BDRPC187**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

TRƯỜNG ĐẠI HỌC SÀI GÒN



KHOA CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI**

**TÌM HIỂU CÔNG NGHỆ LẬP TRÌNH PYTHON**

**CROSS-PLATFORM FRAMEWORK TKINTER, PYQT5**

**Học phần : Các công nghệ lập trình hiện đại**

**Nhóm :** 17

**Sinh viên thực hiện :** Nguyễn Minh Trung – 3119410468

Giảng viên hướng dẫn : Phạm Thi Vương

**Tháng 9 năm 2023**

**LỜI CẢM ƠN**

Lời đầu tiên em xin chân thành cảm ơn các Thầy, Cô trong khoa Công nghệ Thông tin cùng toàn thể các Thầy, Cô trường Đại học Sài Gòn đã giảng dạy, truyền đạt cho em những kiến thức quý báu và tạo điều kiện thuận lợi cho em trong suốt quá trình học tập tại trường cũng như trong thời gian thực hiện đồ án này.

Đặc biệt, em muốn gửi lời cảm ơn tới Thầy Phạm Thi Vương giảng viên trực tiếp hướng dẫn tận tình chỉ bảo giúp em khắc phục những khó khăn, thiếu sót để có thể hoàn thành các phần trong đồ án từ tìm hiểu lý thuyết cho tới thực hành sử dụng công cụ và tính ứng dụng của công nghệ này.

Xin cảm ơn các bạn bè, người thân đã luôn đồng hành cùng tôi trong suốt thời gian qua và cho tôi chỗ dựa vững chắc để tôi đạt được những kết quả như ngày hôm nay. Với hiểu biết tìm tòi của bản thân và sự chỉ bảo hướng dẫn tận tình của giảng viên em đã cố gắng hoàn thành đồ án một cách tốt nhất có thể nhưng cũng không thể tránh được thiếu sót.

Kính mong nhận được sự đóng góp ý kiến từ thầy cô để em có thể nâng cao cũng như bổ sung thêm kiến thức cho bản thân, hoàn thiện đồ án với một kết quả tốt và hoản chỉnh hơn.

Em xin chân thành cảm ơn!

**LỜI MỞ ĐẦU**

Với việc công nghệ thay đổi liên tục và được nâng cấp qua từng ngày thì các ngôn ngữ lập trình đóng vai trò vô cùng quan trọng như một nền tảng của mọi đổi mới. Chúng là nền tảng của một loạt các công cụ được sử dụng bởi mọi doanh nghiệp trong mọi lĩnh vực. Vì vậy có thể nói nếu không có những ngôn ngữ lập trình này thì sẽ không có các công nghệ hiện đại như hiện nay.

Hiện nay có hơn 700 loại ngôn ngữ lập trình trên thế giới, vì vậy việc tiếp cận những kiến thức và ngôn ngữ lập trình mới là điều cần thiết.

Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình và là ngôn ngữ lập trình dễ học; được dùng rộng rãi trong phát triển trí tuệ nhân tạo. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu.

Tkinter là thư viện GUI tiêu chuẩn cho Python. Tkinter trong Python cung cấp một cách nhanh chóng và dễ dàng để tạo các ứng dụng GUI. Tkinter cung cấp giao diện hướng đối tượng cho bộ công cụ GUI.

PyQt5 là bộ công cụ GUI đa nền tảng, một bộ liên kết python cho Qt v5. Người ta có thể phát triển một ứng dụng máy tính tương tác rất dễ dàng nhờ các công cụ và sự đơn giản do thư viện này cung cấp.

Đồ án gồm các nội dung sau :

**Chương I.** Ngôn ngữ Python

**Chương II.** Framework Tkinter

**Chương III.** Framework PyQt5

**Chương IV.** API

**Chương V.** Cross-Platform

**Chương VI.** Áp dụng GUI Framework vào đề tài

**Chương VII.** So sánh giữa 2 GUI framework Tkinter và PyQt5

**MỤC LỤC**

[**CHƯƠNG I. NGÔN NGỮ PYTHON 7**](#_Toc148178146)

[**1. Lịch sử 8**](#_Toc148178147)

[**2. Tính năng 8**](#_Toc148178148)

[**3. Các cú pháp 9**](#_Toc148178149)

[3.1. Biến và kiểu dữ liệu 9](#_Toc148178150)

[3.2. Câu lệnh điều kiện 9](#_Toc148178151)

[3.3. Vòng lặp 9](#_Toc148178152)

[3.4. Hàm 10](#_Toc148178153)

[3.5. Chuỗi 10](#_Toc148178154)

[3.6. Danh sách 10](#_Toc148178155)

[3.7. Import Module 10](#_Toc148178156)

[3.8. Xử lý ngoại lệ 11](#_Toc148178157)

[**4. Thư viện 11**](#_Toc148178158)

[**5. Mục đích và ứng dụng 12**](#_Toc148178159)

[**6. Ưu - nhược điểm 13**](#_Toc148178160)

[6.1. Ưu điểm 13](#_Toc148178161)

[6.2. Nhược điểm 14](#_Toc148178162)

[**CHƯƠNG II. FRAMEWORK TKINTER 15**](#_Toc148178163)

[**1. Giới thiệu 15**](#_Toc148178164)

[1.1. Lịch sử 15](#_Toc148178165)

[1.2. Tổng quát 15](#_Toc148178166)

[1.3. Widget 16](#_Toc148178167)

[1.4. Phương thức bố cục 17](#_Toc148178168)

[**Phương thức pack() 17**](#_Toc148178169)

[**Phương thức grid() 18**](#_Toc148178170)

[**Phương thức place() 18**](#_Toc148178171)

[**2. Ưu - nhược điểm 19**](#_Toc148178172)

[2.1. Ưu điểm 19](#_Toc148178173)

[2.2. Nhược điểm 20](#_Toc148178174)

[**3. Cài đặt 20**](#_Toc148178175)

[3.1. Cài đặt Tkinter trên hệ điều hành Windows 20](#_Toc148178176)

[3.2. Cài đặt Tkinter trên hệ điều hành macOS 21](#_Toc148178177)

[3.3. Cài đặt Tkinter trên hệ điều hành Linux 21](#_Toc148178178)

[**CHƯƠNG III. FRAMEWORK PYQT5 22**](#_Toc148178179)

[**1. Giới thiệu 22**](#_Toc148178180)

[1.1. Lịch sử 22](#_Toc148178181)

[1.2. Tổng quát 23](#_Toc148178182)

[1.3. Widget 24](#_Toc148178183)

[1.4. Class module 25](#_Toc148178184)

[**2. Ưu - nhược điểm 27**](#_Toc148178185)

[2.1. Ưu điểm 27](#_Toc148178186)

[2.2. Nhược điểm 27](#_Toc148178187)

[**3. Cài đặt 28**](#_Toc148178188)

[3.1. Cài đặt trên Windows 28](#_Toc148178189)

[3.2. Cài đặt trên Linux 28](#_Toc148178190)

[**CHƯƠNG IV. API 29**](#_Toc148178191)

[**1. Tổng quát 29**](#_Toc148178192)

[**2. Cách hoạt động 30**](#_Toc148178193)

[**3. Các loại API 31**](#_Toc148178194)

[**3.1. Dựa trên phân ngành nhỏ trong ngành công nghệ thông tin 31**](#_Toc148178195)

[**3.2. Dựa trên quyền hạn truy cập 32**](#_Toc148178196)

[**4. Ưu - nhược điểm 32**](#_Toc148178197)

[4.1. Ưu điểm 32](#_Toc148178198)

[4.2. Nhược điểm 33](#_Toc148178199)

[**5. Web API 33**](#_Toc148178200)

[5.1. Những tính năng nổi bật 34](#_Toc148178201)

[5.2. Hoạt động 35](#_Toc148178202)

[5.3. Ưu - nhược điểm 35](#_Toc148178203)

[**6. Tính bảo mật cho API 36**](#_Toc148178204)

[**CHƯƠNG V. CROSS-PLATFORM 37**](#_Toc148178205)

[**1. Khái niệm thuật ngữ Cross-platfrom 37**](#_Toc148178206)

[**2. Lợi ích 38**](#_Toc148178207)

[**3. Mục tiêu 38**](#_Toc148178208)

[**4. Ưu - nhược điểm 38**](#_Toc148178209)

[4.1. Ưu điểm 38](#_Toc148178210)

[4.2. Nhược điểm 39](#_Toc148178211)

[**CHƯƠNG VI. ÁP DỤNG GUI FRAMEWORK VÀO ĐỀ TÀI 40**](#_Toc148178212)

[**1. Tổng quan đề tài 40**](#_Toc148178213)

[1.1. Gới thiệu chung 40](#_Toc148178214)

[1.2. Lí do chọn đề tài 41](#_Toc148178215)

[1.3. Mục đích, đóng góp và lợi ích 41](#_Toc148178216)

[1.4. Nội dung đề tài 41](#_Toc148178217)

[**2. Cài đặt trên hệ điều hành Windows 44**](#_Toc148178218)

[2.1. Cài đặt môi trường 44](#_Toc148178219)

[2.2. Thiết kế bố cục giao diện cơ bản 45](#_Toc148178220)

[**2.2.1. Tkinter 45**](#_Toc148178221)

[**2.2.2. PyQt5 49**](#_Toc148178222)

