

**HỌC VIỆN KỸ THUẬT MẬT MÃ
KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO MÔN
LẬP TRÌNH GAME ANDROID

Đề tài:
NGHIÊN CỨU VÀ PHÁT TRIỂN GAME BRAVE PILOT

Sinh viên thực hiện: Nhóm 16
Bùi Việt Kiều - CT030428
Phạm Trà My - CT030435
Phan Thị Thúy Hằng - CT030417
Tạ Thị Minh Thu - CT030448

Giảng viên hướng dẫn: Bùi Văn Công

Hà Nội, 2022

DANH MỤC HÌNH ẢNH	4
LỜI CẢM ƠN	5
LỜI MỞ ĐẦU	6
CHƯƠNG 1. GIỚI THIỆU TỔNG QUÁT	7
<i>1.1 Khái niệm Game</i>	<i>7</i>
<i>1.2 Tiêu chí của game.....</i>	<i>7</i>
<i>1.3 Các bước để phát triển một game.....</i>	<i>8</i>
<i>1.4 Công cụ lập trình và phát triển Android Studio</i>	<i>11</i>
1.4.1 Giới thiệu chung.....	11
1.4.2 Tính năng	11
1.4.3 Thư viện	12
<i>1.5 Phạm vi đề tài.....</i>	<i>13</i>
<i>Kết luận chương 1.....</i>	<i>13</i>
CHƯƠNG 2. Phân tích thiết kế.....	14
<i>2.1 Ý tưởng game</i>	<i>14</i>
<i>2.2 Phân tích</i>	<i>14</i>
2.2.1 Biểu đồ ca sử dụng.....	14
2.2.2 Đặc tả biểu đồ ca sử dụng	15
2.2.3 Biểu đồ tuần tự.....	18
2.2.4 Biểu đồ hoạt động	22
<i>2.3 Thiết kế giao diện.....</i>	<i>24</i>
CHƯƠNG 3. THỰC NGHIỆM.....	26
<i>3.1 Dữ liệu game.....</i>	<i>26</i>
3.1.1 Thư mục game.....	26
3.1.2 Một số file và thư mục quan trọng.....	26
<i>3.2 Một số Class quan trọng.....</i>	<i>27</i>
<i>3.3 Hướng dẫn chơi.....</i>	<i>29</i>
<i>3.4 Đối tượng người chơi</i>	<i>29</i>

KẾT LUẬN	30
TÀI LIỆU THAM KHẢO.....	31

DANH MỤC HÌNH ẢNH

Hình 1.1 Android Bumblebee.....	11
Hình 2.1 Biểu đồ ca sử dụng tổng quát	15
Hình 2.2 Biểu đồ tuần tự chức năng bắt đầu game.....	18
Hình 2.3 Biểu đồ tuần tự chức năng di chuyển	19
Hình 2.4 Biểu đồ tuần tự chức năng bắn đạn	20
Hình 2.5 Biểu đồ tuần tự chức năng lưu điểm cao	21
Hình 2.6 Biểu đồ hoạt động chức năng bắt đầu game.....	22
Hình 2.7 Biểu đồ hoạt động chức năng di chuyển	23
Hình 2.8 Giao diện màn giới thiệu	24
Hình 2.9 Giao diện màn chơi game.....	24
Hình 2.10 Giao diện màn điểm cao	25
Hình 3.1 Thư mục game	26
Hình 3.2 Class Flight.....	27
Hình 3.3 Class Bird	27
Hình 3.4 Class Bullet.....	28
Hình 3.5 Class GameView.....	28

LỜI CẢM ƠN

Để hoàn thành bài tập lớn môn học Lập trình game này, bên cạnh sự nỗ lực của nhóm em đã vận dụng và tìm tòi học hỏi, với tình cảm chân thành, chúng em xin dành lời cảm ơn chân thành đối với các giáo viên trong trường Học viện Kỹ Thuật Mật Mã, đặc biệt là giáo viên hướng dẫn trực tiếp Thầy Bùi Văn Công. Cảm ơn thầy đã dành nhiều thời gian cũng như công sức và truyền đạt nhiều kiến thức trong việc theo dõi và hướng dẫn em hoàn thành đề tài “Xây dựng game Brave Pilot”.

Bên cạnh đó, nhóm mình cũng gửi lời cảm ơn đến những người bạn đã cùng học tập, gắn bó và giúp đỡ trong suốt quá trình học tập.

Mặc dù đã cố gắng, nỗ lực trong quá trình làm bài tập lớn nhưng do kiến thức còn hạn hẹp nên báo cáo không tránh khỏi những thiếu sót, em rất mong nhận được sự góp ý kiến, chỉ bảo của thầy và các bạn.

Cuối cùng, em xin kính chúc thầy sức khỏe, đạt được nhiều thành công trong sự nghiệp trồng người cũng như trong nghiên cứu khoa học.

Em xin chân thành cảm ơn!

LỜI MỞ ĐẦU

Hiện nay, với sự phát triển mạnh mẽ của internet và những ứng dụng của nó trong đời sống. Máy tính, điện thoại thông minh không còn là một phương tiện lạ lẫm đối với mọi người mà nó đã dần trở thành một công cụ làm việc và giải trí thông dụng và hữu ích của chúng ta ở mọi nơi nhờ có kết nối internet. Do đó việc xây dựng các ứng dụng cho điện thoại di động đang là một ngành công nghiệp mới đầy tiềm năng và hứa hẹn sự phát triển vượt bậc.

Phần mềm, ứng dụng cho điện thoại di động hiện nay rất đa dạng và phong phú trên các hệ điều hành di động. Các hệ điều hành như Android, IOS, Hybrid, ... đã rất phát triển trên thị trường truyền thông di động.

Cùng với sự phát triển mạnh mẽ của lập trình di động thì vô số tựa game mobile đã ra đời để đáp ứng nhu cầu giải trí của con người. Vì những lý do trên mà nhóm chúng em quyết định lựa chọn đề tài “*Xây dựng game Brave Pilot*” là đề tài báo cáo môn học Phát triển game trên Android.

Nội dung báo cáo gồm:

Chương 1: Giới thiệu tổng quát

Chương này giới thiệu tổng quát về lập trình game, các công cụ hỗ trợ và các bước xây dựng một ứng dụng game mobile

Chương 2: Phân tích thiết kế bài toán

Trong chương này, nhóm đã đề cập đến phần phân tích thiết kế cho hệ thống. Trong đó bao gồm đặc tả bài toán, biểu đồ các ca sử dụng, các biểu đồ tuần tự. Giúp thầy cô và các bạn hiểu rõ hơn về bộ khung của chương trình.

Chương 3: Thực nghiệm

Trong chương này, giới thiệu về game Brave Pilot, hướng dẫn trò chơi, kết luận về đề tài,...

CHƯƠNG 1. GIỚI THIỆU TỔNG QUÁT

1.1 Khái niệm Game

Video game hay còn được gọi tắt là game, là một dạng trò chơi điện tử và tương tác với người dùng thông qua một giao diện để hiển thị hình ảnh (video), nó có thể là: màn hình máy vi tính, màn hình tivi, kính thực tế ảo hay cũng có thể là các máy chơi game thông dụng khác có kích cỡ từ chiếc máy tính đồ sộ cho đến những thiết bị nhỏ gọn cầm tay. Trong đó các thiết bị đầu vào sử dụng để thao tác trong game được gọi là game controller hay thiết bị cầm điều khiển và sẽ có sự thay đổi tùy theo hệ máy.

Lấy một ví dụ cụ thể: một thiết bị game controller chuyên dụng có khi chỉ cần có một nút và một cần là có thể điều khiển được tất cả các chức năng trong game, trong khi những loại khác lại có hàng tá các nút bấm khác nhau và có hơn một cần điều khiển.

Bên cạnh các thiết bị tay cầm điều khiển game, thì đối với dòng game cho máy tính thì để có thể chơi được, người dùng bắt buộc phải có thêm sự hỗ trợ từ chuột và bàn phím. Có khá nhiều cách để các video game có thể tương tác với người dùng của mình, trong đó phổ biến vẫn là âm thanh và sử dụng các thiết bị tái tạo âm thanh, nó có thể là tai nghe và loa, ngoài ra nó cũng có những phản hồi khác để báo động cho người chơi thông qua các thiết bị ngoại vi có chức năng cảm ứng, có thể kể đến như: công nghệ tạo chức năng rung phản hồi/cảnh báo cho tay cầm chơi game hay điện thoại di động.

