

**BÁO CÁO CUỐI KÌ MÔN LẬP TRÌNH NÂNG CAO**

**Giảng viên:** Lê Hoàng Việt Tuấn

**Lớp**: 20BITV02

**Thành viên nhóm:**

Vương Minh Trí 2000000050

Nguyễn Phan Bảo Quý 2000003905

Trần Gia Bảo 2000004968

Phạm Hồ Tấn Đạt 2000005153

**LỜI CẢM ƠN!**

Chúng em xin chân thành cảm ơn sự hướng dẫn tận tình của thầy Lê Hoàng Việt Tuấn trong việc chỉ dạy chúng em trong môn lập trình nâng cao, thầy đã hướng dẫn tận tình và chu đáo để chúng em thành thạo kĩ năng sử dụng Python, nhờ cảm hứng của thầy mà chúng em đã học được rất nhiều điều hay về thiết kế giao diện của Python – ngôn ngữ lập trình tiện lợi nhất hiện tại.

Chúng em xin chân thành cảm ơn về những bài giảng có ích của thầy!”

TP.HCM, ngày 6 tháng 5 năm 2022

**LỜI NÓI ĐẦU**

Công nghệ Thông tin là một trong những ngành khoa học ngày càng được quan tâm và sử dụng rộng rãi trong mọi lĩnh vực của cuộc sống. Với những ưu điểm mạnh, có thể ứng dụng được nhiều trong cuộc sống. Như trong việc tự làm game đơn giản hoặc một game phức tạp như hành động, giải đố, bắn súng…vv. Nhờ có các phần mềm thì công việc trở nên đơn giản hơn rất nhiều.

MỤC LỤC

[PHẦN 1: GIỚI THIỆU LẬP TRÌNH ỨNG DỤNG 6](#_Toc102844279)

[1. Giao diện lập trình ứng dụng là gì? 6](#_Toc102844280)

[2. Có những loại API nào? 6](#_Toc102844281)

[3. Ưu điểm của API 8](#_Toc102844282)

[4. Nhược điểm của API 8](#_Toc102844283)

[PHẦN 2: LẬP TRÌNH GAME 9](#_Toc102844284)

[CHƯƠNG 1: TỔNG QUAN LẬP TRÌNH GAME 9](#_Toc102844285)

[1. Mục đích của con game mà chúng em lập trình ra: 9](#_Toc102844286)

[2. Tổng quang về figma 9](#_Toc102844287)

[3. Tổng quan về python 10](#_Toc102844288)

[CHƯƠNG 2: CÁC CHỨC NĂNG CỦA PHẦN MỀM GAME 12](#_Toc102844289)

[1. Giao diện Game 12](#_Toc102844290)

[2. Các button trên giao diện 13](#_Toc102844291)

[CHƯƠNG 3: HƯỚNG DẪN CÁCH CHƠI 14](#_Toc102844292)

[CHƯƠNG 4: ĐÁNH GIÁ GAME 20](#_Toc102844293)

[Tổng kết 20](#_Toc102844294)

[1. Khả thi về kỹ thuật 20](#_Toc102844295)

[2. Công cụ phát triển 20](#_Toc102844296)

[3. Bảng tổng kết 21](#_Toc102844297)

[4. Yêu cầu bảo mật: Thấp 21](#_Toc102844298)

[5. Ngôn ngữ viết 21](#_Toc102844299)

[KẾT LUẬN 22](#_Toc102844300)

[BẢNG PHÂN CÔNG NHÓM 4 MEMBERS 23](#_Toc102844301)

[NGUỒN THAM KHẢO: 23](#_Toc102844302)

# PHẦN 1: GIỚI THIỆU LẬP TRÌNH ỨNG DỤNG

# Giao diện lập trình ứng dụng là gì?

API (viết tắt từ Application Programming Interface) là một giao diện mà một hệ thống máy tính hay ứng dụng cung cấp cho phép các yêu cầu dịch vụ có thể được tạo ra từ các chương trình máy tính khác, và/hoặc cho phép dữ liệu có thể được trao đổi qua lại giữa chúng. Chẳng hạn, một chương trình máy tính có thể (và thường là phải) dùng các hàm API của [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh" \o "Hệ điều hành) để xin cấp phát bộ nhớ và truy xuất tập tin. Nhiều loại hệ thống và ứng dụng thực hiện API, như các hệ thống đồ họa, cơ sở dữ liệu, mạng, dịch vụ web, và ngay cả một số trò chơi máy tính.

Đây là phần mềm hệ thống cung cấp đầy đủ các chức năng và các tài nguyên mà các lập trình viên có thể rút ra từ đó để tạo nên các tính năng giao tiếp người- máy như: các trình đơn kéo xuống, tên lệnh, hộp hội thoại, lệnh bàn phím và các cửa sổ. Một trình ứng dụng có thể sử dụng nó để yêu cầu và thi hành các dịch vụ cấp thấp do hệ điều hành của máy tính thực hiện. Hệ giao tiếp lập trình ứng dụng giúp ích rất nhiều cho người sử dụng vì nó cho phép tiết kiệm được nhiều thời gian tìm hiểu các chương trình mới, do đó khích lệ mọi người dùng nhiều ứng dụng hơn.

