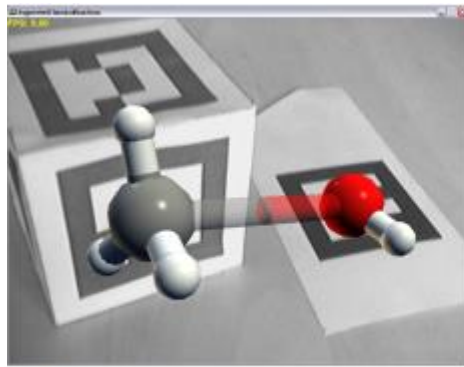
- *Construct3D*: là một công cụ xây dựng hình học ba chiều trên hệ thống thực tế ảo tăng cường “Studiercube”. Giúp học sinh có thể mô phỏng các vật thể hình học trong không gian.



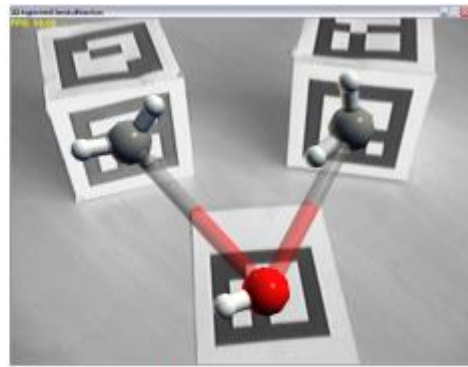
Hình 1.8 Hệ thống mô phỏng hình học không gian Construct3D

- AR có thể giúp học sinh hiểu về hóa học bằng cách cho phép họ hình dung cấu trúc không gian phân tử và tương tác với mô hình ảo mà nó xuất hiện. Qua đó môn học trở nên thú vị hơn cho cả giảng viên lẫn học sinh.





(a) One possible bond between two atoms



(b) More possible bonds

Hình 1.9 Mô phỏng hệ thống mô phỏng cấu trúc phân tử hóa học

- *Anatomy 4D*: Hệ thống giúp học sinh có thể hình dung được các bộ phận trong cơ thể con người.



Hình 1.10 Ứng dụng mô phỏng cơ thể con người Anatomy 4D

❖ Lĩnh vực trò chơi giải trí

Cùng với sự phát triển ngày càng mạnh mẽ của smartphone và tablets, thị trường game cho nền tảng di động càng ngày sôi động. Bên cạnh các game truyền thống, thể loại game AR đã mang đến cho người chơi nhiều trải nghiệm mới mẻ.



Hình 1.11 Game VitaSoccers (<http://arstechnica.com/>)

❖ Lĩnh vực quảng cáo

Các dịch vụ quảng cáo phát triển trên nền tảng AR đã chứng tỏ được tiềm năng của mình trong những năm gần đây. Bên cạnh những ứng dụng nhận diện sản phẩm, các ứng dụng AR hỗ trợ bán hàng qua mạng của các nhà sản xuất lớn cũng có một tầm ảnh hưởng nhất định.



Hình 1.12 Thử đồng hồ bằng ứng dụng AR (www.tissot.ch)

❖ Lĩnh vực bảo trì – sửa chữa sản phẩm

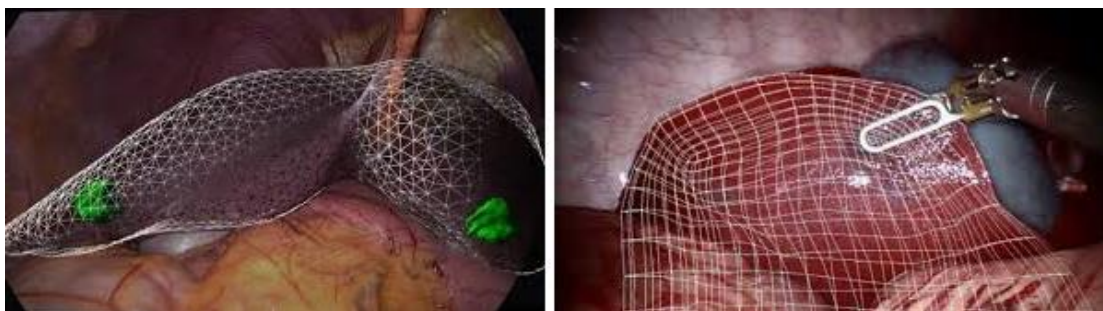
Là tiền đề cho sự ra đời của AR, các ứng dụng bảo trì – sửa chữa sản phẩm ngày càng được phát triển chi tiết hóa các thành phần của sản phẩm.



Hình 1.13 Sửa chữa xe ô tô thông qua các thông số cung cấp từ ứng dụng AR (<http://www.metaio.com/>)

❖ Lĩnh vực y học

Trong các công bố khoa học gần đây, các nhà khoa học máy tính đã kết hợp hỗ trợ các bệnh viện tại Mỹ trong việc ứng dụng AR trong các phẫu thuật ít xâm lấn, nhận diện hình ảnh, bổ sung thông tin liên tục cho các bác sĩ trong quá trình phẫu thuật. Các kết quả thực nghiệm đã chứng minh được tính hiệu quả của các phương pháp này.



Hình 1.14 Tái tạo hình ảnh của gan ứng dụng công nghệ AR trong phẫu thuật ít xâm lấn (<http://medicalaugmentedreality.com/>)

Chương 2. VUFORIA AUGMENTED REALITY SDK VÀ UNITY3D

2.1. Vuforia Augmented reality SDK

2.1.1. Giới thiệu

Thế giới AR là vô tận với rất nhiều kiểu ứng dụng thực tế khác nhau, tuy nhiên để biến những lý thuyết đó thành hiện thực thì cần công cụ để thực hiện. Vuforia là một trong số đó, đây là một nền tảng được Qualcomm phát triển, giúp cho các lập trình viên có thể hiện thực hoá ý tưởng của mình. Bộ công cụ lập trình Vuforia SDK 2.0 hỗ trợ cả hai hệ điều hành iOS và Android với các API được cung cấp tương thích với ngôn ngữ lập trình C++, Java, Objective-C và .Net.



Hình 2.1 Vuforia SDK

Ứng dụng thực tại ảo dựa trên Vuforia SDK dùng màn hình của các thiết bị di động như một “ống kính ma thuật” hoặc thông qua một mặt gương để nhìn vào thế giới ảo, nơi mà thế giới thực và thế giới ảo cùng tồn tại. Nó làm cho một hình ảnh có sẵn thông qua camera để hiển thị ra hình ảnh của thế giới vật chất. Sau đó, các đối tượng 3D ảo sẽ được đặt chồng vào camera của thiết bị và xuất hiện trong thế giới thật.

Thế mạnh của Vuforia đó là thư viện hình ảnh đồ sộ, cùng với khả năng nhận dạng vật thể được tập trung phát triển. Với công cụ này mã hiệu để nhận biết không cần là một hình ảnh vật thể cụ thể mà bạn có thể sử dụng các dữ liệu riêng, các mã QR... Không những thế, với Vuforia thì bạn có thể thiết kế các vật thể 3D một cách đơn giản hơn, không tốn nhiều thời gian và công sức, nhờ vào công cụ Multi-Targets.

Ứng dụng phát triển từ Vuforia sẽ cho phép ta nhiều trải nghiệm hơn:

- Phát hiện nhanh các target.
- Hệ thống điện toán đám mây có thể ghi nhận khoảng 1 triệu target.
- Người dùng có thể tự định nghĩa target khi chạy ứng dụng.
- Tính vững chắc – khả năng bám dính vào target và khó bị mất khi di chuyển thiết bị.
- Có thể cùng lúc hiển thị 5 target.
- Kết quả tốt hơn trong điều kiện thế giới thực – ánh sáng mờ, phủ từng phần target.
- Tối ưu hóa cho việc đồ họa thực tế trên target hiệu quả hơn.

Vuforia Engine cung cấp một thư viện (đối tượng chia sẻ libQCAR.so trên Android và thư viện tĩnh libQCAR.a trên iOS) nhà phát triển có thể liên kết thư viện này với ứng dụng.

Hiện tại Vuforia hỗ trợ phát triển trên hai hệ điều hành là iOS và Android. Ngoài ra nó còn cho phép phát triển ứng dụng AR trên Unity một cách dễ dàng.

Development Enviroment	Development Platform			
	Native SDK		Unity Extension	
	Android	iOS	Android	iOS
Window	Yes	---	Yes multi-platform deployment	
MacOS	Yes	Yes	Yes multi-platform deployment	
Linux	Yes	---	---	--

Bảng 2.1 Nền tảng phát triển ứng dụng

2.1.2. Giấy phép Vuforia

Đối với nhà phát triển hiểu biết giấy phép sử dụng Vuforia là điều hết sức quan trọng.

Qualcomm cung cấp bộ Vuforia SDK hoàn toàn miễn phí, không có giấy phép có phí. Ngay cả khi mục đích sử dụng thương mại cho các ứng dụng dưới nền tảng Android/ iOS cũng miễn phí. Mặt khác là quan trọng khi nhận xét rằng Vuforia hoặc bất kỳ phần mềm của Qualcomm khác đều ràng buộc vào điều khoản của GPL, AGPL, LGPL, EUPL, APSL, CDDL, IPL, EPL, MPL hoặc với những giấy phép mã nguồn mở khác.

Nhưng có một số điều kiện để sử dụng chúng miễn phí và tất cả được giải thích chi tiết ở Giấy phép thỏa thuận có thể xem ở địa chỉ: <https://developer.vuforia.com/legal/license>. Phần quan trọng nhất trong thỏa thuận cấp phép là phần 8 và phần 2.2 (hạn chế sử dụng). Hơn nữa, ứng dụng chỉ có thể chứa các biểu tượng, tên hoặc bất kỳ loại tài liệu tham khảo của Vuforia với sự cho phép của Qualcomm, theo các quy định của Qualcomm.

DEVELOP	STARTER	STANDARD	ENTERPRISE
1,000 Recognitions / Month	10,000 Recognitions / Month	100,000 Recognitions / Month	More than 100,000 Recognitions / Month
1,000 Targets	1,000,000 Targets	1,000,000 Targets	
\$0/MO For Development Use	\$99/MO \$.01 Overage Per Recognition	\$499/MO \$.01 Overage Per Recognition	Contact Us For Pricing

Hình 2.2 Chi phí dịch vụ Cloud Recognition.

Nhà phát triển chỉ phải trả tiền cho việc sử dụng lưu trữ đám mây, vì dịch vụ này là tài sản của Qualcomm và được sự bảo vệ của pháp luật.

2.1.3. Các tính năng chính

Text: Hỗ trợ nhận dạng khoảng 100.00 từ tiếng anh từ bộ dữ liệu chuẩn hoặc do người dùng định nghĩa.

Images: Có thể nhận dạng một thông tin chi tiết từ hình ảnh cụ thể. Chẳng hạn các tạp chí , bao bì sản phẩm...

User-Defined Images: Cho phép người dùng tự định nghĩa các hình ảnh target.

3D Cylindrical Targets: Hỗ trợ nhận dạng và theo dõi các đối tượng 3D như: chai, lọ, cốc, hình hộp,...

Simple 3D Object: Các đối tượng 3D đơn giản là target có thể được nhìn thấy từ bất kỳ góc độ nào từ thiết bị. Điều này bao gồm các đối tượng được xây dựng từ bề mặt phẳng chẳng hạn bao bì , hộp giấy.

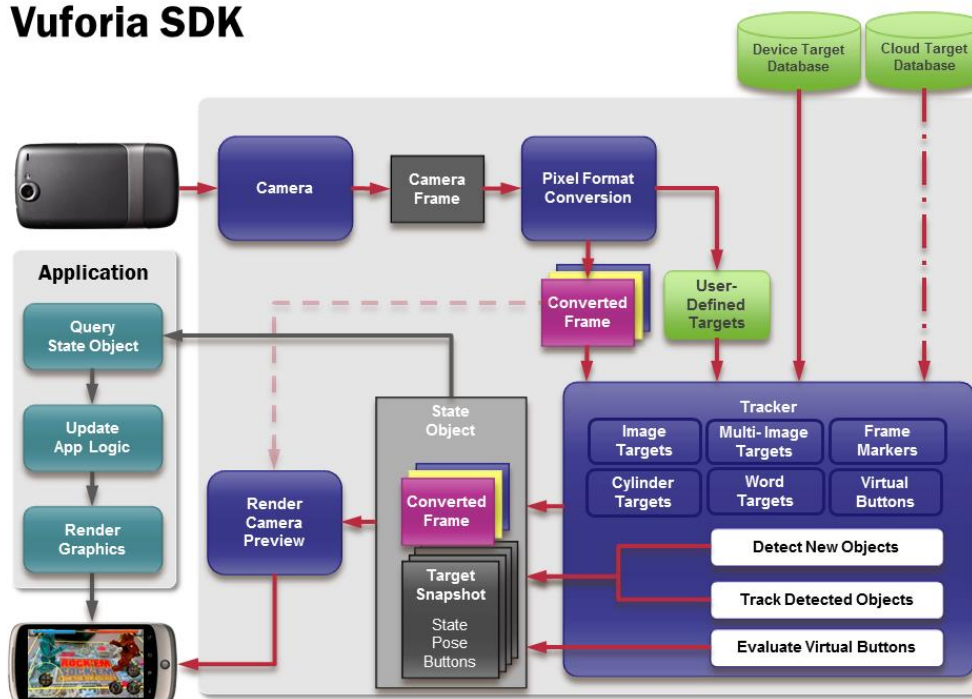
Frame Markers: Đây là các khung hình 2D cụ thể với một mô hình trực quan và có thể sử dụng như thẻ game trong một bảng game.

Device Database: Cơ sở dữ liệu thiết bị trên nền tảng Vuforia cho phép ứng dụng của chúng ta nhanh chóng nhận diện ra các mục tiêu. Cơ sở dữ liệu trên thiết bị có thể hỗ trợ khoảng 80 Image targets được tích hợp trong ứng dụng của của chúng ta lúc cài đặt hoặc cập nhật từ máy chủ.

Cloud Database: Cơ sở dữ liệu điện toán đám mây là điều cần thiết nếu ứng dụng của chúng ta cần một số lượng lớn các hình ảnh và các hình ảnh này thay đổi thường xuyên. Nếu bạn xây dựng ứng dụng bán lẻ hoặc ứng dụng đồng hành với một ấn phẩm in ấn, thì cơ sở dữ liệu điện toán đám mây là điều cần thiết.

2.1.4. Cấu trúc của vuforia

Vuforia SDK



Hình 2.3 Cấu trúc của Vuforia SDK

Một ứng dụng thực tại ảo dựa trên Vuforia SDK là sự tổng hợp của các thành phần cốt lõi dưới đây.

2.1.4.1. Camera

Camera đảm bảo rằng mỗi khung hình được chụp và thông qua một cách nào đó để hiển thị. Nhà phát triển chỉ có nhiệm vụ cho camera biết khi nào hình ảnh bắt đầu hay dừng lại. Khung hình sẽ tự động chuyển vào thiết bị, nó phụ thuộc vào định dạng và kích thước của hình ảnh.

2.1.4.2. Image Converter

Chuyển đổi định dạng pixel giữa thiết bị camera (VD: YUV12) thành một định dạng phù hợp với OpenGL (VD: RGB565) và để theo dõi (VD: luminance). Chuyển đổi này cũng bao gồm việc lấy mẫu để có hình ảnh camera trong độ phân giải khác nhau có sẵn trong ngăn xếp khung chuyển đổi.

2.1.4.3. Tracker

Tracker chứa các thuật toán thị giác máy tính (computer vision) dùng để xác định và theo dõi các đối tượng thế giới thực thông qua các khung hình video của camera. Dựa vào hình ảnh của camera, các thuật toán khác nhau sẽ tự xác định các target hoặc marker mới, hay ước lượng các nút ảo. Kết quả được lưu giữ trong trạng thái một đối tượng được dùng bởi Video background render và có thể được truy cập thẳng từ trong Application code. Tracker có thể tải lên nhiều dataset cùng lúc.

2.1.4.4. Video Background Renderer

Video background renderer làm cho hình ảnh camera được lưu trữ ở trạng thái của đối tượng. Hiệu suất của background video rendering được tối ưu hóa với từng thiết bị cụ thể.

2.1.4.5. Application code

Các nhà phát triển ứng dụng cần phải khởi tạo tất cả các thành phần ở trên và thực hiện 3 bước quan trọng trong Application code. Đối với mỗi khung hình xử lý trạng thái đối tượng phải được cập nhật và gọi đến hàm vẽ. Nhà phát triển ứng dụng cần phải.

- Truy vấn trạng thái đối tượng cho mỗi target, marker hoặc cập nhật trạng thái của những yếu tố xác định mới nhất.
- Update logic ứng dụng với các dữ liệu đầu vào.
- Render đồ họa ảo che phủ lên.

2.1.4.6. Cơ sở dữ liệu của thiết bị - Device Database

Cơ sở dữ liệu của thiết bị được tạo bằng cách dùng trực tuyến Target Manager do Vuforia cung cấp. Cơ sở dữ liệu target của thiết bị tải về chứa trong một file cấu hình XML nó cho phép các nhà phát triển có thể cấu hình một số chức năng của trackable và một file nhị phân chứa cơ sở dữ liệu của trackable.

Những asset này được biên dịch bởi nhà phát triển trong package app installer của ứng dụng và được sử dụng trong thời gian chạy bởi Vuforia AR SDK.

2.1.4.7. Cơ sở dữ liệu đám mây - Cloud Database

Cơ sở dữ liệu đám mây cũng có thể được tạo ra bằng cách dùng Target Manager. Các hình ảnh target có thể được upload thông qua giao diện Web hoặc bằng Vuforia Web Services API.

Các target được truy vấn trong thời gian chạy của ứng dụng bằng cách dùng tính năng nhận dạng đám mây, nó thực hiện tìm kiếm hình ảnh trên đám mây bằng cách gửi hình ảnh qua camera của thiết bị. Ngoài các dữ liệu target, target còn chứa một tập tin metadata được trả về khi truy vấn.

2.1.4.8. Người dùng tự định nghĩa mục tiêu - User Defined Targets

Một cách tiếp cận khác được hỗ trợ là người dùng tự định nghĩa target. Thay vì chuẩn bị sẵn target, chức năng này cho phép tạo ra target từ hình ảnh camera hiện tại.

Một lớp được xây dựng sẵn sẽ được gọi để kích hoạt việc tạo ra một target tự định nghĩa mới. Target trả về sẽ được lưu trữ, nhưng chỉ được giữ lại cho một bản AR.

2.1.5. Trackables

2.1.5.1. Khái niệm Target

Target là đại diện cho đối tượng của thế giới thật, mà có thể xác định và theo dõi bởi Vuforia SDK. Mỗi target thường có tên, định danh, và trạng thái và nó được lưu trong trạng thái của một đối tượng.



Hình 2.4 Minh họa target.

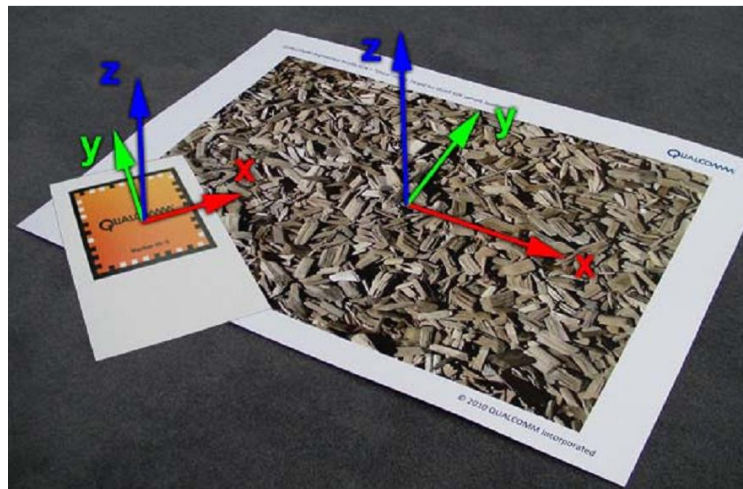
Target bao gồm các kiểu đối tượng khác nhau như:

- Image Target, ví dụ: hình ảnh, bảng trò chơi, các trang tạp chí, bìa sách, bao bì sản phẩm, áp phích, thiệp chúc mừng... chúng kích hoạt hình ảnh ảo.
- Frame Markers: đây là các khung hình 2D cụ thể với một mô hình trực quan và có thể sử dụng như thẻ game trong một bảng game.
- Multi-target: chẳng hạn như các bao bì sản phẩm hoặc các sản phẩm có hình vuông hoặc hình hộp chữ nhật. Chúng kích hoạt các đối tượng 3D đơn giản.
- Virtual Button: nó cho phép ta định nghĩa các button như các vùng hình chữ nhật trên một Image Target.