1.2 Tiêu chí của game

a) Luật chơi

Trong mọi trường hợp luật chơi chính là những thảo luận trước đó giữa người dùng và các chủ admin trước khi bắt đầu vào cuộc chơi. Mỗi game đều có luật chơi khác nhau. Mọi thứ nếu hoạt động không nằm trong luật chơi thì cũng đồng nghĩa nó cũng không được công nhận trong game đó. Bởi thế mà luật chơi cũng được hiểu như linh hồn và trái tim của game vậy.

b) Mục tiêu

Mục tiêu trong game sẽ được chia thành 2 mục tiêu chính là điều kiện để người thắng hay yêu cầu và chiến thuật để người dùng thắng game đó. Ví dụ dễ hình dung nhất là : Trong game cờ vua, điều kiện để người chơi dành chiến thắng chính là ăn được quân

vua của đôi phương. Thế nhưng để có thể đạt được điều kiện này thì người chơi thì người chơi phải tính toán ở từng nước đi và mục tiêu của game ở đây chính là chiến thuật, mà người chơi cần phải thực hiện một cách khôn khéo mới có thể dành chiến thắng.

Trong hàng ngàn các game khác nhau, thì chỉ có một số lượng nhỏ mục tiêu game. Nói một cách dễ hiểu hơn thì nó có nghĩa là hầu hết các game hiện nay đều có một mục tiêu như nhau, thường thì ban đầu điều này sẽ khiến nhiều người khá bất ngờ, thế nhưng khi nhìn nhận lại một cách kỹ càng thì mọi game đều phải có kẻ thua ,người thắng và mục tiêu của game luôn phải là có thể đo lường, nhưng vẫn phải tương đối đơn giản và được diễn đạt một cách dễ hiểu trong game.

c) Tiến trình của game ở mỗi lần chơi luôn khác nhau

Đây là một trong những thuộc tính riêng biệt và chỉ có ở game. Có thể những hoạt động khác như: xem phim, đọc báo, nghe nhạc,... có thể được lặp đi lặp lại ở bất cứ thời điểm nào và tiến trình của nó cũng vì thế mà chẳng có gì thay đổi. Thế nhưng với game thì lại hoàn toàn khác, bạn có thể chơi game vô số lần nhưng chắc chắn tiến trình của mỗi lần chơi không bao giờ giống nhau, nó luôn được thay đổi một cách bí ẩn và chẳng ai có thể đoán trước được phần thắng thuộc về ai. Đây chính là một trong những đặc điểm khiến game trở nên thú vị và dễ dàng thu hút người dùng hơn so với những phương tiện giải trí khác.

d) Cạnh tranh

Ở mỗi game, sự cạnh tranh lại có những cách thể hiện khác nhau, có thể là những người chơi phải cạnh tranh trong một game, có người thua và người thắng. Hay thậm chí sự cạnh tranh có thể xuất hiện từ ngay cả với những người cùng trong một team. Trường hợp này, những người chơi sẽ phải cùng đấu tranh chống lại một tình huống đã được định hướng từ trước đó.

1.3 Các bước để phát triển một game

Để phát triển một game cần trải qua 6 giai đoạn: Lên ý tưởng, thiết kế đồ họa, code, bổ xung âm thanh, hoàn thiện và quảng cáo.

a) Lên ý tưởng

- Môi câu: Điều gì làm game trở nên tuyệt vời? Một game kích thích tư duy? Một game kích thích tính sáng tạo? Một game có kết cục bất ngờ ?

- Cách chơi : Người chơi sẽ phải làm gì? Vì mục đích gì? Đây chính là phần gameplay. Người chơi sẽ di chuyển bằng các phím W,A,S,D và bấm phím Space để nhảy như trong game Mario, người chơi thực hiện sẽ có hàng tấn các combo đã mất như Street Fighter...
- Cốt truyện: Người chơi sẽ nhớ về game như thế nào? Những cảm xúc nào người chơi nên có sau khi hoàn thành game? Có thể là một trong những người sống sót ở thảm họa zombie trong The Walking Dead, hoặc một cuộc hành trình giải cứu con trai trên một hòn đảo hoang trong The Forest...
- Cảm xúc: Trò chơi để lại ấn tượng gì? Hình ảnh? Âm thanh? Ấn tượng đầu tiên rất quan trọng. Ấn tượng đầu tiên sẽ thu hút người chơi vào trò chơi.

b) Thiết kế đồ họa

Đồ họa của các đối tượng trong game như: người chơi, đối phương, lâu đài, ống nước... được gọi với cái tên các khác là Sprite. Thuật ngữ Sprite đã có từ rất lâu, tại thời điểm mà máy vi tính và các máy chơi game cần một phần cứng đặc biệt có tốc độ đủ nhanh để có thể xóa và vẽ lại những đối tượng của một game, để khiến đối tượng đó giống như đang chuyển động vậy.

Đồ họa trong game có thể được thiết kế bằng các phần mềm như Photoshop, Sketch, Gimp, Paint, Tiled... hoặc có tải các file đồ họa trực tiếp từ các nguồn khác <https://opengameart.org/> hoặc <https://kenney.nl/> . Những đồ họa tại 2 trang này có thể sử dụng hoàn toàn miễn phí với mục đích thương mại với điều kiện phải ghi tên tác giả vào phần comment trong những đoạn code game.

c) Code

Để có thể phát triển được game thì cần phải có một ngôn ngữ lập trình. Ví dụ: C, C++, C#, Java, Python...

Những phần code cơ bản để tạo nên một game là:

- Import các module.
- Hàm khởi tạo cửa sổ game.
- Hàm load dữ liệu (Các thư mục của game, hình ảnh trong game, âm thanh trong game...).
- Hàm làm mới game.
- Hàm event (Nhận những thông tin từ chuột, bàn phím, tay cầm...).

- Hàm phát hiện và xử lý các va chạm.
- Hàm xử lý âm thanh (Đễ bật / tắt âm thanh trong game).
- Hàm update (Cập nhật các những thay đổi của game).
- Hàm draw (Hiển thị những hình ảnh ra ngoài màn hình).
- Vòng lặp chính của game.

d) Âm thanh

Âm thanh trong game có vai trò rất quan trọng. Vì nó tác động rất nhiều đến cảm xúc của người chơi. Âm thanh trong game chia làm ba loại chính: Nhạc nền hiệu ứng âm thanh và lồng tiếng của nhân vật.

Những bản nhạc nền êm dịu khi người chơi ở vùng an toàn hoặc ở mức đe dọa thấp, rùng rợn và kinh dị khi người chơi ở những nơi vắng vẻ hoặc ở mức đe dọa cao, hào hùng trong các màn đánh boss hoành tráng.

Hiệu ứng âm thanh của gió lùa qua khe cửa, của những bước chân đi qua đi lại dưới nền gỗ, của các cánh cửa kéo kẹt... trong những game thuộc thể loại kinh dị tạo cho người chơi cảm giác hồi hộp và sợ hãi.

Những nhân vật được lồng tiếng bởi những diễn viên lồng tiếng nổi tiếng cũng là một trong những thứ thu hút người chơi. Có rất nhiều người chơi một game vì game đó có nhân vật được lồng tiếng bởi diễn viên lồng tiếng mà họ hâm mộ.

e) Hoàn thiện

Đến bước này thì việc hai việc là sửa lỗi và chỉnh sửa để game trở nên đẹp hơn. Có ba lỗi thường gặp phải trong quá trình lập trình game:

- Lỗi cú pháp: Xảy ra khi gõ sai cú pháp và game sẽ không thể chạy được khi lỗi cú pháp chưa được sửa. Nhưng ngôn ngữ lập trình sẽ chỉ ra lỗi cú pháp xảy ra ở đâu nên việc sửa lỗi cũng không mất quá nhiều thời gian.
- Lỗi trong khi chạy: Xảy ra khi game đã chạy được một lúc. Đến khi gặp lỗi thì game sẽ dừng lại và không chạy tiếp được nữa. Ngôn ngữ lập trình cũng sẽ thông báo ra vị trí xảy ra lỗi nên việc sửa lỗi cũng không mất quá nhiều thời gian.
- Lỗi logic: Đây là lỗi khó sửa nhất vì game sẽ vẫn chạy mà không có thông báo lỗi gì cả. Nhưng game sẽ chạy sai so với mục đích của người lập trình. Ví dụ: Ấn phím mũi tên sang phải thì người chơi lại đi sang trái, bấm phím nhảy

nhưng người chơi lại không nhảy... Người lập trình buộc phải đọc lại toàn bộ code để tìm ra lỗi sai.

Sau khi sửa lỗi xong thì chỉ cần chỉnh sửa lại một vài thứ trong game cho phù hợp cho phù hợp là được. Ví dụ: Hiệu ứng âm chuẩn hơn, tối ưu hóa game...

