**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH**

Logo

Description automatically generated

**MÔN: THỰC HÀNH CHUYÊN SÂU**

**ĐỀ TÀI: LẬP TRÌNH TRỢ LÝ ẢO ĐỌC WEBSITE VỚI NGÔN NGỮ LẬP TRÌNH PYTHON**

**Giảng viên hướng dẫn: ThS. NGUYỄN VĂN SÁU**

**Sinh viên thực hiện**

HỌ VÀ TÊN: Phạm Trúc Tâm

MÃ SINH VIÊN: N18DCPT058

LỚP: D18CQPU01-N

**Hồ Chí Minh ngày 29 tháng 05 năm 2022**

NHẬN XÉT CỦA GIẢNG VIÊN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT NGÔN NGỮ LẬP TRÌNH PYTHON**

**1. Giới thiệu:**

- [**Python**](https://quantrimang.com/python) là ngôn ngữ lập trình hướng đối tượng, cấp cao, mạnh mẽ, được tạo ra bởi Guido van Rossum. Nó dễ dàng để tìm hiểu và đang nổi lên như một trong những ngôn ngữ lập trình nhập môn tốt nhất cho người lần đầu tiếp xúc với ngôn ngữ lập trình. Python hoàn toàn tạo kiểu động và sử dụng cơ chế cấp phát bộ nhớ tự động. Python có [cấu trúc dữ liệu](https://quantrimang.com/cau-truc-du-lieu-va-giai-thuat) cấp cao mạnh mẽ và cách tiếp cận đơn giản nhưng hiệu quả đối với lập trình hướng đối tượng. Cú pháp lệnh của Python là điểm cộng vô cùng lớn vì sự rõ ràng, dễ hiểu và cách gõ linh động làm cho nó nhanh chóng trở thành một ngôn ngữ lý tưởng để viết script và phát triển ứng dụng trong nhiều lĩnh vực, ở hầu hết các nền tảng.

- **Python** là một ngôn ngữ khá cũ được tạo ra bởi Guido Van Rossum. Thiết kế bắt đầu vào cuối những năm 1980 và được phát hành lần đầu tiên vào tháng 2 năm 1991.