Các tham số:

- Trackable type:
 - UNKNOWN_TYPE: kiểu chưa xác định.
 - IMAGE_TARGET: kiểu Image Target.
 - MULTI_TARGET: kiểu Multi-target.
 - MARKER: kiểu Frame-Marker.
- Trạng thái đối tượng.
 - UNKNOWN: chưa xác định.
 - UNDEFINED: chưa được định nghĩa.

- NOT_FOUND: không tìm thấy trong cơ sở dữ liệu.
 - DETECTED: đã được xác định.
 - TRACKED: đang được theo dõi.
- **Hệ tọa độ:** Vuforia SDK sử dụng hệ trục tọa độ hướng từ trái sang phải. Mỗi target được định nghĩa một hệ tọa độ với (0,0,0) là điểm trung tâm của bức ảnh. Chiều x là hướng sang phải, chiều Y hướng ra sau và chiều z hướng tâm tầm hình ra ngoài được minh họa ở hình dưới.



Hình 2.5 Hệ trục tọa độ.

2.1.5.2. Image Targets

Image target là hình ảnh mà Vuforia SDK có thể xác định và theo dõi. Không như các Marker, code dữ liệu ma trận và code QR tầm thường khác, Image target không cần vùng đen hay trắng đặc biệt hoặc code để nhận hình ảnh.



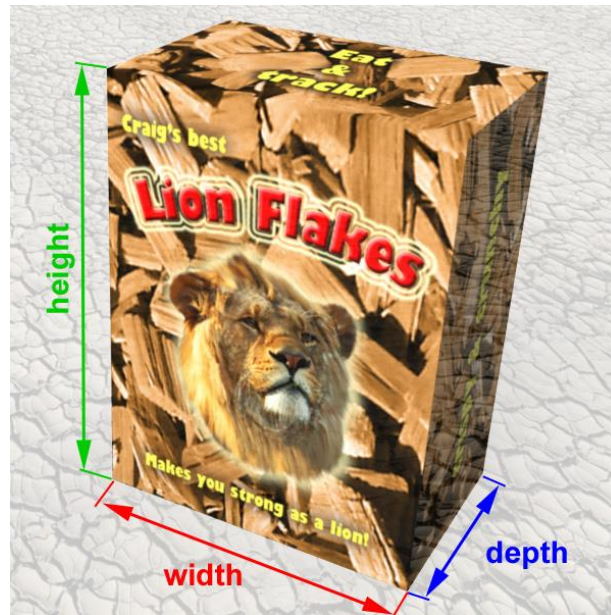
Hình 2.6 Image Targets

Vuforia SDK dùng các thuật toán phức tạp để xác định và theo dõi các điểm đặc trưng được tìm thấy trong hình ảnh. Nó nhận ra Image target bằng cách so sánh những điểm đặc trưng tương ứng với một cơ sở dữ liệu target có sẵn. Khi Image Target được xác định, SDK sẽ theo dõi hình ảnh trong khi nó còn nằm trong tầm nhìn của camera.

Image target có thể được tạo và sử dụng theo 3 cách sau:

- Tạo Image target với Target Manager để dùng trong device database.
- Tạo Image target dùng trong cơ sở dữ liệu nhận dạng đám mây.
- Image target tạo ra khi chạy được lấy từ khung hình video của camera và tự động thêm vào device database.

2.1.5.3. Multi Targets



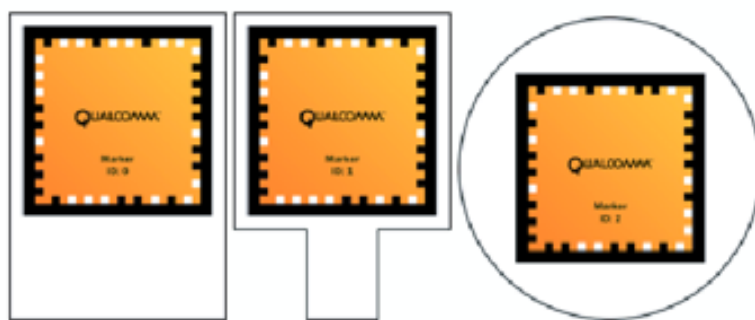
Hình 2.7 Multi Target

Một Multi-target bao gồm nhiều Image target có quan hệ không gian cố định. Một khi một thành phần của Multi-target được xác định, tất cả các thành phần khác có thể được theo dõi như vị trí tương đối của chúng hay định hướng được biết đến. Miễn là một thành phần của target là phần có thể nhìn thấy trong hình ảnh của camera, như vậy Multi-target có thể được theo dõi.

Multi-target được tạo bằng cách định nghĩa một mối quan hệ giữa các Image target đã tồn tại dùng Target Manager hoặc trực tiếp bằng cách điều chỉnh file XML cấu hình Dataset. Mối quan hệ không gian của các phần đơn lẻ được lưu trữ trong file XML bằng cách biết đối đơn giản.

Khi multi-target kết hợp với Image target, chúng phải đáp ứng đủ các thuộc tính của Image target. Ngoài ra, điều quan trọng là phải đảm bảo rằng hình dạng của Multi-target vật lý phải còn nguyên vẹn. Giấy và các cạnh trên đó phải bằng phẳng không có nếp gấp, các mặt trên không được lồi ra.

2.1.5.4. Frame Marker



Hình 2.8 Frame Marker

Ngoài các tính năng dựa trên việc xác định và theo dõi, Vuforia SDK có thể theo dõi một loại marker chuẩn được gọi là Frame marker. ID duy nhất của một Frame marker được mã hóa vào một mô hình nhị phân dọc theo đường viền của hình ảnh marker. Một Frame marker cho phép hình ảnh bất kỳ được đặt trong các viền marker. Nó có nhiều khác biệt như nó nhìn tự nhiên hơn so với nhiều marker chuẩn khác. Vuforia SDK đòi hỏi khung và mô hình nhị phân phải hoàn toàn hiển thị trong hình camera để nhận dạng.

Không như Image target, Frame marker không được tạo bởi Target Manager Tất cả 512 Frame marker được phân phối như một kho lưu trữ trong assets/folder khi cài đặt Vuforia SDK.

Do khả năng xử lý tương đối thấp đòi hỏi việc giải mã marker ID, tất cả 512 Frame marker có thể được sử dụng trong một ứng dụng và khoảng 5 có thể được phát hiện và theo dõi cùng một lúc.

Kích thước của marker phụ thuộc vào môi trường ứng dụng. Trong các ví dụ mẫu, marker có size điển hình là 1-4 inch hoặc 3-10 cm. Size nhỏ cho phép marker dùng như thẻ game, lá bài trong game, hoặc hình động 3D.

2.1.5.5. User-defined Targets

Như các Image target cho phép nhà phát triển chọn một hình ảnh trước để ứng dụng nhận dạng, User-defined Target cho phép người dùng cuối chọn bất kỳ hình ảnh

trong thời gian chạy Camera. Điều này cho phép người dùng trải nghiệm AR “bất cứ lúc nào, bất cứ nơi đâu” bằng cách chọn một hình ảnh- như một bức ảnh, bìa sách hoặc poster- từ môi trường thực của họ mà không cần phải mang theo các target đã định nghĩa trước.



Hình 2.9 User-defined Targets

User-defined target là các Image target, nhưng không giống như image target được xây dựng sẵn bằng Target Manager, chúng được tạo ngay trong thời gian chạy từ lựa chọn của người dùng trong khung hình camera. Đơn giản ta chỉ cần chụp một tấm ảnh và ứng dụng sẽ chuyển tấm ảnh này vào Vuforia Engine.

Target được tạo bằng cách này có đầy đủ thuộc tính của Image Target chuẩn chỉ không hỗ trợ với các Virtual button.

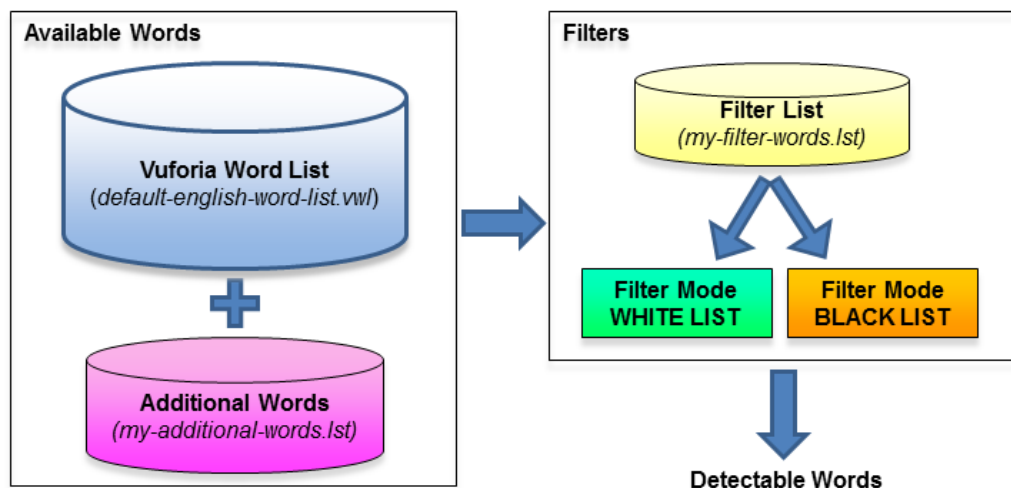
Đối với User-defined target, ứng dụng có trách nhiệm bắt đầu xử lý scan một target, kích hoạt quá trình xây dựng target, sau đó thêm target mới xây dựng này vào một cơ sở dữ liệu để theo dõi.

2.1.5.6. Text-Word Targets

Word Target là những phần tử văn bản mà Vuforia SDK có thể nhận dạng và theo dõi được. Vuforia có thể nhận dạng ra từ nếu từ này có trong dataset hay còn gọi là

“word list”. Hiện tại thì Vuforia chỉ hỗ trợ nhận dạng từ tiếng anh với khoảng 100.00 từ phổ biến có trong “world list”

Hình ảnh dưới đây cho ta một cái nhìn tổng quan về các thành phần liên quan đến việc tạo/chuẩn bị word targets.



Hình 2.10 Text-word Target.

2.1.5.7. Cloud Recognition Target

Cloud Recognition Service, có sẵn trong Vuforia 2.0, cho phép các loại ứng dụng bán hàng hoặc phát hành. Cơ sở dữ liệu với một tập cố định các target không cần phải đính kèm với ứng dụng hoặc load lên khi chạy.



Hình 2.11 Cloud Recognition Target

Một ứng dụng dùng nhận dạng đám mây sẽ truy xuất cơ sở dữ liệu từ hình ảnh của camera (thực tế nhận dạng xảy ra ở trên đám mây) và sau đó xử lý các kết quả trả về phù hợp từ đám mây để thực hiện việc nhận dạng và theo dõi. Không bắt buộc nhà phát triển phải định nghĩa metadata, chúng có thể gắn với mỗi kết quả phù hợp được trả về từ ứng dụng.

Cloud Recognition Target là những hình ảnh được tải vào một cơ sở dữ liệu cloud target, sau đó Vuforia SDK truy xuất trong thời gian chạy ứng dụng. Kết quả thành công sẽ trả về một truy vấn được gọi là một Recognition Event (hoặc “Reco Event”) và cho phép một hình ảnh được theo dõi giống như cách Image target được theo dõi. Target đám mây về cơ bản là Image target từ hệ thống đám mây.

2.1.5.8. Virtual Buttons

Virtual buttons được nhà phát triển định nghĩa là các vùng hình chữ nhật trên Image target, khi chạm hoặc nhấp vào, có thể kích hoạt các sự kiện. Virtual button có thể dùng để thực hiện event như nhấn một button hoặc phát hiện nếu khu vực button đó của image target bị che bởi một đối tượng.



Hình 2.12 Virtual Buttons

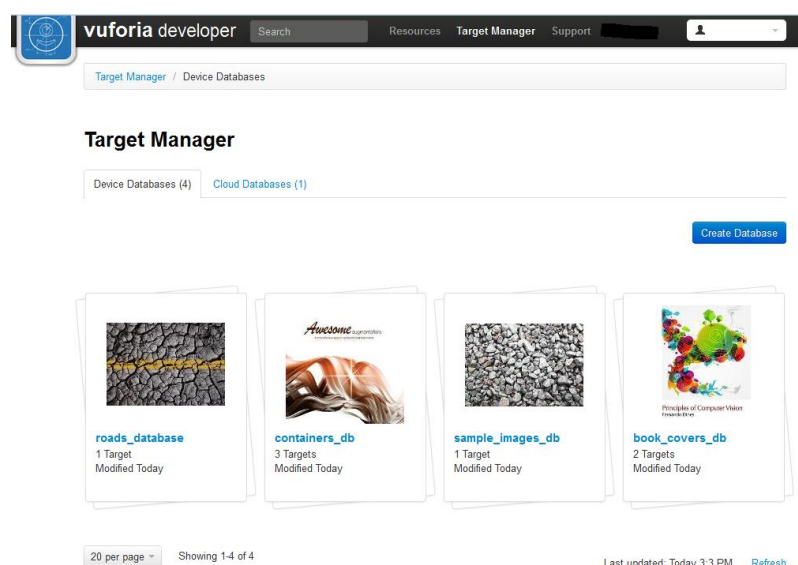
Virtual button được xác định chỉ khi khu vực button nằm trong hình ảnh của camera và camera phải ổn định, Việc xác định Virtual button bị vô hiệu nếu di chuyển camera qua nhanh.

Virtual button chỉ được hỗ trợ cho Image target được tạo bởi Target Manager và được tải về trong cơ sở dữ liệu Target của thiết bị. Cloud Target và User-defined Target không hỗ trợ Virtual button.

2.1.6. Hệ thống quản lý Target

2.1.6.1. Target manager

Vuforia cung cấp cho nhà phát triển công cụ tạo Device database và Cloud database trực tuyến trên giao diện web Target Manager. Nhà phát triển có thể upload hình ảnh lên hệ thống để thêm vào cơ sở dữ liệu cũng như quản lý chúng và tải tập dataset cho Device database nếu muốn thêm nó vào trong ứng dụng.



Hình 2.13 Hệ thống Target Manager

2.1.6.2. Vuforia Web Service APIs

Vuforia Web Service API (VWS API) được tạo ra cho phép nhà phát triển có thể sử dụng hệ thống CMS của cửa hàng kết hợp với hệ thống Cloud Recognition của

Vuforia. Nhà phát triển có thể quản lý dữ liệu trên Cloud Recognition của Vuforia thông qua chính hệ thống CSM của cửa hàng

Vuforia Web Service APIs cung cấp các hàm API cho phép nhà phát triển có thể thao tác quản lý Cloud Target từ xa mà không cần phải truy cập vào giao diện web của Target Management của Vuforia giúp hợp nhất giữa Target Management với hệ thống quản lý tài nguyên của khách hàng.

2.2. Tìm hiểu về game engine

Game engine là một bộ ứng dụng được xây dựng để phát triển game, hỗ trợ các tính năng để quá trình phát triển game trở nên đơn giản, dễ dàng và nhanh chóng. Các tính năng được hỗ trợ bởi các game engine thường bao gồm:

- Dựng hình 2D, 3D
- Chỉnh sửa mã nguồn
- Âm thanh
- Vật lý
- Tạo hiệu ứng động
- Quản lý bộ nhớ
- Xây dựng trí tuệ nhân tạo
- Kết nối mạng
- ...

Game engine giúp tiết kiệm thời gian và chi phí cho các dự án rất đáng kể.

2.2.1. Phân loại game engine

Việc phân loại game engine rất phức tạp vì có rất nhiều tiêu chí để so sánh giữa chúng. Tuy nhiên, dựa vào mức độ chuyên biệt, mức độ hỗ trợ của các engine mà ta chia chúng thành 3 cấp độ khác nhau, từ thấp lên cao: Roll-Your-Own, Most-Ready và Point-And-Click.

❖ **Roll-Your-Own**

Roll-Your-Own engine được xem như những engine ở mức thấp nhất. Thường thì chúng không phổ biến do được phát triển ở mỗi công ty với những mục đích riêng. Những engine này sử dụng những ứng dụng giao diện được công bố rộng rãi, các API như XNA, DirectX, OpenGL, SDL để phát triển phần đồ hoạ. Thêm vào đó, chúng có thể dùng những thư viện từ nhiều nguồn khác nhau để bổ sung các tính năng khác. Những thư viện đó có thể bao gồm thư viện vật lý như Box2D, Chipmunk, thư viện âm thanh như irrKlang, PortAudio và thư viện GUI như AntTweakBar.

Thực tế, Roll-Your-Own engine cho phép những lập trình viên có thể tương tác sâu hơn vào hệ thống, tùy biến theo mong muốn một cách dễ dàng. Tuy nhiên, họ cũng sẽ gặp phải rất nhiều lỗi do sự thiếu tương thích giữa các thành phần với nhau.

❖ **Most-Ready**

Most-Ready engine là những engine tăng trung. Chúng được thiết kế với đầy đủ các tính năng cần thiết, kèm theo đó là rất nhiều công cụ thoải mái hầu hết các mục đích sử dụng của người dùng. Những engine loại này khá phong phú, bao gồm OGRE, Genesis3D, Torge, Gamebryo và cả Unreal, IdTech.

Những lập trình viên cần lập trình thêm để gắn kết các thành phần trong engine lại với nhau để tạo nên một game hoàn chỉnh. Vì tính chuyên biệt cao hơn, nên với loại engine này các lập trình viên sẽ không còn nhiều khả năng tương tác sâu như các engine loại Roll-Your-Own.

❖ **Point-And-Click**

Point-And-Click là loại game engine dễ sử dụng nhất. Nói như vậy không có nghĩa là mọi người đều có thể sử dụng chúng để làm ra game, vẫn cần một ít kiến thức lập trình để có thể làm được, tuy nhiên, lượng công việc đó đã được tối giản xuống mức tối thiểu. Các Point-And-Click engine có chứa hầu như tất cả những thứ cần thiết để làm game, người dùng chỉ cần lựa chọn và kéo thả. Các engine loại này bao gồm GameMaker, Torque Game Builder và Unity.

2.2.2. Một số game engine trên thị trường

❖ CryEngine

CryEngine ban đầu được Crytek phát triển như một bản demo công nghệ cho Nvidia. Khi cảm thấy tiềm năng của CryEngine, Crytek đã phát triển nó và tung ra game đầu tiên Far Cry khá thành công. Qua thời gian CryEngine ngày càng phát triển mạnh mẽ hơn với những tựa game nổi tiếng như Far Cry, Crysis, Crysis Warhead, Crysis 2 và Aion: Tower of Eternity. Phiên bản hiện tại, CryEngine 3 được phát hành vào ngày 4/10/2009. Theo Crytek, CryEngine 3 được tạo ra với tham vọng trở thành công cụ phát triển game tất cả trong một, ứng dụng trên cả Windows, PlayStation 3 và Xbox 360. Riêng đối với nền PC, CryEngine 3 sẽ hỗ trợ tốt cả DirectX 9, 10 và 11.

Điểm nổi trội của CryEngine so với các engine đương thời là tập trung vào mặt xử lý hình ảnh, ngoài ra hiệu ứng âm thanh và chuyển động cũng được mô phỏng xuất sắc.

❖ OGRE

OGRE (Object-Oriented Graphics Rendering Engine) là một engine dựng hình 3D linh hoạt, tập trung vào dựng hình hơn là một công cụ tạo game. Bộ thư viện của OGRE trừu tượng hóa các thư viện ở mức hệ thống như DirectX 3D và OpenGL để cung cấp một giao diện lập trình dựa trên các đối tượng thể giới thực và các lớp cấp cao.