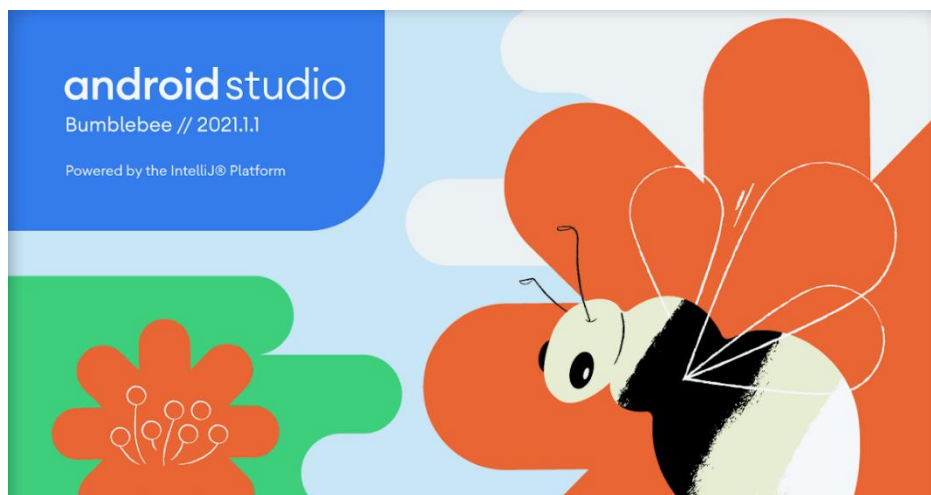
f) Quảng cáo

Một game dù hay đến đâu mà không ai biết tới thì cũng chẳng có tác dụng gì, nên việc quảng cáo game cũng rất quan trọng. Có thể quảng cáo game bằng cách đăng game đó lên những diễn đàn game, hoặc thuê những người nổi tiếng quảng cáo về game...

1.4 Công cụ lập trình và phát triển Android Studio

1.4.1 Giới thiệu chung

Android Studio là môi trường phát triển tích hợp chính thức (IDE) dành cho hệ điều hành Android của Google , được xây dựng trên phần mềm IntelliJ IDEA của JetBrains và được thiết kế đặc biệt để phát triển Android . Nó có sẵn để tải xuống trên các hệ điều hành dựa trên Windows , macOS và Linux hoặc dưới dạng dịch vụ dựa trên đăng ký vào năm 2020. Vào ngày 7 tháng 5 năm 2019, Kotlin đã thay thế Java làm ngôn ngữ ưa thích của Google để phát triển ứng dụng Android. Java vẫn được hỗ trợ, cũng như C++.



Hình 1.1 Android Bumblebee

1.4.2 Tính năng

Một số tính năng sau được cung cấp trong phiên bản ổn định hiện tại:

- Hỗ trợ xây dựng dựa trên Gradle
- Tái cấu trúc dành riêng cho Android và các bản sửa lỗi nhanh chóng
- Các công cụ của Lint để nắm bắt hiệu suất, khả năng sử dụng, khả năng tương thích của phiên bản và các vấn đề khác
- Trình hướng dẫn dựa trên mẫu để tạo các thiết kế và thành phần Android phổ biến
- Trình chỉnh sửa bố cục phong phú cho phép người dùng kéo và thả các thành phần giao diện người dùng, tùy chọn để xem trước bố cục trên nhiều cấu hình màn hình
- Hỗ trợ tích hợp cho Google Cloud Platform, cho phép tích hợp với Firebase Cloud Messaging (“Google Cloud Messaging” trước đây) và Google App Engine
- Sau khi ứng dụng đã được biên dịch bằng Android Studio, ứng dụng đó có thể được xuất bản trên Cửa hàng Google Play. Ứng dụng phải tuân thủ chính sách nội dung dành cho nhà phát triển Cửa hàng Google Play.

1.4.3 Thư viện

Android cung cấp phong phú các thư viện để hỗ trợ lập trình cũng như một nền tảng phát triển ứng dụng vững trãi. Các thư viện này đã đem lại cho Android sức sống mãnh liệt và trên thực tế có vô số các phần mềm chạy trên nền tảng Android.

- util: Gói tiện ích quản lý nhiều lớp ở mức hệ thống như: các lớp (List, Stack...) lớp xử lý chuỗi, lớp xử lý XML,...
- os: Gói hệ điều hành này cung cấp các phương thức cho phép truy cập mức hệ thống như là trao đổi các thông điệp, thông tin chéo, đồng hồ và gỡ lỗi.
- graphics: Gói này cung cấp các lớp đồ họa.
- text: Cung cấp các phương thức xử lý văn bản
- content: Cung cấp phương thức liên quan lập trình nội dung.
- view: Gói khung nhìn người dùng cơ bản nhất. Các chương trình khi tương tác với người dùng đều thông qua các View.
- widget: Gói này được xây dựng dựa trên gói View. Những thành phần giao diện được tạo sẵn đem vào sử dụng để tạo nên giao diện người dùng một cách nhanh chóng.

- app: Gói thư viện này cung cấp phương thức truy cập đến dữ liệu của các ứng dụng.
- provider: Gói thư viện này tạo thuận lợi cho các nhà phát triển truy cập đến các Content Provider tiêu chuẩn cho phép truy cập đến cơ sở dữ liệu chuẩn trong tất cả các bản Android.
- SurfaceView: cung cấp cho lập trình viên 1 mặt phẳng để có thể vẽ (draw) liên tục lên mặt phẳng đó thông qua việc can thiệp tới từng điểm ảnh trên mặt phẳng.
- Canvas: được xem như là một bề mặt (hình dung như tờ giấy, bảng) mà chúng ta có thể vẽ bất cứ thứ gì lên đó. Ví dụ như vẽ một điểm, đường thẳng, hình chữ nhật, đường tròn, elip, văn bản, hay thậm chí là một hình ảnh và các hình ảnh phức tạp khác nữa.
- AudioManager: dùng để điều khiển âm lượng chuông như silent, vibrate, loud
- SoundPool: là một tập hợp các mẫu âm thanh được tải vào bộ nhớ từ resource bên trong file APK hoặc từ hệ thống file, được thiết kế để chơi những file âm thanh ngắn, do đó nó phù hợp cho việc tạo hiệu ứng âm thanh trong game ví dụ đạn nổ, ăn coin, lên level.

1.5 Phạm vi đề tài

Xây dựng một trò chơi hành động phiên lưu trên nền tảng Android bằng Android Studio với các tính năng của game như: di chuyển lên/xuống, bắn để tiêu diệt mục tiêu, các vật cản có thể di chuyển với tốc độ khác nhau, người chơi phải tiêu diệt hết các mục tiêu để có thể đi tiếp, hệ thống lưu điểm cao của người chơi,...

Kết luận chương 1

- Game là một dạng trò chơi điện tử và tương tác với người dùng thông qua một giao diện để hiển thị hình ảnh (video).
- Game gồm 4 tiêu chí: Luật chơi, mục tiêu, tiến trình và cạnh tranh.
- Để phát triển một game cần trải qua 6 giai đoạn: Lên ý tưởng, thiết kế đồ họa, code, bổ xung âm thanh, hoàn thiện và quảng cáo.
- Android Studio là công cụ hữu hiệu trong việc lập trình ứng dụng cũng như game trên hệ điều hành android

CHƯƠNG 2. Phân tích thiết kế

2.1 Ý tưởng game

Người chơi vào vai một anh chàng phi công tham gia chiến tranh, tuy nhiên do bị trúng đạn của địch nên máy bay không thể hoạt động bình thường được nữa, với lượng năng lượng còn lại, người chơi phải cố gắng duy trì, tránh va chạm và trở về căn cứ một cách an toàn. Trên đường đi sẽ gặp những vật cản có thể va vào máy bay làm cho máy bay rơi xuống, nhiệm vụ của người chơi là tiêu diệt các mục tiêu này không cho chúng chạm vào máy bay.