**2. Lịch sử phát triển:**

* Python đã được [Guido van Rossum](https://vi.wikipedia.org/w/index.php?title=Guido_van_Rossum&action=edit&redlink=1) thai nghén vào những năm 1980 tại Trung tâm Toán học – Tin học ([Centrum Wiskunde & Informatica](https://vi.wikipedia.org/w/index.php?title=Centrum_Wiskunde_%26_Informatica&action=edit&redlink=1), CWI) ở [Hà Lan](https://vi.wikipedia.org/wiki/H%C3%A0_Lan) như là một ngôn ngữ kế tục ngôn ngữ [ABC](https://vi.wikipedia.org/w/index.php?title=ABC_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1) – một ngôn ngữ được lấy cảm hứng từ [SETL](https://vi.wikipedia.org/w/index.php?title=SETL&action=edit&redlink=1), có khả năng [xử lí ngoại lệ](https://vi.wikipedia.org/w/index.php?title=X%E1%BB%AD_l%C3%AD_ngo%E1%BA%A1i_l%E1%BB%87&action=edit&redlink=1) và giao tiếp với hệ điều hành [Amoeba](https://vi.wikipedia.org/w/index.php?title=Amoeba_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)&action=edit&redlink=1).
* Nó bất đầu được triển khai vào tháng Mười Hai năm 1989. Van Rossum đã tự mình gánh vác trách nhiệm cho dự án, với vai trò là nhà phát triển chính, cho đến ngày 12 tháng Bảy năm 2018, khi ông thông báo rằng ông sẽ rời bỏ trách nhiệm của ông và cả danh hiệu "[Nhà độc tài nhân từ cho cuộc sống](https://vi.wikipedia.org/w/index.php?title=Nh%C3%A0_%C4%91%E1%BB%99c_t%C3%A0i_nh%C3%A2n_t%E1%BB%AB_cho_cu%E1%BB%99c_s%E1%BB%91ng&action=edit&redlink=1)" của Python, một danh hiệu mà cộng đồng Python đã trao tặng cho ông vì sự tận tuỵ lâu dài của ông với vai trò là người ra quyết định chính cho dự án. Vào tháng Một năm 2019, các nhà phát triển phần lõi Python đã bầu ra một "Hội đồng Chèo lài" gồm năm thành viên để dẫn dắt dự án.
* Python 2.0 được ra mắt vào ngày 16 tháng Mười năm 2000, với nhiều tính năng mới mẻ, bao gồm một [bộ don rác](https://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_don_r%C3%A1c&action=edit&redlink=1) phát hiện theo chu kỳ và khả năng hỗ trợ [Unicode](https://vi.wikipedia.org/wiki/Unicode).
* Python 3.0 được ra mắt vào ngày mùng 3 tháng Mười Hai năm 2008. Đây là một phiên bản lớn của Python không [tương thích ngược](https://vi.wikipedia.org/w/index.php?title=T%C6%B0%C6%A1ng_th%C3%ADch_ng%C6%B0%E1%BB%A3c&action=edit&redlink=1) hoàn toàn. Nhiều tính năng lớn của nó đã được chuyển mã ngược (backport) về loạt phiên bản Python 2.6.x và 2.7.x. Các bản phát hành của Python 3 có đi kèm với công cụ 2to3, có tác dụng tự động hoá việc dịch mã Python 2 sang Python 3.
* Python 3.9.2 và 3.8.8 được xúc tiến[[44]](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)#cite_note-44) 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](https://vi.wikipedia.org/w/index.php?title=Th%E1%BB%B1c_th%E1%BB%8B_m%C3%A3_tu%E1%BB%B3_%C3%BD&action=edit&redlink=1) và ["đầu độc" bộ nhớ đệm](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BA%A7u_%C4%91%E1%BB%99c_b%E1%BB%99_nh%E1%BB%9B_%C4%91%E1%BB%87m&action=edit&redlink=1).

**3. Python được dùng ở đâu:**

* **Lập trình ứng dụng web:** Bạn có thể tạo web app có khả năng mở rộng (scalable) được bằng cách sử dụng framework và CMS (Hệ thống quản trị nội dung) được tích hợp trong Python. Vài nền tảng phổ biến để tạo web app là: Django, Flask, Pyramid, Plone, Django CMS. Các trang như Mozilla, Reddit, Instagram và PBS đều được viết bằng Python.
* **Khoa học và tính toán:** Có nhiều thư viện trong Python cho khoa học và tính toán số liệu, như SciPy và NumPy, được sử dụng cho những mục đích chung chung trong tính toán. Và, có những thư viện cụ thể như: EarthPy cho khoa học trái đất, AstroPy cho Thiên văn học,... Ngoài ra, Python còn được sử dụng nhiều trong machine learning, khai thác dữ liệu và deep learning.
* **Tạo nguyên mẫu phần mềm:** Python chậm hơn khi so sánh với các ngôn ngữ được biên dịch như C++ và Java. Nó có thể không phải là lựa chọn tốt nếu nguồn lực bị giới hạn và yêu cầu về hiệu quả là bắt buộc. Tuy nhiên, Python là ngôn ngữ tuyệt vời để tạo những nguyên mẫu (bản chạy thử - prototype). Ví dụ, bạn có thể sử dụng Pygame (thư viện viết game) để tạo nguyên mẫu game trước. Nếu thích nguyên mẫu đó có thể dùng C++ để viết game thực sự.
* **Ngôn ngữ tốt để dạy lập trình:** Python được nhiều công ty, trường học sử dụng để dạy lập trình cho trẻ em và những người mới lần đầu học lập trình. Bên cạnh những tính năng và khả năng tuyệt vời thì cú pháp đơn giản và dễ sử dụng của nó là lý do chính cho việc này.