Tuy chỉ tập trung vào dựng hình và không cung cấp các thành phần hỗ trợ âm thanh, vật lý, tuy nhiên đó lại được xem là một ưu điểm của OGRE khi nó cho phép các lập trình viên có thể tùy ý lựa chọn các bộ thư viện để tích hợp vào để đáp ứng yêu cầu của từng dự án.

❖ Panda3D

Panda3D là một game engine mã nguồn mở, một thư viện dựng hình 3D và phát triển game được xây dựng bằng ngôn ngữ Python và C++. Bản thân engine được viết

bằng C++ và sử dụng một bộ sinh mã tự động để tạo thành các chức năng hoàn chỉnh cho engine thông qua giao diện của Python. Cách tiếp cận này giúp tận dụng được những ưu điểm của giao diện lập trình Python nhưng vẫn giữ lại được tốc độ của C++ trong nhân của engine. Panda3D thường được dùng trong các dự án ở trường học, hoặc các dự án thương mại cỡ lớn. Những game sử dụng Panda3D như: Toondown Online, Pirates of the Caribbean, A Vampyre Story, Signal Ops.

❖ Unreal Engine

Unreal Engine được phát triển bởi Epic Games, ra mắt vào năm 1998 với game Unreal. Unreal được xem là đối thủ trực tiếp của CryEngine trên thị trường engine làm game FPS. Được xây dựng dựa trên C++ nên Unreal có thể chạy tốt trên hầu hết mọi nền tảng như Windows, Linux, Android, iOS, Mac OS hay các máy chơi game Xbox, PS, Dream Cast.

2.3. Unity

2.3.1. Giới thiệu

Unity là một engine làm game 3D, dựng hình thời gian thực, thuộc loại Point-And-Click, được phát triển bởi Unity Technologies. Unity có thể chạy trên hệ điều hành Windows và Mac OS X. Sản phẩm tạo ra từ Unity có thể chạy trên hầu hết các nền tảng Windows, Mac, Linux, Wii, iOS, Android. Bên cạnh đó, Unity Engine có khả năng phát triển game nền web hỗ trợ hầu hết các trình duyệt chính.

Xét về mặt đồ họa, Unity không phải là engine mạnh nhất, CryEngine vẫn đang dẫn đầu với ưu thế đồ họa 3D cực kỳ chân thực. Tuy nhiên, để có được những trải nghiệm tốt như vậy, CryEngine hay Unreal đều cần những máy cấu hình từ khá đến cao, không giống như Unity, có thể chạy tốt trên những thiết bị cấu hình trung bình. Tuy đồ họa không thể so sánh nhưng chất lượng hình ảnh mà Unity mang lại cũng quá đủ để thỏa mãn những game thủ hiện nay. Hơn nữa, Unity còn có những ưu điểm mà không phải game engine nào cũng có:



Hình 2.14 Game 3D đầu tiên tại Việt Nam – Thủy Hử 3D sử dụng engine Unity3D

❖ **Đa nền tảng**

Unity được phát triển dựa trên kết quả của Mono Project, một dự án mã nguồn mở đa nền tảng trên ngôn ngữ C#. Chính vì điều đó, Unity có thể chạy tốt trên Windows, Mac OS, Linux, PlayStation 3, Xbox 360, Wii U, iOS, Android, Windows Phone, Blackberry 10, và thậm chí cả web và Flash. Với gói phần mềm của Unity, các studio có thể phát triển các game và phát hành trên nhiều nền tảng một cách nhanh nhất và tiết kiệm nhất.



Hình 2.15 Three Kingdoms Online – một game xây dựng bằng engine Unity3D chơi được trên cả Web, iOS và Android

❖ Hỗ trợ 2D và 3D

Unreal Engine và CryEngine đều chỉ hỗ trợ làm game 3D trong khi Unity hỗ trợ cả game 2D. Đây là một trong những ưu điểm vượt trội của Unity cho phép các studio vừa và nhỏ có thể sử dụng để phát triển các game từ nhỏ đến lớn, tùy vào kinh phí của họ. Phiên bản Unity 5 sắp ra mắt vào năm 2015 hứa hẹn sẽ cung cấp khả năng hỗ trợ game 2D tốt hơn nữa.

❖ Dễ sử dụng

Đối tượng hướng đến của Unity là cả các lập trình viên không chuyên và các studio chuyên nghiệp, do vậy nó được thiết kế để dễ dàng sử dụng nhất. Unity cung cấp một hệ thống toàn diện các công cụ từ thiết kế, soạn thảo mã nguồn đến sửa lỗi. Các tính năng được ưa thích của Unity là cho phép các lập trình viên có thể xây dựng các phần plugin để gắn vào editor của Unity và tính năng xem trước, cho phép nhà phát triển vừa kiểm lỗi, vừa thiết kế.

Hơn nữa Unity là một trong những engine phổ biến nhất thế giới, do đó, các nhà phát triển có thể dễ dàng tìm kiếm sự giúp đỡ từ cộng đồng rất nhanh.



Hình 2.16 République – một game viết cho iOS, PC và dòng máy Mac có đồ họa chân thực, sắc nét

- ❖ **Tính kinh tế cao:** Unity Technologies hiện cung cấp bản miễn phí engine Unity3D cho người dùng cá nhân và các doanh nghiệp có doanh thu dưới 100.000 USD/năm. Với bản Pro, người dùng phải trả 1.500 USD/năm – một con số rất khiêm tốn so với những gì engine này mang lại.

2.3.2. Các tính năng chính

Với mong muốn tạo ra công cụ hỗ trợ tốt nhất, Unity giúp người dùng có thể tập trung phát triển ý tưởng của mình hơn là quan tâm giải quyết các vấn đề kỹ thuật, các khó khăn công nghệ.

Các tính năng chính của Unity bao gồm:

❖ **Dựng hình**

- DirectX 11 được hỗ trợ làm cho hiệu suất dựng hình được nâng cao đáng kể so với các phiên bản trước
- Kỹ thuật chiếu sáng Pre-Pass và đa luồng cho ra ánh sáng chân thực nhất
- 100 kiểu shader sẵn có, từ đơn giản đến phức tạp phục vụ mọi mục đích của nhà phát triển

- Cho phép can thiệp xuống tầng thấp của quá trình dựng hình để tối ưu hoá

❖ **Ánh sáng**

- Tích hợp bộ công cụ Beast của Autodesk cho phép thực hiện kỹ thuật Lightmap một cách dễ dàng
- Kỹ thuật Lightmap kép được dùng cho các khung cảnh lớn giúp hạn chế số lượng lightmap được tạo ra
- Hard-shadow và soft-shadow tùy vào phần cứng cho trải nghiệm tốt nhất trên từng thiết bị

❖ **Kỹ thuật đặc biệt**

- Vô số hiệu ứng tiền xử lý toàn màn hình cho phép chất lượng hình ảnh tuyệt vời hơn
- Tăng tính chân thực với các hiệu ứng vẽ lên texture
- Chất lỏng với nhiều trạng thái và màu sắc
- Quản lý nhiều particle cùng lúc hiệu quả với Particle Editor

❖ **Âm thanh**

- Thao tác đưa âm thanh vào game một cách đơn giản, hiệu quả
- Hỗ trợ streaming với file kích thước lớn
- Cung cấp rất nhiều bộ lọc để tạo ra các hiệu ứng âm thanh khác nhau

❖ **Tài nguyên**

- Công nghệ Substance của Allegorithmic giúp giảm kích thước tài nguyên và tạo ra các hiệu ứng ấn tượng

❖ **Địa hình**

- Tạo địa hình đơn giản nhưng hiệu quả hoàn toàn bằng thao tác kéo thả
- Công nghệ Lightmap của Beast cho ánh sáng địa hình được chân thực

❖ **Vật lý**

- Tích hợp engine vật lý NVIDIA PhysX cho hiệu suất xử lý vật lý cực cao
- Thư viện Box2D được sử dụng cho game 2D giúp giảm chi phí tính toán

- Xây dựng game đua xe đơn giản hơn với Wheel Collider

❖ Trí tuệ nhân tạo

- Bộ thư viện tìm đường đi được tích hợp hỗ trợ phát triển các đối tượng tự động

❖ 2D and 3D

- Unity cho phép sử dụng model từ các phần mềm phổ biến như Maya, 3ds Max, Modo, Cinema 4D, Blender
- Xây dựng các màn chơi đơn giản chỉ với thao tác kéo thả
- Công cụ Profiler cho phép kiểm soát tài nguyên, tối ưu hoá hệ thống tốt hơn
- Hỗ trợ nhiều camera cho phép tích hợp các đối tượng 2D và 3D trong cùng một khung cảnh dễ dàng

❖ Hoạt hoạ

Từ phiên bản 4.0, Unity đã cung cấp thêm công cụ xây dựng hoạt hoạ. Các thao tác trực tiếp trên khung xương ở chế độ 2D hay 3D đều đơn giản và dễ dàng.

❖ Hiệu suất

- Giải pháp tiên xử lý Occlusion Culling độc quyền của Unity, phát triển cùng với Umbra Software đảm bảo chỉ những đối tượng được nhìn thấy mới cần phải vẽ
- LODGroups hỗ trợ thực hiện kỹ thuật Level-Of-Detail dễ dàng
- Cơ chế tự động tối ưu bằng cách gán nhiều khối nhỏ thành một khối lớn
- Cơ chế tự động loại bỏ các tài nguyên không sử dụng
- Bộ tối ưu mã GLSL trên mobile giúp tăng tốc độ lên 2-3 lần

❖ Đa nền tảng

- Tất cả các nền tảng mà Unity hỗ trợ bao gồm: iOS, Android, Windows Phone 8, Blackberry 10, Windows, Windows Store Apps, Mac OS, Linux, Web Player, PS3, PS4, PlayStation Vita, PlayStation Mobile, Xbox One, Xbox 360 và Wii U
- Quá trình build đơn giản với một thao tác

❖ Giao diện người dùng

Bộ thư viện GUI mới của Unity hỗ trợ đầy đủ các loại điều khiển từ cơ bản đến phức tạp, hỗ trợ layout và anchor giúp giao diện phù hợp với nhiều kích thước màn hình.

2.3.3. Ngôn ngữ lập trình

Với Unity, ta có thể sử dụng một hoặc kết hợp ba ngôn ngữ để phát triển là: C#, JavaScript và Boo.

Theo một khảo sát của trang <http://forum.unity3d.com> vào tháng 7/2013 thì ngôn ngữ được sử dụng nhiều nhất trong Unity là C#.

Ngôn ngữ	Tỷ lệ
Boo only	3.28%
C# only	40.10%
JavaScript only	27.92%
Boo & C#	1.97%
Boo & JavaScript	0.52%
JavaScript & C#	25.29%
Boo, C# and JavaScript	0.92%

Bảng 2.2 Tỷ lệ sử dụng các ngôn ngữ trong Unity

Giữa 3 ngôn ngữ này, không có sự khác biệt lớn về tốc độ và hiệu năng, tuy nhiên ta có một vài sự khác biệt giữa chúng.

❖ C#

Ưu điểm:

- Lượng người sử dụng lớn.
- Tốc độ nhanh hơn trong một vài trường hợp.

- Ràng buộc nghiêm ngặt, có thể phát hiện lỗi ngay lúc thực thi.
- .NET Framework hỗ trợ nhiều, có cộng đồng lớn ngoài cộng đồng Unity.

Nhược điểm: Chưa thấy

❖ JavaScript

Ưu điểm:

- Dễ học, dễ hiểu.
- Có lượng người sử dụng lớn.

Nhược điểm:

- Không nghiêm ngặt, chỉ có thể phát hiện lỗi lúc chạy.

❖ Boo

Ưu điểm:

- Đẹp, dễ đọc, dễ hiểu.
- Nghiêm ngặt, có thể phát hiện lỗi ngay lúc biên dịch.

Nhược điểm:

- Không có nhiều người sử dụng.

2.3.3.1. Các tính năng

❖ Đa nền tảng

Mono hỗ trợ cho cả hệ thống 32bit và 64bit của tất cả các kiến trúc máy tính. Các hệ điều hành được hỗ trợ bao gồm: Linux, Mac OSX, iOS, Sun Solaris, OpenBSD, FreeBSD, NetBSD, Microsoft Windows, Nintendo Wii, Sony PlayStation 3

❖ Đa ngôn ngữ

Rất nhiều ngôn ngữ có thể được sử dụng với nền tảng Mono. Bất kỳ ngôn ngữ nào có thể biên dịch được sang ngôn ngữ IL (Intermediate Language) đều có thể chạy được trên nền tảng Mono. Danh sách các ngôn ngữ có thể chạy được, bao gồm: C#, F#, Scala, Java, Boo, Nemerle, Visual Basic .NET, JavaScript, Operon, PHP, Object Pascal, Lua, Cobra và nhiều ngôn ngữ khác.

❖ Tương thích với các API của Microsoft .NET Framework

Mono có thể chạy được các ứng dụng ASP.NET, ADO.NET, SilverLight và Windows.Form mà không cần phải biên dịch lại.

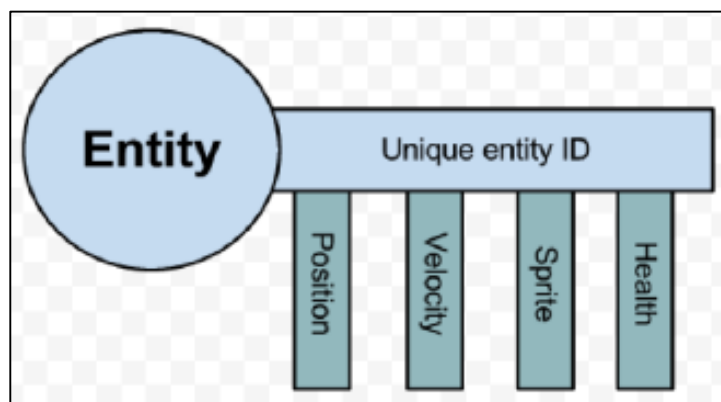
❖ Mã nguồn mở, ứng dụng miễn phí

Mono được phân phối dưới các giấy phép sau:

- C# Compiler xuất bản dưới 2 giấy phép MIT/X11 và GPL
- Các công cụ xuất bản dưới giấy phép GPL
- Các thư viện runtime xuất bản dưới giấy phép LGPL 2.0
- Các lớp thư viện xuất bản dưới giấy phép MIT X11
- Các thư viện liên quan với Microsoft được xuất bản dưới giấy phép Microsoft Permissive License, và một vài thư viện có giấy phép Apache2

2.3.4. Kiến trúc Component

Kiến trúc Component là hướng lập trình hướng dữ liệu (Document-Oriented). Trong đó một đối tượng thường được gọi là entity (thực thể). Các thực thể là những đối tượng rỗng, không chứa bất kỳ dữ liệu nào riêng biệt và không có bất kỳ sự khác nhau nào giữa chúng. Một thực thể có thể gắn thêm nhiều component khác nhau để thực hiện các mục đích riêng. Mỗi thực thể sẽ mang những thuộc tính và hành vi của component được thêm vào, khi đó các thực thể mới thực sự khác nhau. Trong Unity, các thực thể này gọi là GameObject.



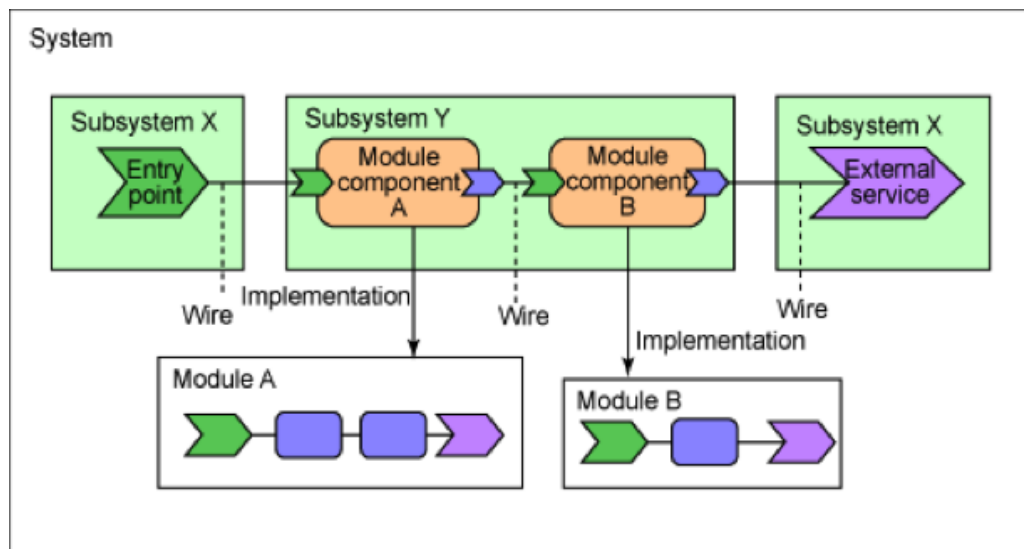
Hình 2.17 Thực thể theo kiến trúc component

Trong thực thể ở trên, có các component là: Position, Velocity, Sprite và Health.

Mỗi component thông thường sẽ có hai thành phần là:

- Attributes: dữ liệu riêng của component.
- Behaviours: các hàm xử lý các dữ liệu.

Mỗi component trong một thực thể sẽ hoàn toàn độc lập với nhau và được thực thi trên những hệ thống riêng biệt, chính vì thế, tốc độ thực thi của kiến trúc Component nhanh hơn kiến trúc cây thừa kế rất nhiều.



Hình 2.18 Các hệ thống xử lý các component

Trong kiến trúc Component hầu như không tạo ra một đối tượng nhất định giống như kiến trúc OOP mà tất cả các đối tượng trong component đều là thực thể. Không có kiến trúc theo cây thừa kế trong kiến trúc Component.

❖ Ưu điểm

- Tái sử dụng code tốt, giảm độ phức tạp của dự án. Một component có thể được sử dụng cho nhiều thực thể khác nhau có chung một số đặc điểm về component đó.
- Tốc độ xử lý cực nhanh vì mỗi component sẽ chạy trên một hệ thống riêng biệt nhau và không phải trả chi phí cho các hàm thừa kế trừu tượng như trong cấu trúc cây thừa kế.

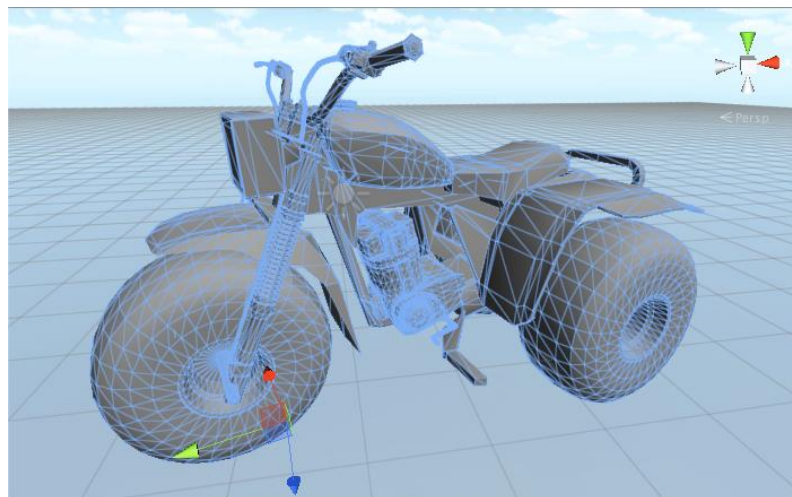
- Dễ vận hành và bảo trì game: mỗi component gần như hoàn toàn độc lập với nhau nên rất dễ thích nghi với những thay đổi trong game, ít ảnh hưởng tới các component còn lại.
- Dễ dàng cho các dự án có nhiều người cùng tham gia khi mỗi người có thể quản lý mỗi component riêng biệt.

❖ **Nhược điểm**

- Không thích hợp với các dự án nhỏ.
- Thời gian phát triển ban đầu lớn.