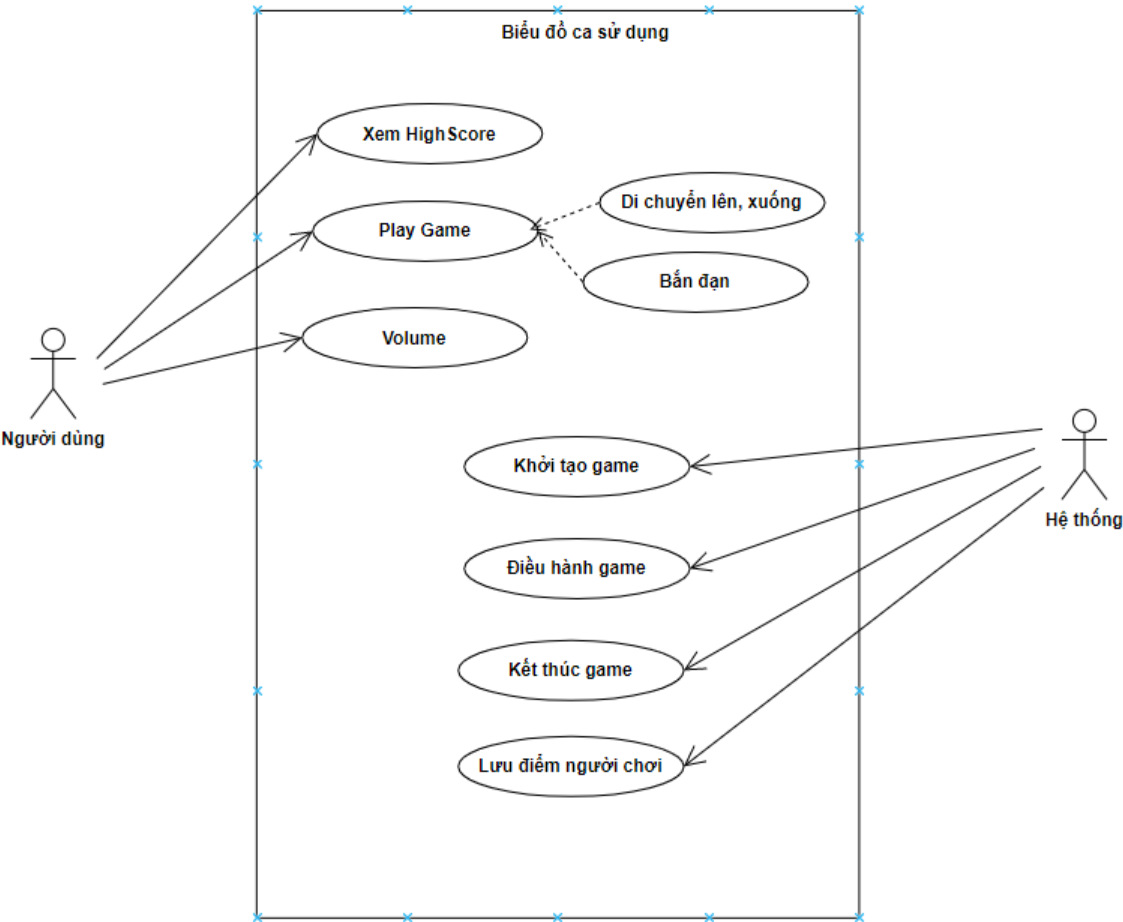
2.2 Phân tích

2.2.1 Biểu đồ ca sử dụng

Xác định Actor và Usecase:

- Hệ thống:
 - Khởi tạo game
 - Điều hành game
 - Kết thúc game
 - Lưu điểm người chơi
- Người chơi:
 - Khởi tạo trò chơi
 - Chơi game
 - Xem điểm cao nhất

Biểu đồ ca sử dụng tổng quát



Hình 2.1 Biểu đồ ca sử dụng tổng quát

2.2.2 Đặc tả biểu đồ ca sử dụng

a) Đặc tả ca sử dụng chơi game

Ca sử dụng	Play game
Tác nhân	Người dùng
Mô tả	Mở màn PlayGame
Luồng sự kiện	1. Người dùng click button Play 2. Mở ra màn Play Game

b) Đặc tả ca sử dụng di chuyển

Ca sử dụng	Di chuyển
Tác nhân	Người dùng
Mô tả	Di chuyển máy bay lên xuống
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng bấm vào nửa trái màn hình => Máy bay di chuyển lên 2. Người dùng không thao tác nửa trái màn hình => Máy bay di chuyển xuống

c) Đặc tả ca sử dụng bắn đạn

Ca sử dụng	Bắn đạn
Tác nhân	Người dùng
Mô tả	Bắn đạn diệt vật cản
Luồng sự kiện	<ol style="list-style-type: none"> 1. Nếu người dùng không bấm vào nửa màn trái: bấm vào nửa màn hình bên phải => Máy bay bắn đạn 2. Nếu người dùng bấm vào nửa màn trái: không click nửa màn hình phải => máy bay không bắn đạn

d) Đặc tả ca sử dụng xem điểm cao

Ca sử dụng	Xem HighScore
Tác nhân	Người dùng
Mô tả	Xem kỷ lục về điểm
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng chưa chơi ván game nào: hiển thị HighScore = 0 2. Người dùng chơi game: Hiển thị điểm cao nhất của trò chơi

e) Đặc tả ca sử dụng khởi tạo game

Ca sử dụng	Khởi tạo game
Tác nhân	Hệ thống
Mô tả	Khởi tạo màn chơi
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng chọn chơi game 2. Hệ thống khởi tạo một màn chơi

f) Đặc tả ca sử dụng điều hành game

Ca sử dụng	Điều hành game
Tác nhân	Hệ thống
Mô tả	Khi người dùng chọn play game, hệ thống sẽ khởi tạo một màn chơi và điều khiển
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng chọn chơi game 2. Hệ thống khởi tạo một màn chơi 3. Hệ thống tạo ngẫu nhiên các vật cản bay về phía người chơi, xử lý các va chạm (giữa đạn và vật cản, giữa người chơi và vật cản).

g) Đặc tả ca sử dụng kết thúc game

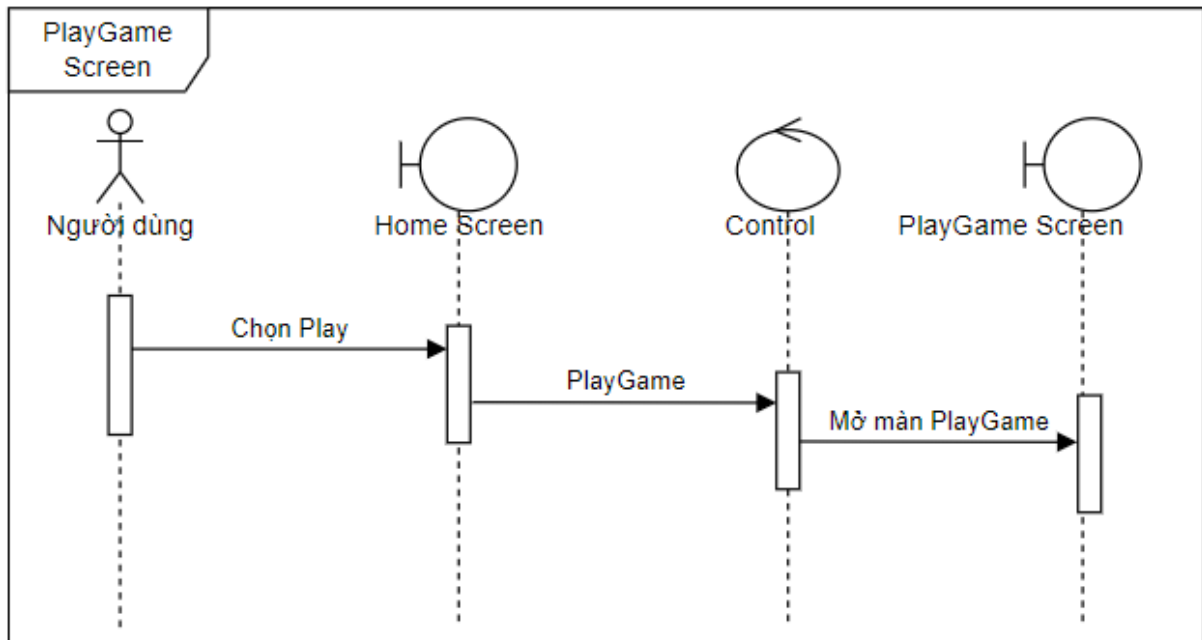
Ca sử dụng	Kết thúc game
Tác nhân	Hệ thống
Mô tả	Khi người dùng va chạm vào vật cản
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng va chạm vào vật cản 2. Hệ thống tính toán điểm và kết thúc màn chơi

h) Đặc tả ca sử dụng lưu điểm cao

Ca sử dụng	Lưu điểm cao
Tác nhân	Hệ thống
Mô tả	Khi người dùng kết thúc màn chơi
Luồng sự kiện	<ol style="list-style-type: none"> 1. Người dùng va chạm vào vật cản 2. Hệ thống sẽ kết thúc màn chơi sau đó tính điểm cho người chơi 3. Nếu điểm người chơi lớn hơn điểm cao mà hệ thống đang lưu thì sẽ thay thế điểm cao bằng điểm của người chơi, ngược lại nếu điểm người chơi nhỏ hơn hoặc bằng điểm cao thì không làm gì.