**4. Triết lý thiết kế và tính năng:**

* Python là một [ngôn ngữ lập trình đa mẫu hình](https://vi.wikipedia.org/wiki/M%E1%BA%ABu_h%C3%ACnh_l%E1%BA%ADp_tr%C3%ACnh). [Lập trình hướng đối tượng](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng) và [lập trình cấu trúc](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_c%E1%BA%A5u_tr%C3%BAc) được hỗ trợ hoàn toàn, và nhiều tính năng của nó cũng hỗ trợ lập trình hàm và [lập trình hướng khía cạnh](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_h%C6%B0%E1%BB%9Bng_kh%C3%ADa_c%E1%BA%A1nh) (bao gồm [siêu lập trình](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_meta) và [siêu đối tượng](https://vi.wikipedia.org/w/index.php?title=Si%C3%AAu_%C4%91%E1%BB%91i_t%C6%B0%E1%BB%A3ng&action=edit&redlink=1) (phương thức thần kì)). Các mẫu hình khác cũng được hỗ trợ thông qua các phần mở rộng, bao gồm [thiết kế theo hợp đồng](https://vi.wikipedia.org/w/index.php?title=Thi%E1%BA%BFt_k%E1%BA%BF_theo_h%E1%BB%A3p_%C4%91%E1%BB%93ng&action=edit&redlink=1) và [lập trình logic](https://vi.wikipedia.org/wiki/L%E1%BA%ADp_tr%C3%ACnh_logic).
* Python sử dụng [kiểu động](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng_ki%E1%BB%83u) và một dạng kết hợp giữa [đếm tham chiếu](https://vi.wikipedia.org/w/index.php?title=%C4%90%E1%BA%BFm_tham_chi%E1%BA%BFu&action=edit&redlink=1) và bộ dọn rác kiểm tra theo chu kì để [quản lí bộ nhớ](https://vi.wikipedia.org/wiki/Qu%E1%BA%A3n_l%C3%AD_b%E1%BB%99_nh%E1%BB%9B). Nó cũng có tính năng [phân giải tên](https://vi.wikipedia.org/w/index.php?title=Ph%C3%A2n_gi%E1%BA%A3i_t%C3%AAn&action=edit&redlink=1) động ([liên kết muộn](https://vi.wikipedia.org/w/index.php?title=Li%C3%AAn_k%E1%BA%BFt_mu%E1%BB%99n&action=edit&redlink=1)), cho phép liên kết các tên biến và phương thức trong quá trình thực thi chương trình.
* Thiết kế của Python cung cấp một số tính năng cho lập trình hàm giống như trong ngôn ngữ [Lisp](https://vi.wikipedia.org/wiki/Lisp). Python có các hàm filter, map và reduce; thông hiểu danh sách (list comprehension), [từ điển](https://vi.wikipedia.org/w/index.php?title=M%E1%BA%A3ng_li%C3%AAn_k%E1%BA%BFt&action=edit&redlink=1) (dictionary), tập hợp (set), và các biểu thức [bộ sinh](https://vi.wikipedia.org/w/index.php?title=B%E1%BB%99_sinh_(l%E1%BA%ADp_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh)&action=edit&redlink=1) (generator). Thư viện chuẩn cũng có hai mô đun (itertools và functools) triển khai các công cụ hàm được vay mượn từ [Haskell](https://vi.wikipedia.org/wiki/Haskell_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) và [Standard ML](https://vi.wikipedia.org/w/index.php?title=Standard_ML&action=edit&redlink=1).
* Triết lý căn bản của ngôn ngữ Python được trình bày trong tài liệu The [Zen of Python](https://vi.wikipedia.org/wiki/Zen_of_Python) (PEP 20), có dạng [thơ Haiku](https://vi.wikipedia.org/wiki/Haiku), tóm gọn như sau:
* Đẹp đẽ tốt hơn xấu xí
* Minh bạch tốt hơn ngầm định
* Đơn giản tốt hơn phức tạp
* Phức tạp tốt hơn rắc rối
* Tính dễ đọc rất quan trọng.
