**ỦY BAN NHÂN DÂN TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue circle with text

Description automatically generated

**BÁO CÁO ĐỒ ÁN**

**MÔN HỌC: LẬP TRÌNH PYTHON**

**ĐỀ BÀI: LẬP TRÌNH GAME BẮN SÚNG BẰNG THƯ VIỆN PYGAME**

**Giảng Viên Hướng Dẫn: ThS.Trương Tấn Khoa**

**Nhóm 3:**

**Trịnh Đình Hiệp - 3122410111**

**Thái Minh Khang - 3122410174**

**Lê Huỳnh Trúc Vy - 3122410489**

**Nguyễn Việt Hoàng - 3122560022**

**Huỳnh Ngọc Tuấn - 3122410446**

**TP. HỒ CHÍ MINH, THÁNG 5 NĂM 2024**

**Lời mở đầu**

Trong thế giới lập trình và game, việc tạo ra các trò chơi đã trở thành một phần không thể thiếu của hành trình khám phá và ứng dụng kiến thức. Đồ án này tập trung vào việc phát triển một trong những trò chơi khám phá, phiêu lưu, “Bắn Súng” - một trò chơi hiện đại với cách chơi đơn giản

Mục tiêu chính của đồ án là tạo ra một phiên bản game “Bắn Súng” sử dụng ngôn ngữ lập trình Python. Bên cạnh việc giúp người chơi thư giãn và giải trí, dự án cũng đặt ra mục tiêu phát triển kỹ năng lập trình cũng như hiểu rõ hơn về cách thức xây dựng một game đơn giản nhưng hấp dẫn.

Trò chơi “Bắn Súng” là một trong những tựa game nhập vai quen thuộc. Trong trò chơi đầy hành động này, bạn sẽ được đưa vào một chiến trường hai chiều có nhịp độ nhanh, nơi kỹ năng bắn súng của bạn sẽ được thử thách tối đa. Là một anh hùng dũng cảm, bạn sẽ điều hướng qua một môi trường năng động với đầy kẻ thù rình rập khắp mọi ngóc ngách. Hãy sống sót qua làn sóng đối thủ không ngừng nghỉ trong khi tích lũy điểm và leo lên hàng ngũ thiện xạ ưu tú. Nhưng hãy cẩn thận, nguy hiểm luôn rình rập ở mọi ngã rẽ. Bạn sẽ cần phản xạ nhanh như chớp và độ chính xác chính xác để vượt qua kẻ thù và giành chiến thắng.

Thông qua đồ án này, chúng ta sẽ tìm hiểu cách xây dựng một game nói chung và game “Bắn Súng” nói riêng bằng ngôn ngữ lập trình Python. Báo cáo sẽ trình bày quá trình phát triển, cũng như các kỹ thuật và công nghệ được sử dụng để hoàn thiện sản phẩm.

Hy vọng rằng dự án sẽ mang lại sự thú vị, hào hứng, vui vẻ đến cho mọi người.

**Lời cảm ơn**

Chúng tôi muốn dành một chút thời gian để bày tỏ lòng biết ơn sâu sắc nhất đến Giảng viên, ThS. Trương Tấn Khoa cũng như tất cả các thành viên trong nhóm đã tận tình, chăm chỉ thực hiện đồ án. Tiếp theo chúng tôi muốn gửi lời cảm ơn đến các bạn, đã đọc và l. Sự hỗ trợ và cống hiến không ngừng nghỉ của thầy và các bạn là động lực thúc đẩy sự thành công của trò chơi của chúng tôi và chúng tôi thực sự rất cảm kích trước sự nhiệt tình của ban.

Chúng tôi vô cùng háo hứng về tương lai của trò chơi và những cuộc phiêu lưu phía trước.. Cùng nhau, hãy tiếp tục tạo nên những kỷ niệm, chinh phục thử thách và ăn mừng chiến thắng trong thế giới trò chơi bắn súng 2D của chúng tôi. Phản hồi của bạn đối với trò chơi là một đóng góp có vai trò quan trọng trong việc định hình sự phát triển của trò chơi và cải thiện trải nghiệm chơi trò chơi cho mọi người.

Cảm ơn các bạn, từ tận đáy lòng, vì sự hỗ trợ, cống hiến và nhiệt tình không ngừng nghỉ của bạn. Và cuối cùng hãy tận hưởng thành quả cùng nhau !

DANH MỤC HÌNH ẢNH

[Hình 2.1: Hộp thoại system properties. 11](#_Toc166158047)

[Hình 2.2: Hộp thoại Environment Variables. 12](#_Toc166158048)

[Hình 2.3: Hộp thoại Edit Environment Variables. 13](#_Toc166158049)

[Hình 2.4: Hộp thoại Run. 14](#_Toc166158050)

[Hình 2.5: Trình biên dịch Python. 14](#_Toc166158051)

[Hình 2.6: Cript python bằng Visual Studio Code. 15](#_Toc166158052)

[Hình 2.7: Hộp thoại Run mở cmd. 16](#_Toc166158053)

[Hình 2.8: Hộp thoại cmd cài đặt pygame. 16](#_Toc166158054)

[Hình 2.9: Kiểm tra đã cài đặt thư viện pygame. 16](#_Toc166158055)

[Hình 3.1: Flowchart Diagram. 18](#_Toc166158056)

[Hình 3.2: Classdiagram 27](#_Toc166158057)

[Hình 3.3: Giao diện khi vừa khởi chạy game. 28](#_Toc166158058)

[Hình 3.4: Giao diện game play. 29](#_Toc166158059)

[Hình 3.5: Giao diện game over. 30](#_Toc166158060)

MỤC LỤC

**[CHƯƠNG I: TỒNG QUAN](#_Toc166158067)** [6](#_Toc166158067)

[I.Mục tiêu và phạm vi. 6](#_Toc166158068)

[1.1. Mục tiêu. 6](#_Toc166158069)

[2.2. Phạm vi. 6](#_Toc166158070)

[II. Mô tả về game. 7](#_Toc166158071)

[2.1. Giới thiệu. 7](#_Toc166158072)

[2.2 Gameplay. 7](#_Toc166158073)

[2.3. Âm thanh và đồ họa. 8](#_Toc166158074)

**[CHƯƠNG II: GIỚI THIỆU NGÔN NGỮ PYTHON VÀ THƯ VIỆN PYGAME](#_Toc166158075)** [9](#_Toc166158075)

[I.Giới thiệu. 9](#_Toc166158076)