2.3.5. Các Component trong Unity

❖ **Nhóm Mesh**



Nhóm Mesh dùng để hiển thị các model 3D trong Unity.

Bao gồm các component:

- Mesh
Lớp dữ liệu, dùng để lưu các model 3D được import từ assets.
- Mesh Filter
Dùng để truyền một Mesh từ assets vào cho Mesh Renderer hiển thị.
- Text Mesh
Một Mesh đặc biệt dùng để hiển thị chữ trong không gian 3D.

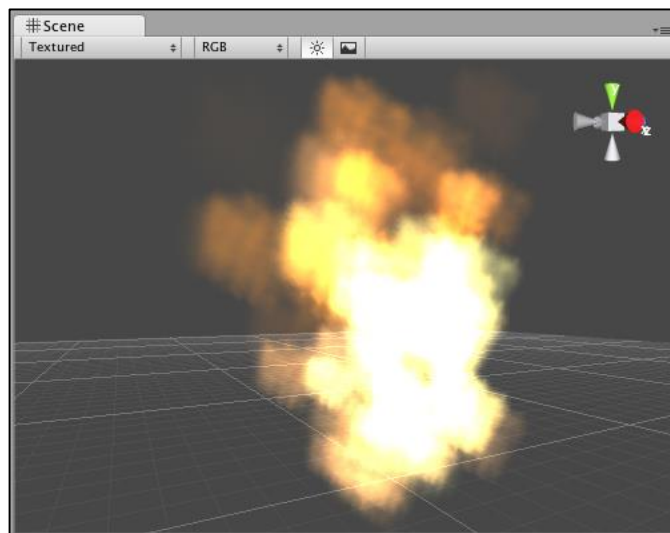
- Mesh Render

Dùng để hiển thị Mesh và Text Mesh trong không gian 3D. Một GameObject phải được gắn component Mesh Renderer mới có thể hiển thị được model 3D trong không gian.

Các định dạng model 3D được Unity hỗ trợ gồm 2 loại:

- Các định dạng đã kết xuất: .FBX, .OBJ
- Các định dạng chưa kết xuất: .MAX, .BLEND nhận từ các phần mềm 3D Studio Max hoặc Blender

❖ **Nhóm Effect**



Tạo ra các hiệu ứng trong game như: khói, hơi nước, hiệu ứng khí quyển... Nhóm này bao gồm các component:

- Particle System

Dùng để hiện thực một hiệu ứng particle nào đó.

- Trail Renderer

Tạo hiệu ứng lưu vết phía sau cho các đối tượng di chuyển trong không gian.

- Line Renderer

Tạo ra các đường vẽ trong không gian, dựa vào các điểm được xác định. Đường được vẽ không phải là các điểm mà là các texture có độ dày nhất định.

- Lens Flare

Được dùng để thêm vào một nguồn sáng để tạo hiệu ứng cho nguồn sáng đó. Hiệu ứng được tạo ra giống như khi một camera quay trực tiếp vào một nguồn sáng mạnh.

- Halo

Tạo hiệu ứng vầng sáng xung quanh một đối tượng. Halo được dùng chủ yếu cho hiệu ứng của một điểm sáng (Point Light).

- Projector

Tạo ra hiệu ứng đổ bóng lên một mặt phẳng.

❖ Nhóm Renderring



Tạo các thành phần hiển thị trong không gian 3D, hoặc trong giao diện 2D. Nhóm này bao gồm các component:

- Camera

Dùng để quay lại tất cả các đối tượng có trong không gian đưa lên màn hình. Đây là component không thể thiếu trong game. Nếu không có đối tượng nào chứa component này, sẽ không có gì được thể hiện lên màn hình.

- Flare Layer

Được gắn vào Camera để tạo ra hiệu ứng Lens Flare.

- GUI Layer

Tương tự Flare Layer, cần được gắn vào Camera để thể hiện các GUI Text và GUI Texture.

- GUI Text

Dùng để hiển thị chữ lên màn hình 2D.

- GUI Texture

Dùng để hiển thị Texture lên màn hình 2D.

- Light

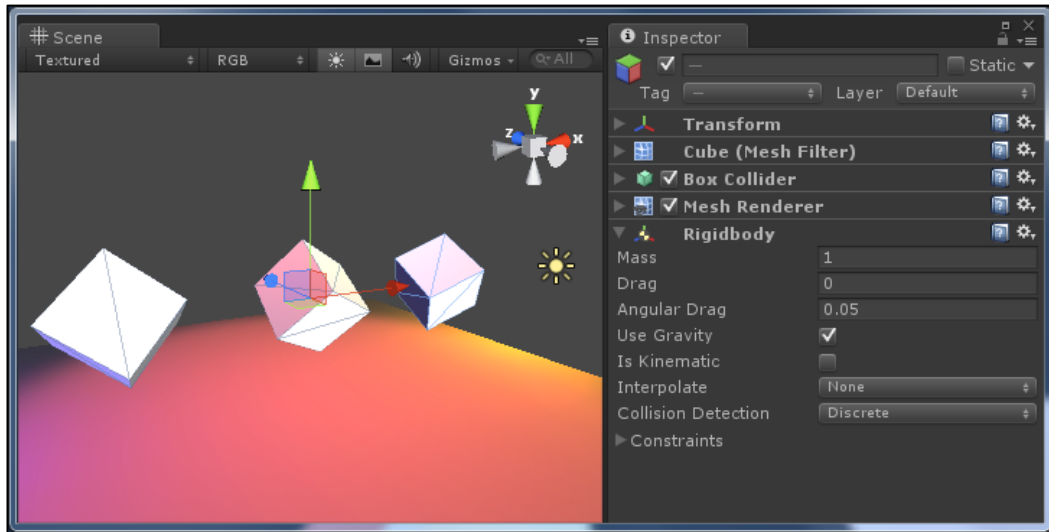
Tạo ra một nguồn sáng, chiếu sáng các thành phần hoặc toàn bộ các đối tượng có trong không gian. Có 4 loại Light:

- Direction: nguồn sáng song song, chiếu từ một nơi rất xa, có tác dụng với tất cả các đối tượng trong không gian. Ví dụ: ánh sáng mặt trời.
- Point: điểm sáng, chiếu từ 1 điểm ra tất cả các hướng. Ví dụ: bóng đèn tròn.
- Spot: nguồn sáng hình chóp. Ví dụ: đèn pin.
- Area: Chỉ có tác dụng khi tạo light mapping, và không có tác dụng lúc thực thi.

- Skybox

Dùng để giả lập một không gian vô hạn bao quanh toàn bộ không gian. Ví dụ: bầu trời.

❖ Nhóm Physics



Unity sử dụng NVIDIA PhysX để giả lập các tương tác vật lý trong game. Để tạo các chuyển động vật lý, va chạm, phản hồi, ... cần gắn một component physic vào đối tượng. Các component nhóm này bao gồm:

- Rigidbody

Cho phép các đối tượng có thể chịu tác động của lực, bao gồm cả trọng lực trái đất, lực cản không khí... Đối với các đối tượng tĩnh, không cần phản ứng với lực thì không cần tới component này.

- Character Controller

Được dùng để gắn vào các đối tượng mà người chơi điều khiển trong game như: nhân vật góc nhìn thứ nhất, góc nhìn thứ 3. Đối tượng này không chịu sự tác động của các lực vật lý để đảm bảo người chơi có thể di chuyển giống thực, nhưng nó có khả năng gây ra lực lên các đối tượng khác khi va chạm.

- Constant Force

Tạo ra một lực lập tức tác động liên tục lên đối tượng ở mọi khung hình thay vì chỉ một khung hình nếu sử dụng hàm *Rigidbody.AddForce*. Được dùng nhiều cho các loại tên lửa hay viên đạn bay.

- Collider

Cho phép các đối tượng có thể va chạm với nhau. Nếu không có một Collider nào được gắn vào đối tượng, thì chúng sẽ di chuyển xuyên qua nhau. Có các loại Collider sau:

- Box Collider: Khối lập phương
- Sphere Collider: Khối cầu
- Capsule Collider: Khối trụ có 2 đầu tròn.
- Mesh Collider: Khối được tạo nên từ một Mesh.
- Wheel Collider: Một dạng đặc biệt, dùng cho các loại xe.

❖ Nhóm Audio

Cung cấp các component liên quan đến âm thanh trong game. Bao gồm các component:

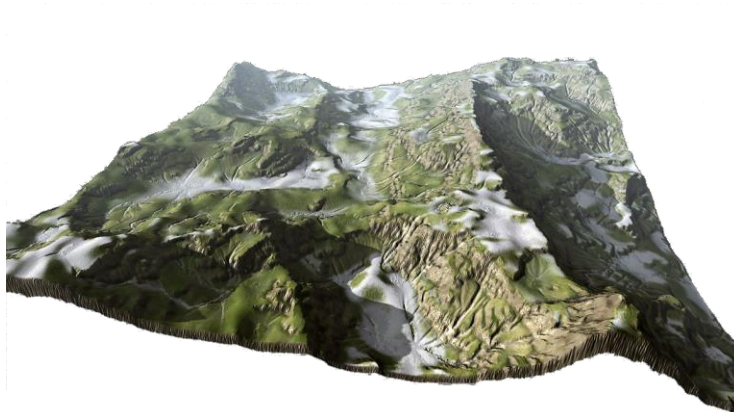
- Audio Listener

Thể hiện một chiếc micro để lắng nghe các âm thanh có trong không gian 3D và thường được gắn vào Camera chính. Trong môi trường 2D thì có thể gắn ở bất kỳ đâu cũng được. Ở một scene, chỉ được phép tồn tại một Audio Listener.

- Audio Source

Đại diện cho một đối tượng phát ra âm thanh trong không gian. Trong môi trường 3D, nếu một Audio Source có vị trí càng gần Audio Listener thì âm thanh mà người dùng nghe được sẽ càng to hơn.

❖ Nhóm Terrain



Tạo ra địa hình cho game như mặt đất, đường, đồi núi, cây cối, biển, sông suối,... Nhóm này chỉ có một component là Terrain. Các thao tác chỉnh Terrain đều được thực hiện trên giao diện đồ họa.

2.3.6. Script

Trong Unity, một file script định nghĩa một class cũng được xem là một component khi class đó thừa kế từ class MonoBehaviour.

Vì script là một phần không thể thiếu và được dùng vô cùng phổ biến, nên MonoBehaviour cũng là một class vô cùng quan trọng. Đây là class sẽ được hầu hết các class của người dùng thừa kế. Với JavaScript, tất cả các file script đều tự động thừa kế từ class này, tuy nhiên, với C# và Boo thì việc thừa kế phải được thực hiện tường minh.

Các thành phần quan trọng của class này bao gồm.

Thuộc tính:

Tên	Ý nghĩa
enabled	Nếu enabled thì hàm Update của MonoBehaviour sẽ được gọi mỗi frame.
Tag	Tag của GameObject, dùng để phân nhóm các GameObject.
name	Tên của GameObject, dùng để định danh các GameObject.

Bảng 2.3 Các thuộc tính của MonoBehaviour

Phương thức:

Tên	Ý nghĩa
GetComponent	Trả về component được gắn vào GameObject.
GetComponentInChildren	Trả về component được gắn vào GameObject hoặc bất kỳ đối tượng con của GameObject.

SendMessage	Gọi phương thức của GameObject theo tên có trong bất kỳ MonoBehaviour nào của đối tượng đó.
Invoke	Gọi phương thức theo tên sau một khoảng thời gian.
InvokeRepeating	Gọi lặp lại phương thức theo tên sau một khoảng thời gian.
CancelInvoke	Hủy hoặc dừng việc thực hiện 2 hàm trên.
IsInvoking	Kiểm tra phương thức có được chờ để gọi.
StartCoroutine	Bắt đầu một coroutine (tương tự thread).
StopCoroutine	Dừng tất cả coroutine chạy trên phương thức có tên này.
StopAllCoroutines	Dừng tất cả coroutine chạy trên đối tượng này.

Bảng 2.4 Các phương thức của MonoBehaviour

Bên cạnh đó, có một vài hàm rất quan trọng, thường được sử dụng để thừa kế lại trong các script của người dùng.

Tên	Ý nghĩa
Update	Sẽ được gọi ở mỗi frame, nếu MonoBehaviour được kích hoạt.
LateUpdate	Sẽ được gọi ở mỗi frame, sau hàm Update nếu MonoBehaviour được kích hoạt.
FixedUpdate	Được gọi ở mỗi frame cố định, nếu MonoBehaviour được kích hoạt, được dùng chủ yếu cho các xử lý vật lý, cần thời gian chính xác.
Awake	Được gọi khi đối tượng được load (trước hàm Start) và một lần duy nhất cho một đối tượng, thường được dùng để thực hiện các tiền xử lý.

Start	Được gọi đầu tiên (sau hàm Awake), thường được dùng để khởi tạo đối tượng.
Reset	Thiết lập lại các giá trị mặc định.
OnMouseEnter	Sẽ được gọi một lần khi đưa chuột vào GUIElement hoặc Collider.
OnMouseOver	Sẽ được gọi mỗi frame khi chuột nằm trên GUIElement hoặc Collider.
OnMouseExit	Sẽ được gọi một lần khi đưa chuột ra khỏi GUIElement hoặc Collider.
OnMouseDown	Sẽ được gọi lúc người dùng nhấn chuột và chuột đang nằm trên GUIElement hoặc Collider.
OnMouseUp	Sẽ được gọi khi thả chuột.
OnMouseDownDrag	Được gọi khi click lên GUIElement hoặc Collider và giữ trạng thái đó rồi di chuyển chuột.
OnTriggerEnter	Được gọi một lần khi các collider khác bắt đầu va chạm với trigger.
OnTriggerStay	Được gọi một lần cho từng frame cho tất cả các collider khác đang chạm vào trigger.
OnTriggerExit	Được gọi một lần khi collider khác không chạm vào trigger nữa.
OnCollisionEnter	Được gọi một lần khi collider/rigidbody bắt đầu chạm vào một collider/rigidbody khác.
OnCollisionStay	Được gọi một lần mỗi frame cho tất cả các collider/rigidbody khác va chạm với collider/rigidbody này.

OnCollisionExit	Được gọi một lần khi collider/rigidbody khác không còn va chạm với collider/rigidbody này.
-----------------	--

Bảng 2.5 Các phương thức cần thừa kế của MonoBehaviour

2.3.7. Physics

2.3.7.1. Tổng quan về Physics trong Unity

Vật lý trong game là tất cả những thứ liên quan đến sự chuyển động và tác động qua lại của các vật thể nhằm mô phỏng như trong thế giới thật, bao gồm sự chuyển động phức tạp của các vật thể, sự va chạm, phản hồi, ma sát, dòng chảy... dựa trên sự tác động của lực đẩy, trọng lực, gió...

Thông thường, CPU đảm nhiệm vai trò tính toán logic và vật lý trong game, GPU chỉ nhận nhiệm vụ dựng hình cho game. Chỉ những máy tính chơi game chuyên dụng, có card vật lý (PPU) thì các tính toán vật lý mới được thực hiện chuyên nghiệp ở đây.

PhysX	Core 2 Quad	GeForce GTX 280
Cores	4	240
GFLOPS	96	930
Fluids	1	15
Soft Bodies	1	12x
Cloth	1	13x

Bảng 2.6 So sánh tốc độ xử lý vật lý của CPU và GPU

Một lý do chính để chuyển việc tính toán vật lý từ CPU sang GPU là do sức mạnh to lớn trong tính toán của GPU cao hơn CPU rất nhiều lần. Qua bảng so sánh bên trên có thể thấy ưu thế rõ rệt của GPU so với CPU trong việc xử lý các hiệu ứng vật lý.

Hiện nay chỉ duy nhất engine vật lý PhysX của NVIDIA là có thể vừa được xử lý thông qua CPU lại vừa được xử lý thông qua GPU trong khi những engine như Havok, Box2D, Chipmunk lại chỉ có một cách xử lý duy nhất là thông qua CPU.

Unity sử dụng engine vật lý NVIDIA PhysX, điều này cho phép Unity có thể trình diễn những chức năng và hiệu ứng vật lý rất mượt mà và độc đáo.

Các tính năng của PhysX bao gồm:

- Giả lập đối tượng vật lý
- Phát hiện va chạm
- Điều khiển nhân vật
- Hiệu ứng particles
- Phương tiện giao thông
- Các loại trang phục

Với việc tích hợp hệ thống engine NVIDIA PhysX, Unity đã kế thừa được sức mạnh của hệ thống này để hỗ trợ các xử lý vật lý của mình mượt mà nhất.

2.3.7.2. Các Component vật lý trong Unity

❖ Rigidbody

Để làm cho một đối tượng chịu ảnh hưởng của các yếu tố vật lý (trọng lực, va chạm,...) thì ta chỉ cần thêm component Rigidbody cho đối tượng đó.

Rigidbody có các thuộc tính vật lý, được dùng cho những GameObject có khả năng chịu tác động lực và tác động lực lên đối tượng khác.

Các thuộc tính bao gồm:

Tên	Ý nghĩa
Mass	Khối lượng của đối tượng.
Drag	Sức cản không khí lên đối tượng khi di chuyển.
Angular Drag	Sức cản không khí lên đối tượng khi quay.
Use Gravity	Nếu được kích hoạt, đối tượng sẽ chịu tác động bởi trọng lực.
Is Kinematic	Nếu được kích hoạt, đối tượng sẽ không chịu tác động bởi hệ thống vật lý, nó chỉ được điều khiển thông qua Transform.

Interpolate	Chế độ làm mượt chuyển động. Gồm các tham số: <ul style="list-style-type: none"> - None: Không áp dụng - Interpolate: Transform sẽ mượt mà dựa trên Transform của khung hình trước đó. - Extrapolate: Transform sẽ mượt mà dựa trên dự đoán Transform của khung hình kế tiếp.
Collision Detetion	Được dùng để phát hiện va chạm trong các trường hợp vật di chuyển quá nhanh.
Constraints	Giới hạn phạm vi chịu tác động lực của các Rigidbody, gồm giới hạn di chuyển và giới hạn quay theo 3 trục.

Bảng 2.7 Các thuộc tính của Rigidbody

❖ Các component Collider

Collider là một loại component phải được thêm vào cùng với rigidbody để cho phép va chạm xảy ra. Nếu 2 rigidbody va đập vào nhau, hệ thống vật lý sẽ không tính toán va chạm trừ khi cả hai đối tượng đều có gắn một component Collider.

Trong các trường hợp khác, Collider được sử dụng để xác định vùng không gian xung quanh vật thể để nhận các sự kiện mouse, touch hoặc để hàm raycast xác định được chính xác đối tượng.

Có các loại Collider gồm:

Tên	Miêu tả
Box Collider	Hình dáng cơ bản của một khối hộp.
Sphere Collider	Hình dáng cơ bản của một khối cầu.
Capsule Collider	Hình dáng cơ bản của một khối trụ, hai đầu tròn (hình con nhộng).

Mesh Collider	Một Collider được tạo ra từ một Mesh của GameObject, không thể va chạm với Mesh Collider khác.
Wheel Collider	Một loại Collider đặc biệt sử dụng cho đối tượng xe cộ trong game.

Bảng 2.8 Các loại Collider

❖ Joint

Joint được dùng để kết nối các đối tượng vật lý trong Unity lại với nhau, tạo thành một hệ chuyển động, tương tác với nhau. Một ví dụ như: con lắc đơn, lò xo, ...

Có ba loại Joint cơ bản: Fixed Joint, Spring Joint và Hinge Joint với các mục đích sử dụng khác nhau.

- Fixed Joint: Gắn liền các đối tượng lại với nhau
- Spring Joint: Kết nối các đối tượng dạng lò xo
- Hinge Joint: Kết nối các đối tượng dạng sợi dây

❖ Constant Force

Constant Force giúp ta tạo lực tác động liên tục vào Rigidbody. Điều này hoạt động tuyệt vời cho những đối tượng tăng tốc tức thời như tên lửa, viên đạn, ...

Thuộc tính:

- *Force*: Vector lực tác dụng trong hệ tọa độ thế giới.
- *Relative*: Vector lực tác dụng trong hệ tọa độ địa phương.
- *Torque*: Vector của mô-men xoắn tác dụng trong hệ tọa độ thế giới.