* Thay vì tích hợp hết tất cả các tính năng vào phần cốt lõi, Python được thiết kế để [dễ dàng mở rộng](https://vi.wikipedia.org/w/index.php?title=T%C3%ADnh_m%E1%BB%9F_r%E1%BB%99ng&action=edit&redlink=1) (bằng các mô đun). Tính mô đun nhỏ gọn này đã làm cho Python trở nên phổ biến như là một cách thêm các giao diện lập trình được vào các ứng dụng hiện có. Tầm nhìn của Van Rossum về một ngôn ngữ có phần lõi nhỏ với một thứ viện chuẩn rộng lớn và một trình thông dịch dễ dàng mở rộng bắt nguồn từ việc ông nản lòng trước [ABC](https://vi.wikipedia.org/w/index.php?title=ABC_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)&action=edit&redlink=1), một ngôn ngữ lập trình tán thành hướng tiếp cận ngược lại. Python thường được mô tả là một ngôn ngữ "tặng kèm pin" nhờ vào [thư viện chuẩn](https://vi.wikipedia.org/wiki/Th%C6%B0_vi%E1%BB%87n_chu%E1%BA%A9n) bao quát của nó.
* Python nỗ lực hướng đến một cú pháp đơn giản hơn, gọn gàng hơn trong khi vẫn cho các nhà phát triển lựa chọn phương pháp viết mã của họ. Đối lập với khẩu hiệu "[có nhiều hơn một cách để làm việc này](https://vi.wikipedia.org/w/index.php?title=C%C3%B3_nhi%E1%BB%81u_h%C6%A1n_m%E1%BB%99t_c%C3%A1ch_%C4%91%E1%BB%83_l%C3%A0m_vi%E1%BB%87c_n%C3%A0y&action=edit&redlink=1)", triết lý thiết kể của Python lại nằm trong châm ngôn "chỉ nên có một— và tốt nhất là chỉ một—cách rõ ràng để làm việc này". [Alex Martelli](https://vi.wikipedia.org/w/index.php?title=Alex_Martelli&action=edit&redlink=1), một Viện sĩ (Fellow) tại Tổ chức Phần mềm Python (Python Software Foundation) và là một tác giả viết sách Python, viết rằng "Mô tả một thứ gì đó là "tài tình" không được coi là một lời khen ngợi trong văn hoá Python."[[58]](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)#cite_note-AutoNT-202-58)
* Các nhà phát triển Python nỗ lực tránh xa việc [tối ưu hoá quá sớm](https://vi.wikipedia.org/w/index.php?title=T%E1%BB%91i_%C6%B0u_ho%C3%A1_ch%C6%B0%C6%A1ng_tr%C3%ACnh&action=edit&redlink=1) và không chấp nhận các bản vá không cải thiện đáng kể tốc độ mà lại làm mất đi tính rõ ràng lên những phần không thiết yếu của bản thực hiện tham khảo [CPython](https://vi.wikipedia.org/w/index.php?title=CPython&action=edit&redlink=1).[[58]](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)#cite_note-AutoNT-202-58) Khi cần đến tốc độ, một lập trình viên Python có thể di chuyển các hàm bị giới hạn về thời gian sang các mô đun mở rộng được viết bằng những ngôn ngữ như C, hoặc sử dụng [PyPy](https://vi.wikipedia.org/w/index.php?title=PyPy&action=edit&redlink=1), một [trình biên dịch tức thời](https://vi.wikipedia.org/w/index.php?title=Bi%C3%AAn_d%E1%BB%8Bch_t%E1%BB%A9c_th%E1%BB%9Di&action=edit&redlink=1). [Cython](https://vi.wikipedia.org/w/index.php?title=Cython&action=edit&redlink=1) cũng có thể được dùng để phiên dịch một tập lệnh Python sang C và tạo ra các lệnh gọi API ở cấp độ của C trực tiếp vào trình thông dịch Python.
* Mục tiêu mà các nhà phát triển Python hướng đến là đem đến niềm vui khi sử dụng ngôn ngữ này. Điều này được thể hiện qua bản thân cái tên – một sự tôn vinh dành cho nhóm hài người Anh [Monty Python](https://vi.wikipedia.org/w/index.php?title=Monty_Python&action=edit&redlink=1)[[59]](https://vi.wikipedia.org/wiki/Python_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)#cite_note-AutoNT-24-59) – và trong một số cách tiếp cận thi thoảng vui tươi trong hướng dẫn và các tài liệu tham khảo, chẳng hạn như một vi dụ có đề cập đến trứng và [spam](https://vi.wikipedia.org/wiki/Spam_(th%E1%BB%B1c_ph%E1%BA%A9m)) (gợi nhắc đến một [tiểu phẩm trong Monty Python](https://vi.wikipedia.org/w/index.php?title=Spam_(Monty_Python)&action=edit&redlink=1)) thay cho [foo](https://vi.wikipedia.org/wiki/Foo) và [bar](https://vi.wikipedia.org/wiki/Bar_(khoa_h%E1%BB%8Dc_m%C3%A1y_t%C3%ADnh)) tiêu chuẩn.