[1.1. Ngôn ngữ python. 9](#_Toc166158077)

[1.2. Thư viện pygame. 10](#_Toc166158078)

[II.Hướng dẫn cài đặt python và thư viện pygame. 10](#_Toc166158079)

[2.1. Cài đặt python. 10](#_Toc166158080)

**[CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ](#_Toc166158081)** [17](#_Toc166158081)

[I.PHÂN TÍCH VÀ THIẾT KẾ. 17](#_Toc166158082)

[1.1. Phân tích hệ thống. 17](#_Toc166158083)

[1.2. Thiết kế hệ thống. 18](#_Toc166158084)

[1.3. Tổng kết. 27](#_Toc166158085)

[II.Xây dựng game. 28](#_Toc166158086)

**[CHƯƠNG IV: KẾT LUẬN.](#_Toc166158087)** [31](#_Toc166158087)

[I. Các vấn đề đã thực hiện được trong đề tài. 31](#_Toc166158088)

[II.Hướng phát triển. 31](#_Toc166158089)

# **CHƯƠNG I: TỒNG QUAN**

## I.Mục tiêu và phạm vi.

### 1.1. Mục tiêu.

**-**Tìm hiểu về công nghệ:

+Tìm hiểu về ngôn ngữ Python cơ bản.

+Tìm hiểu về Pygame.

+Tìm hiểu về lập trình game bằng Python.

+Cài đặt và hướng dẫn.

-Xây dựng game bắn súng đồ họa 2D:

+Sử dụng thư viện Pygame trong Python lập trình game.

### 2.2. Phạm vi.

Xây dựng game bắn súng 2D (Shoot) offline bằng ngôn ngữ Python trên máy tính.

## II. Mô tả về game.

### 2.1. Giới thiệu.

“Bắn súng” là một trò chơi hiện đại và đơn giản, được thiết kế để giải trí và của người chơi. Trong trò chơi này, bạn sẽ điều khiển một nhân vật di chuyển trong một bản đồ không gian hai chiều, cố gắng tiêu diệt kẻ địch và vượt ải.

### 2.2 Gameplay.

#### a) Mục tiêu.

Người chơi cần điều kiển nhân vật bằng các nút A, D , W, SPACE, Q để di chuyển qua các địa hình, chướng ngại vật, bắn súng và ném boom kẻ thù. Cố gắng vượt qua các mản để tiếp tục những thử thách tiếp theo.

#### b) Menu chính

Gồm 2 chế độ: START, EXIT.

+Khi chọn chế độ PLAY trò chơi sẽ bắt đầu và màn hình chơi sẽ được hiển thị.

+ Khi chọn chế độ EXIT, người chơi có thể thoát khỏi trò chơi (hoặc bấm nút X bên phải góc trên màn hình).

#### c) Cách chơi.

- Di chuyển: sử dụng các nút: A qua trái, D qua phải, W nhảy.

- Tấn công: SPACE bắn súng, Q ném boom. (lưu số đạn và số boom có giới hạn)

- Vật phẩm: trên đường đi sẽ có những túi vật phẩm như máu, đạn dược và boom, lấy chúng để bổ sung những thứ cần thiết.

- Tránh và né: cần tránh những vũng nước, các hố, vực thẩm, và đạn từ kẻ địch, nếu bị kẻ địch bắn sẽ mất máu và rơi xuống sẽ mất mạng.

#### d) Các ải.

- Điểm số sẽ được tính bằng các ải vượt qua.

#### e) Kết thúc game.

-Trò chơi kết thúc khi bạn hết máu, có thể do đi vào nước, vựa thẩm hoặc bị kẻ địch bắn.

- Màn hình chính sẽ cho bạn 2 lựa chọn, hoặc là bắt đầu lại cuộc phiêu lưu hoặc thoát khỏi trò chơi.

### 2.3. Âm thanh và đồ họa.

#### a) Âm thanh.

- Âm thanh nền game mang hướng hành động, tiếng đạn, tiếng nổ..

#### b) Đồ họa.

-Giao diện bắt đầu với màu xanh và kết thúc với màu đỏ.

-Giao diện trò chơi có: Nền vùng núi thiên nhiên, các địa hình là đất, bãi cỏ, nước và các thùng trang trí.

-Giao diện nhân vật và kẻ địch: nhân vật có màu xanh, cầm súng và kẻ địch có màu đỏ.

-Giao diện các hòm vật phẩm: máu-hòm màu vàng có hình chữ thập đỏ ở giữa, hòm boom-màu vàng có hình lựu đạn ở giữa, hòm đạn-màu vàng có hình viên đạn ở giữa.

# **CHƯƠNG II: GIỚI THIỆU NGÔN NGỮ PYTHON VÀ THƯ VIỆN PYGAME**

## I.Giới thiệu.

### 1.1. Ngôn ngữ python.

#### a) Python là gì?

Python là một ngôn ngữ lập trình thông dịch, đa mục đích và mã nguồn mở. Nó được thiết kế với ưu tiên đọc dễ hiểu và cú pháp gọn nhẹ, giúp người lập trình tập trung vào giải quyết vấn đề thay vì mất thời gian vào cú pháp phức tạp.

#### b) Lịch sử phát triển.

1989-1991: Python được tạo ra bởi Guido van Rossum tại CWI (Centrum Wiskunde & Informatica) ở Hà Lan. Phiên bản đầu tiên, Python 0.9.0, được phát hành vào tháng 2 năm 1991.

1994: Python 1.0 ra mắt với nhiều tính năng mới như lambda, map, filter, và reduce.

2000: Python 2.0 giới thiệu Garbage Collection và hỗ trợ Unicode. Python ngày càng trở nên phổ biến, đặc biệt trong cộng đồng lập trình web.

2008: Python 3.0 (hay Python 3000 hoặc “Py3k”) được phát hành, với sự thay đổi lớn trong cú pháp và thiết kế để cải thiện sự đơn giản, đồng thời giữ lại sự tương thích ngược với Python 2.

Ngày nay: Python đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất thế giới, được sử dụng rộng rãi trong nhiều lĩnh vực như phân tích dữ liệu, trí tuệ nhân tạo, phát triển web, và hệ thống.