Relative Torque: Vector của mô-men xoắn tác dụng trong hệ tọa độ địa phương.

Chương 3. ỨNG DỤNG AUGEMENTED REALITY VÀO PHÁT TRIỂN GAME

3.1. Tổng quan

3.1.1. Khảo sát các sản phẩm tương tự

Hiện tại, trên thế giới đã có một số ứng dụng, game phục vụ cho việc giáo dục, khơi mào và phát triển tư duy của trẻ, góp phần nâng cao được hiệu quả giáo dục.

3.1.1.1. Game Words



Địa chỉ	https://www.playosmo.com/en/words/
Mô tả	Đây là game học từ vựng tiếng anh do Tangible Play phát triển, chạy trên nền tảng iOS.
Cách chơi	<p>Người chơi đoán từ khóa dựa vào gợi ý là một bức ảnh diễn tả nội dung của từ khóa đó.</p> <p>Bằng cách chọn những chữ cái mà bạn đoán là có trong từ khóa và ném nó vào, chương trình sẽ tự nhận diện và phân tích, và hãy làm điều đó thật nhanh so với đối thủ của bạn</p>
Ưu điểm	<p>Nội dung đầy đủ và cập nhật</p> <p>Giúp nhớ từ vựng tốt hơn thông qua những hình ảnh sinh động</p> <p>Game mượt, hiệu ứng game tuyệt vời.</p>
Nhược điểm	<p>Chỉ hỗ trợ trên hệ điều hành IOS</p> <p>Giá thành cao</p>

	Chưa được phổ biến rộng rãi, chỉ áp dụng cho những gia đình có điều kiện
--	--

Bảng 3.1 Mô tả game Words

3.1.1.2. Game hỗ trợ Learn English Vocabulary Daily



Địa chỉ	https://play.google.com/store/apps/details?id=com.tflat.english.vocabulary
Mô tả	Đây là ứng dụng học tiếng anh do TFLAT-GROUP phát hành
Cách chơi	Phiên bản hiện tại gồm 5 trò chơi nhỏ: Bể cá 1, Bể cá 2, Éch từ vựng 1, Éch từ vựng 2, Giải cứu gấu trúc
Ưu điểm	Nội dung bám sát với bài học, từ vựng khá đa dạng với nhiều chủ đề khác nhau. Đồ họa game tương đối đẹp.
Nhược điểm	Game play chưa thật sự thú vị với người chơi

Bảng 3.2 Mô tả Learn English Vocabulary Daily

3.1.1.3. Game Tangram



Địa chỉ	https://www.playosmo.com/en/tangram/
Mô tả	Đây là game xếp hình do Tangible Play phát triển, chạy trên hệ điều hành iOS
Cách chơi	Người chơi sẽ ghép những mảnh nhỏ thành mô hình đúng với mô hình được cho sẵn
Ưu điểm	Nội dung đầy đủ và cập nhật Giúp phát triển khả năng tư duy logic của trẻ Game mượt, hiệu ứng game tuyệt vời.
Nhược điểm	Chỉ hỗ trợ trên hệ điều hành IOS Giá thành cao Chưa được phổ biến rộng rãi, chỉ áp dụng cho những gia đình có điều kiện

Bảng 3.3 Mô tả game Tangram

3.1.2. Bối cảnh nghiên cứu

Đa phần trẻ em ngày nay không thích đọc sách in. Các em có xu hướng đam mê các thiết bị số như iPad, điện thoại thông minh... Chúng có thể tìm thấy mọi thứ tiện ích khi sử dụng phương tiện này, thậm chí xem những bộ phim hoạt hình yêu thích hàng giờ liền mà không phải chuyển kênh như khi xem tivi. Điều này sẽ gây ra nhiều tác hại nghiêm trọng.



Hình 3.1 Sự quan tâm của trẻ em với thiết bị số

❖ **Khiến trẻ lười hoạt động thể chất**

Những thiết bị kỹ thuật số có thể là kho thông tin hữu ích nhưng không mang lại lợi ích gì cho sức khỏe của trẻ. Các em sử dụng hầu hết thời gian dán mắt vào màn hình sẽ lười vận động, không thích đi ra ngoài để hít thở không khí trong lành.

❖ **Khiến trẻ không còn hứng thú với đọc sách in**

Thiết bị thông minh giúp ích trong việc thu thập thông tin, nhưng những đứa trẻ được khuyến khích đọc sách in sẽ có lợi cho sự phát triển trí não hơn. Cách dễ nhất để trẻ thấm nhuần bài học là thông qua việc đọc sách. Những thiết bị kỹ thuật có thể giúp ích trong việc đọc sách điện tử nhưng chúng cũng có thể đưa đến sự xao lãng vì bị "cắm đố" bởi những trò chơi điện tử trong đó. Thực tế, một đứa trẻ nghiện game hiếm khi đọc sách.



Hình 3.2 Minh họa trẻ em chơi video game

❖ **Giảm đi giá trị của những mối quan hệ**

Đưa trẻ nên hiểu về giá trị của những mối quan hệ ngay từ khi còn ấu thơ. Nếu một đứa trẻ kết thân với những thiết bị tiện ích hoặc một anh hùng ảo như một người bạn thân, chúng sẽ chẳng bao giờ hiểu được giá trị thực của mối liên đới ngoài đời thực. Các nhà tâm lý cho rằng, bất kỳ một người nào cũng nên làm thân với những người bạn thật ngoài đời để lớn lên và phát triển hơn trong các mối quan hệ.

❖ **Giết chết năng lực tập trung**

Nghiên cứu đã chỉ ra rằng khi có quá nhiều thứ kích thích giác quan như âm nhạc lớn, tivi, hoặc Internet có thể gây hại đến khả năng tập trung của trẻ. Sự phiền nhiễu này có thể cũng ảnh hưởng xấu đến hành vi của các em. Vì thế, cha mẹ được khuyên nên tránh cho trẻ sử dụng những tiện ích này.

❖ **Làm nghèo đi trí tưởng tượng và khả năng sáng tạo**

Nghiên cứu đã chỉ ra rằng khi có quá nhiều thứ kích thích giác quan như âm nhạc lớn, tivi, hoặc Internet có thể gây hại đến khả năng tập trung của trẻ. Sự phiền nhiễu này có thể cũng ảnh hưởng xấu đến hành vi của các em. Vì thế, cha mẹ được khuyên nên tránh cho trẻ sử dụng những tiện ích này.

3.1.3. Động lực nghiên cứu

Các ứng dụng giải trí hiện nay, đa phần đều mang tính giải trí là chủ yếu, việc học hỏi được những kiến thức xã hội hầu như rất ít. Trẻ em đang dần bị cuốn vào thế giới ảo này và xa rời với việc học tập, mang đến sự lo ngại không nhỏ với các bậc phụ huynh.

Chính vì vậy, cần tạo ra một môi trường giải trí lành mạnh, không áp lực và tạo điều kiện học tập tốt cho trẻ đồng thời các bậc phụ huynh có thể giám sát con em mình tốt hơn, hướng dẫn chúng ra khỏi vòng xoáy của những trò chơi trực tuyến.

Nhưng làm thế nào để có thể tạo ra được những sản phẩm tốt và thật sự mang đến được sự hứng thú đối với trẻ là một vấn đề lớn. Bên cạnh đó, làm sao có thể kết hợp được những kiến thức xã hội vào các trò chơi đặt ra một thách thức không nhỏ.

3.1.4. Mục tiêu nghiên cứu

Công nghệ thực tế ảo tăng cường AR là sự kết hợp của thế giới thật và vật thể ảo. Với nhu cầu ngày càng lớn về tương tác với thế giới ảo trong môi trường thật, công nghệ thực tế ảo sẽ ngày càng phát triển và sẽ thành xu hướng của thế giới công nghệ trong tương lai.

Hiện nay có rất nhiều hãng lớn đã bước chân vào nghiên cứu lĩnh vực này, như Google, Facebook, Samsung, Microsoft... Và trong số đó có hãng công nghệ Qualcomm. Qualcomm đã phát triển một công nghệ mang tên Vuforia, kết hợp với công cụ Unity sẽ giúp chúng ta tạo ra các ứng dụng thực tế ảo.

Trong bối cảnh đó, nhóm sẽ phát triển một kho ứng dụng giải trí dành cho trẻ em, kết hợp với những kiến thức xã hội. Khóa luận này chỉ dừng lại ở việc phát triển một ứng dụng giải trí đó là:

Game myWords: Học từ vựng tiếng anh thông qua hình ảnh bằng cách sắp xếp những chữ cái trong bộ hai mươi sáu chữ cái thành từ khóa đúng.

3.2. Thiết kế game

3.2.1. Tổng quan về game

❖ Ý tưởng

Tiếng Anh là một ngôn ngữ thông dụng, phổ biến nhất trên thế giới, việc học nó không phải là dễ dàng, xuất phát từ thực tế đó, để giúp trẻ em có thể học tốt hơn, không nhàm chán và phát triển được tư duy. Nhóm đã nghiên cứu, phân tích và quyết định phát triển sản phẩm này.

Thông qua chơi game, trẻ em sẽ được học những từ vựng tiếng anh từ cách đọc, cách viết, ngữ nghĩa của từ thông qua các hình ảnh, âm thanh thực tế.

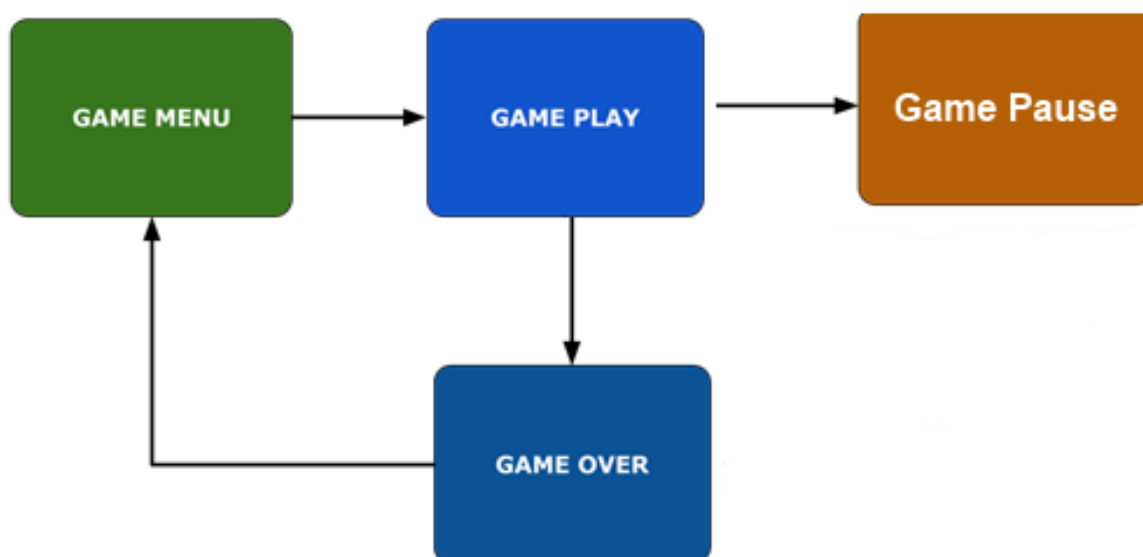
❖ Thể loại game

Game thuộc thể loại giải đố, thách thức người chơi, yêu cầu họ phải hoàn thành các thử thách đưa ra trong khoảng thời gian sớm nhất.

❖ Đối tượng hướng tới

Trẻ em đã đi học, được đào tạo trong môi trường có dạy ngoại ngữ là tiếng anh, độ tuổi từ 6 – 15 tuổi.

❖ Tóm tắt hướng đi của game



Hình 3.3 Tổng quan về follow game

❖ **Cảm nhận về game**

Nhìn nhận ban đầu, game sẽ mang tới thách thức cho người chơi qua các bức hình với mức độ khó dễ khác nhau, âm thanh sống động và hiệu ứng mượt mà là những gì mà game sẽ mang lại cho người chơi.

3.2.2. Game play và Mechanics

3.2.2.1. Game play

❖ **Tiến trình game**

Trong mỗi màn chơi sẽ chọn ngẫu nhiên một bức ảnh thuộc một chủ đề nào đó có thể là về tự nhiên, con người, con vật, đồ vật, cây cối...

Nhiệm vụ của người chơi trong một màn chơi là xác định đúng từ khóa trong màn chơi đó, từ khóa sẽ được gợi ý bằng cái ô chữ.

Game sẽ kèm theo một bộ phụ kiện là những ký tự được in ra, chỉ bộ ký tự đi cùng với Game mới có thể dùng được. Nhiệm vụ người chơi là đặt những ký tự này trước màn hình thiết bị thông minh có cài Game, để thiết bị nhận diện được những ký tự này.

Mỗi ký tự đúng người chơi được cộng điểm tương ứng với từng ký tự.

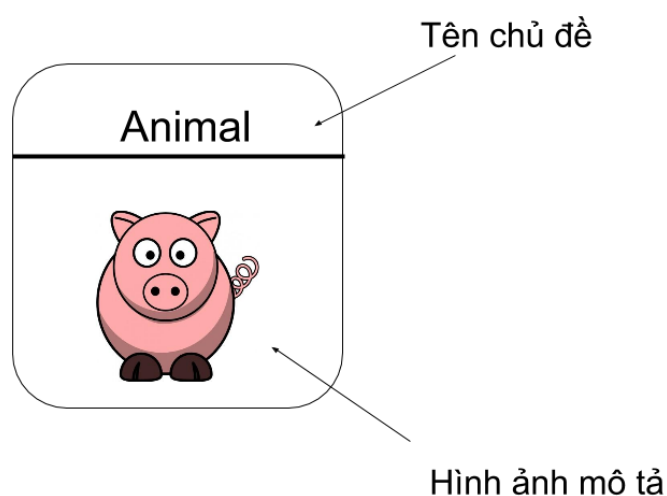
❖ **Độ thách thức**

Tùy thuộc vào chế độ chơi mà đòi hỏi những kỹ năng riêng, đòi hỏi sự nhạy bén, suy nghĩ nhanh, nhanh tay, nhanh mắt, chọn đúng những ký tự có trong từ khóa mà người chơi đoán. Tuy nhiên vào chế độ khó hơn, đòi hỏi người chơi phải phối hợp những kỹ năng với nhau để hoàn thành được từ khóa đã cho.

❖ Các đối tượng

– Category

Category là chủ đề trong game, với mỗi chủ đề sẽ bao gồm nhiều từ khóa có liên quan tới nó. Có rất nhiều chủ đề hay và hấp dẫn trong game cùng với đó là những hình ảnh chân thực, mô tả được đúng nội dung từ khóa mà bức ảnh muốn truyền tải đến cho người chơi như: động vật, gia đình, đồ dùng học tập, công việc, hành động...



Hình 3.4 Chủ đề trong game myWords

Việc vượt qua được các bộ từ trong game, người chơi có thể học được các từ vựng tiếng Anh một cách nhanh chóng và nhớ lâu hơn.

Với kho dữ liệu lớn, nhiều đề tài hấp dẫn và đặc biệt là luôn được nâng cấp sẽ mang đến người chơi những trải nghiệm mới, thích thú hơn đặc biệt là đối tượng trẻ em.

– **Photo**

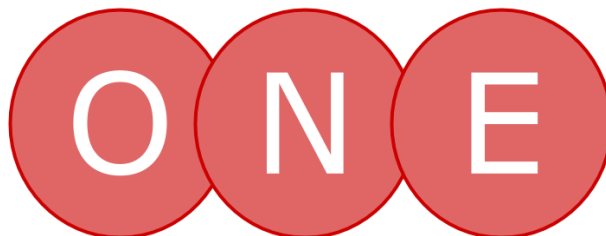
Photo là một bức ảnh diễn tả nội dung từ khóa mà người chơi cần đoán, hình ảnh được sưu tầm từ nhiều nguồn chất lượng, phù hợp với đối tượng trẻ em.



Hình 3.5 Hình ảnh trong game myWords

– **Word**

Word là một từ khóa yêu cầu người chơi phải đoán được từ bức ảnh được gợi ý, nó là một từ tiếng anh, không vượt quá 11 ký tự.



Hình 3.6 Từ khóa trong game myWords

– *Character*

Character là một ký tự trong từ khóa mà người chơi có thể chọn, nằm trong bộ 26 chữ cái LaTinh.



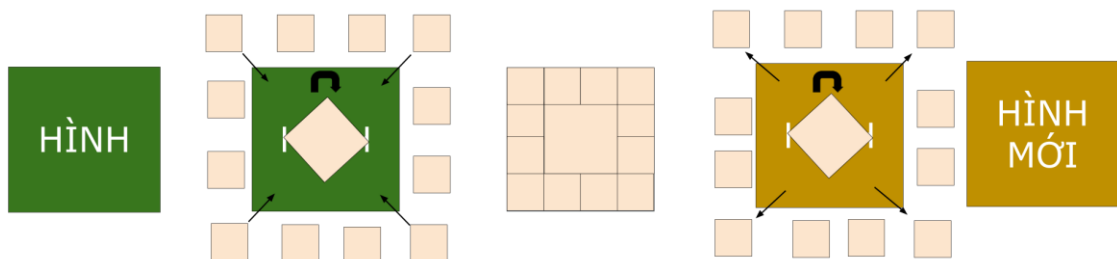
Hình 3.7 Ký tự trong game myWords

3.2.2.2. Mechanics – Cơ chế game

❖ **Hiệu ứng game**

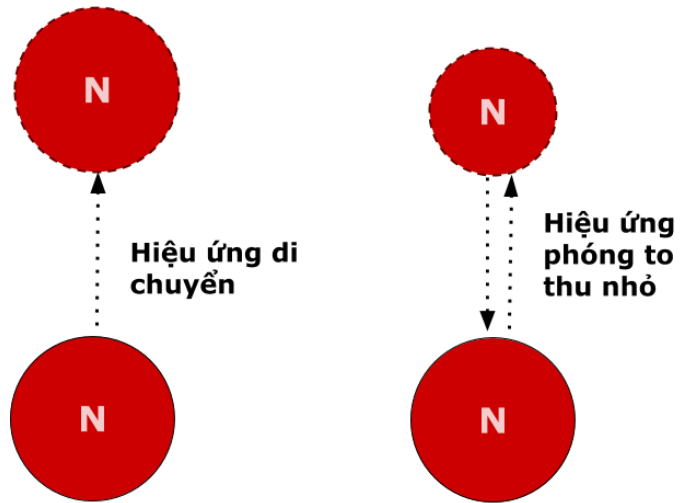
Các hiệu ứng được sử dụng trong game, mang đến trải nghiệm khác lạ cho người chơi, các đối tượng không cứng nhắc tại một chỗ, mà di động lên xuống qua lại kết hợp, hòa quyện vào nhau tạo nên một thế giới game sinh động.