**5. Ưu nhược điểm:**

**Ưu điểm:**

* Là một ngôn ngữ có hình thức sáng sủa, cấu trúc roc ràng, cú pháp ngắn gọn
* Có trên tất cả các nền tảng hệ điều hành từ UNIX, MS – DOS, Mac OS, Windows và Linix và các OS khác thuộc họ Unix.
* Tương thích mạnh mẽ với Unix, hardware, thirt-party software với số lượng thư viện khổng lồ (400 triệu người sử dụng)
* [Python](http://itplus-academy.edu.vn/Khoa-hoc-lap-trinh-Python-PYTHON-FOR-EVERYBODY.html) với tốc độ xử lý cực nhanh, python có thể tạo ra những chương trình từ những script siêu nhỏ tới những phần mềm cực lớn như Biender 3D.

**Nhược điểm:**

[Python](http://itplus-academy.edu.vn/Khoa-hoc-lap-trinh-Python-PYTHON-FOR-EVERYBODY.html) không có các thuộc tính như: protected, private hay public, không có vòng lặp do…while và switch….case.

Mặc dù tốc độ xử lý của Python nhanh hơn [PHP](http://itplus-academy.edu.vn/Lap-trinh-CC-cung-FPT-Software.html) nhưng không bằng[JAVA](http://itplus-academy.edu.vn/Khoa-hoc-lap-trinh-java-co-ban.html) và[C++](http://itplus-academy.edu.vn/Lap-trinh-CC-cung-FPT-Software.html).

**CHƯƠNG 2: THIẾT KẾ**

**1. Ý tưởng thực hiện:**

Người dùng nói tôi muốn mua giày adidas trên shopee thì mình mở shopee và tìm giày adidas cho người dùng rồi mình lần lượt đọc các kết quả, ví dụ khi mình đọc tới món hàng giày adidas "chính hãng" giá 350k, người dùng kêu coi chi tiết đôi này thì mình bấm chi tiết lên đọc cho người dùng nghe, nếu người dùng kêu mua thì mình bấm vô mua,...

**2. Các thư viện được sử dụng:**

**-** gTTS

- Speech Recognition

- selenium

- pygame

- playsound

- …

**3. Các Pip yêu cầu để thực hiện code cho trợ lý ảo:**

import os

import time

import speech\_recognition as sr

import webbrowser

import re

import playsound

import pygame

from pygame import mixer

from gtts import gTTS

import selenium

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.chrome.options import Options

from webdriver\_manager.chrome import ChromeDriverManager

from time import strftime

**4. Các bước dự kiến thực hiện:**

Bước 1: tạo hàm speak() để chuyển từ chữ sang âm thanh

Bước 2: tạo hàm get\_voice() để thu giọng nói của người dùng

Bước 3: tạo hàm get\_text() để chuyển từ giọng nói của người dùng thành text

Bước 4: tạo các chức năng mở website hoặc search google

…

**CHƯƠNG 3: THỰC HIỆN**

**1. Lập trình phần nghe cho trợ lý ảo:**

def get\_voice():

    r = sr.Recognizer()

    with sr.Microphone() as source:

        print("Me: ", *end*='')

        audio = r.listen(source, *phrase\_time\_limit*=5)

        try:

            text = r.recognize\_google(audio, *language*="vi-VN")

            print(text)

            return text

        except:

            print("...")