#### c) Đặc điểm nổi bật của python.

Dễ đọc và dễ hiểu: Cú pháp của Python tập trung vào sự đơn giản và rõ ràng, giúp người lập trình tập trung vào giải quyết vấn đề thay vì cú pháp phức tạp.

Mã nguồn mở: Python có giấy phép mã nguồn mở, khuyến khích sự hợp tác và đóng góp từ cộng đồng.

Đa mục đích: Python có thể sử dụng cho mọi thứ từ lập trình web, phân tích dữ liệu, trí tuệ nhân tạo đến lập trình nhúng và đám mây

Thư viện và Framework: Python có thư viện phong phú và các framework như Django, Flask, NumPy, và Pandas, giúp tăng cường khả năng phát triển.

Cộng đồng mạnh mẽ: Cộng đồng Python rất lớn, với hàng ngàn người đóng góp vào sự phát triển, hỗ trợ qua diễn đàn, tài liệu và dự án mã nguồn mở.

### 1.2. Thư viện pygame.

#### a) Giới thiệu chung.

Pygame là một thư viện lập trình game cho Python, được xâydựng trên nền tảng Simple DirectMedia Layer (SDL). Nó cung cấp cáccông cụ và chức năng để phát triển trò chơi đồ họa 2D một cách đơngiản và linh hoạt. Pygame giúp người lập trình tạo ra các trải nghiệmgame mà không cần mất nhiều thời gian và công sức.

#### b) Đặc điểm nổi bật.

* **Dễ Học và Sử Dụng**: Pygame được thiết kế để làm cho việc phát triển trò chơi đơn giản và dễ tiếp cận cho người mới học lập trình.
* **Đồ Họa 2D**: Pygame chủ yếu được sử dụng cho phát triển trò chơi đồ họa 2D, với khả năng xử lý sprite, hình ảnh, và âm thanh.
* **Sử Dụng SDL**: SDL là một thư viện đa nền tảng (cross-platform) giúp tương tác với phần cứng máy tính, điều này làm cho Pygame có khả năng chạy trên nhiều hệ điều hành khác nhau.
* **Thư Viện Mạnh Mẽ**: Pygame đi kèm với nhiều thư viện hỗ trợ cho đồ họa, âm thanh, đầu vào từ bàn phím và chuột, giúp đơn giản hóa việc phát triển game.
* **Cộng Đồng Hoạt Động**: Pygame có một cộng đồng sôi nổi với nhiều tài nguyên, hướng dẫn và ví dụ trực tuyến giúp người lập trình giải quyết vấn đề và phát triển kỹ năng.

#### c)Các phần chính của pygame.

* + **pygame.display**: Quản lý cửa sổ hiển thị game.
  + **pygame.sprite**: Hỗ trợ quản lý sprite và collision detection.
  + **pygame.image**: Cho phép làm việc với hình ảnh.
  + **pygame.mixer**: Điều khiển âm thanh và âm nhạc.
  + **pygame.event**: Xử lý sự kiện từ bàn phím, chuột, và các nguồn khác.
  + **pygame.font**: Cho phép vẽ văn bản trực tiếp trên cửa sổ game.

## II.Hướng dẫn cài đặt python và thư viện pygame.

### 2.1. Cài đặt python.

Để cài đặt **Python**, bạn vào trang chủ của Python tại https://python.org/ và tải về phiên bản phù hợp với hệ điều hành đang dùng. Ở đây tôi không đi vào chi tiết cách cài đặt, cá nhân tôi sử dụng phiên bản 3.11 cho Windows 64 bit và cài vào thư mục C:\Python11, chỉ lưu ý các bạn khi cài đặt nên tích chọn để đưa **Python** vào biến môi trường (**System Path**). Nếu không, bạn phải thêm thư mục Python vào **System Path** một cách thủ công như sau:

• Bấm chuột phải vào **My Computer** (hoặc **This PC**) ngoài Desktop và chọn **Properties**;

hoặc bấm tổ hợp phím ÿ + Break; hoặc vào **Control Panel\System and**

**Security\System**.

• Chọn thẻ **Advanced System Setting** để mở hộp thoại **System Properties.**

A screenshot of a computer program

Description automatically generated

*Hình : SEQ Hình\_: \\* ARABIC 1 Hộp thoại system properties.*

[Hình 2.1: Hộp thoại system properties.](about:blank)

• Chọn thẻ **Advanced** rồi chọn nút **Environment Variables…**

A screenshot of a computer

Description automatically generated

*Hình : SEQ Hình\_: \\* ARABIC 2Hộp thoại Environment Variables.*

[Hình 2.2: Hộp thoại Environment Variables.](about:blank)

• Trong thẻ **System variables**, chọn dòng **Path** và bấm **Edit.**

• Tiếp tục chọn **New** và gõ vào đường dẫn đến thư mục cài đặt Python, ở đây, của tôi là

**C:\Users\hau66\AppData\Local\Programs\Python\Python311**

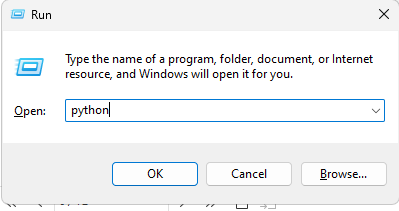
• Chọn tiếp **New** và thêm tiếp thư mục chứa các Scripts, ở đây, máy của tôi là **C:\Users\hau66\AppData\Local\Programs\Python\Python311\Scripts**

• Bấm **OK.**

A screenshot of a computer program

Description automatically generated

[Hình 2.3: Hộp thoại Edit Environment Variables.](#_1pxezwc)

Để kiểm tra đã thêm Python vào **System Path** chưa, bạn mở1 hộp thoại **Run** của Windows và gõ python, sau đó bấm Enter:

*Hình : SEQ Hình\_: \\* ARABIC 4 Hộp thoại Run.*

[Hình 2.4: Hộp thoại Run.](about:blank)

Nếu hiện cửa sổ như sau là thành công:A computer screen with white text

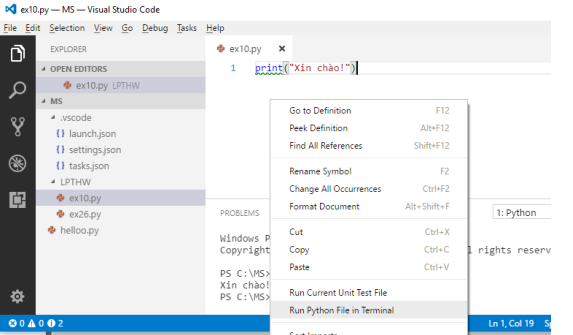
Description automatically generated

*Hình : SEQ Hình\_: \\* ARABIC 5 Trình biên dịch Python.*

[Hình 2.5: Trình biên dịch Python.](about:blank)