2.2.3 Biểu đồ tuần tự

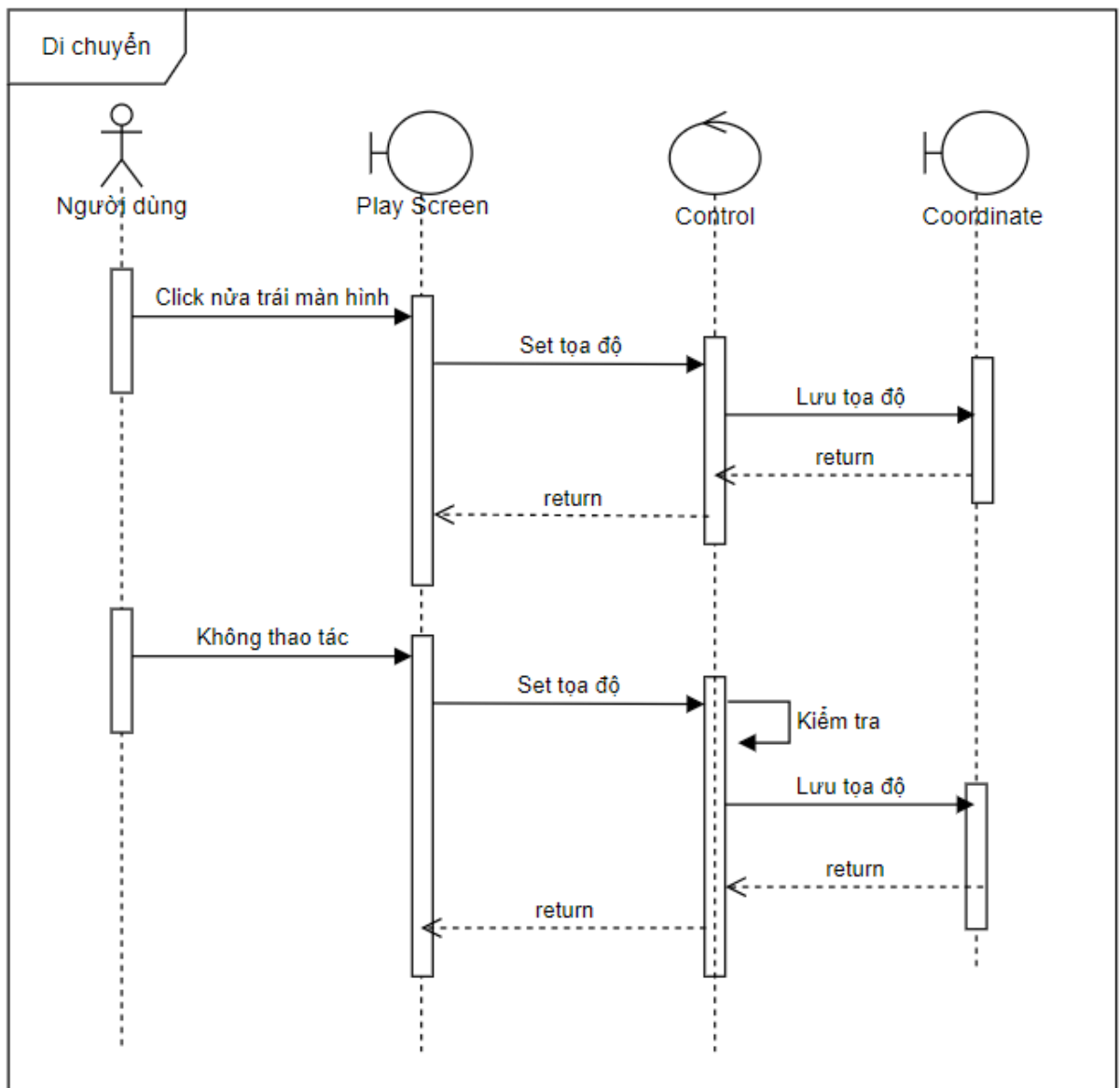
a) Biểu đồ tuần tự chức năng bắt đầu game



Hình 2.2 Biểu đồ tuần tự chức năng bắt đầu game

Khi người dùng chọn Play, sẽ gửi yêu cầu play đến hệ thống, hệ thống xử lý và mở màn chơi game.

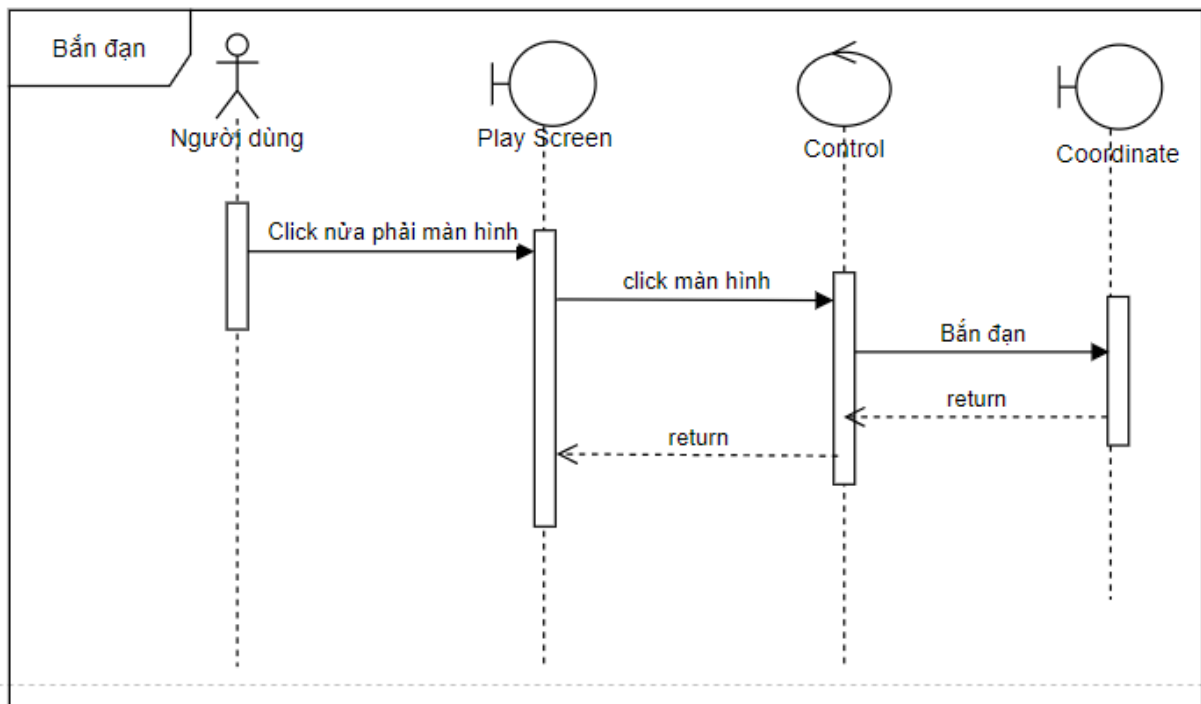
b) Biểu đồ tuần tự chức năng di chuyển



Hình 2.3 Biểu đồ tuần tự chức năng di chuyển

Khi người chơi vào một màn chơi, người chơi sẽ điều khiển máy bay di chuyển bằng cách bấm vào nửa trái màn hình, khi đó hệ thống sẽ khi nhận thao tác và đặt tọa độ máy bay tương ứng với chức năng đi lên rồi lưu tọa độ đó để hiển thị ra giao diện cho người chơi. Ngược lại nếu người chơi không thao tác thì máy bay sẽ di chuyển xuống, tương tự hệ thống cũng sẽ đặt tọa độ máy bay và hiển thị ra giao diện.