            return 0

* **Sử dụng thư viện Speech Recognition để thực hiện việc này**

**2. Lập trình phần nói cho trợ lý ảo:**

def speak(*text*):

    tts = gTTS(*text*=*text*, *lang*=LANGUAGE, *slow*=False)

    tts.save("sound.mp3")

    playsound.playsound("sound.mp3", True)

    os.remove("sound.mp3")

* **Đầu tiên là ta phải tạo ra file âm thanh bằng cách sử dụng thư viện gTTS sau đó dùng hàm playsound() của thư viện PlaySound để phát âm thanh đó, sau khi phát xong thì remove nó.**

**3. Lập trình hàm stop() cho trợ lý ảo:**

def stop():

    speak("Hẹn gặp lại bạn nhé!")

**4. Lập trình hàm get\_text() cho trợ lý ảo:**

def get\_text():

    for i in range(5):

        text = get\_voice()

        if text:

            return text.lower()

        elif i < 4:

            speak("Tớ không nghe rõ. Bạn nói lại nhé?")

    time.sleep(1)

    stop()

    return 0

* **Hàm get\_text() để chuyển từ giọng nói của người dùng sang dạng text**

**5. Lập trình chức năng mở website bất kì:**

def open\_website(*text*):

    regex = re.search('mở (.+)', *text*)

    if regex:

        domain = regex.group(1)

        url = 'https://www.' + domain

        webbrowser.open(url)

        speak("Trang web của bạn đã được mở lên!")

        return True

    else:

        return False

* **Khi người dùng muốn mở trang web nào thì chỉ cần nói “mở youtube.com” thì chức năng sẽ mở youtube.com**

**6. Lập trình chức năng search google cho trợ lý ảo:**

def google\_search(*text*):

    clears()

    search\_for = *text*.split("kiếm", 1)[1]

    speak("được thôi")

    global driver

    driver = webdriver.Chrome(*service*=ser, *options*=op)

    driver.get("http://www.google.com")

    driver.maximize\_window()

    query = driver.find\_element(By.XPATH, "//input[@name='q']")

    query.send\_keys(str(search\_for))

    query.send\_keys(Keys.RETURN)

    elements = driver.find\_elements(By.CSS\_SELECTOR, "#rso>div")

*text* = ""

    for el in elements:

        try:

            titles.append(el.find\_element(By.CSS\_SELECTOR, 'a>h3').text)

            links.append(el.find\_element(

                By.CSS\_SELECTOR, 'a').get\_attribute('href'))

        except Exception:

            pass

    for i in range(3):

*text* += titles[i] + "\n"

* **Khi người dùng nói “mở và tìm kiếm Sơn Tùng” thì hàm sẽ tách câu nói của người dùng để lấy thông tin cần search (là sau từ ‘kiếm’) sau đó thực hiện mở google và search với từ khóa đó.**

**7. Lập trình chức năng mở liên kết và thực hiện đọc trang web đó:**

* **Hàm mở liên kết:**

def open\_link(*driver*, *url*):

*driver*.get(*url*)

**Khi người dùng nói mở liên kết thứ … thì hàn sẽ được thực thi**

* **Hàm đọc website:**

def read\_news():

    text = driver.find\_element(By.XPATH, "html").text

    arr = text.split('\n')

    for i in range(len(arr)):

        speak(str(arr[i]))

**Đầu tiên sẽ thực hiện việc lấy text của toàn bộ trang web, sau đó sẽ thực hiện việc chia nhỏ text ra để việc đọc được thực thi nhanh sau đó thực hiện đọc web.**