Sau khi cài đặt xong trình biên dịch Python, mặc định sẽ có một trình soạn thảo đi kèm là **IDLE**, tuy nhiên trình soạn thảo này khá cơ bản và không hỗ trợ nhiều cho người sử dụng như gợi ý các từ khóa, quản lý project, gỡ lỗi… nên tôi khuyên bạn nên sử dụng thêm một trình soạn thảo như **Notepad++, Sublime Text**, **Visual Studio Code**, **Pycharm**, **Eclipse**… Có rất nhiều chương trình như vậy, cả miễn phí và trả phí, nhưng cá nhân tôi thường sử dụng **Visual Studio Code** của Microsoft, đôi khi cũng sử dụng thêm cả **Sublime Text 3**.

Nếu mới làm quen với Python, bạn có thể cài đặt **Anaconda** tại https://www.continuum.io là một môi trường Python đã bao gồm cả trình dịch Python, trình soạn thảo với rất nhiều tính năng cao cấp chuyên dụng giành cho **Data Science,** và được cài sẵn rất nhiều thư viện, đặc biệt là các thư viện cho **Machine Learning**, **Data Science** như numpy, jupyter, matplotlib..

Ví dụ: chạy 1 chương trình python bằng **Visual Studio Code** bấm chuột phải vào vùng soạn thảo và chọn **Run Python File in Terminal.**

*Hình : SEQ Hình\_: \\* ARABIC 6 Cript python bằng Visual Studio Code.*

[Hình 2.6: Cript python bằng Visual Studio Code.](about:blank)

2.2: Cài đặt thư viện Pygame.

Để cài đặt Pygame trên máy tính của bạn, bạn có thể sử dụng pip, trình quản lý gói Python.

* Ở đây chỉ hướng dẫn cách cài thư viện trên hệ điều hành Window.
  + Đầu tiên , bạn mở hộp thoại **Run** và gõ lệnh **cmd**.A screenshot of a computer error

    Description automatically generated

*Hình : SEQ Hình\_: \\* ARABIC 7 Hộp thoại Run mở cmd.*

[Hình 2.7: Hộp thoại Run mở cmd.](about:blank)

* + Trong hộp thoại **Cmd** bạn gõ lệnh **pip install pygame,** sau đó nhấn **enter.**A screenshot of a computer

    Description automatically generated

*Hình : SEQ Hình\_: \\* ARABIC 8 Hộp thoại cmd cài đặt pygame.*

[Hình 2.8: Hộp thoại cmd cài đặt pygame.](about:blank)

* + Sau đó để kiểm tra đã cài đặt **pygame** chưa, bạn tạo cript và chạy như ảnh sau là thành công

A screen shot of a computer program

Description automatically generated[Hình 2.9: Kiểm tra đã cài đặt thư viện pygame.](about:blank)

*Hình : SEQ Hình\_: \\* ARABIC 9 Kiểm tra đã cài đặt thư viện pygame.*

# **CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ**

## I.PHÂN TÍCH VÀ THIẾT KẾ.

### 1.1. Phân tích hệ thống.

Hệ thống chính là trò chơi “Bắn Súng” phiêu lưu, trong đó người chơi điều khiển nhân vật di chuyển vượt chướng ngại vật, tiêu diệt kẻ địch và vượt qua các ải một cách an toàn.

Các file chính bao gồm: **RUN.py, button.py**.

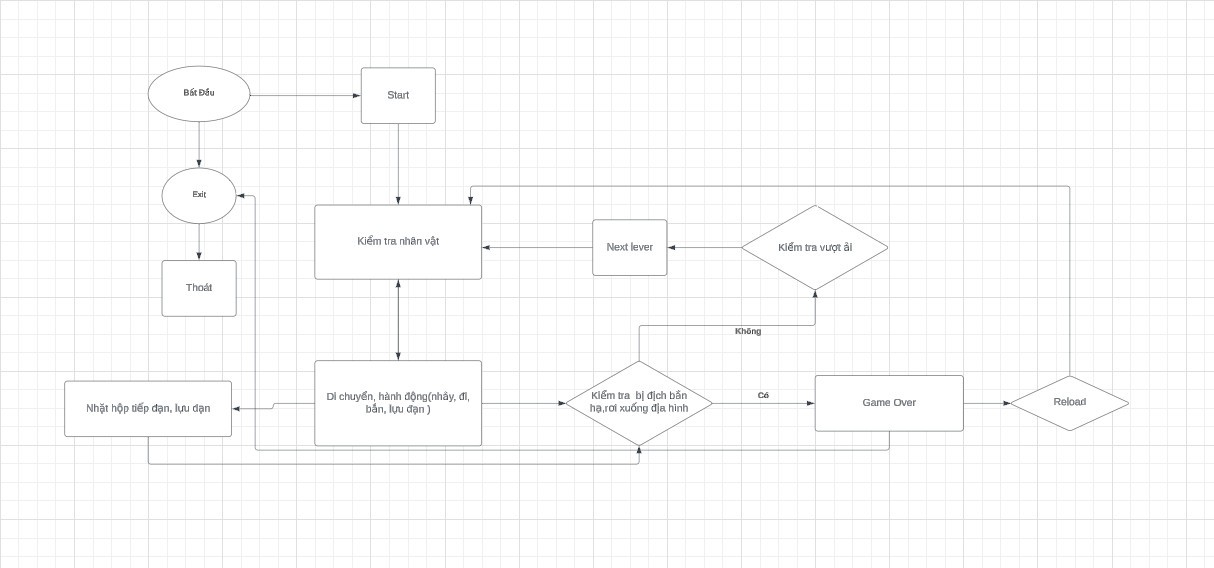
Bên cạnh đó là các thư mục để lưu các tệp hình ảnh, âm thanh và font chữ.

**RUN.py** là file chính để chạy trò chơi, chứa tất cả đối tượng trong game, khi chạy file sẽ có các nút START để bắt đầu trò chơi, EXIT khi muốn thoát game và RELOAD khi nhân vật mà người chơi điểu khiển chết trong trò chơi.