# Có những loại API nào?

* Dựa trên phân ngành trong ngành công nghệ thông tin thì API có các loại sau:

#### ****Web API – hệ thống API trên nền tảng web****

Loại này rất phổ biến, các website lớn đều [thiết kế web-app](https://freelancervietnam.vn/thiet-ke-web-application-dich-vu-freelancer-lap-trinh-web-app-co-phai-lua-chon-tot-danh-cho-ban/) với nền tảng hệ thống API giúp bạn kết nối, lấy dữ liệu hoặc đôi khi là cập nhật dữ liệu vào hệ thống.

#### ****Hệ thống API trên hệ điều hành****

**Windows**cũng như **Linux** cung cấp các tài liệu API đặc tả các hàm, phương thức và các giao thức kết nối. Nhờ đó, lập trình viên có thể tạo được các phần mềm ứng dụng có chức năng tương tác với hệ điều hành. Đối với Windows thì bạn có thể [tạo ứng dụng chạy trên máy tính bằng C++ và Win32 API](https://docs.microsoft.com/en-us/windows/win32/). Còn với Linux bạn có thể tham khảo thử [Electron API](https://electronjs.org/).

#### ****API của thư viện phần mềm (framework)****

API mô tả cũng như quy định các hoạt động mong muốn mà thư viện cung ứng. Một API có thể có nhiều cách triển khai hoạt động khác nhau. API cũng có thể giúp cho một chương trình được viết bằng ngôn ngữ này nhưng có thể sử dụng được thư viện được viết bằng ngôn ngữ khác.

* Dựa trên quyền truy cập thì có thể phân API thành các loại sau:
* **API mở (Open API):** Có sẵn, công khai, không hạn chế quyền truy cập.
* **API đối tác (Partner API):** Cần có quyền hoặc giấy phép cụ thể mới truy cập được.
* **API nội bộ (Internal API):** Chỉ dùng cho các hệ thống nội bộ (chẳng hạn như công ty, tổ chức). Các đội ngũ phát triển nội bộ khác nhau có thể sử dụng chúng để cải thiện cho các sản phẩm hay dịch vụ chỉ phục vụ cho riêng nhân viên công ty.

# **Ưu điểm của API**

**- Tính tự động hóa rất cao:** API có thể thay thế chúng ta quản lý công việc cực kì hiệu quả. API giúp các cơ quan có thể cập nhật, xử lý hoàn thiện công việc nhanh và chất lượng hơn.

**- Ứng dụng cực kỳ linh hoạt:** API có thể truy cập vào các thành phần ứng dụng giúp việc cung cấp dịch vụ và thông tin linh hoạt hơn nhiều.

**- Khả năng thích ứng tốt:** API có chức năng thay đổi cũng như dự đoán thay đổi theo thời gian cho nên dữ liệu được truyền tốt hơn, thông tin được chọn lọc kĩ hơn, dịch vụ tốt hơn.

**- Cá nhân hóa:** Người dùng có thể tinh chỉnh API cho phù hợp nhu cầu sử dụng.

**- Dữ liệu có sẵn:** API cho phép mọi thông tin được tạo ở dạng chính chủ luôn có sẵn cho mọi người xem được.

**- Phạm vi:** Ngoài dữ liệu mới có sẵn được chia sẻ rộng rãi, người dùng còn có thể hiệu chỉnh web API để cung cấp dịch vụ, thông tin cá nhân hóa.

# **Nhược điểm của API**

- API **chỉ hỗ trợ mặc định get, post** chưa hoàn toàn là restful service.

- Cấu hình cố hữu của WFC làm cấu hình của **API cực kì khó nhớ**. Nếu mới sử dụng, người dùng khó mà nhớ và dùng dễ dàng như các ứng dụng khác.

*WCF (Windows Communication Foundations) là mô hình phát triển ứng dụng hướng dịch vụ trên nền tảng của Microsoft.*

**- Web API tốn kha khá chi phí** vận hành, phát triển, hiệu chỉnh và đòi hỏi kiến thức chuyên sâu. Khi bị tấn công hệ thống, người dùng đôi lúc gặp rắc rối về bảo mật.

# PHẦN 2: LẬP TRÌNH GAME

# CHƯƠNG 1: TỔNG QUAN LẬP TRÌNH GAME

Để lập trình game hoàn chỉnh chúng ta cần trải qua 5 giai đoạn: Khảo sát, phân tích game, thiết kế giao diện game, cài đặt về máy, trải nghiệm thử game.

Để tạo giao diện game chúng ta thường dùng figma và để lập trình game thì thường dùng python.

# **Mục đích của con game mà chúng em lập trình ra:**

* Game có những câu hỏi rất thú vị gồm có những kiến thức liên quan tới lịch sử, thể thao, ca dao tục ngữ, …
* Game có thể chơi cùng bạn bè để có thể phá đảo những câu hỏi có thể khó khi chơi cá nhân những lại cực kì dễ khi chơi với một nhóm.