**8. Chương trình hoàn chỉnh và chạy thử**

import os

import subprocess

import time

from turtle import st

import speech\_recognition as sr

import webbrowser

import re

import playsound

import pygame

from pygame import mixer

from gtts import gTTS

import selenium

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

from selenium.webdriver.common.by import By

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.chrome.options import Options

from webdriver\_manager.chrome import ChromeDriverManager

from time import strftime

LANGUAGE = 'vi'

path = ChromeDriverManager().install()

ser = Service(path)

op = webdriver.ChromeOptions()

titles = []

links = []

return\_keywords = ""

driver = ""

def speak(*text*):

    tts = gTTS(*text*=text, *lang*=LANGUAGE, *slow*=False)

    tts.save("sound.mp3")

    playsound.playsound("sound.mp3", True)

    os.remove("sound.mp3")

def get\_voice():

    r = sr.Recognizer()

    with sr.Microphone() as source:

        print("Me: ", end='')

        audio = r.listen(source, *phrase\_time\_limit*=5)

        try:

            text = r.recognize\_google(audio, *language*="vi-VN")

            print(text)

            return text

        except:

            print("...")

            return 0

def stop():

    speak("Hẹn gặp lại bạn nhé!")

def get\_text():

    for i in range(5):

        text = get\_voice()

        if text:

            return text.lower()

        elif i < 4:

            speak("Tớ không nghe rõ. Bạn nói lại nhé?")

    time.sleep(1)

    stop()

    return 0

def open\_website(*text*):

    regex = re.search('mở (.+)', text)

    if regex:

        domain = regex.group(1)

        url = 'https://www.' + domain

        webbrowser.open(url)

        speak("Trang web của bạn đã được mở lên!")

        return True

    else:

        return False

def clears():

    links.clear()

    titles.clear()

def google\_search(*text*):

    clears()

    search\_for = text.split("kiếm", 1)[1]

    speak("được thôi")

    global driver

    driver = webdriver.Chrome(*service*=ser, *options*=op)

    driver.get("http://www.google.com")

    driver.maximize\_window()

    query = driver.find\_element(By.XPATH, "//input[@name='q']")

    query.send\_keys(*str*(search\_for))

    query.send\_keys(Keys.RETURN)

    elements = driver.find\_elements(By.CSS\_SELECTOR, "#rso>div")

    text = ""

    for el in elements:

        try:

            titles.append(el.find\_element(By.CSS\_SELECTOR, 'a>h3').text)

            links.append(el.find\_element(

                By.CSS\_SELECTOR, 'a').get\_attribute('href'))

        except *Exception*:

            pass

    for i in range(3):

        text += titles[i] + "\n"

    speak(text)

# Đọc full text bằng selenium

def open\_link(*driver*, *url*):

    driver.get(url)

def read\_news():

    text = driver.find\_element(By.XPATH, "html").text

    arr = text.split('\n')

    for i in range(len(arr)):

        speak(*str*(arr[i]))

def call():

    speak("Bạn cần giúp gì ạ?")

    global return\_keywords

    while True:

        try:

            text = get\_text()

            if "mở" in text and "liên kết" in text:

                for i in range(len(links)):

                    if *str*(i) in text:

                        open\_link(driver, links[i])

                        break

            elif "trở về" in text:

                google\_search(return\_keywords)

                driver.close()

            elif "mở" in text:

                if "tìm kiếm":

                    return\_keywords = text

                    google\_search(return\_keywords)

                else:

                    open\_website(text)

            elif "đọc" in text:

                read\_news()

            elif "dừng" in text or "thôi" in text or "stop" in text or "đóng" in text:

                if driver == '':

                    stop()

                else:

                    stop()

                    driver.close()

                break

        except:

            speak("Đã có lỗi")

call()

**9. Hạn chế:**

**-** Chưa thực hiện được việc đang đọc báo mà người dùng muốn dừng lại.

- Chưa đi sâu vào thực hiện nhiều chức năng khác.

**10. Khắc phục:**

**-** Thực hiện thêm chức năng trợ lý đang đọc mà người dùng muốn dừng lại

- Thêm nhiều chức năng khác.