– *Hiệu ứng đổi hình*



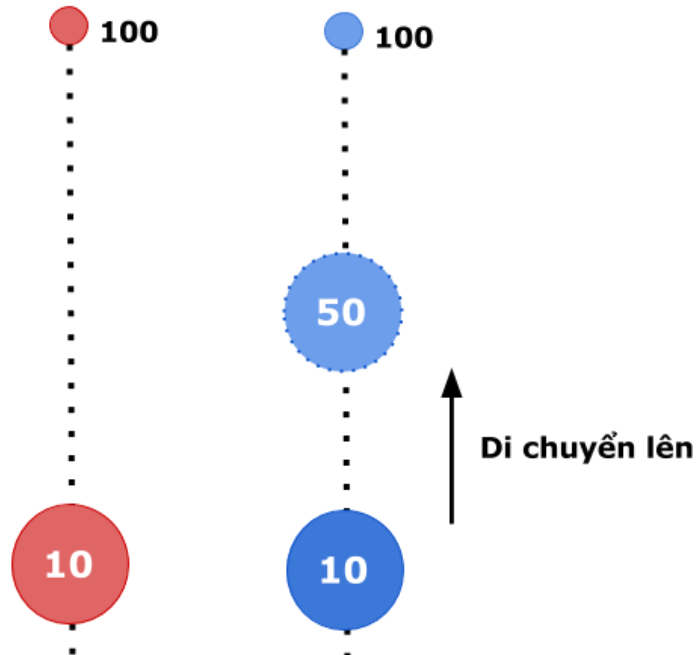
Hình 3.8 Hiệu ứng đổi hình

– *Hiệu ứng di chuyển chữ*



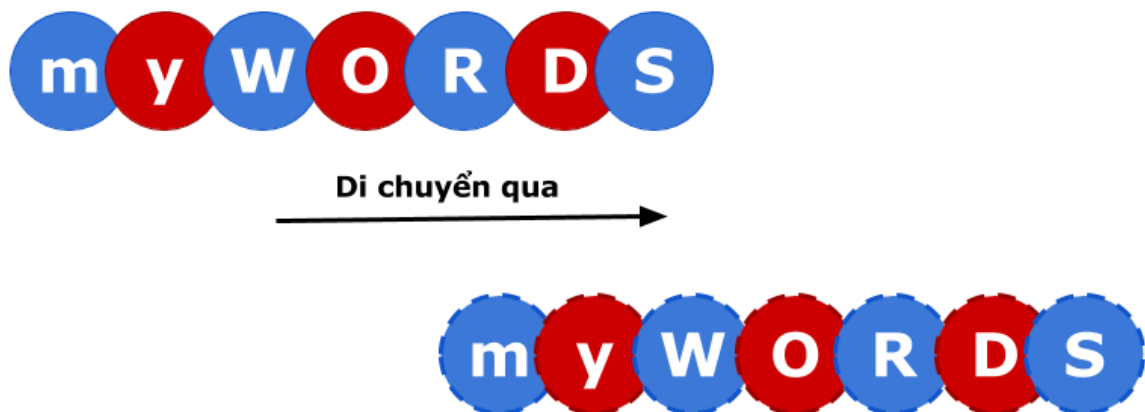
Hình 3.9 Hiệu ứng di chuyển chữ

– *Hiệu ứng di chuyển điểm*



Hình 3.10 Hiệu ứng di chuyển điểm

– *Hiệu ứng di chuyển word*



Hình 3.11 Hiệu ứng di chuyển từ

3.2.3. Thiết kế màn chơi

❖ Chọn bộ từ

Người chơi sẽ chọn một bộ từ bất kỳ trong game, với mỗi bộ từ đi kèm là những từ khóa mà người dùng phải vượt qua. Việc vượt qua được một bộ từ người chơi sẽ cải thiện được điểm số của người chơi, điểm càng cao, càng gần với mức tối đa có thể đạt được thì người chơi đang càng gần với đích đến của mình đó là học hết được tất cả các từ vựng trong đó.

❖ Chế độ người chơi

Trong game có hai chế độ người chơi chính là **Single Player Mode** và **Multi Player Mode**.

- **Single Player Mode:** Ở chế độ chơi này, người chơi sẽ lựa chọn các chế độ màn chơi khác nhau: *Easy*, *Normal*, *Hard*. Với mỗi chế độ màn chơi khác nhau thì cách chơi game và tính điểm mỗi màn cũng sẽ tương ứng khác nhau.
 - o **Easy:** Ở chế độ màn chơi này, người chơi không bị ràng buộc về vấn đề thời gian, có thể chọn một ký tự bất kỳ trong từ khóa được cho là đúng. Khi sai hết số lượng ký tự cho phép thì sẽ tự động

chuyển sang từ khóa khác và được xem là sai một từ khóa. Nếu người chơi sai hết số lượng từ cho phép thì sẽ thua cuộc.

- **Normal:** Ở chế độ màn chơi này, game sẽ ràng buộc thêm vấn đề thời gian, nghĩa là trong khoảng thời gian cho phép người chơi phải hoàn thành được từ khóa hiện hành, nếu hết thời gian thì xem như sai một từ khóa. Nếu người chơi sai hết số lượng từ cho phép thì sẽ thua cuộc.
- **Hard:** Khác với hai chế độ chơi trên, ở chế độ chơi này sẽ khó hơn, nghĩa là trong khoảng thời gian cho phép người dùng phải đoán đúng từ khóa và phải theo tuần tự từng ký tự, nếu các ký tự sai vị trí xem như sai một từ khóa.
- **Multi Player Mode:** Ở chế độ chơi này, sẽ có hai đội chơi thi đấu với nhau và tính điểm, khi hoàn thành bộ từ, đội chơi nào đạt điểm số cao hơn thì đội chơi đó sẽ thắng.

❖ Cơ chế tính điểm

Sau khi hoàn thành một từ khóa, số điểm của người chơi sẽ được tính theo công thức: $a * x - y$

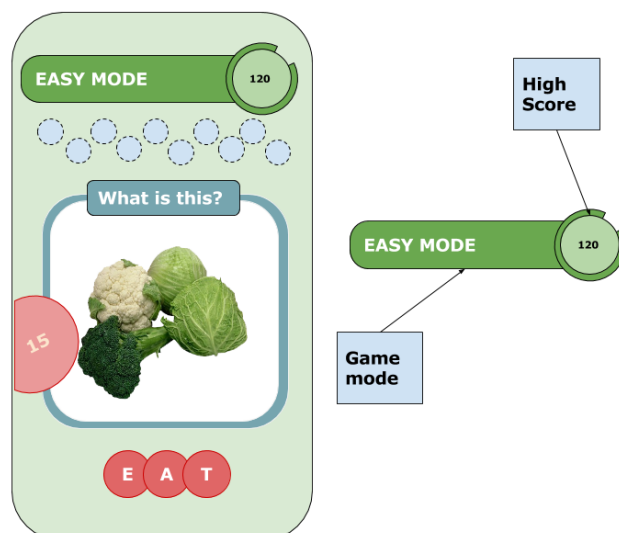
Trong đó:

- a là hệ số điểm phụ thuộc vào chế độ chơi(Easy: 2, Normal: 3, Hard: 4)
- x là số từ đúng
- y là số từ sai

Khi hoàn thành xong một bộ từ, nếu số điểm là cao hơn so với điểm hiện tại thì sẽ được lưu lại. Điểm số này thể hiện được mức độ hoàn thiện của người chơi.

3.2.4. Hệ thống giao diện

❖ Hệ thống giao diện trực quan - Visual system



Hình 3.12 Hệ thống HUD của game

Thanh trạng thái của game hiển thị chế độ người chơi hiện tại và hiển thị điểm số cao nhất của người chơi đó.

❖ Âm thanh, nhạc nền

Hệ thống nhạc nền, hiệu ứng âm thanh được sưu tầm thì nhiều nguồn khác nhau, âm thanh của game phù hợp với độ tuổi của trẻ em.



Âm thanh trong game được chia thành 2 phần:

- Nhạc nền: bản nhạc không lời được sưu tầm từ những trang sound effect nổi tiếng, được miễn phí bản quyền.

- Hiệu ứng âm thanh: bao gồm các hiệu ứng như: nhận diện chữ, âm thanh chuyển hình ảnh, âm thanh báo chiến thắng, âm thanh báo thua...

❖ Hệ thống trợ giúp

Hệ thống trợ giúp của game bao gồm hướng dẫn sử dụng game, thiết lập cấu hình âm thanh của game.



3.2.5. Môi trường và công cụ phát triển

❖ Môi trường phát triển

- Unity3D 5.2
- Visual Studio 2013
- MonoDevelop

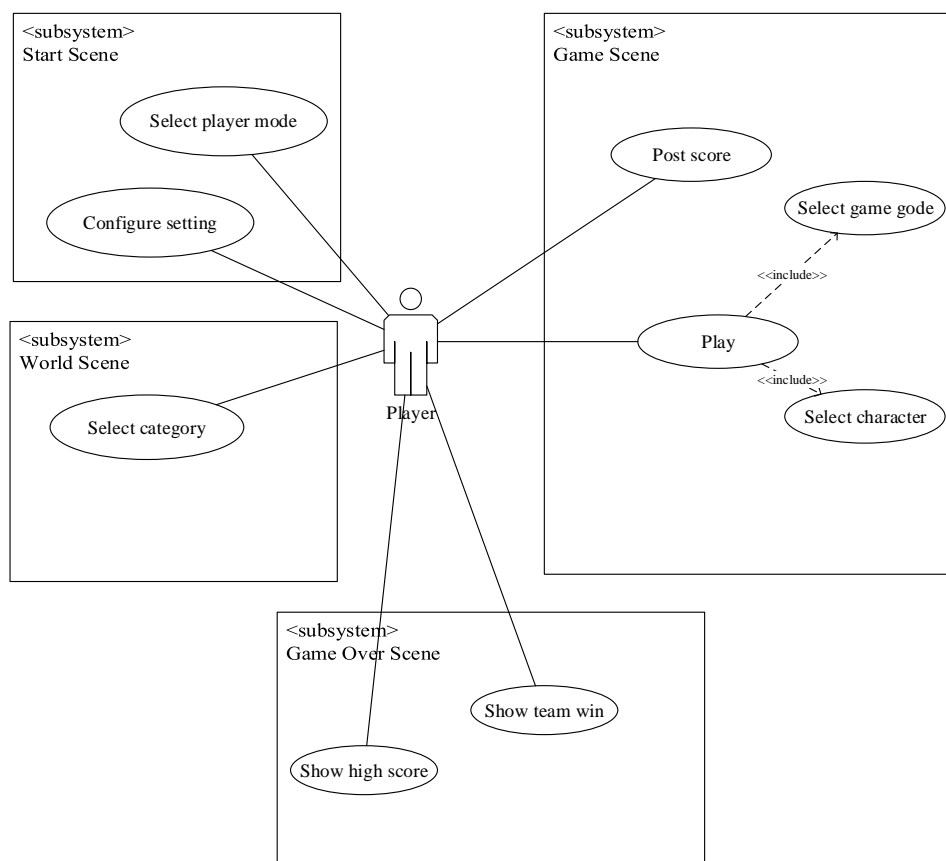
❖ Thư viện hỗ trợ

- Vuforia SDK 5.0

3.2.6. Sơ đồ thiết kế

3.2.6.1. Sơ đồ use case

Game myWords có 4 màn hình chính, do đó, sơ đồ use case được chia thành 4 subsystem tương ứng với các màn hình: StartScene, WorldScene, GameScene, GameOverScene.



Hình 3.13 Sơ đồ use case của game myWords

Danh sách các Actor:

STT	Tên Actor	Ý nghĩa
01	Player	Người chơi chính

Hình 3.14 Danh sách các Actor trong sơ đồ use case của game myWords

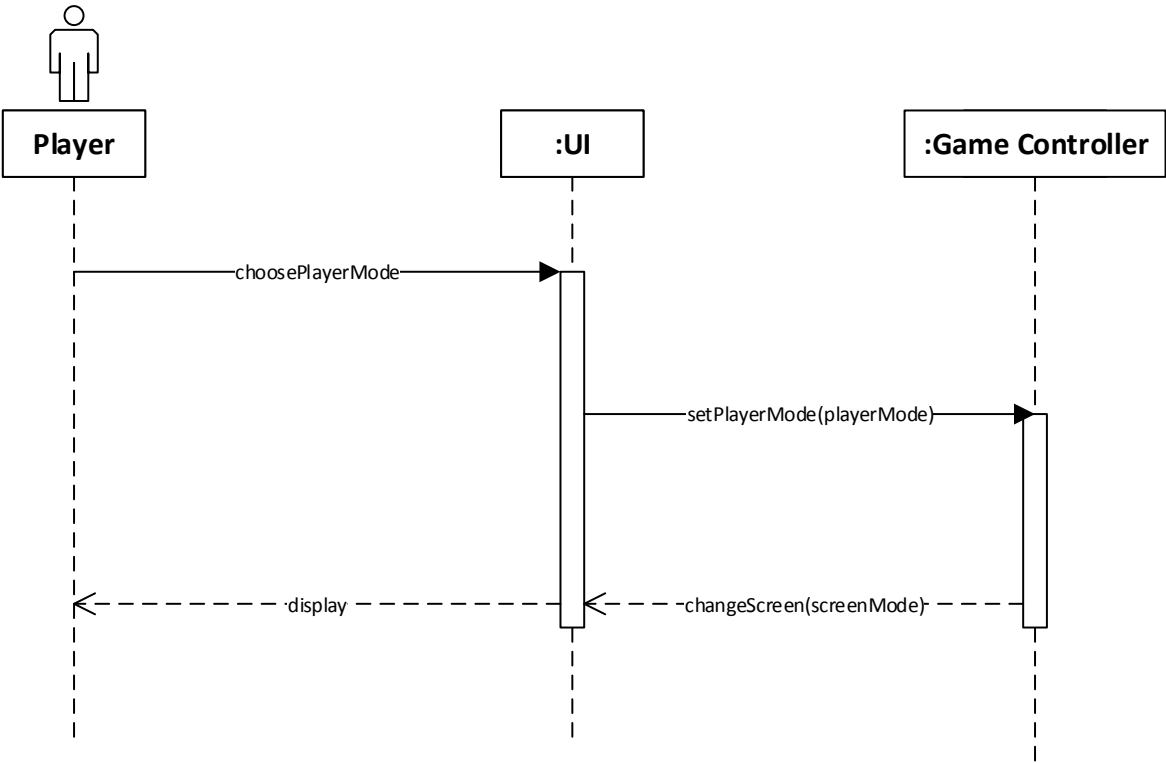
Danh sách các use case:

STT	Tên use case	Ý nghĩa
01	Select player mode	Chọn chế độ người chơi, gồm có 2 chế độ chơi chính là: Single player mode và Multi player mode
02	Configure setting	Thiết lập cài đặt âm thanh
03	Select category	Chọn gói từ
04	Play	Các thao tác màn hình chơi chính
05	Select game mode	Chọn chế độ màn chơi
06	Select character	Chọn ký tự và đưa vào màn hình
07	Show team win	Hiển thị thông tin đội thắng cuộc
08	Show high score	Hiển thị điểm số cao nhất của người chơi
09	Post score	Lưu lại điểm xuống vùng nhớ

Bảng 3.4 Danh sách các use case trong sơ đồ use case của game myWords

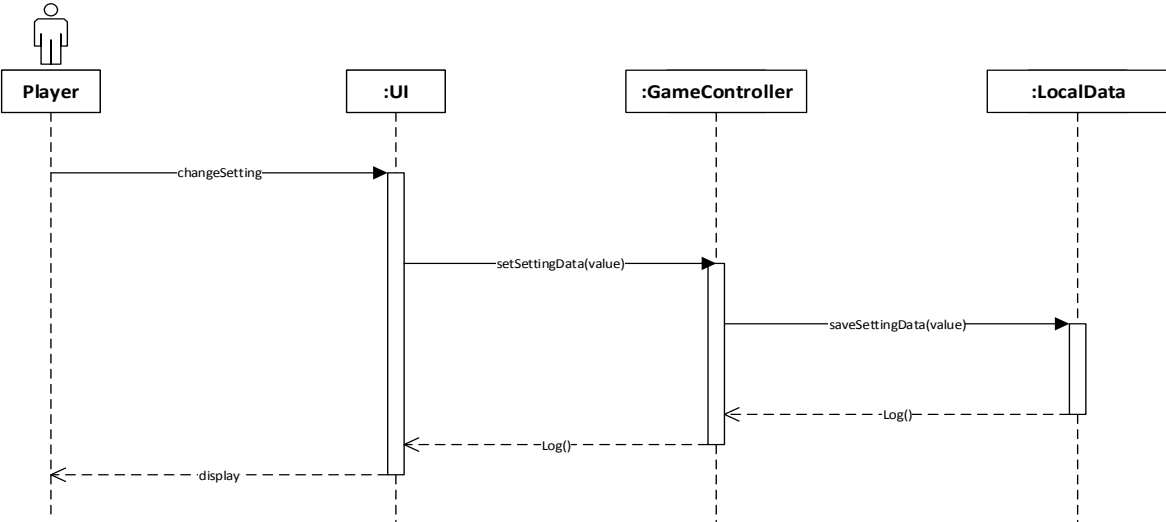
3.2.6.2. Sơ đồ tuần tự

❖ Sơ đồ tuần tự use case Select player mode



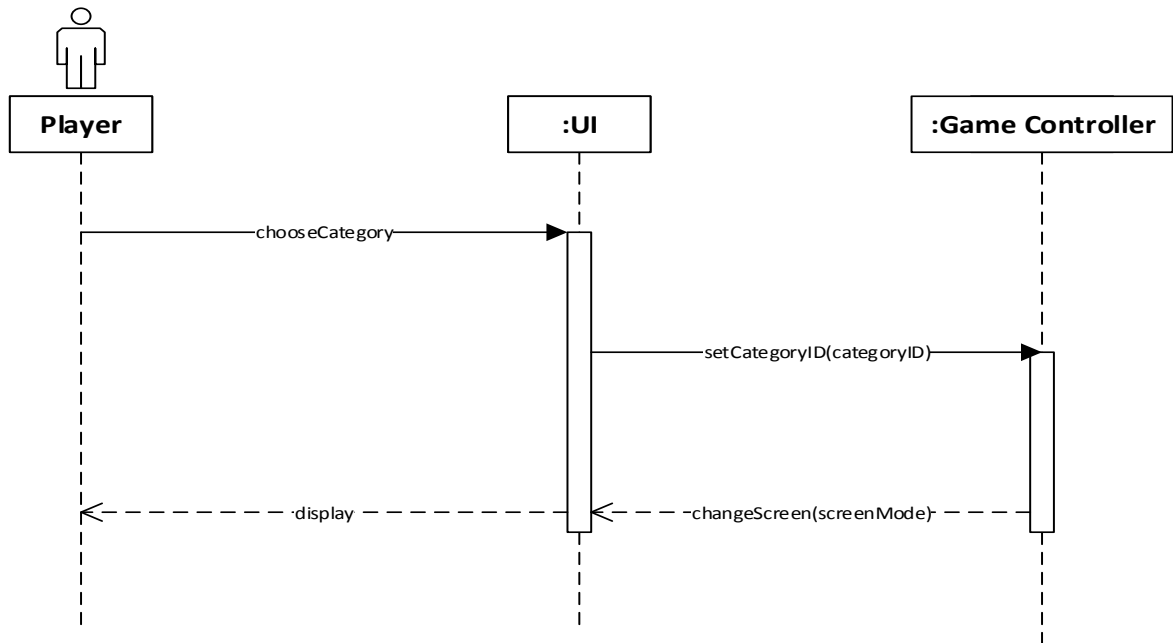
Hình 3.15 Sơ đồ tuần tự use case Slect player mode

❖ Sơ đồ tuần tự use case Configure setting



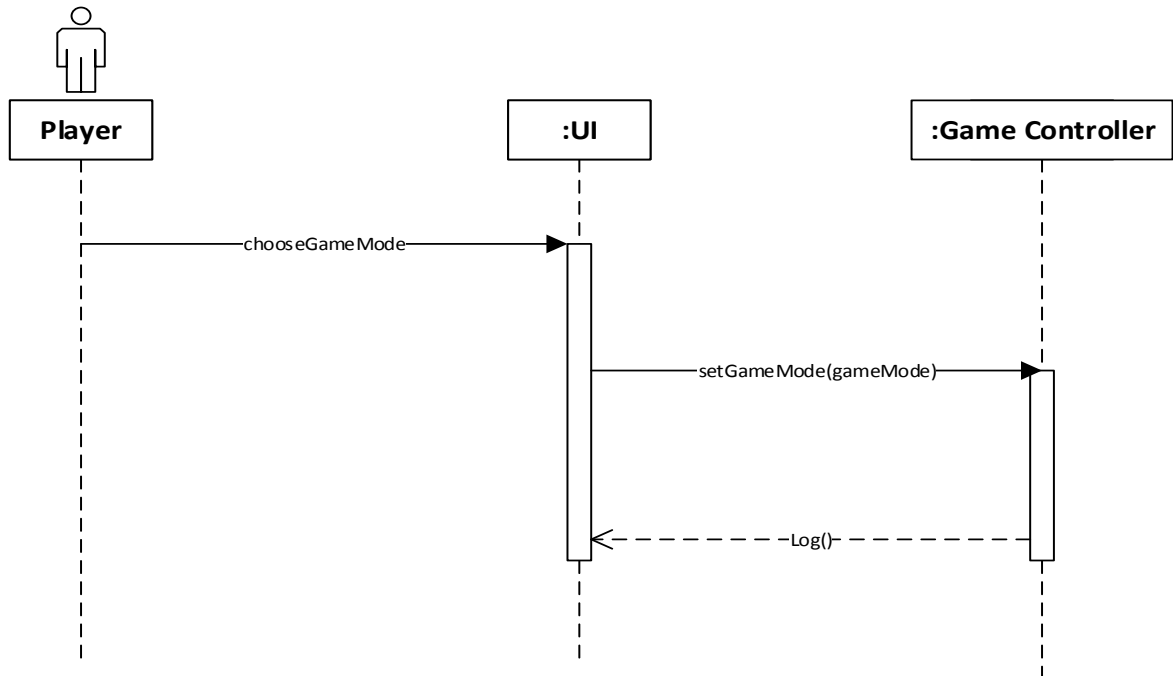
Hình 3.16 Sơ đồ tuần tự use case Configure setting