# **Tổng quang về figma**

Figma là một công cụ chỉnh sửa giao diện Website/App hoàn toàn thực hiện trên trình duyệt mà không cần bất kỳ thao tác tải về nào. Bên cạnh đó, Figma còn có thể lưu trữ dữ liệu thông qua hệ thống đám mây vô cùng đơn giản. Đây là một ứng dụng vô cùng phù hợp với các bạn mới bắt đầu trở thành Thiết kế UX/UI mà không cần sử dụng những phần mềm quá cồng kềnh khác.

Là một công cụ luôn được lựa chọn ưu tiên hàng đầu của các Designer chắc hẳn sẽ có những điểm nổi bật khiến Figma trở nên phổ biến và thông dụng, có thể kể đến:

* Khả năng tương thích cao, có thể truy cập vào Figma trên bất cứ hệ điều hành nào, chỉ cần bạn có một trình duyệt web là có thể sử dụng Figma.
* Giao diện tối giản, thân thiện với người dùng.
* Cho phép chế độ Feedbacks, Comments ngay trên chính mỗi Project từ đó giúp cho các bạn thiết kế UI sẽ dễ dàng nhận được những góp ý đến từ Project Manager, User…, giúp cho công việc tiết kiệm được nhiều thời gian và nhanh chóng hoàn thành.
* Hỗ trợ lưu trữ dữ liệu đám mây cực kỳ tiện lợi, không cần tốn quá nhiều tài nguyên trên máy.
* Những tính năng độc đáo chuyên biệt cho thiết kế với bộ phím tắt đơn giản, nhanh chóng giúp các Designer tiết kiệm được nhiều thời gian cho một Project.