**button.py** để tạo ra các button(nút) và kiểm tra xem các hành động như chuột có nằm trong phạm vi của button hay không, nếu chuột thực hiện click vào button thì sẽ trả về action và self.clicked là True, ngược lại sẽ trả về False.

Các file này sử dụng thư viện Pygame để tạo giao diện đồ họa và xử lý sự kiện.

Dưới đây là mô hình hoạt động sao khi phân tích.



[Hình 3.1: Flowchart Diagram.](#_23ckvvd)

### 1.2. Thiết kế hệ thống.

Hệ thống được thiết kế theo mô hình lập trình hướng đối tượng (OOP) để tạo ra các đối tượng có khả năng tái sử dụng và dễ dàng bảo trì.

1. Lớp Soldier (người lính):
2. Lớp World (bản đồ):
3. Lớp Decoration (trang trí):
4. Lớp Water (vùng nước)
5. Lớp Exit (thoát):
6. Lớp ItemBox (hòm phụ kiện):
7. Lớp HealthBar (thanh máu):
8. Lớp Bullet (đạn dược):
9. Lớp Grenade (lựu đạn):
10. Lớp Explosion (boom nổ):
11. Lớp ScreenFade:

\*Soldier (Nhân vật):

-Mô tả:

Đại diện cho nhân vật trong trò chơi, bao gồm cả người chơi và kẻ địch.

-Thuộc tính:

+char\_type: Loại nhân vật (player hoặc enemy).

+speed: Tốc độ di chuyển.

+ammo: Số lượng đạn.

+grenades: Số lượng quả bom.

+health: Mức sống.

+max\_health: Mức sống tối đa.

+direction: Hướng di chuyển (-1: trái, 1: phải).

+vel\_y: Vận tốc theo trục y.

+jump: Trạng thái nhảy.

+in\_air: Trạng thái đang ở trong không trung.

+flip: Trạng thái lật hình ảnh.

+animation\_list: Danh sách hình ảnh cho các hành động khác nhau của nhân vật.

+frame\_index: Chỉ số frame hiện tại của animation.

+action: Hành động hiện tại của nhân vật.

+update\_time: Thời điểm cập nhật animation cuối cùng.

-Phương thức:

+update(): Cập nhật trạng thái của nhân vật.

+move(): Di chuyển nhân vật.

+shoot(): Bắn đạn.

+ai(): Điều khiển hành động của kẻ địch.

+update\_animation(): Cập nhật animation của nhân vật.

+update\_action(): Cập nhật hành động của nhân vật.

+check\_alive(): Kiểm tra xem nhân vật còn sống không.

+draw(): Vẽ nhân vật lên màn hình.

\*World (Thế giới):

-Mô tả:

Đại diện cho thế giới của trò chơi, bao gồm các đối tượng và vật thể.

-Thuộc tính:

+obstacle\_list: Danh sách các vật thể trong thế giới.

+level\_length: Độ dài của mỗi cấp độ.

-Phương thức:

+process\_data(): Xử lý dữ liệu của mỗi cấp độ và tạo ra các đối tượng trong thế giới.

+draw(): Vẽ các vật thể lên màn hình.

\*Decoration (Trang trí):

-Mô tả:

Lớp Decoration là một lớp con của pygame.sprite.Sprite, được sử dụng để hiển thị các hình ảnh trang trí trong trò chơi. Các hình ảnh này thường là các phần trang trí không tương tác trong cảnh chơi.

-Thuộc tính:

+img: Hình ảnh trang trí.

+x: Tọa độ x của hình ảnh trang trí trên màn hình.

+y: Tọa độ y của hình ảnh trang trí trên màn hình.

-Phương thức:

+\_\_init\_\_(self, img, x, y): Phương thức khởi tạo lớp Decoration. Nó nhận vào một hình ảnh, và các tọa độ x, y của hình ảnh trên màn hình. Trong phương thức này:

pygame.sprite.Sprite.\_\_init\_\_(self): Gọi phương thức khởi tạo của lớp cơ sở pygame.sprite.Sprite.

self.image = img: Gán hình ảnh trang trí cho thuộc tính image.

self.rect = self.image.get\_rect(): Lấy hình chữ nhật bao quanh hình ảnh để xác định vị trí và kích thước của hình ảnh.

self.rect.midtop: Đặt vị trí của hình ảnh trên màn hình.

+update(self): Phương thức này được gọi để cập nhật vị trí của đối tượng trang trí khi màn hình cuộn. Trong phương thức này:

self.rect.x += screen\_scroll: Cập nhật tọa độ x của đối tượng trang trí dựa trên giá trị screen\_scroll, giúp tạo hiệu ứng cuộn màn hình.

\*Water (Nước):

-Mô tả:

Lớp Water là một lớp con của pygame.sprite.Sprite, được sử dụng để hiển thị các phần nước trong trò chơi. Điều này có thể bao gồm các hình ảnh động hoặc đối tượng nước tĩnh, tùy thuộc vào cách triển khai.

-Thuộc tính:

+img: Hình ảnh biểu diễn nước.

+x: Tọa độ x của hình ảnh nước trên màn hình.

+y: Tọa độ y của hình ảnh nước trên màn hình.

-Phương thức:

+\_\_init\_\_(self, img, x, y): Phương thức khởi tạo lớp Water. Nó nhận vào một hình ảnh, và các tọa độ x, y của hình ảnh trên màn hình. Trong phương thức này:

pygame.sprite.Sprite.\_\_init\_\_(self): Gọi phương thức khởi tạo của lớp cơ sở pygame.sprite.Sprite.

self.image = img: Gán hình ảnh nước cho thuộc tính image.

self.rect = self.image.get\_rect(): Lấy hình chữ nhật bao quanh hình ảnh để xác định vị trí và kích thước của hình ảnh.

self.rect.midtop: Đặt vị trí của hình ảnh trên màn hình.

+update(self): Phương thức này được gọi để cập nhật vị trí của đối tượng nước khi màn hình cuộn.

Trong phương thức này:

self.rect.x += screen\_scroll: Cập nhật tọa độ x của đối tượng nước dựa trên giá trị screen\_scroll, giúp tạo hiệu ứng cuộn màn hình.