[2.3. Thực hiện chức năng 53](#_Toc148178223)

[**2.3.1. API và SQL 53**](#_Toc148178224)

[**2.3.2. Tkinter 54**](#_Toc148178225)

[**Tìm kiếm 54**](#_Toc148178226)

[**Hiển thị 55**](#_Toc148178227)

[**Yêu thích 59**](#_Toc148178228)

[**2.3.3. PyQt5 61**](#_Toc148178229)

[**Tìm kiếm 61**](#_Toc148178230)

[**Hiển thị 63**](#_Toc148178231)

[**Yêu thích 65**](#_Toc148178232)

[**3. Cài đặt trên hệ điều hành Linux 67**](#_Toc148178233)

[3.1. Cài đặt môi trường 67](#_Toc148178234)

[3.2. Cài đặt cơ sở dữ liệu 68](#_Toc148178235)

[**CHƯƠNG VII. SO SÁNH GIỮA 2 GUI FRAMEWORK TKINTER VÀ PYQT5 70**](#_Toc148178236)

# **CHƯƠNG I. NGÔN NGỮ PYTHON**

Python là ngôn ngữ lập trình máy tính bậc cao thường được sử dụng để xây dựng trang web và phần mềm, tự động hóa các tác vụ và tiến hành phân tích dữ liệu. Python là ngôn ngữ có mục đích chung, nghĩa là nó có thể được sử dụng để tạo nhiều chương trình khác nhau và không chuyên biệt cho bất kỳ vấn đề cụ thể nào.

Tính linh hoạt này, cùng với sự thân thiện với người mới bắt đầu, đã khiến nó trở thành một trong những ngôn ngữ lập trình được sử dụng nhiều nhất hiện nay.

## **1. Lịch sử**

Python đã được hình thành vào cuối những năm 1980 và được bắt đầu thực hiện vào tháng 12/1989 bởi Guido van Rossum tại CWI tại Hà Lan như là người kế thừa của ngôn ngữ ABC (tự lấy cảm hứng từ SETL) có khả năng xử lý ngoại lệ và giao tiếp với hệ điều hành Amoeba. Van Rossum là tác giả chính của Python, và vai trò trung tâm của ông tiếp tục trong việc quyết định hướng phát triển của Python được phản ánh trong tiêu đề mà cộng đồng Python dành cho ông “Độc tài nhân từ cho cuộc sống” ( Benevolent dictator for life - BDFL ).

Python 2. 0 được phát hành vào ngày 16/10/2000, với nhiều tính năng chính mới bao gồm một bộ dọn rác đầy đủ và hỗ trợ Unicode. Với phiên bản này, quá trình phát triển đã được thay đổi và trở thành minh bạch hơn và được cộng đồng ủng hộ.

Python 3. 0 (còn được gọi là Python 3000 hoặc Py3k), một bản phát hành

lớn, không tương thích ngược, được phát hành vào ngày 03/12/2008 sau một thời

Python 3. 0 (còn được gọi là Python 3000 hoặc Py3k), một bản phát hành lớn, không tương thích ngược, được phát hành vào ngày 03/12/2008 sau một thời gian dài thử nghiệm. Nhiều trong số các tính năng chính của nó đã được điều chỉnh để tương thích ngược với Python 2. 6 và 2. 7.

Python 3.9.2 và 3.8.8 được xúc tiến vì tất cả các phiên bản trước của Python (bao gồm cả 2.7) gặp một số vấn đề bảo mật, có thể dẫn đến thực thị mã từ xa và "đầu độc" bộ nhớ đệm.

Trong năm 2022, Python 3.10.4 và 3.9.12 được xúc tiến cùng với 3.8.13 và 3.7.13, nguyên nhân là do một vài vấn đề về bảo mật. Khi Python 3.9.13 được phát hành vào tháng Năm năm 2022, loạt phiên bản 3.9 (cùng với loạt 3.8 và 3.7) được thông báo rằng sẽ chỉ nhận được các bản vá bảo mật trong tương lai.

## **2. Tính năng**

* Python hỗ trợ nhiều nền tảng khác nhau (Windows, Mac, Linux, Raspberry Pi, etc). Python có cú pháp đơn giản, dễ đọc hiểu và rất gần gũi với tiếng Anh.
* Cú pháp của Python giúp lập trình viên sử dụng ít dòng code để lập trình cùng một thuật toán hơn so với các ngôn ngữ lập trình khác.
* Python sử dụng trình thông dịch để thực thi các dòng code. Do đó, những dòng code có thể được thực thi ngay lập tức mà không cần biên dịch toàn bộ chương trình. Như vậy giúp chúng ta kiểm tra code nhanh hơn.
* Python cũng hỗ trợ hàm, thủ tục, hay kể cả lập trình hướng đối tượng.

## **3. Mục đích và ứng dụng**

* **Phát triển Web:** Frameworks như Django và Flask giúp xây dựng ứng dụng web một cách nhanh chóng và dễ dàng.Python cũng được sử dụng để phát triển các ứng dụng web front-end thông qua các thư viện như Flask và Django REST framework.
* **Xử lý Dữ liệu và Phân Tích:** Python là một công cụ mạnh mẽ cho xử lý và phân tích dữ liệu với các thư viện như NumPy, pandas, SciPy, và Matplotlib.Jupyter Notebook cung cấp môi trường tương tác cho phân tích dữ liệu và viết báo cáo.
* **Trí Tuệ Nhân Tạo (AI) và Học Máy (Machine Learning):** Python là một trong những ngôn ngữ hàng đầu cho phát triển và triển khai các mô hình học máy và trí tuệ nhân tạo.Thư viện như TensorFlow, PyTorch, scikit-learn và Keras hỗ trợ trong việc xây dựng các dự án AI và ML.
* **Sử dụng trong Lĩnh Vực Khoa Học và Nghiên Cứu:** Python là một công cụ quan trọng trong các lĩnh vực khoa học, nghiên cứu, và tính toán khoa học. Các thư viện như NumPy và SciPy giúp trong tính toán và phân tích dữ liệu khoa học.
* **Phát triển Ứng Dụng Máy Tính Desktop và Game:** Pygame cho phép phát triển các trò chơi sử dụng Python.Thư viện như PyQt và Tkinter giúp xây dựng ứng dụng máy tính desktop đa nền tảng.
* **Điều Khiển Thiết Bị Vật Lý và IoT (Internet of Things):** Python có thư viện GPIO cho Raspberry Pi để điều khiển các thiết bị vật lý.Nó được sử dụng rộng rãi trong các dự án IoT và điều khiển thiết bị nhúng.
* **Phát triển Ứng Dụng Mạng và Hệ Thống:** Python có thư viện và framework cho phát triển ứng dụng mạng và hệ thống như Twisted và Ansible.
* **Thiết kế Đồ Họa và Trình Duyệt 3D:** Python có thư viện như PyOpenGL và Blender để tạo đồ họa 3D và hoạt ảnh.
* **Trình Bày Dữ Liệu và Báo Cáo:** Python có các thư viện như Matplotlib và Seaborn để tạo biểu đồ và đồ thị.Jupyter Notebook cho phép tạo báo cáo tương tác với mã và kết quả.
* **Automation và Scripting:** Python thường được sử dụng để tự động hóa các tác vụ, xử lý tệp, và tạo các script cho hệ thống và quy trình làm việc hàng ngày.

## **4. Ưu - nhược điểm**

### **4.1. Ưu điểm**

* **Dễ Đọc và Dễ Viết:** Python có cú pháp đơn giản và gần giống ngôn ngữ tự nhiên, giúp mã nguồn dễ đọc, dễ viết và dễ hiểu. Điều này làm cho Python rất thích hợp cho người mới học lập trình.
* **Cộng Đồng Lớn:** Python có một cộng đồng đông đảo và nhiệt tình, với nhiều tài liệu, thư viện, và hỗ trợ trực tuyến. Bất kỳ vấn đề nào bạn gặp phải, bạn cũng có thể tìm thấy giải pháp dễ dàng.
* **Đa Năng:** Python được sử dụng trong nhiều lĩnh vực khác nhau, bao gồm phát triển web, xử lý dữ liệu, trí tuệ nhân tạo, học máy, IoT, game, thiết kế đồ họa, và nhiều ứng dụng khác.
* **Thư Viện và Frameworks Đa Dạng:** Python có một hệ sinh thái phong phú của các thư viện và frameworks. Ví dụ, Django và Flask cho phát triển web, NumPy và pandas cho xử lý dữ liệu, TensorFlow và PyTorch cho học máy, Matplotlib cho vẽ đồ thị, và nhiều thư viện khác.
* **Tương Tác:** Python hỗ trợ mô hình tương tác thông qua trình thông dịch (interpreter) nơi bạn có thể thử nghiệm và kiểm tra mã một cách nhanh chóng. Jupyter Notebook là một ví dụ tiêu biểu cho môi trường này.
* **Hệ Sinh Thái Lớn:** Python có một loạt các công cụ phát triển và trình quản lý gói (package manager) như pip, virtualenv, và conda, giúp quản lý dự án và thư viện dễ dàng.
* **Tích Hợp Dễ Dàng:** Python dễ dàng tích hợp với các ngôn ngữ khác như C, C++, và Java. Điều này cho phép bạn sử dụng các thư viện chuyên biệt khi cần.
* **Hỗ Trợ Đa Nền Tảng:** Python là một ngôn ngữ đa nền tảng, có sẵn trên nhiều hệ điều hành như Windows, macOS, và Linux.
* **Mã Mở và Miễn Phí:** Python là một ngôn ngữ mã nguồn mở và hoàn toàn miễn phí. Bạn có thể sử dụng, phân phối và sửa đổi mã nguồn Python mà không cần trả bất kỳ phí nào.
* **Mã Mở và Miễn Phí:** Python là một ngôn ngữ mã nguồn mở và hoàn toàn miễn phí. Bạn có thể sử dụng, phân phối và sửa đổi mã nguồn Python mà không cần trả bất kỳ phí nào.