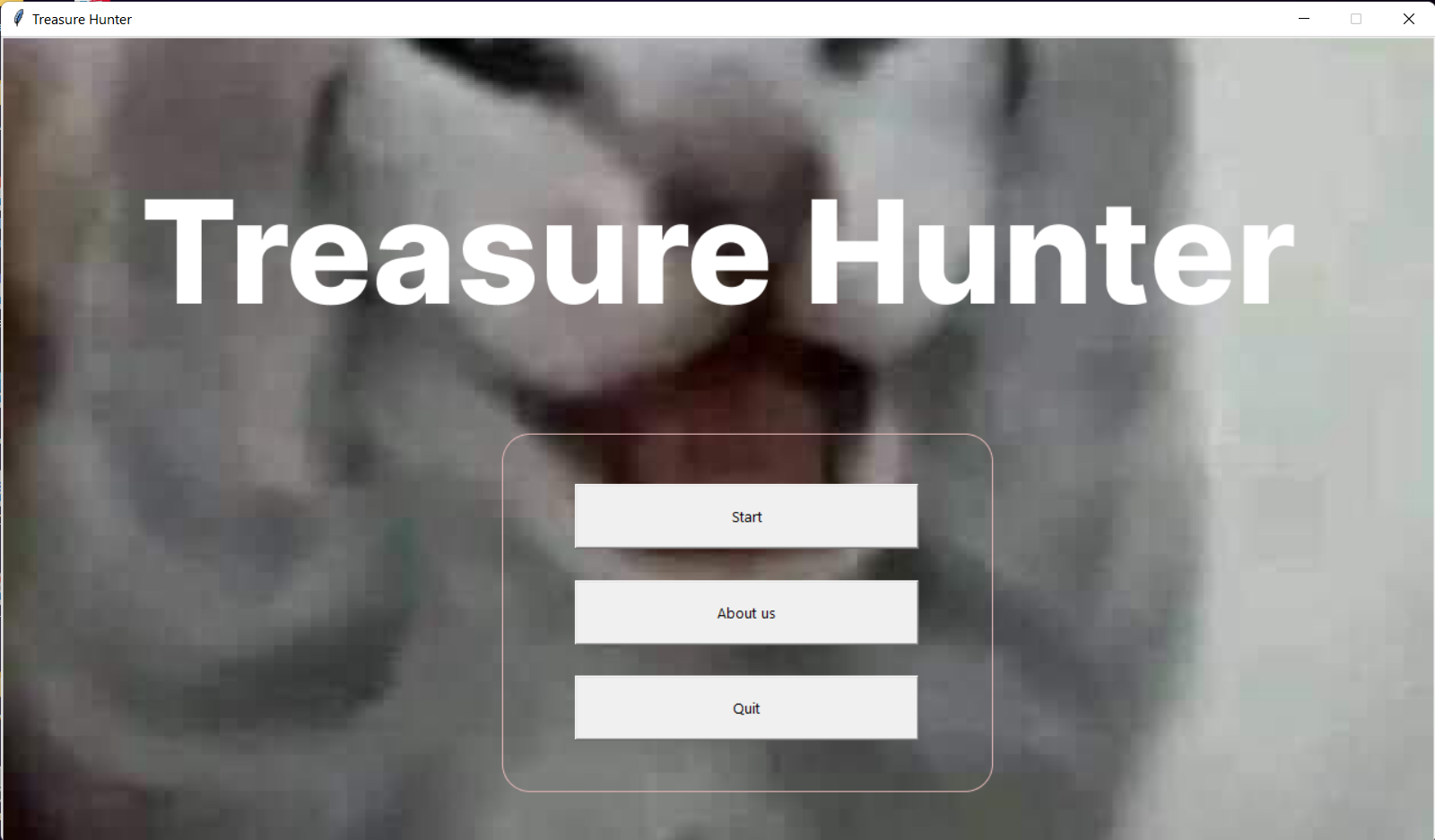
# **Tổng quan về python**

Là ngôn ngữ lập trình bậc cao, phục vụ cho các mục đích lập trình đa năng. Ưu điểm nổi bật nhất đó chính là dễ đọc, dễ nhớ, dễ học. Python là ngôn ngữ có cấu trúc tương đối rõ ràng, thuận tiện cho người mới học lập trình. Cấu trúc của python còn cho phép người dùng sử dụng để viết mã lệnh với số lần gõ phím tối thiểu. Nói cách khác, khái niệm lập trình python là gì còn được hiểu là cửa ngõ để mọi người bước vào thế giới lập trình máy tính hay là một phương tiện để bạn nhận được một khoản lương “đáng mơ ước” từ một công việc sáng tạo. Python mang bản chất là ngôn ngữ lập trình bậc cao, được tạo ra bởi Guido van Rossum. Ngôn ngữ python được thiết kế hướng tới đối tượng với cấu trúc hàng và cách xử lý dữ liệu đơn giản, dễ đọc. Nó sẽ giúp người dùng tạo ra những chương trình hay với số lượng dòng code ít nhất. Python hiện có 2 version phổ biến đó chính là python 2 và python 3. Một số HĐH Linux cũ không được hỗ trợ python 3 mà cần phải cài đặt. Do đó người dùng nên lựa chọn một phiên bản của python để viết lập trình và bạn nên sử dụng python 3.

Một trong những ưu điểm lớn nhất của Python để phát triển trò chơi là ngôn ngữ này khá dễ dàng vì cách viết giống tiếng Anh. Python là một trong những ngôn ngữ lập trình đơn giản nhất để viết và đọc, khiến đây là điểm khởi đầu tốt nhất cho các hoạt động viết mã còn lại. Đối với những công ty Game lớn, họ cần sử dụng ngôn ngữ này tạo một nguyên mẫu trò chơi một cách nhanh chóng để xem xét có nên phát triển tiếp hay không. Sử dụng Python sẽ tiết kiệm thời gian hơn so với các ngôn ngữ khác như C++. Đối với những nhà lập trình Game mới bắt đầu, thời gian và tốc độ lập trình rất quan trọng. Sử dụng một ngôn ngữ đơn giản hơn để tạo ra một trò chơi sẽ giúp tạo nên sự khác biệt thời gian lớn thay vì lập trình trong nhiều ngày, thậm chí là nhiều tuần liền.

# CHƯƠNG 2: CÁC CHỨC NĂNG CỦA PHẦN MỀM GAME

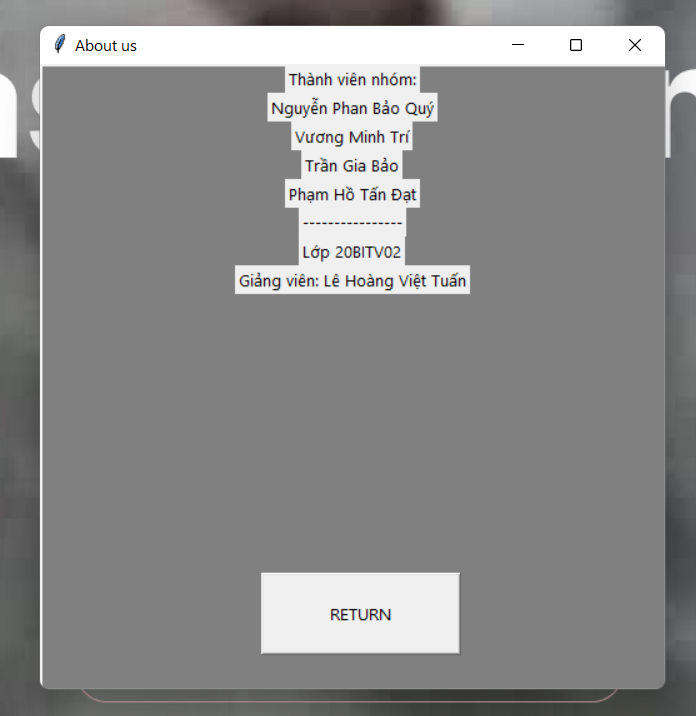
# **Giao diện Game**



Đây là giao diện khi bạn mới vừa mở game lên, giao diện khá đơn giản và dễ nhìn gồm có tên game là “**Treasure Hunter**” và các button cơ bản là “**Start**”, “**About us**” và “**Quit**”.

# **Các button trên giao diện**

* Button “**Start**”: Nhấn vào button này thì chắn chắn sẽ vô được màn đầu tiên. Chi tiết sẽ nói thêm ở **Chương 4. Hướng dẫn cách chơi.**
* Button “**About us**”: Nhấn vào button này sẽ hiển thị ra những thành viên sáng lập ra game này, mã lớp và giáo viên hưỡng dẫn. Nếu muốn quay lại giao diện ban đầu thì bạn chỉ cần nhấn “**RETURN**” là sẽ trở về giao diện đầu tiên.