❖ Sơ đồ tuần tự use case Select category



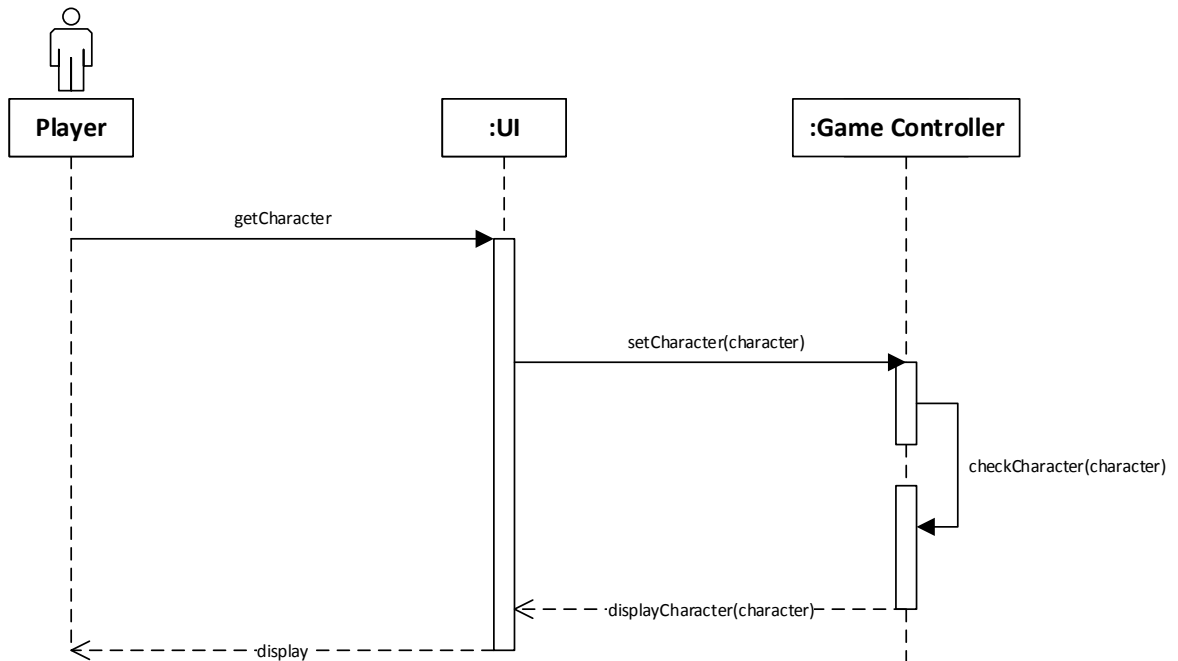
Hình 3.17 Sơ đồ tuần tự use case Select category

❖ Sơ đồ tuần tự use case Select game mode



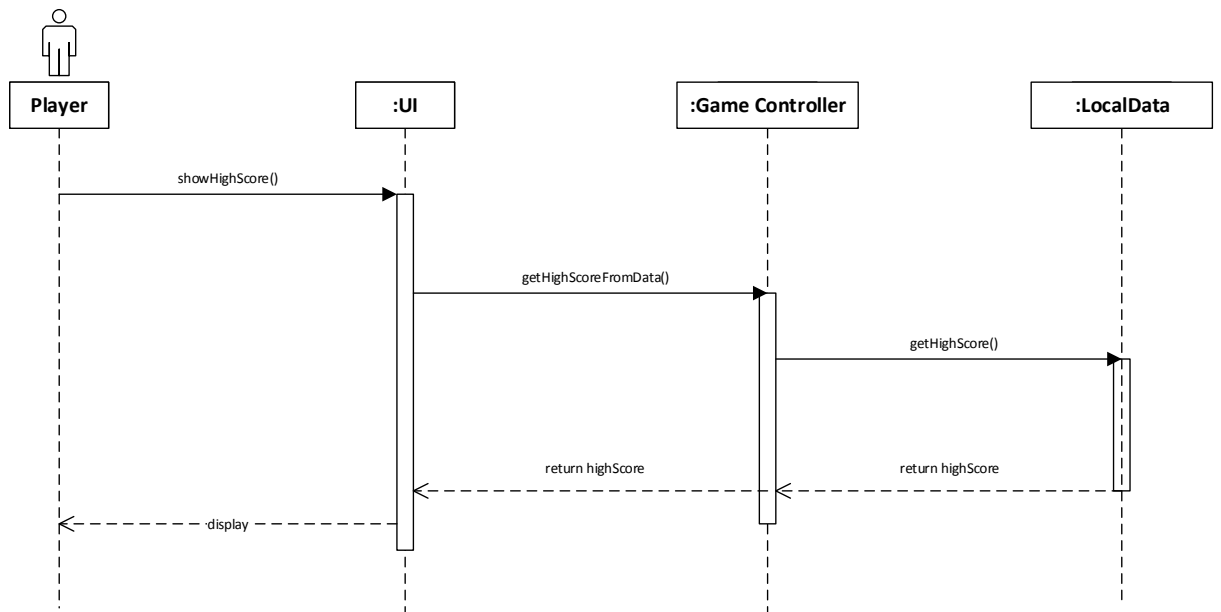
Hình 3.18 Sơ đồ tuần tự use case Select game mode

❖ Sơ đồ tuần tự use case Select character



Hình 3.19 Sơ đồ tuần tự use case Select character

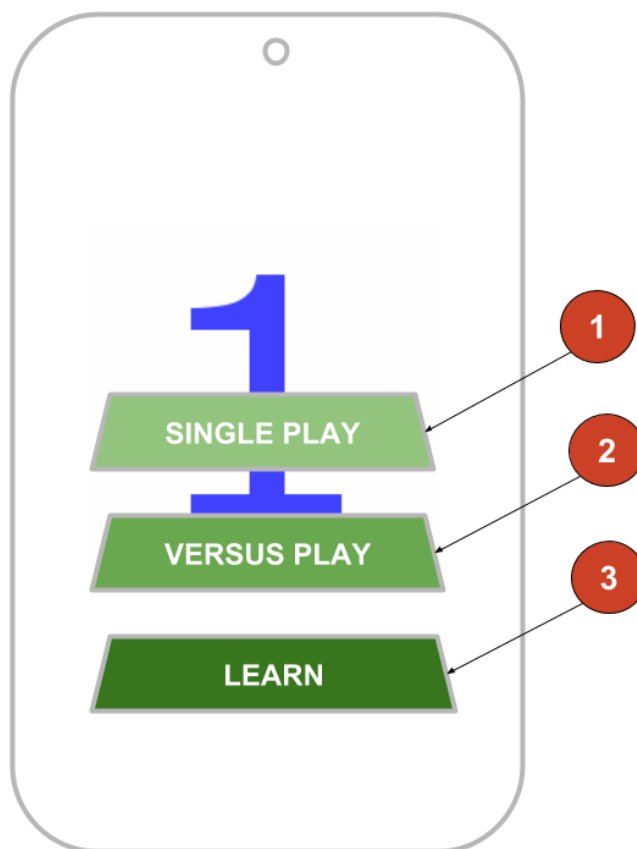
❖ Sơ đồ tuần tự use case Show high score



Hình 3.20 Sơ đồ tuần tự use case Show high score

3.2.7. Thiết kế giao diện

❖ Màn hình Start Game



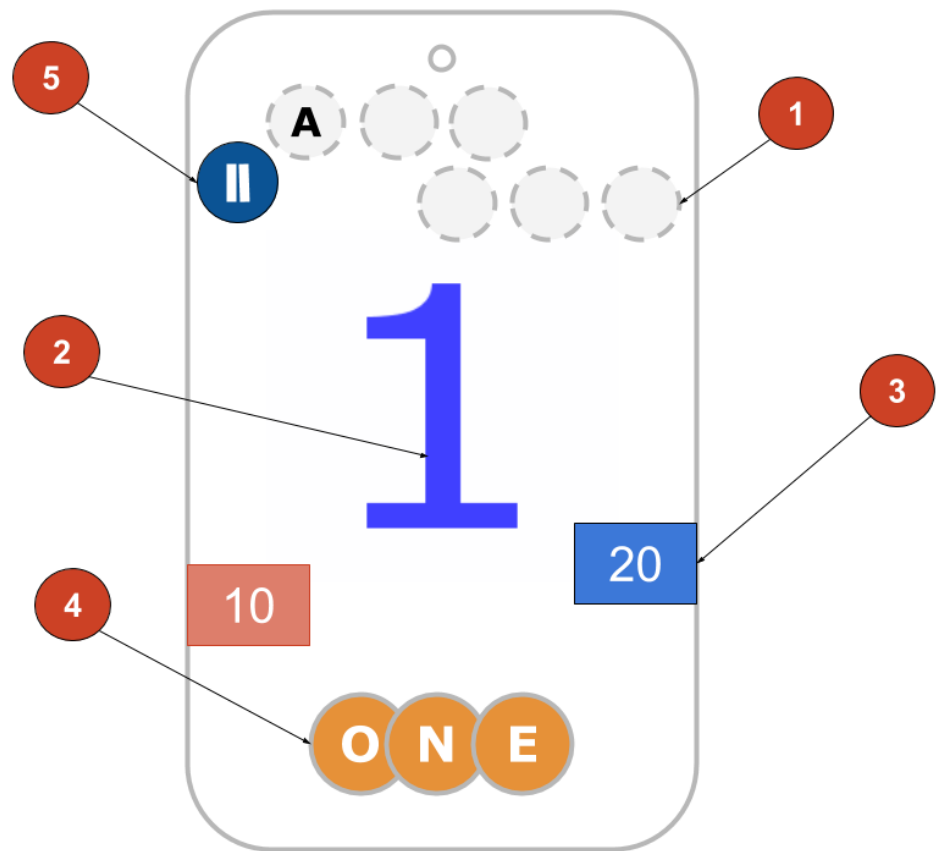
Hình 3.21 Màn hình start game myWords

Chú giải giao diện:

STT	Tên	Chức năng	Ghi chú
01	Single Play	Chế độ một người chơi	
02	Versus Play	Chế độ chơi đối kháng	
03	Learn	Chế độ học từ	

Bảng 3.5 Các thành phần trong màn hình game start - myWords

❖ Màn hình Play Game



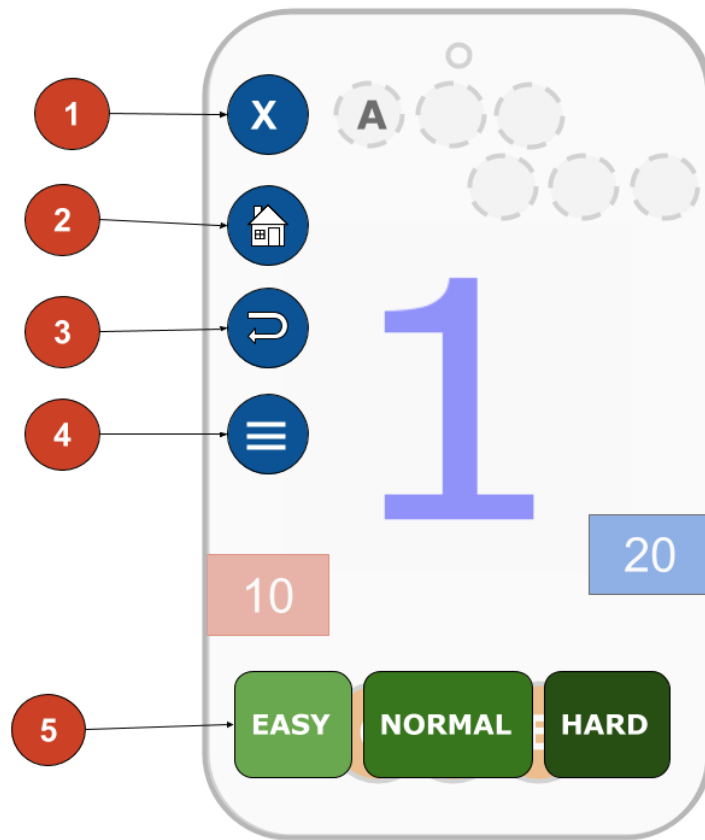
Hình 3.22 Màn hình world game myWords

Chú giải giao diện

STT	Tên	Chức năng	Ghi chú
01	List wrong word	Danh sách các từ sai	
02	Photo	Hình ảnh	
03	Score	Điểm	
04	List word	Danh sách từ đúng	
05	Pause	Nút tạm dừng game	

Bảng 3.6 Các thành phần trong màn hình game play – myWords

❖ Màn hình Pause Game



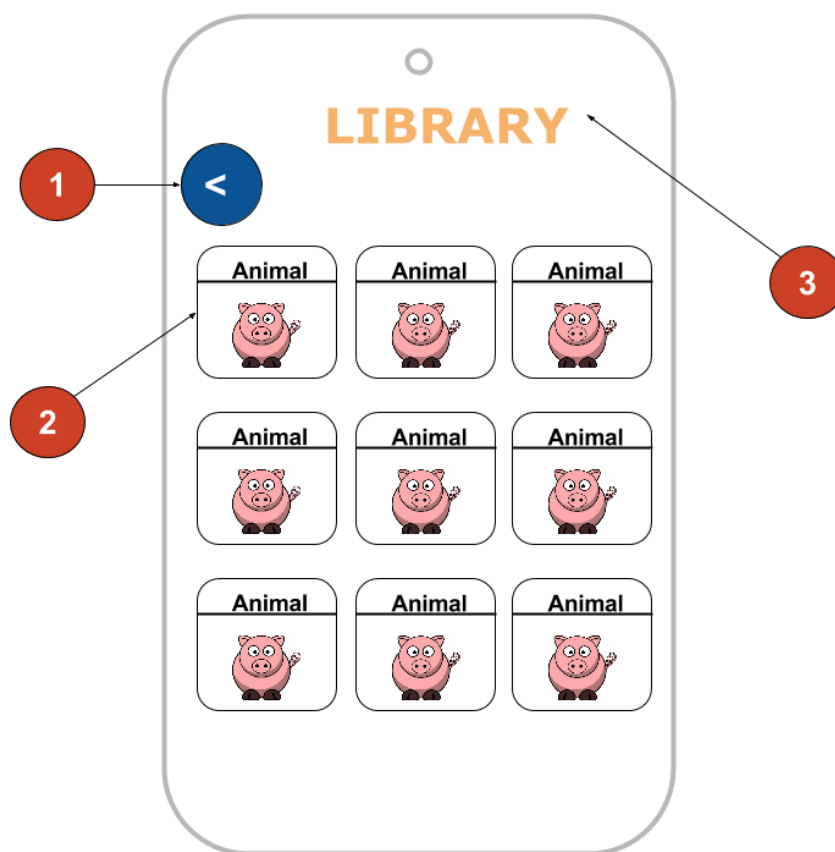
Hình 3.23 Màn hình world game myWords

Chú giải giao diện

STT	Tên	Chức năng	Ghi chú
01	Exit	Thoát màn hình pause	
02	Home	Quay trở lại màn hình ban đầu	
03	Restart	Khởi động lại màn chơi	
04	Library	Chọn bộ từ	
05	Mode	Chế độ chơi	

Bảng 3.7 Các thành phần trong màn hình game pause– myWords

❖ Màn hình Library



Hình 3.24 Màn hình library game myWords

Chú giải giao diện:



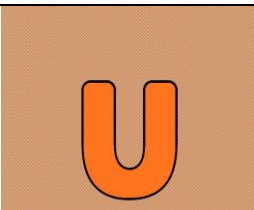
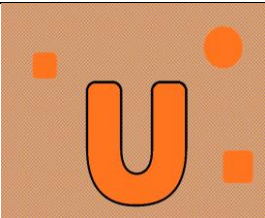

STT	Tên	Chức năng	Ghi chú
01	Back	Nút quay lại	
02	Category	Bộ chủ đề	
03	Title	Tiêu đề	

Bảng 3.8 Các thành phần trong màn hình world game – myWords

3.3. Thiết kế bộ công cụ hỗ trợ

3.3.1. Bộ chữ cái

Để tạo ra những bộ từ phù hợp với nhu cầu của game, nhóm tiến hành phân tích và thử nghiệm để chọn ra những bức ảnh có khả năng nhận diện tốt nhất. Dưới đây là bảng mô tả về quá trình thực nghiệm, qua đó đánh giá được mức độ nhận diện ảnh của Vuforia.

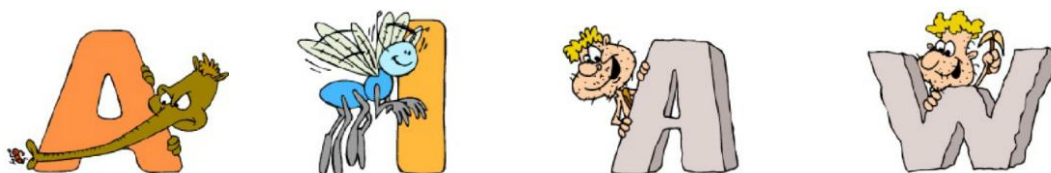
STT	Hình ảnh	Mô tả	Đánh giá
1		Hình ảnh được lựa chọn có ít chi tiết.	★ ★ ★ ★ ★
2		Hình ảnh được thêm các chi tiết phụ như: các hình học	★ ★ ★ ★ ★
3		Hình ảnh được thêm nền với nhiều chi tiết	★ ★ ★ ★ ★
4		Hình ảnh được thêm nền và các chi tiết chữ viết và hình ảnh	★ ★ ★ ★ ★
6		Hình ảnh được thêm hình hoạt họa nhiều chi tiết	★ ★ ★ ★ ★

Bảng 3.9 Trường hợp kiểm thử










Sau khi tiết hành kiểm thử trên thiết bị thật với nhiều trường hợp và các hình ảnh khác nhau thì rút ra được những kết luận khi lựa chọn đầu vào cho ứng dụng với những yếu tố giúp cho việc nhận dạng diễn ra nhanh chóng và chính xác.

- Những hình ảnh sử dụng để nhận dạng nên có nhiều chi tiết.
- Hình ảnh không phải là những mẫu lặp lại ví dụ: cánh đồng cỏ...
- Ảnh phải có đủ độ sáng, và tương phản tốt.
- Phải là ảnh 8 bit hoặc 24 bit định dạng PNG, JPG dung lượng nhỏ hơn 2M, JPG phải là RGB hoặc màu xám (không CMYK).