### **4.2. Nhược điểm**

* **Chậm:** Python thường không nhanh bằng các ngôn ngữ lập trình tĩnh như C++ hoặc Java. Điều này có thể là một vấn đề khi bạn cần hiệu suất cao cho các ứng dụng yêu cầu xử lý số lớn hoặc thời gian thực.
* **Ghi Chú Chậm (Global Interpreter Lock - GIL):** GIL là một khóa tự động bên trong CPython (phiên bản Python chính thống) để đảm bảo rằng chỉ có một luồng được thực thi tại một thời điểm. Điều này có thể gây ra vấn đề khi bạn muốn tận dụng lợi ích của đa luồng (multithreading) để tăng hiệu suất cho các ứng dụng đa nhiệm.
* **Khoản Thời Gian Khởi Động Chậm:** Ứng dụng Python thường có khoản thời gian khởi động chậm hơn so với các ngôn ngữ khác, do việc tải và biên dịch mã nguồn.
* **Kích Thước Ứng Dụng Lớn:** Ứng dụng Python có thể có kích thước lớn hơn so với các ứng dụng tương tự viết bằng C++ hoặc Go, do sự bao gồm của trình thông dịch Python và thư viện chuẩn.
* **Khả Năng Điều Khiển Cấu Hình Hệ Thống Hạn Chế:** Python không phải là lựa chọn tốt cho các ứng dụng nhúng hoặc hệ thống cần kiểm soát phần cứng một cách trực tiếp.
* **Thư Viện Của Cộng Đồng Không Đồng Đều:** Mặc dù Python có nhiều thư viện và frameworks tốt, nhưng không phải lúc nào chúng cũng đồng đều về mức độ phát triển và hỗ trợ.
* **Phiên Bản Python không tương thích ngược:** Có sự thay đổi trong cú pháp và hành vi của Python giữa các phiên bản chính, do đó, việc nâng cấp từ một phiên bản cũ lên một phiên bản mới có thể gây ra sự khắc phục mã nguồn.
* **Khó Sử Dụng cho Ứng Dụng Mobile:** Python không phải là ngôn ngữ phát triển chính cho ứng dụng di động và có hạn chế trong việc phát triển ứng dụng di động chuyên nghiệp.

# **CHƯƠNG II. FRAMEWORK TKINTER**

## **1. Giới thiệu**

### **1.1. Lịch sử**

* **Sự Xuất Hiện Ban Đầu (1988):** Tkinter dựa trên thư viện Tk (Toolkit) ban đầu được phát triển bởi John Ousterhout. Tkinter bắt đầu như một giao diện Tcl/Tk (Tool Command Language/ToolKit) cho ngôn ngữ lập trình Tcl.
* **Tcl/Tk và Phát Triển Ban Đầu:** Tcl/Tk đã thu hút sự quan tâm từ cộng đồng phát triển phần mềm ngay từ đầu như một framework đa nền tảng cho việc phát triển GUI. Tkinter xuất hiện khi Python chưa phải là một ngôn ngữ lập trình phát triển GUI chính thống và được tạo ra để tạo giao diện người dùng cho Python sử dụng Tk.
* **Tkinter trong Python (1991):** Tkinter được tích hợp chính thức vào Python từ phiên bản Python 1.4, phát hành vào năm 1991. Việc này đánh dấu sự ra đời của Tkinter như một thư viện GUI mặc định cho Python, và từ đó, nó đã trở thành một phần quan trọng của hệ sinh thái Python.
* **Phát Triển và Sự Thay Đổi (Từ Năm 1990 Đến Nay):** Tkinter đã trải qua nhiều phiên bản và cải tiến kể từ khi được tích hợp vào Python. Nó đã được cải tiến để hỗ trợ các tính năng và widgets mới, tương thích với các phiên bản Python mới, và cải thiện hiệu suất và tích hợp.
* **Sự Phổ Biến và Sử Dụng Rộng Rãi:** Tkinter đã trở thành một công cụ quan trọng trong việc phát triển ứng dụng desktop GUI trong Python. Sự đơn giản và tích hợp dễ dàng với Python đã giúp Tkinter phổ biến trong cộng đồng phát triển Python và trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng desktop đơn giản và trung bình.

### **1.2. Tổng quát**

Tkinter là một thư viện trong Python sử dụng để phát triển ứng dụng desktop GUI (Graphical User Interface) đơn giản và trực quan. Tên gốc của Tkinter là "Tk Interface," bởi vì nó là một giao diện Python đối với thư viện Tk, một thư viện GUI được phát triển ban đầu cho ngôn ngữ lập trình Tcl/Tk. Tkinter cung cấp một tập hợp các công cụ và widgets để tạo giao diện người dùng trực quan cho ứng dụng của bạn.

* **Widgets:** Tkinter cung cấp một loạt các widgets như cửa sổ (window), nút (button), hộp văn bản (text box), danh sách (listbox), menu, và nhiều widgets khác để bạn có thể sử dụng trong ứng dụng của mình.
* **Tương Tác Tốt Với Python:** Tkinter tích hợp tốt với Python và cho phép bạn tạo ứng dụng có tính tương tác cao, bao gồm cả xử lý sự kiện như bấm nút, nhập liệu từ bàn phím, và nhiều sự kiện khác.
* **Phù Hợp Cho Ứng Dụng Nhỏ và Trung Bình:** Tkinter thường được sử dụng cho các ứng dụng GUI đơn giản và trung bình. Đối với các ứng dụng lớn hơn và phức tạp hơn, có thể cần sử dụng các thư viện GUI mạnh mẽ hơn.

Tkinter là một công cụ phát triển ứng dụng GUI nhanh chóng và hiệu quả trong Python. Nó thường được sử dụng để phát triển ứng dụng như các công cụ quản lý dữ liệu, ứng dụng cơ sở dữ liệu đơn giản, giao diện người dùng cho ứng dụng máy tính, và nhiều ứng dụng desktop khác.

## **2. Ưu - nhược điểm**

### **2.1. Ưu điểm**

* **Dễ Học và Sử Dụng:** Tkinter có cú pháp đơn giản và dễ đọc, điều này làm cho nó thích hợp cho người mới học lập trình GUI. Một số widgets cơ bản có thể được tạo và tùy chỉnh một cách dễ dàng.
* **Widgets Đa Dạng:** Tkinter cung cấp một loạt các widgets như cửa sổ, nút, hộp văn bản, danh sách, menu, và nhiều widgets khác để bạn có thể sử dụng trong ứng dụng của mình. Điều này giúp bạn xây dựng giao diện người dùng phong phú và đa dạng.
* **Tương Tác Tốt Với Python:** Tkinter tích hợp tốt với Python và cho phép bạn tạo ứng dụng có tính tương tác cao. Bạn có thể dễ dàng thực hiện xử lý sự kiện như bấm nút, nhập liệu từ bàn phím, và nhiều sự kiện khác.
* **Cộng Đồng Hỗ Trợ Rộng Rãi:** Cộng đồng Python rất lớn và nhiệt tình, vì vậy bạn có thể tìm thấy nhiều tài liệu, ví dụ, và hỗ trợ trực tuyến liên quan đến Tkinter.
* **Thư Viện Tiêu Chuẩn:** Tkinter là một thư viện tiêu chuẩn của Python, nghĩa là bạn không cần cài đặt bất kỳ phần mềm bên ngoài nào để bắt đầu phát triển ứng dụng GUI. Nó đi kèm với Python khi bạn cài đặt Python trên máy tính của mình.
* **Đa Nền Tảng:** Tkinter được tích hợp trong Python và hỗ trợ trên nhiều hệ điều hành, bao gồm Windows, macOS và Linux. Điều này đảm bảo rằng ứng dụng bạn phát triển có thể chạy trên nhiều nền tảng mà không cần thay đổi mã nguồn.

### **2.2. Nhược điểm**

* **Đa Nền Tảng :** Tkinter được tích hợp trong Python và hỗ trợ trên nhiều hệ điều hành, bao gồm Windows, macOS và Linux. Điều này đảm bảo rằng ứng dụng bạn phát triển có thể chạy trên nhiều nền tảng mà không cần thay đổi mã nguồn.
* **Hiệu Suất Hạn Chế :** Tkinter có thể không thích hợp cho các ứng dụng đòi hỏi hiệu suất cao, đặc biệt là khi có nhiều phần tử đồ họa hoặc xử lý số lớn. Tkinter có một số hạn chế về hiệu suất do việc sử dụng GIL (Global Interpreter Lock) của Python.
* **Khả Năng Tùy Chỉnh Khó Khăn** : Một số tùy chỉnh giao diện phức tạp hơn trong Tkinter so với các thư viện GUI khác, đặc biệt là trong việc tạo các phần tử giao diện tùy chỉnh.
* **Không Phù Hợp Cho Ứng Dụng Lớn và Phức Tạp :** Tkinter thường không được sử dụng cho các ứng dụng lớn và phức tạp như các ứng dụng đồ họa 3D hoặc ứng dụng chơi video game.
* **Tài Liệu Hạn Chế :** Mặc dù có sẵn tài liệu và nguồn hỗ trợ cho Tkinter, tài liệu không phong phú và chi tiết bằng một số thư viện GUI khác. Điều này có thể làm cho việc tìm hiểu và xử lý các vấn đề phức tạp trở nên khó khăn hơn.