\*Exit (Lối ra):

-Mô tả:

Lớp Exit là một lớp con của pygame.sprite.Sprite, được sử dụng để đại diện cho đối tượng thoát ra khỏi màn chơi, chẳng hạn như cổng hoặc lối thoát.

-Thuộc tính:

+img: Hình ảnh biểu diễn cho đối tượng thoát ra.

+x: Tọa độ x của hình ảnh đối tượng thoát ra trên màn hình.

+y: Tọa độ y của hình ảnh đối tượng thoát ra trên màn hình.

-Phương thức:

+\_\_init\_\_(self, img, x, y): Phương thức khởi tạo của lớp Exit. Nó nhận vào một hình ảnh và các tọa độ x, y của đối tượng thoát ra trên màn hình. Trong phương thức này:

pygame.sprite.Sprite.\_\_init\_\_(self): Gọi phương thức khởi tạo của lớp cơ sở pygame.sprite.Sprite.

self.image = img: Gán hình ảnh của đối tượng thoát ra cho thuộc tính image.

self.rect = self.image.get\_rect(): Lấy hình chữ nhật bao quanh hình ảnh để xác định vị trí và kích thước của đối tượng thoát ra.

self.rect.midtop: Đặt vị trí của hình ảnh thoát ra trên màn hình.

+update(self): Phương thức này được gọi để cập nhật vị trí của đối tượng thoát ra khi màn hình cuộn.

Trong phương thức này:

self.rect.x += screen\_scroll: Cập nhật tọa độ x của đối tượng thoát ra dựa trên giá trị screen\_scroll, giúp tạo hiệu ứng cuộn màn hình.

\*ItemBox (Hộp vật phẩm):

-Mô tả:

Lớp ItemBox là một lớp con của pygame.sprite.Sprite, được sử dụng để đại diện cho hộp chứa vật phẩm mà người chơi có thể nhặt được trong trò chơi.

-Thuộc tính:

+item\_type: Loại vật phẩm trong hộp, có thể là "Health" (sức khỏe), "Ammo" (đạn) hoặc "Grenade" (lựu đạn).

+x: Tọa độ x của hộp vật phẩm trên màn hình.

+y: Tọa độ y của hộp vật phẩm trên màn hình.

-Phương thức:

+\_\_init\_\_(self, item\_type, x, y): Phương thức khởi tạo của lớp ItemBox. Nó nhận vào loại vật phẩm, và các tọa độ x, y của hộp vật phẩm trên màn hình. Trong phương thức này:

pygame.sprite.Sprite.\_\_init\_\_(self): Gọi phương thức khởi tạo của lớp cơ sở pygame.sprite.Sprite.

self.image = item\_boxes[self.item\_type]: Lấy hình ảnh tương ứng với loại vật phẩm từ một bộ sưu tập hình ảnh được đặt tên item\_boxes, và gán cho thuộc tính image.

self.rect = self.image.get\_rect(): Lấy hình chữ nhật bao quanh hình ảnh để xác định vị trí và kích thước của hộp vật phẩm.

self.rect.midtop: Đặt vị trí của hộp vật phẩm trên màn hình.

+update(self): Phương thức này được gọi để cập nhật vị trí của hộp vật phẩm khi màn hình cuộn và kiểm tra xem người chơi đã nhặt hộp vật phẩm hay chưa. Trong phương thức này:

self.rect.x += screen\_scroll: Cập nhật tọa độ x của hộp vật phẩm dựa trên giá trị screen\_scroll, giúp tạo hiệu ứng cuộn màn hình.

pygame.sprite.collide\_rect(self, player): Kiểm tra va chạm giữa hộp vật phẩm và người chơi.

-Nếu người chơi va chạm với hộp vật phẩm, phương thức sẽ kiểm tra loại hộp vật phẩm và thực hiện các hành động tương ứng:

-Nếu là hộp "Health": Tăng sức khỏe của người chơi lên 25 điểm, nhưng không vượt quá sức khỏe tối đa.

-Nếu là hộp "Ammo": Tăng số đạn của người chơi lên 15 viên.

-Nếu là hộp "Grenade": Tăng số lựu đạn của người chơi lên 3.

Cuối cùng, phương thức sẽ xóa hộp vật phẩm khỏi danh sách sprite nếu người chơi đã nhặt.

\*HealthBar (Thanh máu):

-Mô tả:

Lớp HealthBar được sử dụng để hiển thị thanh máu của người chơi trên màn hình.

-Thuộc tính:

+x: Tọa độ x của thanh máu trên màn hình.

+y: Tọa độ y của thanh máu trên màn hình.

+health: Số lượng máu hiện tại của người chơi.

+max\_health: Số lượng máu tối đa mà người chơi có thể có.

-Phương thức:

+\_\_init\_\_(self, x, y, health, max\_health): Phương thức khởi tạo của lớp HealthBar. Nó nhận vào các tham số là tọa độ x, y của thanh máu, số lượng máu hiện tại và số lượng máu tối đa của người chơi.

+draw(self, health): Phương thức này được sử dụng để vẽ thanh máu lên màn hình. Nó nhận vào số lượng máu mới của người chơi và cập nhật thanh máu dựa trên giá trị đó. Trong phương thức này:

self.health = health: Cập nhật số lượng máu hiện tại của người chơi.

ratio = self.health / self.max\_health: Tính tỉ lệ máu hiện tại so với máu tối đa.

pygame.draw.rect(screen, BLACK, (self.x - 2, self.y - 2, 154, 24)): Vẽ một hình chữ nhật đen để làm nền cho thanh máu.

pygame.draw.rect(screen, RED, (self.x, self.y, 150, 20)): Vẽ một hình chữ nhật màu đỏ để đại diện cho toàn bộ thanh máu.

pygame.draw.rect(screen, GREEN, (self.x, self.y, 150 \* ratio, 20)): Vẽ một hình chữ nhật màu xanh lá cây, chiều rộng của nó tương ứng với tỉ lệ máu hiện tại so với máu tối đa, từ 0 đến 150 pixel.

\*Bullet (Đạn):

-Mô tả:

Lớp Bullet đại diện cho đạn được bắn từ vũ khí của người chơi hoặc kẻ địch.

-Thuộc tính:

+x: Tọa độ x ban đầu của đạn.

+y: Tọa độ y ban đầu của đạn.

+direction: Hướng di chuyển của đạn (trái hoặc phải).