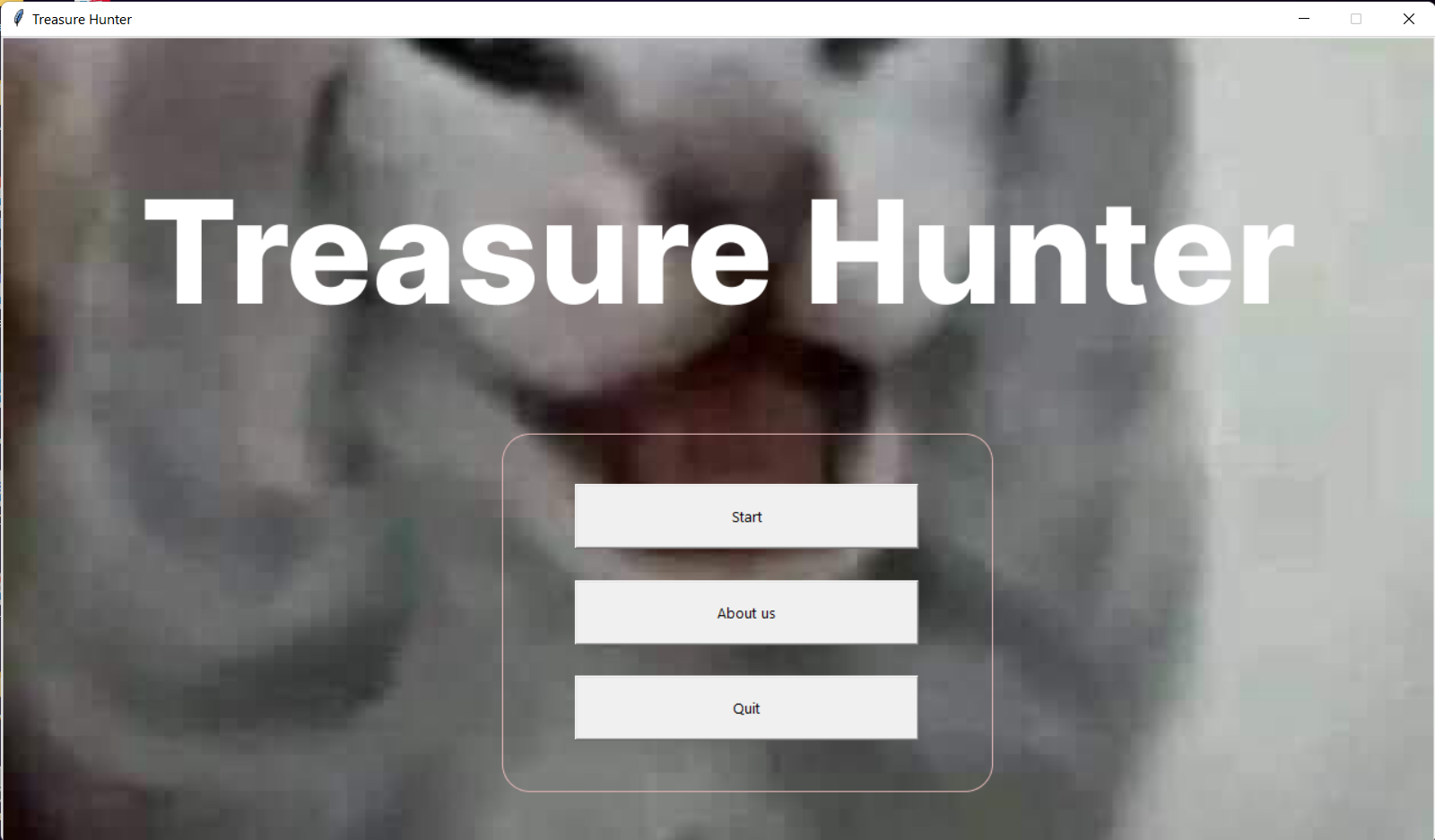


* Button “**Exit**”: Nhấn button này thì bạn sẽ thoát game.