Kết quả cuối cùng nhóm đã tạo ra được 2 bộ chữ cái, mỗi bộ gồm 26 ký tự. Sau đây là một vài hình ảnh về thành quả mà nhóm đã đạt được:



Hình 3.25 Một số chữ cái trong bộ 26 chữ cái ABC

<input type="checkbox"/>		R_Z	Single Image	★★★★★	Active	Dec 10, 2015 21:15
<input type="checkbox"/>		R_Y	Single Image	★★★★★	Active	Dec 10, 2015 21:15
<input type="checkbox"/>		R_X	Single Image	★★★★★	Active	Dec 10, 2015 21:15
<input type="checkbox"/>		R_W	Single Image	★★★★★	Active	Dec 10, 2015 21:14
<input type="checkbox"/>		R_V	Single Image	★★★★★	Active	Dec 10, 2015 21:14
<input type="checkbox"/>		R_U	Single Image	★★★★★	Active	Dec 10, 2015 21:14
<input type="checkbox"/>		R_T	Single Image	★★★★★	Active	Dec 10, 2015 21:14
<input type="checkbox"/>		R_S	Single Image	★★★★★	Active	Dec 10, 2015 21:13
<input type="checkbox"/>		R_R	Single Image	★★★★★	Active	Dec 10, 2015 21:13

Hình 3.26 Mô tả về bộ 26 chữ cái khi đăng ký trên vuforia

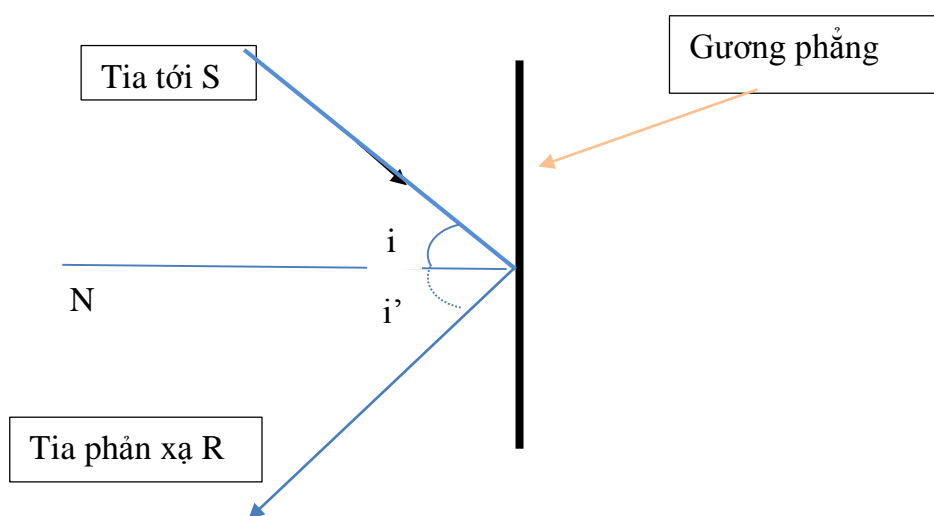
3.3.2. Lăng kính

Lăng kính có tác dụng đổi hướng chiếu của camera tạo sự thuận lợi và thoải mái cho người chơi. Giúp người chơi có thể thực hiện những thao tác dễ dàng hơn trong lúc chơi.

Cấu tạo:

- Gương phẳng kích thước khoảng (2cm x 4cm).
- Khung xốp giữ gương phẳng

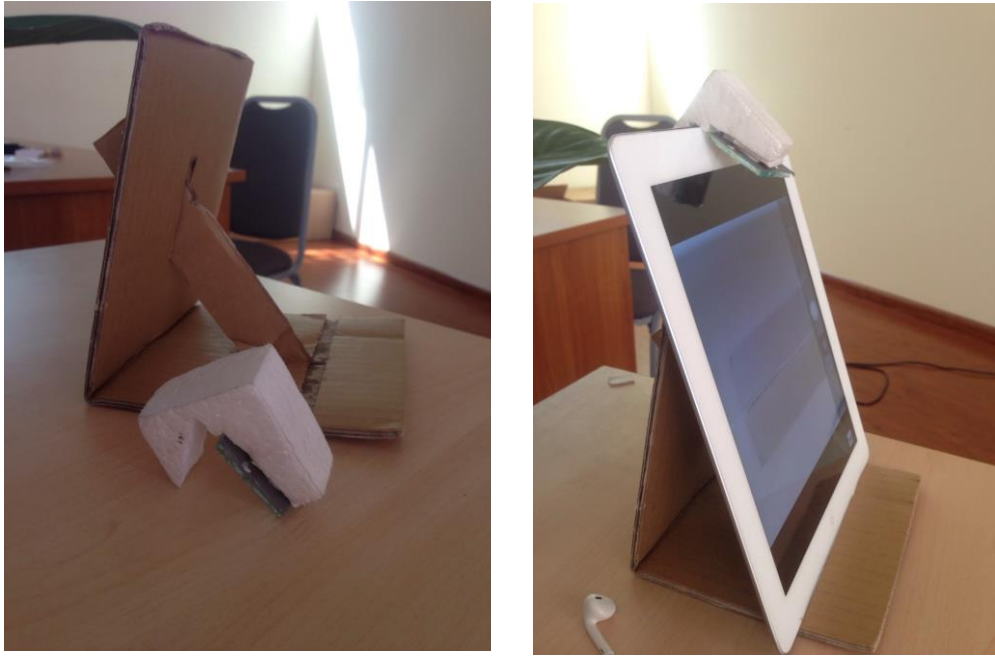
Nguyên lý: Sử dụng định luật phản xạ ánh sáng



Hình 3.27 Định luật phản xạ ánh sáng

Góc tới i	Góc phản xạ i'
30°	30°
45°	45°
60°	60°

Bảng 3.10 Thông số góc tới và góc phản xạ



Hình 3.28 Lăng kính và ứng dụng thực tiễn

3.3.3. Đế giữ thiết bị

Để giữ thiết bị đứng thẳng tạo sự thoải mái trong lúc sử dụng và chơi game, nhóm đã tạo ra một sản phẩm đơn giản từ những vật liệu có sẵn và dễ kiếm, đó là bìa carton.



Hình 3.29 Đế giữ thiết bị

Chương 4. ỨNG DỤNG THỰC TIỄN

4.1. Cài đặt thực tiễn

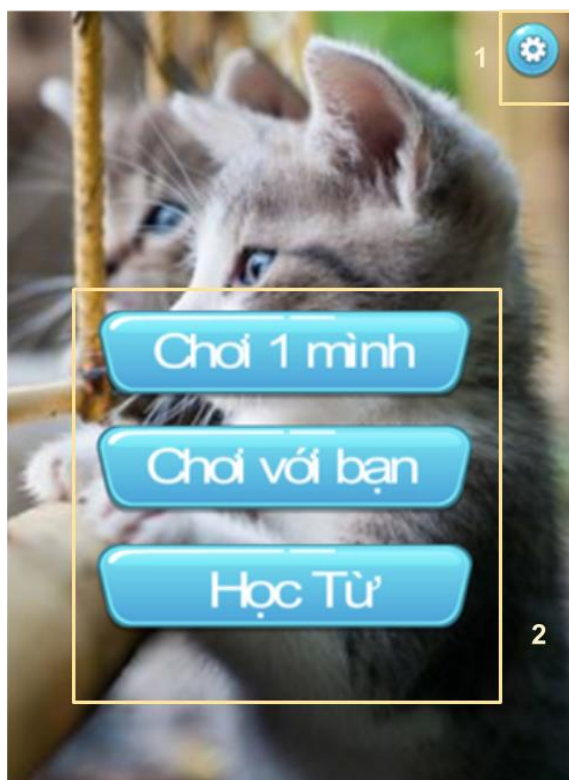
Nền tảng hỗ trợ: Android từ 2.3 trở lên và IOS 7

Kích thước màn hình: Game thích hợp cho màn hình từ 5.0 inch trở lên, màn hình lớn tạo điều kiện thao tác người dùng được dễ dàng hơn

Môi trường áp dụng: Game thích hợp cho đối tượng trẻ em đã từng được học qua tiếng anh. Có thể áp dụng trong các cuộc thi học thuật tại trường tiểu học, giúp trẻ em có thể nắm được kiến thức đã học đồng thời bổ sung được những kiến thức mới.

4.2. Hướng dẫn chơi game

❖ Màn hình chính



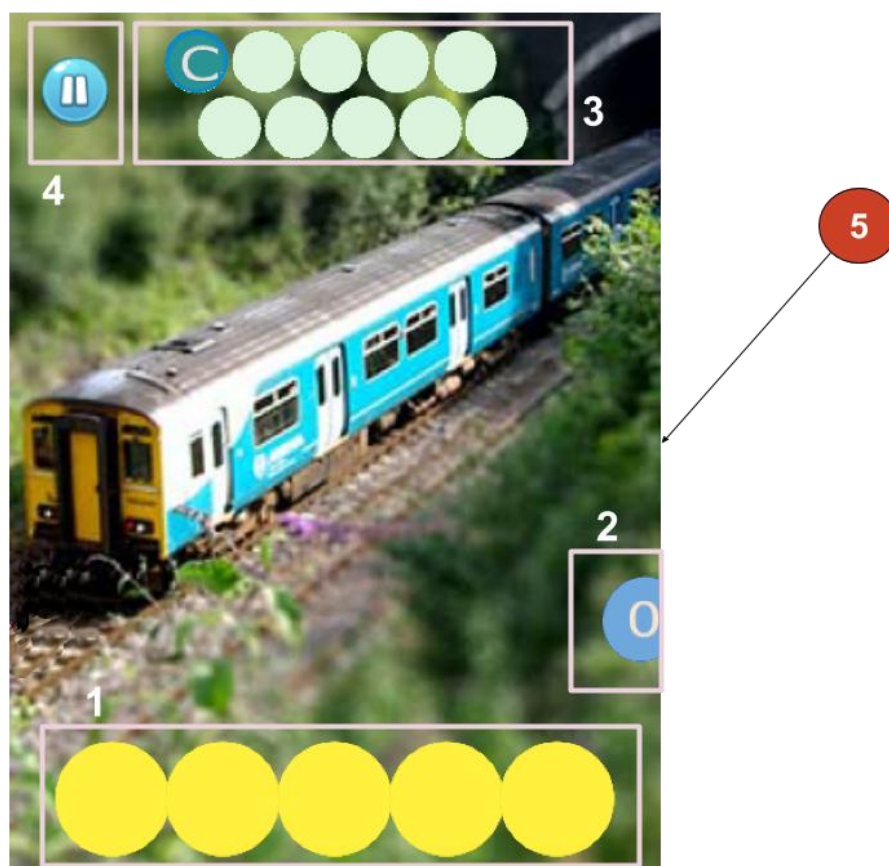
Hình 4.1 Màn hình chính game myWords

Danh sách các thành phần:

STT	Tên	Ý nghĩa
1	Setting	Cấu hình game
2	Mode	Chọn chế độ game

Bảng 4.1 Mô tả thành phần màn hình chính

❖ **Màn hình game play**



Hình 4.2 Các thành phần trong màn hình game play

Danh sách các thành phần:

STT	Tên	Ý nghĩa
1	Answer Word	Hiện từ đoán đúng
2	Score	Hiện thị điểm
3	Wrong Word	Hiện thị từ đoán sai
4	Pause	Nút tạm dừng game
5	Picture	Hình ảnh mô tả từ cần đoán

Bảng 4.2 Mô tả thành phần màn hình game play

❖ **Màn hình tạm dừng game**



Hình 4.3 Các thành phần trong màn hình pause game

Danh sách các thành phần:

STT	Tên	Ý nghĩa
1	Function	Chức năng người dùng
2	Mode	Chế độ chơi

Bảng 4.3 Mô tả thành phần màn hình pause game

❖ **Màn hình thư viện từ**



Hình 4.4 Màn hình thư viện từ

Danh sách các thành phần:

STT	Tên	Ý nghĩa
1	Back	Quay lại màn hình game play
2	Category	Chọn gói từ

Bảng 4.4 Mô tả thành phần màn hình thư viện từ

4.3. Đánh giá người dùng

4.3.1. Số liệu thống kê

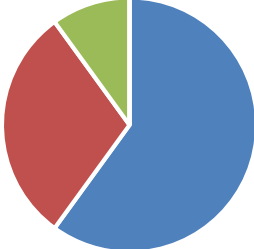
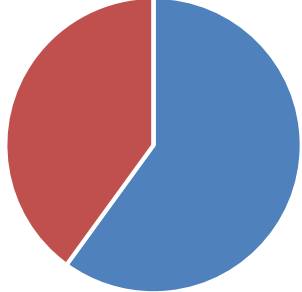
Đối tượng khảo sát: Trẻ em độ tuổi từ 6 – 15

Nơi khảo sát: Trung tâm anh ngữ Thần Đồng - 52 Lê Văn Việt, P. Hiệp Phú, Q.9, TP HCM, nhóm trẻ em tại Đại lộ II, Đỗ Xuân Hợp, Q9, TP HCM.

Số lượng trẻ em tham gia làm đánh giá khảo sát: 10

Các tiêu chí đánh giá:

STT	Tiêu chí đánh giá	Kết quả
1	Hình ảnh trong game	<p>Hình ảnh trong game</p>  <p>■ Đẹp mắt và sinh động ■ Tương đối đẹp ■ Chấp nhận được ■ Hình ảnh chưa đẹp</p>
2	Âm thanh trong game	<p>Âm thanh trong game</p>  <p>■ Vui tai gây hứng thú ■ Bình thường ■ Khó chịu</p>

3	Cảm nhận của trẻ em về game	<p>Cảm nhận về game</p>  <p>■ Rất thoải mái ■ Thoải mái ■ Bình thường ■ Khó chịu ■ Rất khó chịu</p>
4	Sự hứng thú của trẻ em với game	<p>Sự hứng thú với game</p>  <p>■ Rất thích ■ Thích ■ Không thích</p>
5	Cảm nhận riêng của người dùng	Đa số các bạn đều thích chơi game, đánh giá game hấp dẫn và vui nhộn

Bảng 4.5 Số liệu thống kê đánh giá người dùng về game

4.3.2. Phân tích số liệu và đưa ra kết luận

Theo số liệu thống kê được, nhóm đã tiến hành phân tích và đưa ra một số kết luận như sau:

- Nội dung game phù hợp với lứa tuổi, các từ vựng theo đúng chương trình học và hầu như là không quá khó với trẻ.
- Hình ảnh và âm thanh trong game tương đối phù hợp với trẻ.
- Đa phần trẻ rất thích chơi game và muốn chơi game ở các lần tiếp theo.
- Thái độ trẻ lúc chơi game là rất tích cực.

Chương 5. TỔNG KẾT

Trong thời gian thực hiện đề tài, nhóm tác giả đã áp dụng những kiến thức đã học do thầy cô truyền đạt và tích lũy trong nhiều năm qua vào việc xây dựng dự án này. Qua đó, củng cố những kiến thức đã học cũng như tìm hiểu thêm được nhiều kiến thức mới. Từ đó, nhóm tác giả đã hoàn thành được đề tài và nội dung nghiên cứu. Ở chương này, tác giả tổng kết lại những kết quả thực hiện đề tài về những thuận lợi, khó khăn và hướng phát triển trong thời gian tới. Bên cạnh đó, còn những mặt hạn chế chủ quan và khách quan. Tuy nhiên trong thời gian tới nhóm tác giả sẽ tiếp tục phát triển và đưa đề tài nghiên cứu vào áp dụng thực tiễn tại Việt Nam.

5.1. Kết quả đạt được

❖ Về mặt lý thuyết

Qua đề tài này, nhóm tác giả đã củng cố được những kiến thức đã được học tại trường và gần như hiểu sâu các kiến thức chuyên ngành về phát triển game trên thiết bị di động, quy trình phát triển game và nhiều kiến thức liên quan đến chuyên môn. Bên cạnh đó nhóm tác giả còn được tiếp xúc với công nghệ mới về lập trình game trên thiết bị di động:

- **Engine game Unity3D:** Nhóm tác giả nắm tương đối vững engine Unity, nền tảng phát triển và những hỗ trợ của công cụ này. Đây cũng là trong một engine mạnh để phát triển game trong thế giới hiện nay.
- **Công nghệ nhận diện Vuforia:** Qua quá trình tìm hiểu về công nghệ, nhóm tác giả đã nắm vững được cách áp dụng công nghệ này vào để xây dựng và phát triển đề tài.

❖ Về thực tiễn

Khóa luận đã tương đối hoàn thành được các mục tiêu mà đề cương đã đặt ra. Ngoài việc tìm hiểu và áp dụng công nghệ Vuforia vào Unity3D, nhóm tác giả đã hoàn thành được một phần công việc, tạo ra game myWords:

- **Game myWords:** Việc học tập tiếng anh cho trẻ em đã phát triển theo một chiều hướng mới nhờ áp dụng công nghệ vào dạy học cùng với đó là vui chơi giải trí lành mạnh. Ứng dụng game myWords đã mang lại sự hứng thú cho trẻ em, giúp chúng có thể tiếp thu tốt hơn những kiến thức đã học, thu nạp thêm những kiến thức mới vô cùng hiệu quả thông qua các hình ảnh chân thật, sống động. Đó cũng là mong muốn tốt nhất mà nhóm tác giả muốn hướng tới.

Bên cạnh đó nhóm cũng đã nghiên cứu và xây dựng được một bộ phụ trợ để chơi game bao gồm:

- **Bộ ký tự chữ:** Nhóm đã phát triển 2 bộ ký phụ trợ cho việc chơi game, gồm một bộ chữ cho đội xanh và bộ còn lại cho đội đỏ, bộ chữ này được dùng trong game myWords.
- **Bộ ký tự số:** Bộ kí tự số được dùng trong game myNumber
- **Kính đổi chiều camera:** Qua nhiều lần nghiên cứu, thử nghiệm, nhóm đã tạo ra thành công lăng kính có thể đổi hướng camera, góc chiếu phù hợp hơn cho việc chơi game.
- **Giá đỡ thiết bị**

5.2. Khó khăn

❖ Công nghệ:

Khá là mới và không quá phổ biến rộng rãi trong nước. Tài liệu có nhưng không chi tiết. Nên khi gặp vấn đề khó khăn thì nhóm phải tốn thời gian nghiên cứu và thử nghiệm nhiều.

❖ Công cụ hỗ trợ:

Trong game nhóm có sử dụng kính để hỗ trợ việc chơi game nhưng chưa thực hiện được. Hy vọng sẽ hoàn thành được kính trước lúc báo cáo. Và những bộ chữ và số sử dụng để chơi trong game thì phải thiết kế và chỉnh sửa rất nhiều để giúp game nhận diện tốt.

❖ Thiết kế game:

Do chưa có nhiều kinh nghiệm thiết kế game cho trẻ em, nên một số tính năng của game cũng chưa hoàn thiện.

5.3. Hướng phát triển

Trong thời gian tới, nhóm sẽ tiếp tục nghiên cứu và hoàn thiện hơn sản phẩm đã có, bổ sung về nội dung và hình thức chơi.

Mục tiêu hướng tới của nhóm sau này là phát triển thêm nhiều ứng dụng game nữa để tạo nên một kho ứng dụng dành cho trẻ em với mục đích là tạo nên một môi trường giải trí lành mạnh cho trẻ, giúp trẻ phát triển tư duy và tiếp thu được những kiến thức xã hội đã học.

Xa hơn nữa là nhóm muốn phát triển luận văn này thành bộ những sản phẩm thương mại hướng tới cộng đồng những người sử dụng thiết bị thông minh và đặc biệt là đối tượng trẻ em.

TÀI LIỆU THAM KHẢO

- [1] Patrick Maier, Gudrun Klinker, Marcus Tonnies(Technische Universität München, Germany) - Augmented Reality for teaching spatial relations, 2009
- [2] Dr. Javier Fombona Cadavieco, Dra. María Ángeles Pascual Sevillano, Dra. María Filomena Madeira Ferreira Amador - Augmented reality, an evolution of the application of mobile devices, 2012
- [3] Giới thiệu về thực tế ảo tăng cường:
<http://www.stdio.vn/articles/read/237/tong-quan-ve-thuc-tai-tang-cuong-augmented-reality>
- [4] Giới thiệu về vuforia: <https://www.qualcomm.com/products/vuforia>
- [5] Thư viện vuforia: <http://developer.vuforia.com/library/>
- [6] Một số sản phẩm sử dụng thực tế ảo tăng cường: <http://ismar.vgts.org>
- [7] Thực tế ảo tăng cường và ứng dụng:
https://en.wikipedia.org/wiki/Augmented_reality#Applications
- [8] Dạy học lập trình với Unity3D: <http://docs.unity3d.com/Manual/index.html>