-Phương thức:

+\_\_init\_\_(self, x, y, direction): Phương thức khởi tạo của lớp Bullet. Nhận vào các tham số là tọa độ ban đầu (x, y) và hướng di chuyển của đạn (trái hoặc phải).

+update(self): Phương thức này được gọi mỗi lần cập nhật vòng lặp chính. Trong phương thức này:

Đạn sẽ di chuyển theo hướng đã cho (self.direction) với tốc độ (self.speed).

Nếu đạn đi ra khỏi màn hình, nó sẽ bị hủy.

Kiểm tra va chạm với các vật cản trong môi trường game, nếu có va chạm, đạn sẽ bị hủy.

Kiểm tra va chạm với nhân vật, nếu có va chạm, đạn sẽ gây sát thương cho nhân vật và bị hủy. Nếu là đạn của người chơi, sát thương sẽ được gây cho kẻ địch và ngược lại.

\*Grenade (Quả bom):

-Mô tả:

Lớp Grenade đại diện cho quả bom được ném từ vũ khí của người chơi hoặc kẻ địch.

-Thuộc tính:

+x: Tọa độ x ban đầu của quả bom.

+y: Tọa độ y ban đầu của quả bom.

+direction: Hướng di chuyển của quả bom (trái hoặc phải).

+timer: Thời gian còn lại trước khi quả bom phát nổ.

+vel\_y: Vận tốc theo trục y của quả bom.

+speed: Tốc độ di chuyển của quả bom.

+Phương thức:

+\_\_init\_\_(self, x, y, direction): Phương thức khởi tạo của lớp Grenade. Nhận vào các tham số là tọa độ ban đầu (x, y) và hướng di chuyển của quả bom (trái hoặc phải).

+update(self): Phương thức này được gọi mỗi lần cập nhật vòng lặp chính. Trong phương thức này:

Quả bom sẽ di chuyển theo hướng đã cho và với vận tốc xác định.

Kiểm tra va chạm với các vật cản trong môi trường game để xác định hướng di chuyển và tốc độ mới của quả bom.

Cập nhật vị trí của quả bom sau mỗi lần cập nhật.

Đếm ngược thời gian đến khi quả bom phát nổ. Khi hết thời gian, quả bom sẽ bị hủy và gây sát thương cho nhân vật và kẻ địch nếu ở gần.

\*Explosion (Vụ nổ):

-Mô tả:

Lớp Explosion đại diện cho hiệu ứng nổ, thường được kích hoạt khi một quả bom phát nổ hoặc khi một sự kiện tương tự xảy ra trong trò chơi.

-Thuộc tính:

+x: Tọa độ x của vị trí ban đầu của hiệu ứng nổ.

+y: Tọa độ y của vị trí ban đầu của hiệu ứng nổ.

+scale: Tỉ lệ co giãn của hiệu ứng nổ so với kích thước gốc.

-Phương thức:

+\_\_init\_\_(self, x, y, scale): Phương thức khởi tạo của lớp Explosion. Nhận vào các tham số là tọa độ ban đầu (x, y) và tỉ lệ co giãn của hiệu ứng nổ.

+update(self): Phương thức này được gọi mỗi lần cập nhật vòng lặp chính. Trong phương thức này:

Hiệu ứng nổ sẽ di chuyển theo hướng của màn hình.

Cập nhật hình ảnh của hiệu ứng nổ để tạo ra hiệu ứng chớp chớp.

Khi hoàn thành hiệu ứng nổ (tất cả các hình ảnh đã được sử dụng), hiệu ứng nổ sẽ bị hủy.

\*ScreenFade (Làm mờ):

-Mô tả:

Lớp ScreenFade được sử dụng để tạo hiệu ứng làm mờ màn hình trong trò chơi, với khả năng làm mờ toàn bộ màn hình hoặc làm mờ một phần của màn hình theo hướng dọc.

-Thuộc tính:

+Direction: Hướng của hiệu ứng làm mờ (1 cho làm mờ toàn bộ màn hình, 2 cho làm mờ theo hướng dọc).

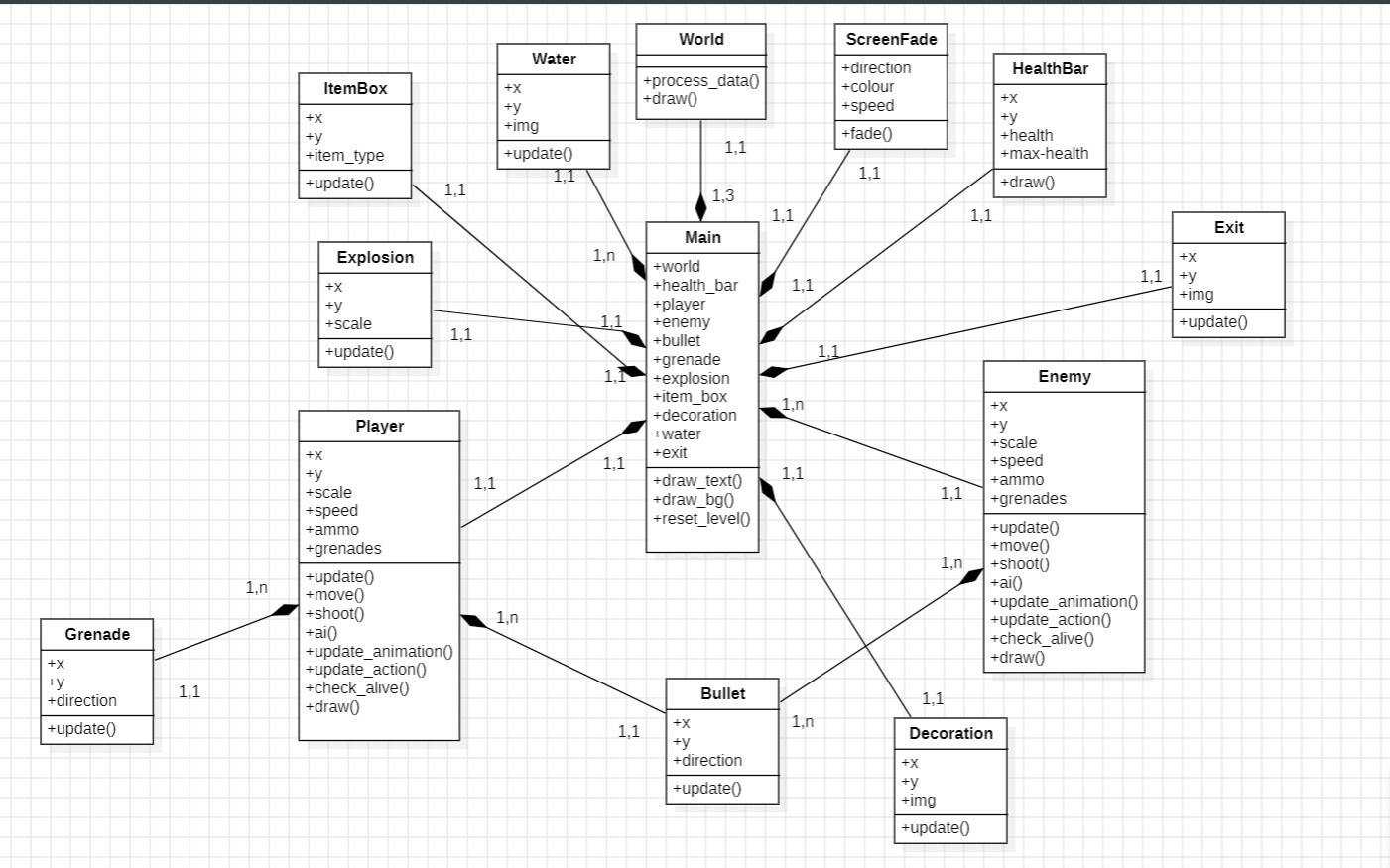
+Colour: Màu sắc của hiệu ứng làm mờ.

Speed: Tốc độ của hiệu ứng làm mờ.

+Fade\_counter: Biến đếm được sử dụng để điều chỉnh mức độ làm mờ.

-Phương thức:

+fade(): Phương thức này cập nhật hiệu ứng làm mờ dựa trên hướng, màu sắc và tốc độ được chỉ định. Nó vẽ các hình chữ nhật lên màn hình để tạo hiệu ứng làm mờ. Nếu hiệu ứng đã hoàn thành (đã làm mờ đến mức độ mong muốn), phương thức này trả về True, ngược lại trả về False.



Hình 3.2: Classdiagram

Ngoài ra còn có các class khác trong file Menu.py với các thuộc tính và phương thức để quản lý trạng thái trò chơi như: bắt đầu trò chơi, cập nhật trạng thái và thoát khỏi trò chơi. File Menu.py tạo một giao diện menu game và gọi phương thức start để bắt đầu trò chơi.

Sử dụng Pygame để tạo giao diện đồ họa và xử lý sự kiện, như vẽ các đối tượng lên màn hình, xử lý phím nhấn để thay đổi hướng di chuyển của rắn và kiểm tra va chạm với mồi hoặc với chính nó.

### 1.3. Tổng kết.

Hệ thống được thiết kế để tạo ra trò chơi bắn súng-shooter bằng Python.

File chính bao gồm RUN.py đảm nhận vai trò quản lý một phần của trò chơi.

Sử dụng mô hình OOP giúp tạo ra các đối tượng có khả năng tái sử dụng và dễ dàng bảo trì.

Các lớp Solier, World và Main được tạo ra để quản lý các thành phần chính của trò chơi, bao gồm di chuyển, thay đổi hướng, kiểm tra va chạm của nhân vật, xây dựng thế giới và quản lý trạng thái trò chơi.

Sử dụng thư viện Pygame để tạo giao diện đồ họa và xử lý sự kiện.

Qui trình thực thi code từ các file này giúp quản lý và xử lý dữ liệu một cách có tổ chức và dễ dàng mở rộng cho các tính năng và chức năng khác trong trò chơi rắn săn mồi.

## II.Xây dựng game.

-Giao diện khi người dùng mở game:

* Start: bắt đầu.
* Exit: thoát game.

*Hình : 12 Giao diện khi vừa khởi chạy game.*

* -Giao diện khi bắt đầu trò chơi

[Hình 3.3: Giao diện khi vừa khởi chạy game.](about:blank)

Ảnh có chứa ảnh chụp màn hình, bầu trời, cây cối

Mô tả được tạo tự động

[Hình 3.4: Giao diện game play.](#_41mghml)

-Giao diện game over khi chết:

* Restart: Bắt đầu lại màn đang chơi.
* Exit: Thoát game.



*Hình : 14 Giao diện game over khi va chạm.*

[Hình 3.5: Giao diện game over.](about:blank)

# **CHƯƠNG IV: KẾT LUẬN.**

## I. Các vấn đề đã thực hiện được trong đề tài.

Qua đề tài “Tìm hiểu ngôn ngữ lập trình Python và lập trình game bắn súng qua thư viện Pygame” chúng em đã hoàn thành được những mục tiêu của mình :

-Tìm hiểu được ngôn ngữ Python:

* Python là gì?
* Lịch sử phát triển.
* Ưu điểm nổi bật.

-Lập trình game bằng Python.

-Cài đặt ngôn ngữ Python.

-Tìm hiểu được về Pygame.

-Phân tích thiết kế và xây dựng game bắn súng bằng ngôn ngữ Python:

* Phân tích và thiết kế hệ thống.
* Xây dựng nên giao diện game.

## II.Hướng phát triển.

Chúng em không tránh được những sai sót của mình nhưng sẽ cố gắng phát triển và hoàn thiện hơn để có thể làm đồ án tốt nghiệp sau này.  
Các phần chưa thực hiện: phần option tùy tùy chỉnh giao diện, âm thanh, …

**TÀI LIỆU THAM KHẢO**

[1] Lập trình python cơ bản

[Python Tutorial (w3schools.com)](https://www.w3schools.com/python/).

[2] Lập trình game với python

<https://www.pygame.org/news>.

[3] Trang điện tử: youtube.com.

[4] Các group cộng đồng.

[5] Pygame Font Page – pygame v2.6.0 documentation

<https://www.pygame.org/docs/>.

[6] Pixel Joint – The Internet Pixel Art Gallery

<https://pixeljoint.com/>.

[7] OpenGameArt

<https://opengameart.org/>

[8] Real Python - Pygame Tutorial Series

<https://realpython.com/pygame-a-primer/>

[9] Pygame Tutorials on Programiz

<https://www.programiz.com/python-programming/pip>

[10] Tech With Tim - Pygame Tutorials

<https://www.techwithtim.net/tutorials/game-development-with-python/pygame-tutorial>.