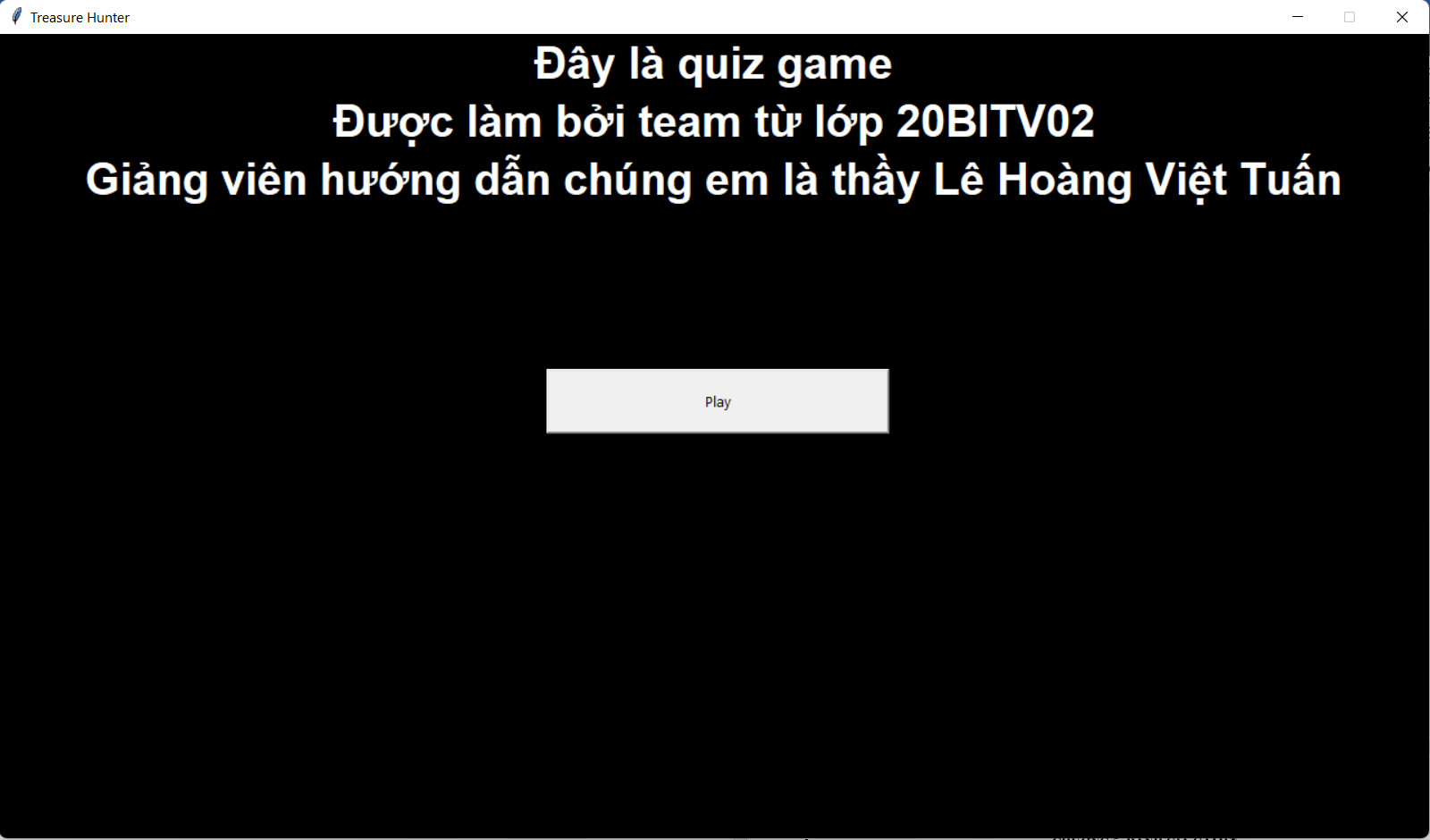
# CHƯƠNG 3: HƯỚNG DẪN CÁCH CHƠI

Sau đây nhóm sẽ hướng dẫn sử dụng cách chơi như sau:

Đầu tiên là màn hình chính của game. Chỉ cần nhấp vào nút **Start** là có thể chơi được game.



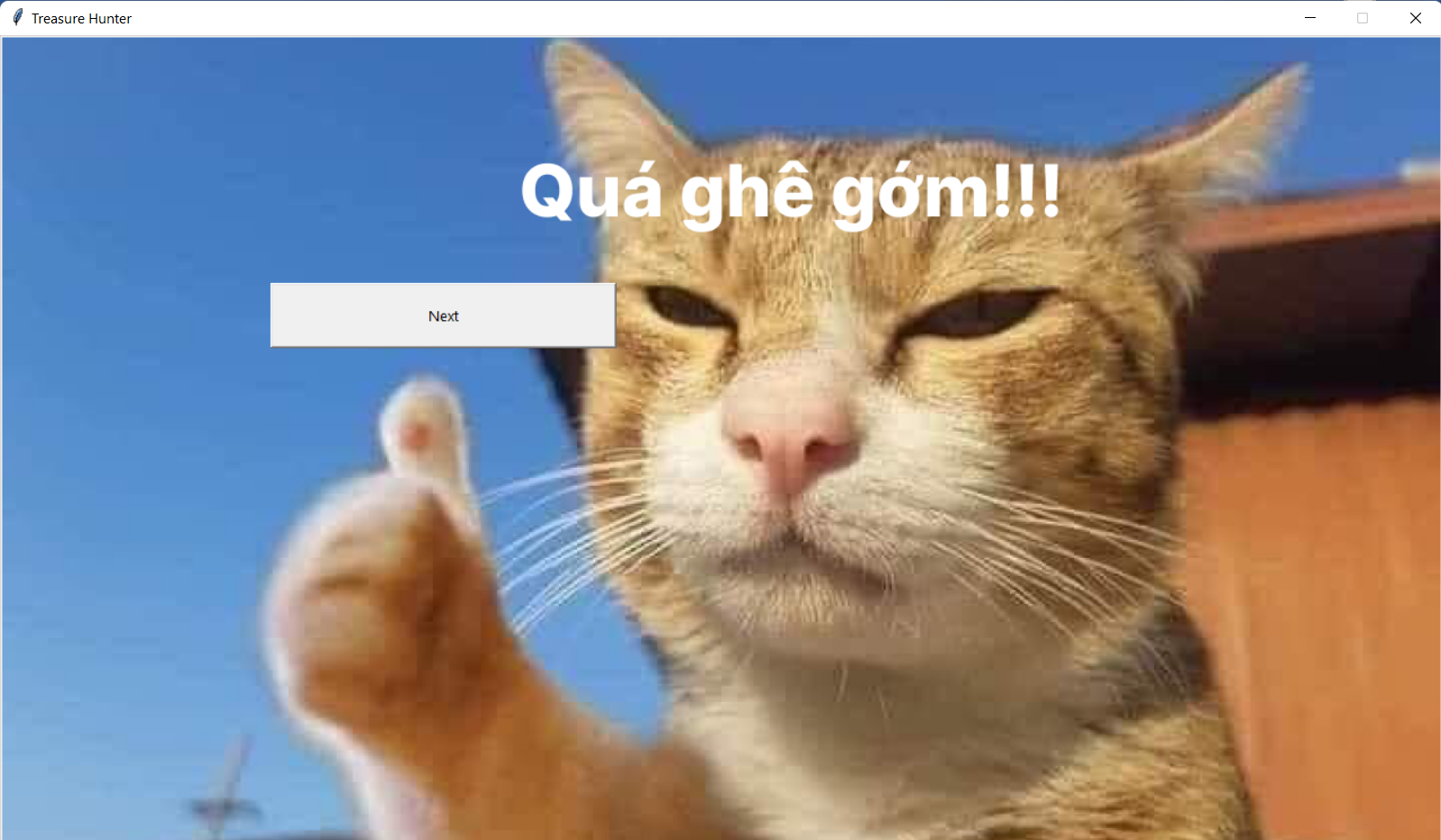
Đây là giao diện câu hỏi đầu tiên khi bạn click button Start Game. Tới đây là lời chào đầu tiên và tiếp theo bạn chỉ cần nhấn **Play** là chơi được ngay.