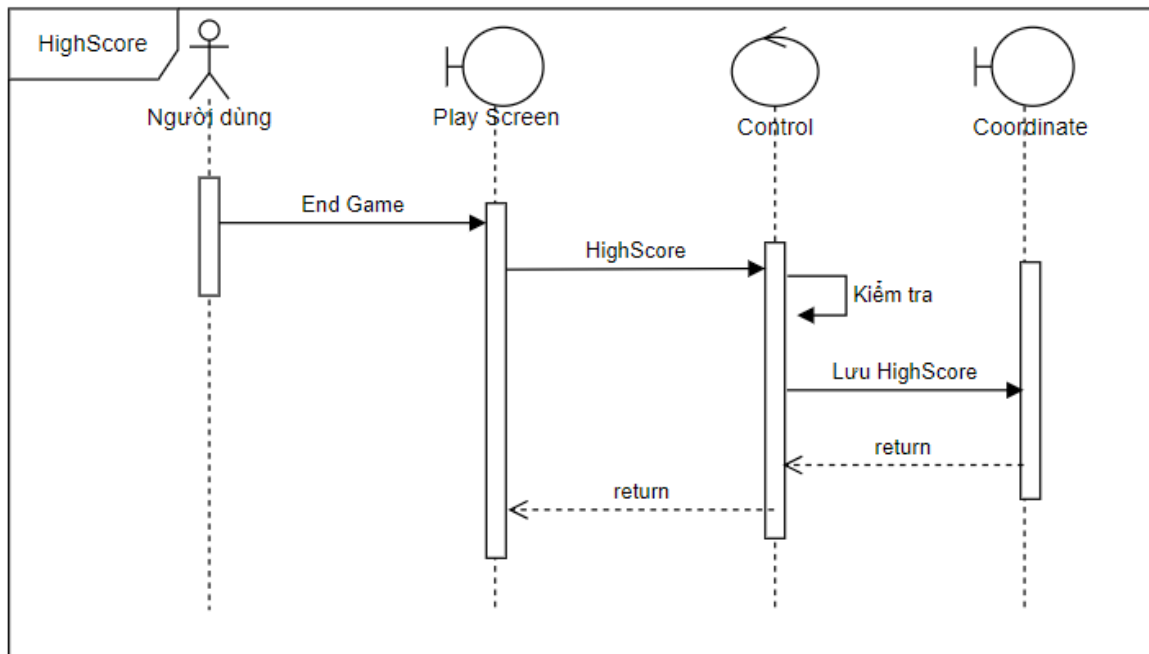
c) Biểu đồ tuần tự chức năng bắn đạn



Hình 2.4 Biểu đồ tuần tự chức năng bắn đạn

Khi người dùng bấm vào nửa bên phải màn hình, hệ thống sẽ xử lý yêu cầu và bắn ra 1 viên đạn, rồi liên tục cập nhật tọa độ của viên đạn đó theo phương nằm ngang cho đến khi đi hết màn hình thì sẽ remove.

d) Biểu đồ tuần tự chức năng lưu điểm cao

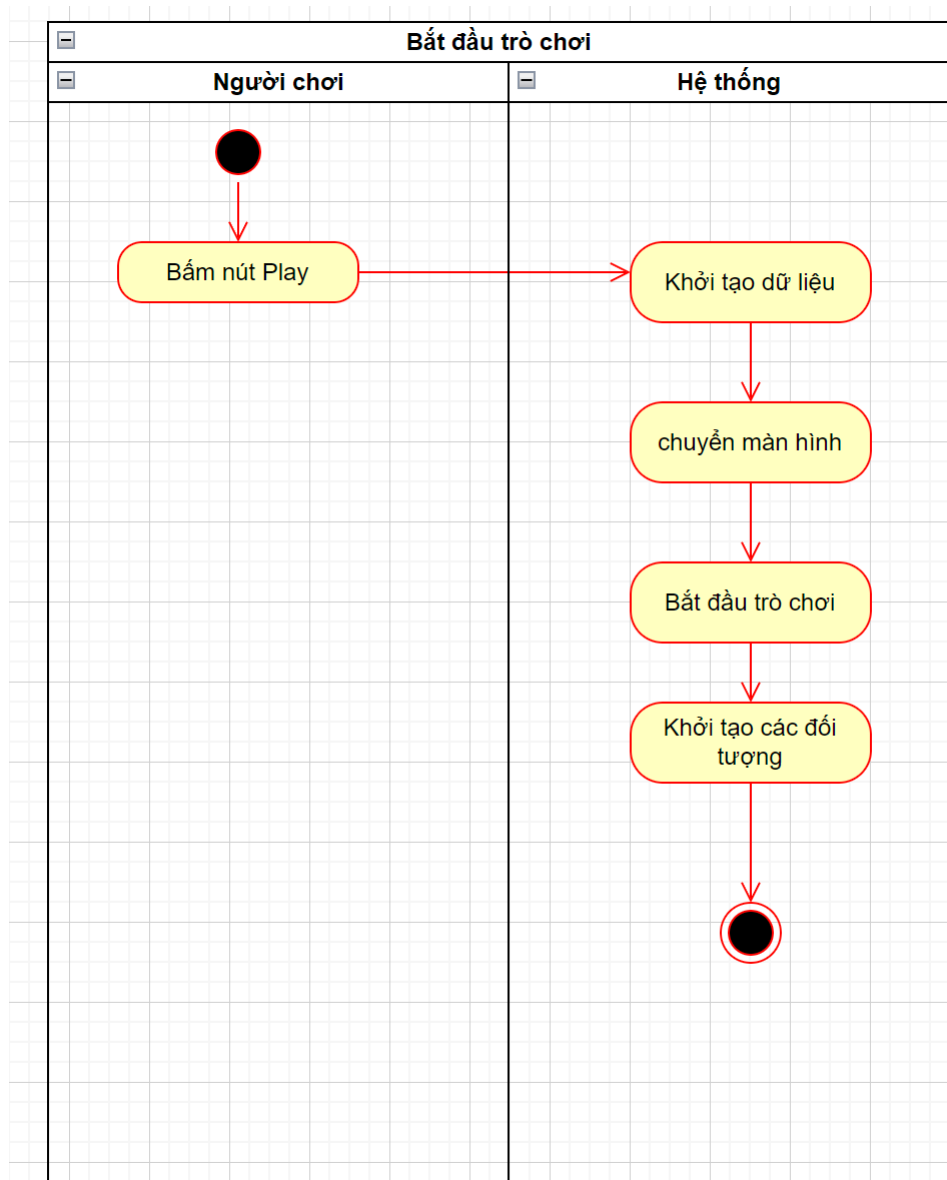


Hình 2.5 Biểu đồ tuần tự chức năng lưu điểm cao

Khi người chơi kết thúc một màn, hệ thống sẽ tính toán điểm cho người chơi, sau đó kiểm tra so sánh với điểm cao hiện tại, nếu điểm người chơi cao hơn điểm cao hiện tại thì sẽ lưu lại, ngược lại thì giữ nguyên điểm cao hiện tại.

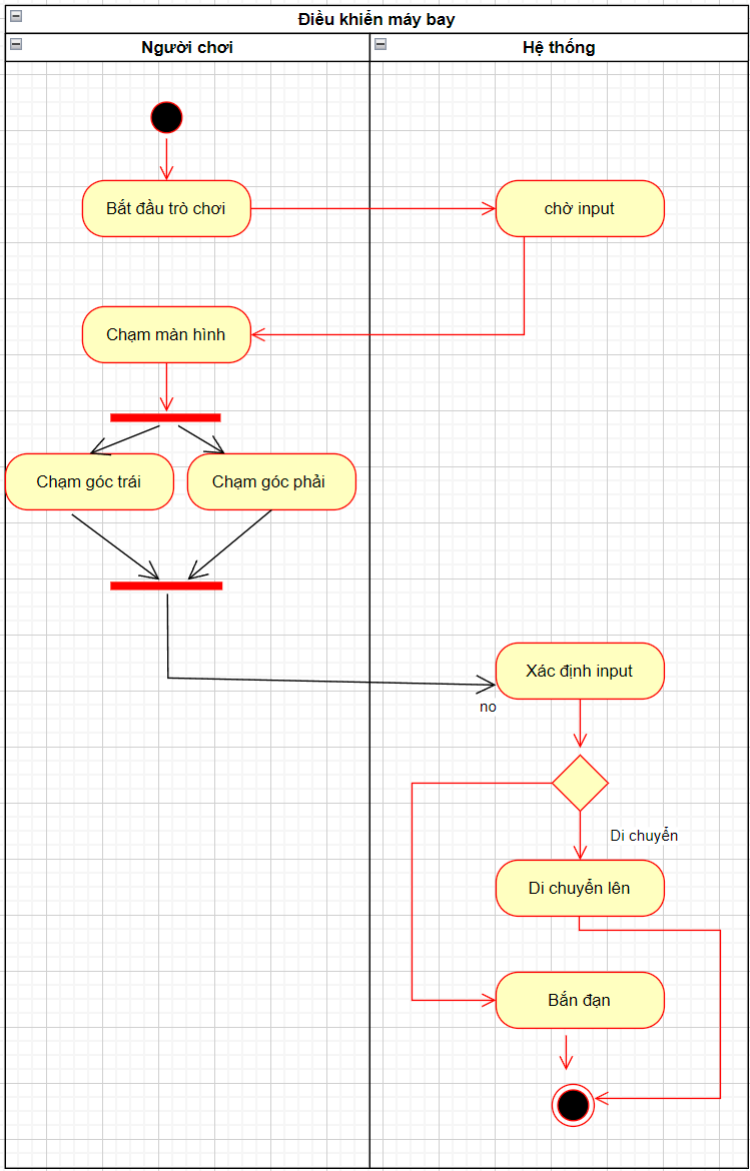
2.2.4 Biểu đồ hoạt động

Biểu đồ hoạt động chức năng bắt đầu game



Hình 2.6 Biểu đồ hoạt động chức năng bắt đầu game

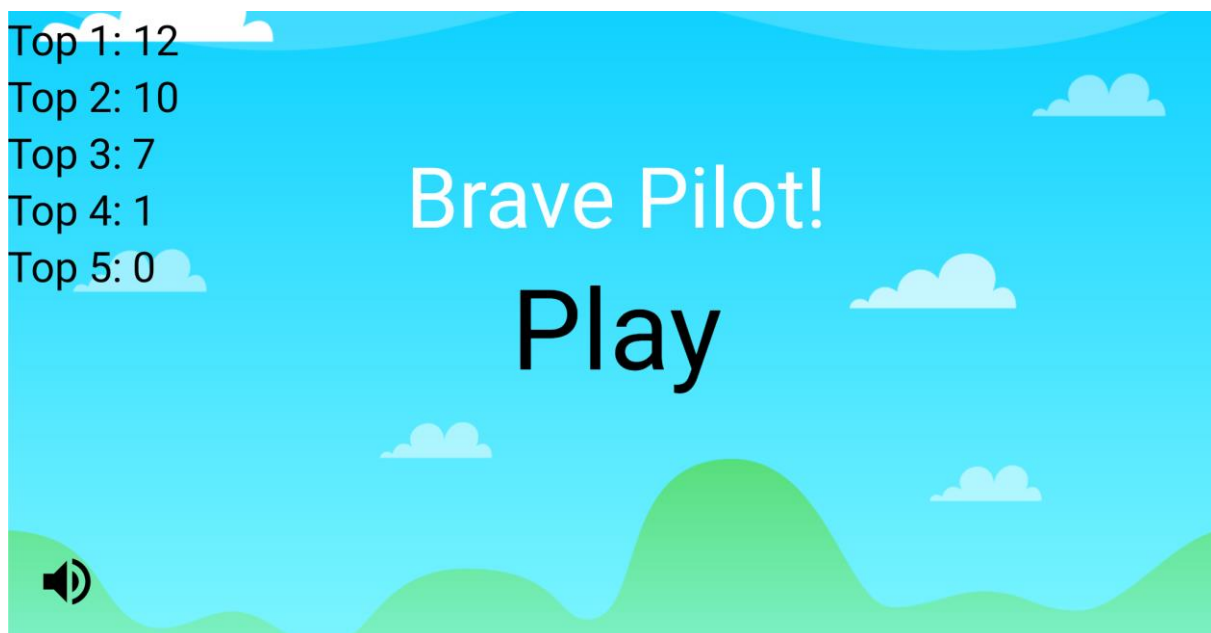
Biểu đồ hoạt động chức năng điều khiển



Hình 2.7 Biểu đồ hoạt động chức năng di chuyển

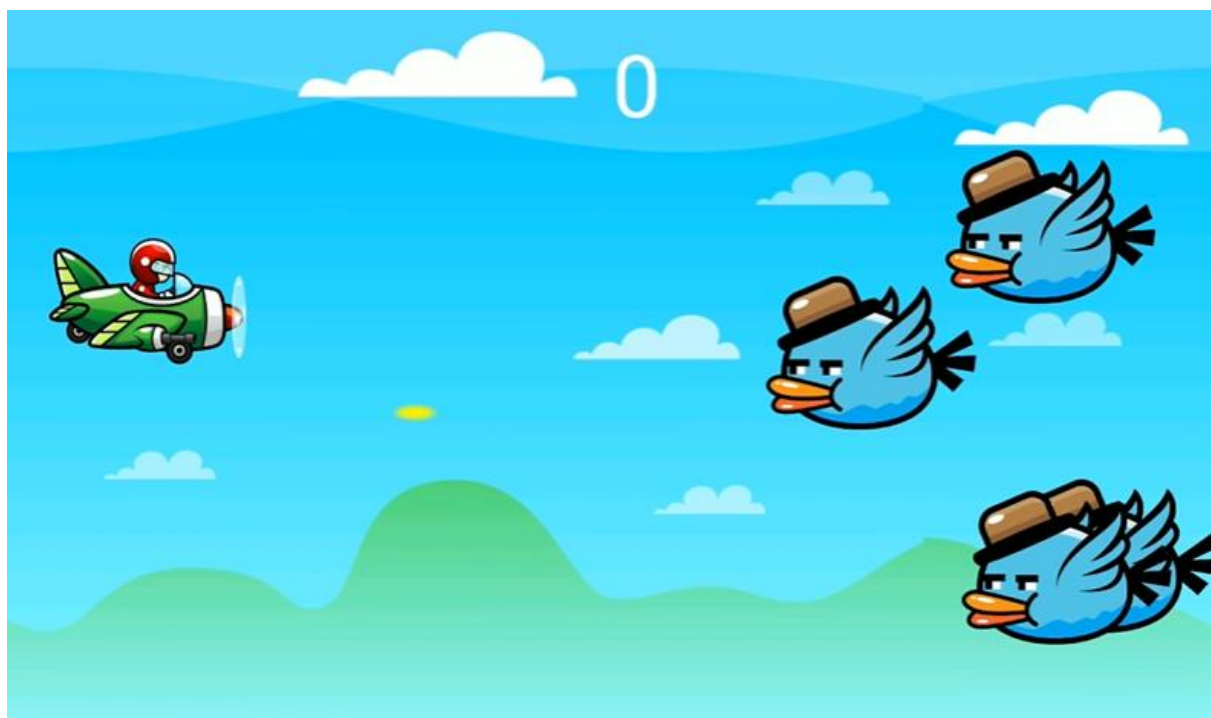
2.3 Thiết kế giao diện

a) Giao diện màn giới thiệu



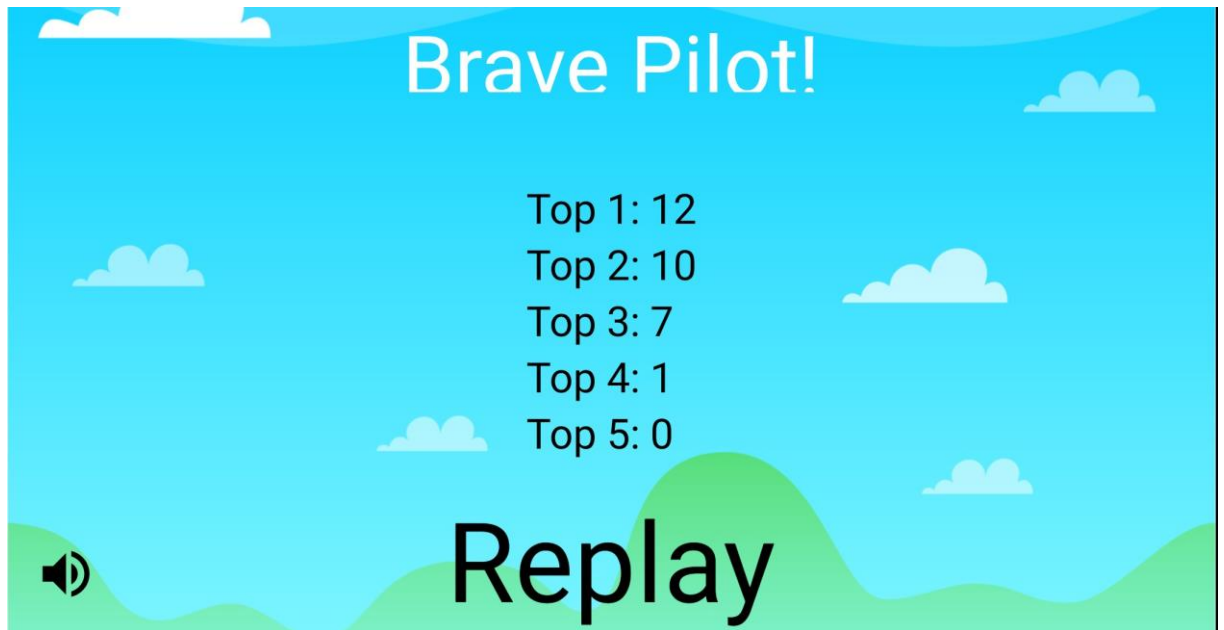
Hình 2.8 Giao diện màn giới thiệu

b) Giao diện màn chơi game



Hình 2.9 Giao diện màn chơi game

c) Giao diện màn điểm cao

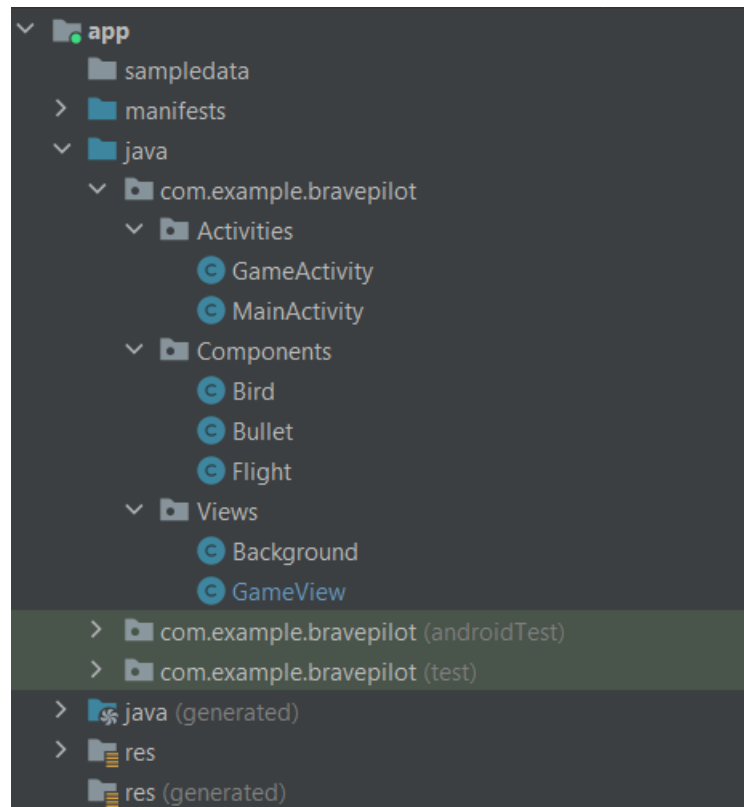


Hình 2.10 Giao diện màn điểm cao

CHƯƠNG 3. THỰC NGHIỆM

3.1 Dữ liệu game

3.1.1 Thư mục game



Hình 3.1 Thư mục game

3.1.2 Một số file và thư mục quan trọng

a) File GameView.java

- Khởi tạo màn chơi
- Nhận và xử lý dữ liệu từ các thao tác
- Chứa vòng lặp chính của game
- Liên kết các phần tử của game

b) Thư mục Components

- Chứa các component của game: Flight, Bird, Bullet
- Các thuộc tính của các component

c) Thư mục Activities

- Chứa các Activity của game: MainActivity, GameActivity
- Điều hướng khi khởi tạo game và khi kết thúc game

3.2 Một số Class quan trọng

a) Class Flight (Người chơi)

```
public class Flight {
    public int toShoot = 0;
    public boolean isGoingUp = false;
    public int x;
    public int y;
    public int width;
    public int height;
    public int wingCounter = 0;
    public int shootCounter = 1;
    Bitmap flight1, flight2, shoot1, shoot2, shoot3, shoot4, shoot5, dead;
    private GameView gameView;

    public Flight(GameView gameView, int screenY, Resources res) {...}

    public Bitmap getFlight() {...}

    public Rect getCollisionShape() { return new Rect(x, y, right: x + width, bottom: y + height); }
    public Bitmap getDead() { return dead; }
}
```

Hình 3.2 Class Flight

b) Class Bird

```
public class Bird {
    public int speed = 20;
    public boolean wasShot = true;
    public int x = 0;
    public int y;
    public int width;
    public int height;
    public int birdCounter = 1;
    public Bitmap bird1, bird2, bird3, bird4;

    public Bird(Resources res) {...}

    public Bitmap getBird() {...}

    public Rect getCollisionShape() { return new Rect(x, y, right: x + width, bottom: y + height); }
}
```

Hình 3.3 Class Bird

c) Class Bullet

```
public class Bullet {
    public int x;
    public int y;
    public int width;
    public int height;
    public Bitmap bullet;

    public Bullet(Resources res) {...}

    public Rect getCollisionShape() { return new Rect(x, y, right: x + width, bottom: y + height); }
}
```

Hình 3.4 Class Bullet

d) Class GameView

```
public class GameView extends SurfaceView implements Runnable {
    private Thread thread;
    private boolean isPlaying, isGameOver = false;
    private int screenX, screenY, score = 0;
    public static float screenRatioX, screenRatioY;
    private Paint paint;
    private Bird[] birds;
    private SharedPreferences prefs;
    private Random random;