# **CHƯƠNG III. FRAMEWORK PYQT5**

## **1. Giới thiệu**

### **1.1. Lịch sử**

* **Bắt Đầu với PyQt (1996):** PyQt5 là một phần của dự án PyQt, đã bắt đầu từ năm 1996 bởi Phil Thompson. PyQt ban đầu hỗ trợ Qt 1.x và Qt 2.x, và nó đã giúp Python có một cách tiếp cận tốt hơn đối với Qt, một framework đa nền tảng mạnh mẽ cho việc phát triển giao diện người dùng đồ họa.
* **Qt 3 và PyQt3 (2000):** Với sự ra mắt của Qt 3, phiên bản PyQt3 đã được phát hành để hỗ trợ Qt 3. PyQt3 đã cung cấp khả năng phát triển ứng dụng desktop GUI với Qt 3 bằng Python.
* **Phát Triển PyQt4 (2005):** PyQt4 ra mắt vào năm 2005, hỗ trợ cho Qt 4. PyQt4 đã cung cấp nhiều cải tiến và tính năng mới, bao gồm sự hỗ trợ tốt hơn cho các chức năng của Qt 4 và Python 2.x.
* **Sự Thay Đổi Cho Python 3 (2010):** Với sự ra đời của Python 3, PyQt5 đã được phát triển để hỗ trợ Python 3.x. PyQt5 được thiết kế để sử dụng cùng với Qt 5, phiên bản mới nhất của framework Qt.
* **PyQt5 (2016):** PyQt5 đã được phát hành vào năm 2016 và trở thành một trong những phiên bản phổ biến và được sử dụng rộng rãi nhất của PyQt. Nó tích hợp tốt với Python 3 và Qt 5, cung cấp khả năng phát triển ứng dụng desktop GUI đa nền tảng và đa dạng.
* **Sự Thành Công và Sự Phát Triển Liên Tục:** PyQt5 đã đạt được sự thành công lớn trong cộng đồng phát triển phần mềm và là một công cụ quan trọng cho việc phát triển ứng dụng desktop GUI trong Python. Sự phát triển của PyQt5 vẫn tiếp tục, với việc cập nhật định kỳ để hỗ trợ phiên bản mới của Qt và Python.

### **1.2. Tổng quát**

PyQt5 là một thư viện Python dùng để phát triển ứng dụng desktop GUI sử dụng framework Qt. Qt là một framework đa nền tảng mạnh mẽ cho việc phát triển giao diện người dùng đồ họa (GUI), được sử dụng rộng rãi trong công nghiệp phần mềm. PyQt5 cho phép bạn sử dụng tất cả các tính năng và widget của Qt trong Python để tạo các ứng dụng desktop đa nền tảng và đa dạng.

* **Được Phát Triển Bởi Riverbank Computing:** PyQt5 được phát triển và duy trì bởi Riverbank Computing. Nó là một phiên bản Python của PyQt, phiên bản Python cho Qt.
* **Hỗ Trợ Cho Đa Nền Tảng :** PyQt5 hỗ trợ đa nền tảng, có sẵn cho Windows, macOS và Linux, giúp bạn phát triển ứng dụng đa nền tảng một cách dễ dàng.
* **Thư Viện Widgets :** PyQt5 cung cấp một loạt các widget đồ họa và công cụ cho việc xây dựng giao diện người dùng đa dạng. Điều này bao gồm nút, hộp văn bản, danh sách, menu, cửa sổ, và nhiều widget khác.
* **Giấy Phép Kéo dài GPL:** PyQt5 có hai phiên bản giấy phép: giấy phép GPL (General Public License) và giấy phép thương mại. Bạn có thể chọn giấy phép phù hợp với dự án của mình.
* **Phát Triển Tùy Chỉnh :** PyQt5 cho phép bạn tạo các widget tùy chỉnh và tạo giao diện người dùng phức tạp với sự tùy chỉnh cao.
* **Hỗ Trợ Cho Qt Designer :** Qt Designer là một công cụ đồ họa cho phép bạn thiết kế giao diện người dùng một cách trực quan và sau đó tạo mã Python tương ứng. PyQt5 hoàn toàn tương thích với Qt Designer.

PyQt5 là một lựa chọn mạnh mẽ cho việc phát triển ứng dụng desktop GUI đa nền tảng với Python. Nó cung cấp khả năng tùy chỉnh cao và tích hợp mạnh mẽ với Python, làm cho nó trở thành một công cụ phát triển GUI phổ biến trong cộng đồng phát triển phần mềm.

### **1.3. Widget**

|  |  |
| --- | --- |
| QLabel | Widget này hiển thị văn bản hoặc hình ảnh tĩnh. Nó thường được sử dụng để hiển thị nhãn cho các phần tử khác trong giao diện người dùng. |
| QPushButton | Nút này được sử dụng để tạo các nút bấm mà người dùng có thể nhấn để thực hiện các hành động hoặc chức năng. |
| QLineEdit | Widget này cho phép người dùng nhập dữ liệu văn bản đơn giản. Nó thường được sử dụng để lấy đầu vào từ người dùng. |
| QTextEdit | Đây là một widget cho phép hiển thị và chỉnh sửa đoạn văn bản dài hơn. Nó thường được sử dụng cho việc nhập và hiển thị văn bản đa dòng. |
| QCheckBox | Widget này cho phép người dùng chọn hoặc bỏ chọn một hoặc nhiều tùy chọn. Nó thường được sử dụng trong các cài đặt hoặc tùy chọn |
| QRadioButton | Widget này cung cấp một danh sách các tùy chọn, nhưng chỉ cho phép người dùng chọn một tùy chọn trong danh sách đó. |
| QComboBox | Widget này tạo ra một hộp đổ xuống cho phép người dùng chọn từ danh sách các mục. Nó thường được sử dụng khi có một số lượng lớn các tùy chọn để chọn. |
| QSlider | Đây là một thanh trượt cho phép người dùng chọn một giá trị số liên tục trong một phạm vi. |
| QProgressBar | Widget này hiển thị tiến trình hoàn thành một tác vụ nào đó. Nó thường được sử dụng để theo dõi tiến trình của các tác vụ dài hạn |
| QSpinBox và QDoubleSpinBox | Các widget này cho phép người dùng chọn một giá trị số nguyên hoặc số thập phân từ một dãy giá trị cho trước. |
| QDateEdit và QTimeEdit | Các widget này cho phép người dùng chọn ngày và thời gian, tương ứng. |
| QTableWidget và QListView | Các widget này được sử dụng để hiển thị dữ liệu trong bảng hoặc danh sách. |
| QGraphicsView | Widget này cho phép hiển thị và tương tác với đồ họa vector và hình ảnh. |
| QWebView | Widget này cho phép bạn hiển thị và tương tác với trình duyệt web bên trong ứng dụng của bạn. |
| QFileDialog | Widget này cho phép bạn hiển thị hộp thoại lựa chọn tệp và thư mục để người dùng chọn tệp hoặc đường dẫn. |

### **1.4. Class module**

|  |  |
| --- | --- |
| QtCore | Là module bao gồm phần lõi không thuộc chức năng GUI, ví dụ dùng để làm việc với thời gian, file và thư mục, các loại dữ liệu, streams, urls, mime type, threads hoặc processes. |
| QtGui | Bao gồm các class dùng cho việc lập trình giao diện (windowing system integration), event handling, 2D graphics, basic imaging, fonts và text. |
| QtWidgets | Bao gồm các class cho widget, ví dụ : button, hộp thoại, … được sử dụng để tạo nên giao diện người dùng cơ bản nhất. |
| QtMultimedia | Thư viện cho việc sử dụng âm thanh, hình ảnh, camera,… |
| QtBluetooth | Bao gồm các class giúp tìm kiếm và kết nối với các thiết bị có giao tiếp với phần mềm. |
| QtNetwork | Bao gồm các class dùng cho việc lập trình mạng, hỗ trợ lập trình TCP/IP và UDP client , server hỗ trợ việc lập trình mạng. |
| QtPositioning | Bao gồm các class giúp việc hỗ trợ xác định vị. |
| Enginio | Module giúp các client truy cập các Cloud Services của Qt. |
| QtWebSockets | Cung cấp các công cụ cho websocket protocol. |
| QtWebKit | Cung cấp các class dùng cho làm việc với các trình duyệt Web, dựa trên thư viện webkit2. |
| QtWebKitWidgets | Các widget cho webkit. |
| QtXml | Các class dùng cho làm việc với XML file. |
| QtSvg | Dùng cho hiển thị các thành phần của SVG file. |
| QtSql | Cung cấp các class dùng cho việc làm việc với dữ liệu. |
| QtTest | Cung cấp các công cụ cho phép test các đơn vị của ứng dụng với pyqt5. |