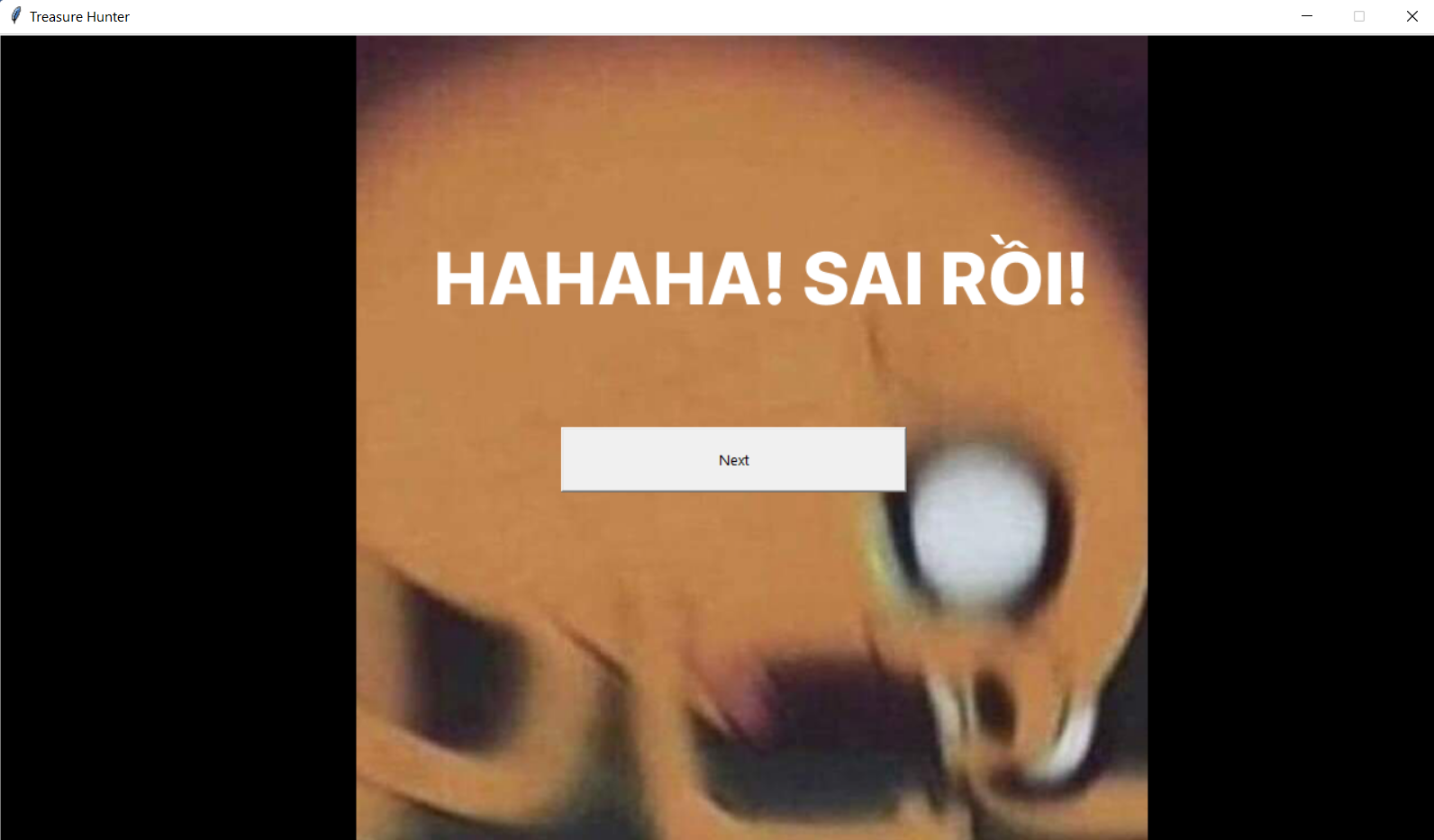
Đây là màn chơi đầu tiên, để chơi chỉ cần click vào đáp án mà bạn cho là đúng.