    private SoundPool soundPool;
    private List<Bullet> bullets;
    private int sound;
    private Flight flight;
    private GameActivity activity;
    private Background background1, background2;
    private int mActivePointerId;

    public GameView(GameActivity gameActivity, int screenX, int screenY) {...}

    @Override
    public void run() {...}
    private void update() {...}
    private void draw() {...}
    private void waitBeforeExiting() {...}
    private void saveIfHighScore() {...}
    private void sleep() {...}
    public void resume() {...}
    public void pause() {...}

    @Override
    public boolean onTouchEvent(MotionEvent event) {...}
    public void newBullet() {...}
}
```

Hình 3.5 Class GameView

3.3 Hướng dẫn chơi

Khi mở ứng dụng người dùng có thể

- ON/OFF âm thanh trò chơi
- Xem HighScore(kỷ lục về điểm) của trò chơi

Đầu tiên, Chọn button “Play” để vào game: người chơi sẽ sở hữu một máy bay, có nhiệm vụ dọn dẹp các vật cản khác bằng cách tiêu diệt các chúng bằng lượng đạn của mình. Trong khi làm nhiệm vụ, chúng ta chỉ có 1 mạng để sẵn sàng đối mặt với những con quái vật cứng đầu sắp tới. Người chơi cần tiêu diệt vật cản và tránh va vào vật cản. Tiêu diệt được càng nhiều vật cản thì điểm của người chơi càng cao.

Khi vào màn PlayGame:

- Người chơi bấm vào nửa phải màn hình để di chuyển lên xuống.
 - Khi bấm: di chuyển lên.
 - Khi không bấm: di chuyển xuống.
- Người chơi tab nửa trái màn hình để bắn đạn.
- Không thể vừa di chuyển vừa bắn đạn.

Khi máy bay chạm vật cản trò chơi sẽ kết thúc trở lại màn Play và số điểm của trò chơi sẽ được lưu lại nếu > HighScore trước đó.

Người dùng có thể xem được điểm số của mình.

3.4 Đối tượng người chơi

- Những người muốn chơi game để giải trí sau những giờ làm việc và học tập căng thẳng.
- Khi đường truyền mạng đang gặp trục trặc thì có thể chơi game để đợi đường truyền mạng được khắc phục.
- Chơi với bạn bè và đồng nghiệp để giải trí.

KẾT LUẬN

Sau quá trình phân tích và xây dựng game Brave Pilot thì hiện tại sản phẩm của chúng em đã hoàn thiện được một game Android đáp ứng mục tiêu đặt ra ban đầu. Game đáp ứng được nhu cầu giải trí của người chơi, kích thích người chơi để vượt qua mức điểm kỉ lục.

Tuy nhiên vẫn gặp một số khó khăn

- Khó khăn khi chơi game: .
 - Chỉ có 1 mạng, dễ chết.
 - Tốc độ game cao, yêu cầu người chơi phải có phản xạ nhanh nhẹn.
- Khó khăn khi phát triển:
 - Ít kinh nghiệm chuyên môn.
 - Ít kinh nghiệm thiết kế game.
 - Thiếu thiết bị để test trên nhiều loại máy khác nhau.
- Hướng phát triển
 - Phát triển tính năng đa người chơi.
 - Phát triển nhiều Items trong game hơn.
 - Phát triển đồ hoạ đẹp hơn.
 - Làm mịn, làm mượt .
 - Tương thích nhiều cấu hình.
 - Cho lựa chọn độ khó của game

TÀI LIỆU THAM KHẢO

- [1] <https://openplanning.net/10521/huong-dan-lap-trinh-android-game-2d-cho-nguoi-moi-bat-dau>
- [2] <https://khoapham.vn/khoa-hoc-lap-trinh-android-tot-nhat.html>

Hà nội, ngày 20 tháng 12 năm 2022

XÁC NHẬN CỦA GIẢNG VIÊN HƯỚNG DẪN

Bùi Văn Công