## **2. Ưu - nhược điểm**

### **2.1. Ưu điểm**

* **Hiệu Suất Tốt:** PyQt5 cung cấp hiệu suất tốt cho các ứng dụng GUI, và nó được tối ưu hóa để hoạt động nhanh chóng và mượt mà.
* **Tài Liệu Phong Phú:** PyQt5 có một tài liệu phong phú và một cộng đồng nhiệt tình. Có nhiều tài liệu hướng dẫn, ví dụ và tài liệu trực tuyến liên quan đến PyQt5, giúp bạn học và phát triển dễ dàng hơn.
* **Tương Tác Tốt Với Qt Designer:** Qt Designer là một công cụ đồ họa cho phép bạn thiết kế giao diện người dùng một cách trực quan và sau đó tạo mã Python tương ứng. PyQt5 hoàn toàn tương thích với Qt Designer, giúp bạn tạo giao diện một cách nhanh chóng và dễ dàng.
* **Khả Năng Tùy Chỉnh Cao:** PyQt5 cho phép bạn tạo các widget tùy chỉnh và tạo giao diện người dùng phức tạp với sự tùy chỉnh cao. Bạn có thể điều chỉnh giao diện người dùng theo ý muốn và tạo các ứng dụng độc đáo.
* **Widgets Đa Dạng:** PyQt5 cung cấp một loạt các widgets đồ họa và công cụ cho việc xây dựng giao diện người dùng đa dạng. Nó bao gồm nút, hộp văn bản, danh sách, menu, cửa sổ, và nhiều widget khác để bạn có thể sử dụng.

### **2.2. Nhược điểm**

* **Giấy Phép:** PyQt5 có giấy phép kéo dài GPL (General Public License), có nghĩa là nếu bạn sử dụng PyQt5 trong một ứng dụng thương mại, bạn cần xem xét việc trả tiền cho việc sử dụng phiên bản thương mại của PyQt5. Tuy nhiên, có một phiên bản PyQt5 dưới dạng giấy phép GPL và giấy phép thương mại, cho phép bạn lựa chọn giấy phép phù hợp với dự án của bạn.
* **Khó Học:** PyQt5 có độ phức tạp cao hơn so với Tkinter và một số thư viện GUI Python khác, đặc biệt đối với người mới học lập trình GUI. Học PyQt5 có thể đòi hỏi thời gian và kiên nhẫn.
* **Kích Cỡ Lớn:** Ứng dụng PyQt5 có thể có kích thước lớn hơn so với các ứng dụng được xây dựng bằng các thư viện khác do thư viện PyQt5 và Qt cơ bản có kích thước khá lớn.
* **Phụ Thuộc Vào Thư Viện Bên Ngoài:** PyQt5 phụ thuộc vào framework Qt, và việc cài đặt và quản lý các phụ thuộc này có thể gây khó khăn trong một số trường hợp.

# **CHƯƠNG IV. WEB API**

Web API hay ASP.NET Web API là một framework dùng để xây dựng và lập trình các dịch vụ web HTTP. Nó có dạng là một RESTful API hiện đại, hội tụ đủ các điều kiện của REST cũng như các tiêu chuẩn tương tự, được tối ưu cho các dịch vụ trực tuyến cũng như ứng dụng web hiện nay. Web API sở hữu những ưu điểm vượt trội mà một API cũ có cùng nhiều ưu điểm mới tuyệt vời, giúp nó nhận được nhiều sự tin tưởng từ các nhà phát triển và lập trình website.



Web API và ứng dụng trong thiết kế website hiện nay.

### **1. Những tính năng nổi bật**

* **Tự động hóa sản phẩm**

Đối với Web API, sẽ giúp người dùng có thể dễ dàng tự động quản lý được công việc. Cập nhật được luồng công việc tạo hiệu quả tốt hơn trong công việc.

* **Tích hợp linh động**

API cho phép lấy nội dung ở bất kỳ Website hay ứng dụng nào đó một cách dễ dàng, khiến trải nghiệm người dùng được tăng lên. API giúp các công ty chia sẻ được những thông tin được chọn và tránh được những yêu cầu không mong muốn xảy ra.

* **Cập nhật thông tin theo thời gian thực**

API giúp thay đổi và cập nhật những thông tin mới theo thời gian thực. Công nghệ này sẽ giúp những thông tin truyền đi tốt hơn, chính xác hơn và dịch vụ cung cấp cũng được linh hoạt hơn.

### **2. Hoạt động**

Khi website thực hiện một lệnh API để lấy thông tin thì nó sẽ trả về một nội dung dạng JSON hoặc XML. Tuy nhiên, 4 bước dưới đây sẽ giúp bạn hiểu rõ hơn về cách hoạt động lấy dữ liệu thông qua Web API.

1. Trước tiên, xây dựng URL API để gửi tới máy chủ cung cấp nội dung, dịch vụ thông qua giao thức HTTP hoặc HTTPS.
2. Sau khi nhận được thông tin, phía máy chủ cung cấp, ứng dụng nguồn sẽ thực hiện kiểm tra xác thực nếu có và tìm tới nguồn nội dung phù hợp để tạo nội dung trả về phù hợp nhất.
3. Lúc này phía server sẽ gửi lại thông tin theo định dạng JSON hoặc XML thông qua giao thức HTTP hoặc HTTPS.
4. Phía website yêu cầu sẽ phân tích các dữ liệu JSON/SML được gửi tới để thực hiện tiếp các hoạt động như lưu dữ liệu xuống cơ sở dữ liệu và hiển thị thông tin đó ra ngoài cho người dùng đọc.

### **3. Ưu - nhược điểm**

* **Ưu điểm**
* Web API được sử dụng khá rộng rãi ở trên các ứng dụng như: Desktop, mobile và cả ứng dụng ở Website.
* Linh hoạt đối với các dạng dữ liệu trả về Client: Json, XML hay những định dạng khác nữa.
* Dễ dàng xây dựng được HTTP service: URI, URI, request/response headers, caching, versioning, content formats và cả host trong ứng dụng.
* Với mã nguồn mở có thể giúp hỗ trợ những chức năng của Restful một cách đầy đủ.
* Hỗ trợ về thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, dependency injection, unit test.
* Giao tiếp 2 chiều được xác nhận, vì vậy các giao dịch có thể đảm bảo được độ tin cậy cao hơn.
* **Nhược điểm**
* Web API chưa được gọi là Restful Service bởi nó chỉ mới hỗ trợ mặc định Get, Post.
* Nếu muốn sử dụng tốt nhất bạn cần có kiến thức và am hiểu thật sự về backend.
* Khá mất thời gian cho việc phát triển cũng như nâng cấp, vận hành.
* Hệ thống có thể bị tấn công nếu như không giới hạn chức năng hay điều kiện.

# **CHƯƠNG V. CROSS-PLATFORM**

## **1. Khái niệm thuật ngữ Cross-platfrom**

Cross-platform là một khái niệm trong lĩnh vực phát triển phần mềm, đề cập đến khả năng của một ứng dụng hoặc sản phẩm làm việc trên nhiều hệ điều hành hoặc nền tảng khác nhau mà không cần phải viết lại mã nguồn hoàn toàn cho mỗi nền tảng. Mục tiêu của việc phát triển cross-platform là tối ưu hóa sự chia sẻ mã nguồn và tài nguyên để giảm công sức và thời gian phát triển, cũng như để đảm bảo tính nhất quán và khả năng tương thích trên nhiều nền tảng.

**Một số cách để đạt được tính cross-platform trong phát triển phần mềm:**

* Sử dụng ngôn ngữ lập trình đa nền tảng: Sử dụng các ngôn ngữ lập trình như JavaScript, Python, hoặc C# để viết mã nguồn mà có thể chạy trên nhiều hệ điều hành hoặc nền tảng khác nhau
* Sử dụng framework và thư viện đa nền tảng: Sử dụng các framework và thư viện như React Native, Flutter, Xamarin, Electron, và Qt để xây dựng ứng dụng hoạt động trên nhiều nền tảng.
* Sử dụng công cụ cross-compiler: Sử dụng các công cụ cross-compiler để biên dịch mã nguồn từ một nền tảng sang nền tảng khác mà không cần sửa đổi mã nguồn gốc.
* Virtualization và containerization: Sử dụng công nghệ ảo hóa và containerization như Docker để chạy ứng dụng trong môi trường cô lập trên nhiều hệ điều hành.

## **2. Lợi ích**

* **Phát triển ứng dụng nhanh hơn:** Việc sử dụng một bản code cho các nền tảng khác nhau sẽ loại bỏ việc nhà phát triển phải viết lại code từ đầu và cho phép tái sử dụng, giúp cắt giảm thời gian, tự động tăng tốc toàn bộ quá trình phát triển ứng dụng.
* **Tiếp cận thị trường lớn hơn:** Ứng dụng đa nền tảng mang lại sự linh hoạt trên nhiều nền tảng như iOS, Android, Windows,... Do đó, doanh nghiệp có phạm vi tiếp cận đối tượng rộng hơn.
* **Giảm chi phí phát triển:** Hầu hết các doanh nghiệp chưa tự xây dựng được ứng dụng vì chi phí. Tuy nhiên, phát triển Cross Platform App ít tốn kém vì chỉ sử dụng một bản code cho nhiều nền tảng.

## **3. Mục tiêu**

Mục tiêu chính của cross-platform là tạo ra các ứng dụng hoặc sản phẩm có khả năng hoạt động trên nhiều nền tảng khác nhau mà không cần phải viết lại mã nguồn hoàn toàn cho từng nền tảng.

Tối ưu hóa quá trình phát triển phần mềm, giúp tiết kiệm thời gian, nguồn lực, và tiền bạc, đồng thời cung cấp ứng dụng nhất quán và tiếp cận rộng rãi cho nhiều người dùng trên nhiều nền tảng khác nhau.

## **4. Ưu - nhược điểm**

### **4.1. Ưu điểm**

* **Kiến trúc mạch lạc:** Khi xây dựng ứng dụng Cross Platform, nhà phát triển cần đảm bảo tuân thủ và chặt chẽ theo đúng mô hình MVC hoặc MVVM để tránh phụ thuộc OS. Tuy điều này gây áp lực cho nhà phát triển app nhưng bù lại ứng dụng sẽ có hiệu năng cao, bảo trì dễ dàng trong tương lai.
* **Ít code:** Bạn chỉ cần tốn một ít thời gian và công sức để viết code một bộ mã máy duy nhất, xây dựng và chạy code đó trên nhiều nền tảng khác nhau.
* **Bảo trì và mở rộng:** Việc kiến trúc mạch lạc đã góp phần hỗ trợ điều tra và sửa lỗi trở nên dễ dàng hơn. Từ đó, việc bảo trì hay mở rộng cũng sẽ không gặp nhiều khó khăn nữa.
* **Tiết kiệm thời gian và nguồn lực:** Phát triển một lần và triển khai trên nhiều nền tảng giúp tiết kiệm thời gian và nguồn lực phát triển so với việc viết mã nguồn riêng biệt cho từng nền tảng.
* **Chi phí thấp hơn:** Đối với doanh nghiệp và các dự án phát triển, việc duy trì một mã nguồn chung cho nhiều nền tảng có thể giúp giảm chi phí phát triển và duy trì ứng dụng.
* **Tính nhất quán:** Ứng dụng cross-platform thường có tính nhất quán cao hơn giữa các phiên bản trên các nền tảng khác nhau, vì chúng chia sẻ một mã nguồn chung.

### **4.2. Nhược điểm**

* **Hiệu suất thấp hơn:** Ứng dụng cross-platform thường có hiệu suất thấp hơn so với các ứng dụng được phát triển riêng cho mỗi nền tảng. Điều này đặc biệt đúng trong các ứng dụng đòi hỏi hiệu suất cao như game hoặc ứng dụng đồ họa.
* **Giới hạn tính năng:** Một số tính năng đặc thù của một nền tảng có thể không được hỗ trợ hoặc hỗ trợ kém hiệu quả trong ứng dụng cross-platform. Điều này có thể đặt ra giới hạn cho khả năng sáng tạo và phát triển ứng dụng.
* **Phụ thuộc vào các framework và công cụ:** Phát triển ứng dụng cross-platform thường phải phụ thuộc vào các framework và công cụ cụ thể, và việc này có thể tạo ra các hạn chế và khó khăn trong việc quản lý mã nguồn.
* **Kích thước ứng dụng lớn hơn:** Ứng dụng cross-platform có thể có kích thước lớn hơn do việc bao gồm các thư viện và tài nguyên cho nhiều nền tảng.
* **Khó khăn trong việc tối ưu hóa giao diện người dùng:** Tạo giao diện người dùng hấp dẫn và tối ưu cho từng nền tảng có thể là một thách thức, đặc biệt khi sử dụng một mã nguồn chung.

# **CHƯƠNG VI. ÁP DỤNG GUI FRAMEWORK VÀO ĐỀ TÀI**

## **1. Tổng quan đề tài**

### **1.1. Gới thiệu chung**

Các công nghệ lập trình và framework hiện đại đã trở thành một xu hướng phổ biến trong ngành công nghệ thông tin trong những năm gần đây. Với đề tài này, chúng ta sẽ tìm hiểu cách phát triển một phần mềm cross-platfrom với công nghệ hiện đại để tra cứu thông tin về các Pokemon khác nhau.

Trước hết chúng ta sẽ biết thêm một chút về Pokemon (viết tắt của "Pocket Monsters" - Quái vật trong túi) là một thương hiệu nổi tiếng của Nhật Bản, được phát triển bởi công ty Game Freak và công ty Nintendo. Pokemon bắt đầu là một trò chơi video ra mắt trên hệ máy Game Boy của Nintendo vào năm 1996 và sau đó trở thành một hiện tượng văn hóa toàn cầu.



Trong thế giới Pokemon, người chơi bắt đầu cuộc hành trình để bắt, huấn luyện và chiến đấu với các loài Pokemon khác nhau. Mỗi loài Pokemon có các đặc điểm riêng, khả năng độc đáo và thuộc tính khác nhau. Mục tiêu chính của người chơi là trở thành "Huấn luyện viên Pokemon tốt nhất" bằng cách thu thập và chiến đấu với các Pokemon khác nhau, tham gia các cuộc thi, và hoàn thành nhiệm vụ trong thế giới Pokemon.

Và Pokedex là một thiết bị trong thế giới của trò chơi và anime Pokémon, được thiết kế để theo dõi và lưu trữ thông tin về các loài Pokémon khác nhau. Một công cụ hữu ích trong việc huấn luyện và chinh phục các loài Pokemon khác nhau. Nó cung cấp cho huấn luyện viên thông tin chi tiết về các loài Pokemon, giúp họ tìm hiểu cách tương tác với chúng và sử dụng chiến thuật phù hợp để chiến đấu và đánh bại chúng.

### **1.2. Lí do chọn đề tài**

Với sự phát triển nhanh chóng của các trò chơi Pokemon, việc cung cấp một công cụ để tra cứu thông tin về các Pokemon sẽ trở thành một nhu cầu thiết yếu. Đồng thời, sử dụng mã nguồn mở cũng giúp chúng ta phát triển một phần mềm hiệu quả và dễ dàng mở rộng trong tương lai.

Đồng thời cũng sẽ cho mọi người thấy được các công nghệ hiện đại sẽ hỗ trợ rất nhiều cho các sản phẩm như thế là rất cần thiết như là có thể hiển thị chúng qua giao diện, hình ảnh, lưu dữ liệu,…

Và ngôn ngữ Python là một phương án tốt cho việc lập trình để lập trình bởi những cú pháp được cho là rất ngắn và dễ hiểu và dễ sử dụng, cùng với đó sẽ là các GUI framework giúp thể hiện chúng lên giao diện người dùng.

### **1.3. Mục đích, đóng góp và lợi ích**

* **Áp dụng các công nghệ mới:** Chia sẻ rộng rãi hơn cho mọi người về cách tiếp xúc với các loại công nghệ mới để xây dựng nên một phần mềm, thấy được độ hiệu quả, cách thức hoạt động và năng suất đạt được của nó, cùng với đó là học tập thêm để có thể luôn cập nhật các loại công nghệ mới.
* **Tăng cường kiến thức về Python:** Xây dựng Pokedex bằng Python sẽ giúp bạn nâng cao kiến thức của mình về ngôn ngữ lập trình Python.
* **Học cách trích xuất và xử lý dữ liệu:** Trong quá trình xây dựng Pokedex, bạn sẽ học cách trích xuất và xử lý dữ liệu từ các nguồn khác nhau, bao gồm các trang web và tệp tin dữ liệu.
* **Phát triển kỹ năng lập trình:** Xây dựng Pokedex sẽ giúp bạn phát triển kỹ năng lập trình của mình, bao gồm cả các kỹ năng thiết kế và triển khai ứng dụng.
* **Tạo ra một sản phẩm hoàn chỉnh:** Nếu bạn hoàn thành việc xây dựng Pokedex, bạn sẽ tạo ra một sản phẩm hoàn chỉnh có thể được sử dụng và phát triển tiếp trong tương lai. Nó cũng có thể được chia sẻ và sử dụng bởi cộng đồng lập trình viên và các fan của trò chơi Pokémon.

### **1.4. Nội dung đề tài**

Chức năng chính của phần mềm là tra cứu thông tin, cho phép người dùng xem được các thông tin chi tiết của từng pokemon, kỹ năng của chúng, vật phẩm trong game.

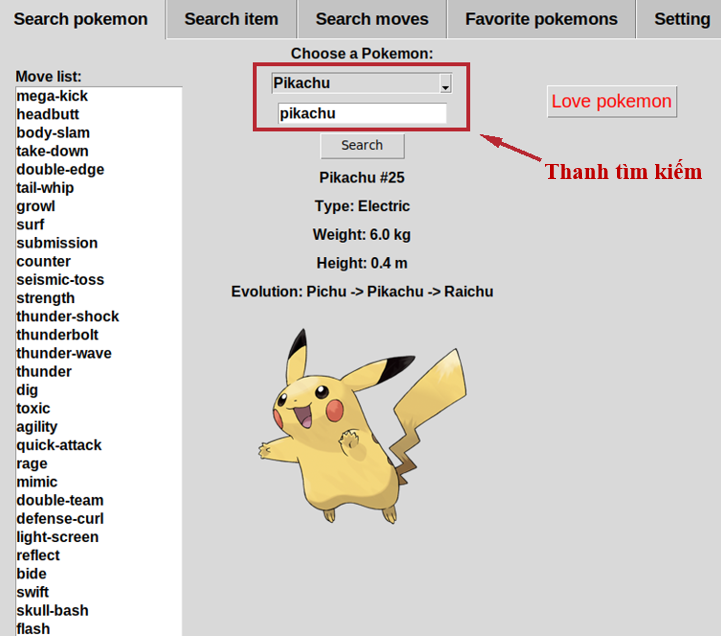
Đây là chức năng quan trọng nhất giúp cho người dùng trải nghiệm Pokemon hài lòng nhất.

Sử dụng API của **pokeapi.co** để lấy tất cả dữ liệu về Pokemon để cung cấp cho việc xây dựng phần mềm tra cứu thông tin Pokedex.

**a) Tìm kiếm Pokemon**

Hiển thị danh sách các Pokemon và thông tin chi tiết của từng Pokemon gồm :

* Mã Pokemon
* Tên Pokemon
* Thuộc tính
* Hình ảnh
* Chiều cao
* Cân nặng
* Tiến hóa
* Kỹ năng



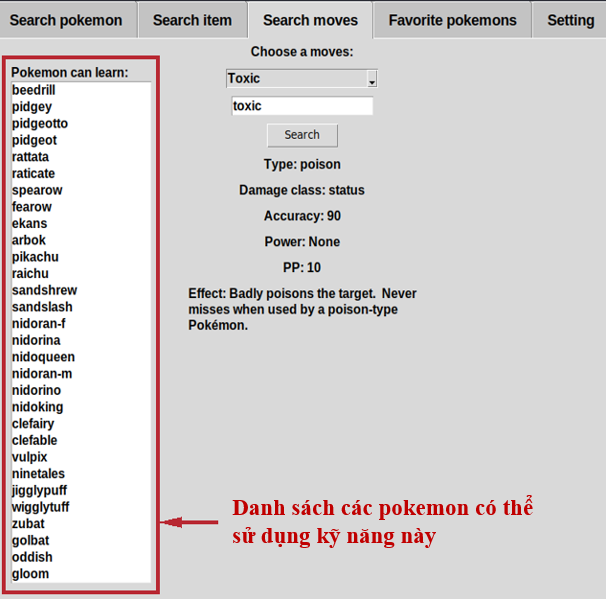
Giao diện tìm kiếm Pokemon

Với danh sách sidebar bên trái cho hiển thị các chiêu thức mà pokemon có thể học được – rút gọn được thời gian tìm kiếm của người dùng.

“Love Pokemon” là chức năng thêm Pokemon đang hiển thị vào danh sách yêu thích để tiện cho việc rút ngắn thời gian khi họ cần xem thông tin này thường xuyên.

**b) Tìm kiếm kỹ năng**

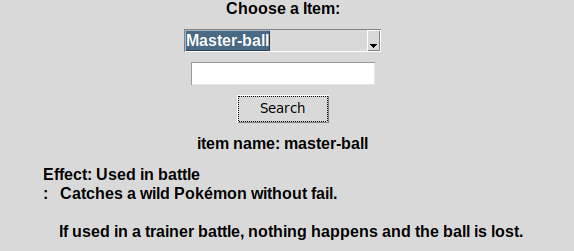
Tương tự như chức năng trên, đây là giao diện nơi cung cấp việc tìm kiếm tất cả kỹ năng, và có thể hiện thị luôn cả các Pokemon có thể sử dụng được kỹ năng ấy.



Giao diện tìm kiếm kỹ năng

**c) Tìm kiếm vật phẩm**

Giao diện tìm kiếm các vật phẩm có trong game như vật phẩm hồi máu, tăng chỉ số, quả cầu thu phục pokemon,…



Giao diện tìm kiếm vật phẩm

**d) Danh sách yêu thích**

Giao diện lưu trữ các pokemon yêu thích của người dùng – thuận tiện và rút ngắn thời gian tìm kiếm cho người dùng. Pokemon sẽ được thêm vào danh sách khi được ấn vào button “Love Pokemon” ở giao diện tìm kiếm pokemon.



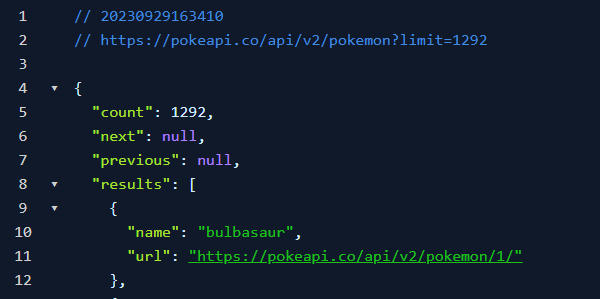
Danh sách Pokemon yêu thích

## **2. Cài đặt trên hệ điều hành Windows**

#### **2.1. API**

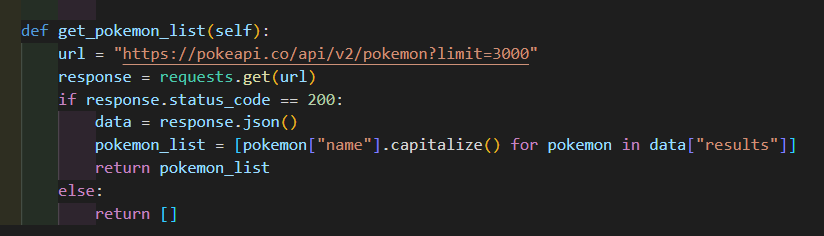
Đề tài sử dụng API của PokeAPI : https://pokeapi.co/api/v2/pokemon/ditto



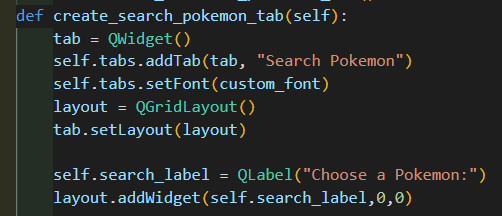
  
Trang API

#### **2.2. PyQt5**

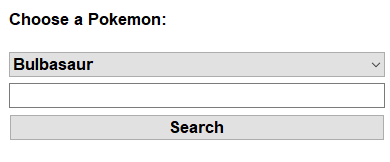
##### **Tìm kiếm Pokemon**

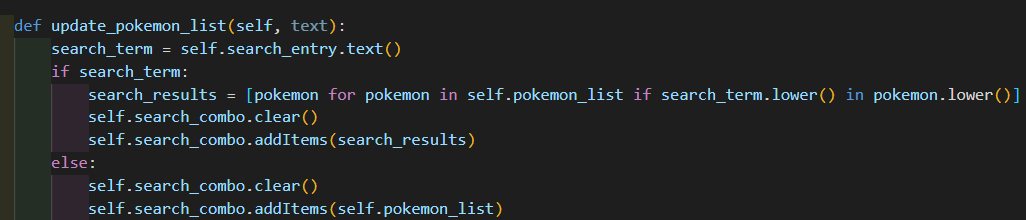


Trả kết quả các dữ liệu về pokemon\_list( danh sách tất cả các pokemon trên API )

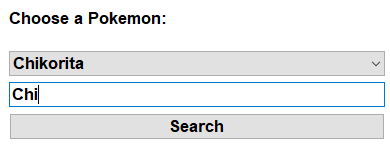


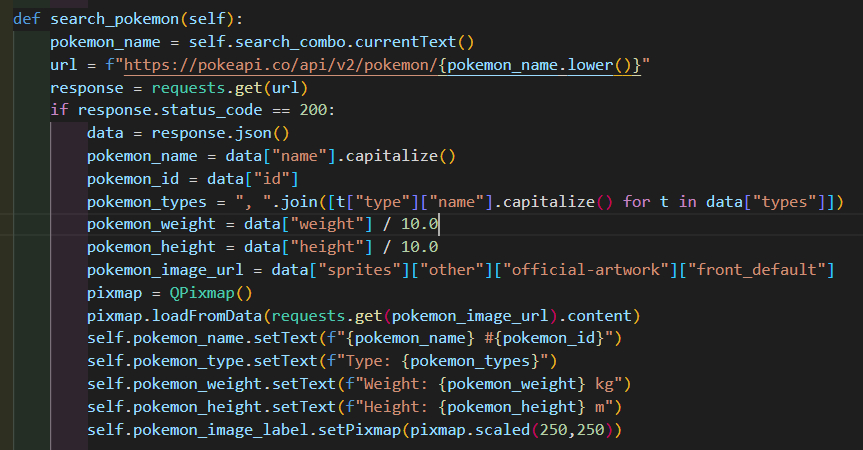
Tìm kiếm Pokemon

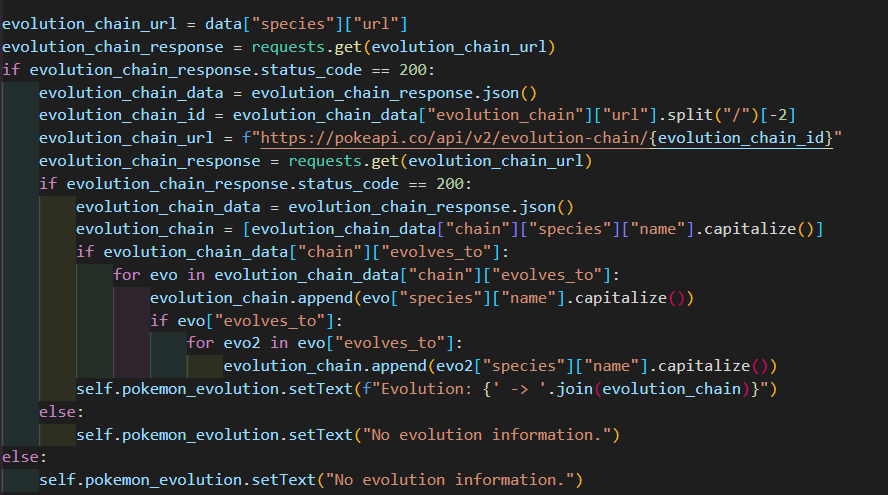
  
Kết quả giao diện tìm kiếm Pokemon



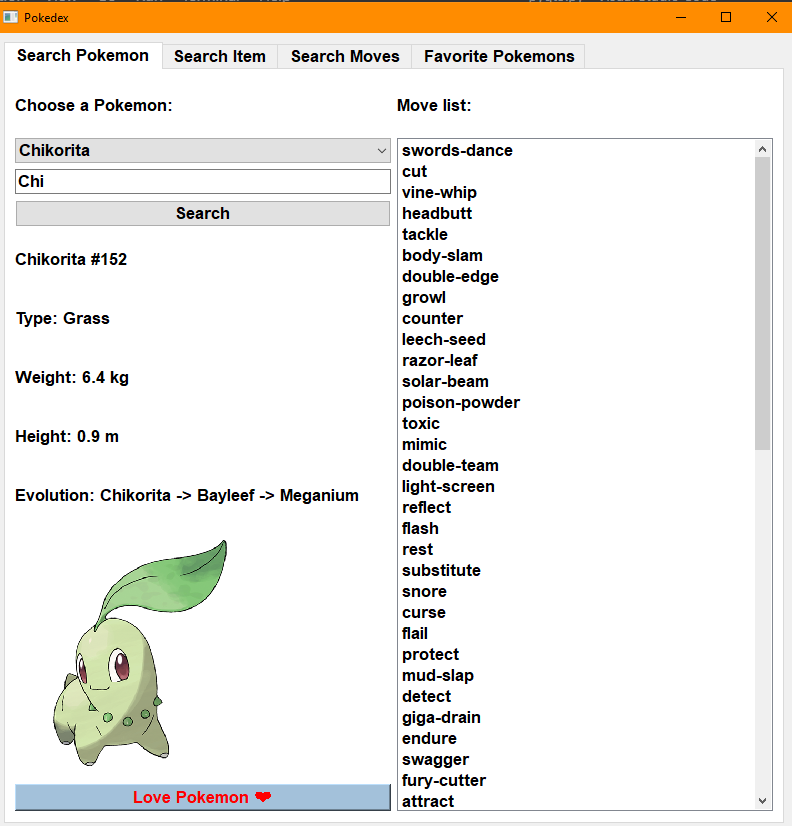
Khi thực hiện tìm từ khóa trên thanh tìm kiếm thì sẽ cập nhật trên combobox

  
Giao diện minh họa cho việc tìm kiếm theo tên

Lấy dữ liệu JSON thuộc tính cơ bản từ API về và xử lý lên giao diện



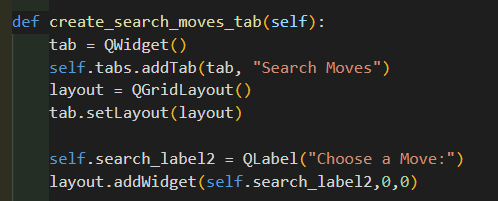
Lấy dữ liệu JSON về chuỗi tiến hóa từ API và xử lý hiển thị lên giao diện



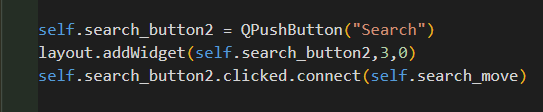
##### **Tìm kiếm kỹ năng**

Thực hiện tìm kiếm kỹ năng cụ thể, cho biết được thông tin của kỹ năng đó như :

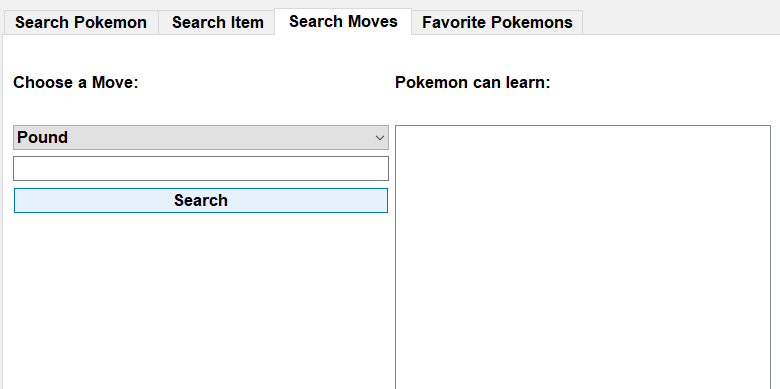
* **Type :** Loại kĩ năng.
* **Damage class :** Loại sát thương.
* **Accuracy :** Độ chính xác.
* **Power :** Sức mạnh.
* **PP :** Số lần sử dụng.
* **Effect :** hiệu ứng gây ra.



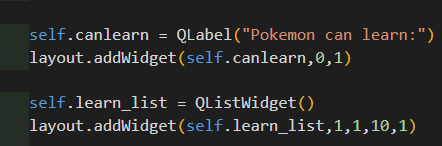
Giao diện tìm kiếm kỹ năng



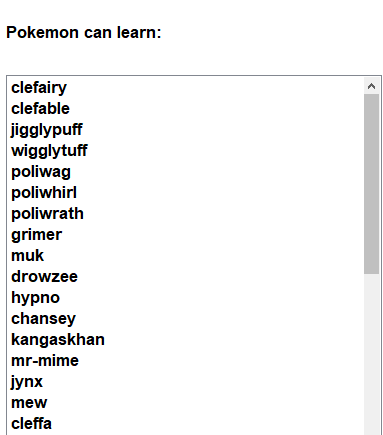
Nút tìm kiếm

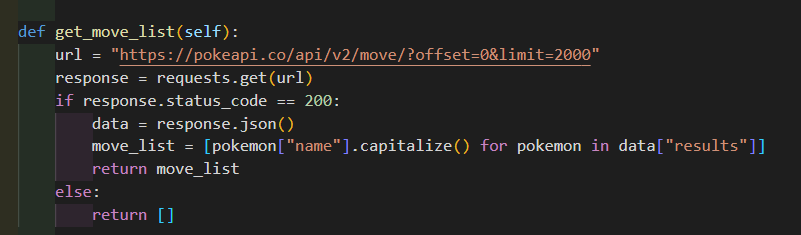


Giao diện tìm kiếm kỹ năng Pokemon

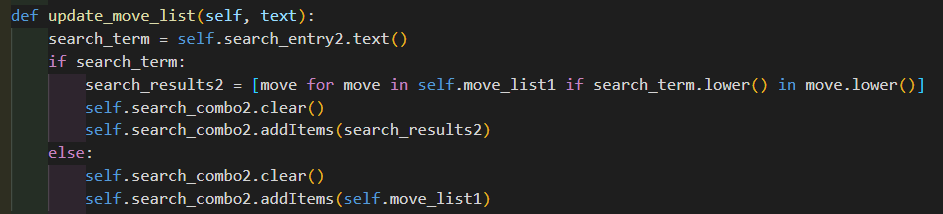


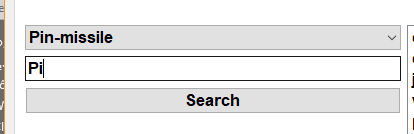
Nơi hiển thị danh sách các pokemon có thể học được kĩ năng này

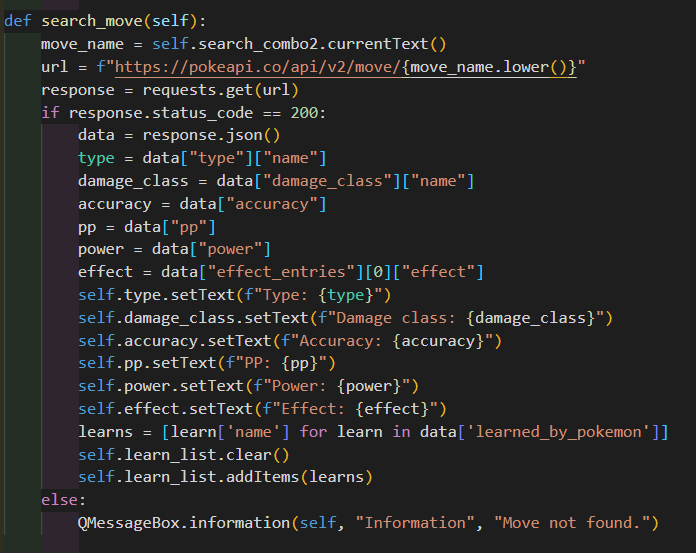




Trả kết quả các dữ liệu về move\_list( danh sách tất cả các kĩ năng trên API )



Thực hiện lọc danh sách các kĩ năng theo từ khóa đã nhập để hiển thị giao diện  




Thực hiện xử lý dữ liệu được lấy về từ API và hiển thị lên giao diện