Nếu bạn chọn đáp án đúng sẽ hiển thị ra giao diện này. Nhấn **Next** để tiếp tục màn chơi tiếp theo.



Nếu bạn chọn đáp án sai thì sẽ hiển thị ra gieo diện này. Nhấn **Next** để chơi tiếp.



Đây là giao diện cuối cùng khi bạn hoàn thành tất cả những màn chơi. Nếu muốn chơi lại thì bạn nhân chọn **Replay**, còn không muốn thì bạn chỉ cần nhấn **Return to Menu** hoặc tắt hẳn game.



Trung bình học kì theo học kì

# CHƯƠNG 4: ĐÁNH GIÁ GAME

# Tổng kết

# 1. **Khả thi về kỹ thuật**

-Thành viên tham gia dự án: 4 thành viên

- Thời gian thực hiện: 2 tuần

- Kích thước hệ thống: trung bình

# 2. **Công cụ phát triển**

- Hệ điều hành Windows 11

- Công cụ lập trình: Python 3, tkinter, figma

- Công cụ khác: Word, PPT…

# 3. **Bảng tổng kết**

| **STT** | **Rủi ro** | **Mức độ** | **Mô tả đánh giá** | **Chú thích** |
| --- | --- | --- | --- | --- |
| 1 | Câu hỏi dễ gây ức chế người chơi | Thấp | Những câu hỏi đặc biệt | Những câu hỏi xoay quanh nhiều lĩnh vực như thể thao, lịch sử, địa lý, … |
| 2 | Người chơi nhìn giao diện của game sẽ thấy không thoải mái | Cao | Giao diện 1280\*720 | Nếu để giao diện full màn hình thì chưa nghĩ ra những chi tiết để khỏa lấp những khoảng trống còn thiếu trên màn hình giao diện. |
| 3 | Người chơi sẽ cảm thấy hơi nhàm chán | Cao | Khá ít câu hỏi (5 câu hỏi) | Tốn khá nhiều thời gian để tạo them giao diện cho từng câu hỏi cũng như soạn câu hỏi. |

1. Yêu cầu bảo mật: Thấp

Người chơi chỉ cần nhập tên game là có thể vào chơi, không có save game vì game chơi khá nhanh ( tầm 30 giây là có thể phá đảo game).

# **Ngôn ngữ viết**

* Python
* Giao diện Tkinter
* Web design Figma

# KẾT LUẬN

* Ưu điểm:
* Các chức năng cơ bản của chương trình đã hoàn thành.
* Giao diện thân thiện, dễ sử dụng.
* Cài đặt trên nhiều hệ điều hành
* Nhược điểm
* Game có khá ít màn chơi (5 màn)
* Thời gian hạn chế và nhóm chỉ có 4 thành viên cho nên trong quá trình xây dựng phần mềm còn gặp nhiều khó khăn.
* Chưa có kinh nghiệm nhiều trong việc lập trình game nên sẽ có vài lỗi khi chơi game
* Hầu hết các chức năng hệ thống đã hoàn chỉnh.

* Kinh nghiệm thu được:
* Hiểu được nhiều kinh nghiệm trong khoản lập trình 1 ứng dụng nó sẽ như thế nào.
* Có cơ hội để nâng cao kiến thức về mảng làm giao diện.
* Hướng giải quyết:

Phần mềm được viết với các chức năng cơ bản và có thể dựa vào nó để nâng cấp lên thêm nhiều chế độ chơi hơn có thể cập nhật thêm nhiều câu hỏi không chỉ về lịch sử mà còn có cả toán, địa lý, hóa, …

# BẢNG PHÂN CÔNG NHÓM 4 MEMBERS

|  |  |  |
| --- | --- | --- |
| **MSSV** | **Họ tên** | **Nhiệm vụ** |
| 2000003905 | Nguyễn Phan Bảo Quý | Hỗ trợ bạn viết code và viết báo cáo |
| 2000000050 | Vương Minh Trí | Hỗ trợ bạn viết code và viết báo cáo |
| 2000005153 | Phạm Hồ Tấn Đạt | Viết code cho những trang giao diện liên kết được và tạo thành 1 ứng dụng game hoàn chỉnh. |
| 2000004968 | Trần Gia Bảo | Tìm kiếm câu hỏi và tạo những trang giao diện từ figma. |

# NGUỒN THAM KHẢO:

Link Github của nhóm 4 Members tụi em: