

**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP
KHOA SƯ PHẠM TOÁN – TIN**



0020410724 – LƯƠNG NGUYỄN ANH KHOA

**BÁO CÁO BÀI TẬP LỚN
HỌC PHẦN NGÔN NGỮ LẬP TRÌNH PYTHON**

**XÂY DỰNG CHƯƠNG TRÌNH
TẠO MÃ, QUÉT MÃ QR CODE/BARCODE,
DỊCH MÁY, GAME CƠ BẢN VỚI PYTHON**

NGÀNH: KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT20B

Đồng Tháp, tháng 01 năm 2023

**TRƯỜNG ĐẠI HỌC ĐỒNG THÁP
KHOA SƯ PHẠM TOÁN – TIN**



0020410724 – LƯƠNG NGUYỄN ANH KHOA

**BÁO CÁO BÀI TẬP LỚN
HỌC PHẦN NGÔN NGỮ LẬP TRÌNH PYTHON**

**XÂY DỰNG CHƯƠNG TRÌNH
TẠO MÃ, QUÉT MÃ QR CODE/BARCODE,
DỊCH MÁY, GAME CƠ BẢN VỚI PYTHON**

NGÀNH: KHOA HỌC MÁY TÍNH

LỚP: ĐHCNTT20B

GIẢNG VIÊN GIẢNG DẠY: TS. ĐẶNG QUỐC BẢO

DANH SÁCH SINH VIÊN

STT	MSSV	HỌ TÊN	GHI CHÚ
1	0020410724	Lương Nguyễn Anh Khoa	Nhóm trưởng ĐT: 0763238045 E-mail: hi.luongkhoa@gmail.com

Bảng phân công công việc:

STT	CÔNG VIỆC	GHI CHÚ
1	Tìm hiểu các thư viện GoogleTrans, Tkinter, Pyttsx3, Speech_recognition, Pygame, Qrcode, CV2, Pyzbar, PIL	Lương Nguyễn Anh Khoa
2	Xây dựng chương trình tạo mã và quét, đọc mã Qrcode/Barcode	Lương Nguyễn Anh Khoa
3	Xây dựng chương trình Dịch máy	Lương Nguyễn Anh Khoa
4	Xây dựng game Flappy Bird	Lương Nguyễn Anh Khoa
5	Thực hiện tài liệu báo cáo	Lương Nguyễn Anh Khoa

Ngày nộp:

Người nộp

Lương Nguyễn Anh Khoa

MỤC LỤC

PHẦN 1: GIỚI THIỆU	1
1.1. Giới thiệu đề tài	1
1.2. Nội dung thực hiện	1
1.3. Mục tiêu của đề tài.....	1
PHẦN 2: CƠ SỞ LÝ THUYẾT	2
2.1. Ngôn ngữ lập trình Python.....	2
2.1.1. Giới thiệu	2
2.1.1. Đặc điểm	2
2.1.2. Ứng dụng	2
2.2. Mã QRCode/BarCode.....	3
2.3. Dịch máy	3
PHẦN 3: XÂY DỰNG CHƯƠNG TRÌNH	4
3.1. Đặt vấn đề	4
3.2. Phương pháp giải quyết vấn đề.....	4
3.3. Thiết kế, cài đặt thư viện và xây dựng các hàm xử lý	4
3.3.1. Cài đặt các thư viện	4
3.3.1.1. Tạo mã, quét mã QRCode/BarCode	5
3.3.1.2. Dịch máy	8
3.3.1.3. Game Flappy Bird.....	10
3.3.2. Các hàm xử lý tiêu biểu	11
3.3.2.1. Tạo mã, quét mã QRCode/BarCode	11
3.3.2.2. Dịch máy	13
3.3.2.3. Game Flappy Bird.....	14
3.4. Giới thiệu Demo và hướng dẫn sử dụng.....	17
3.4.1. Menu chính	17
3.4.2. Tạo mã, quét mã QRCode/BarCode	17
3.4.3. Dịch máy	19
3.4.4. Game Flappy Bird.....	20

PHẦN 4: KẾT LUẬN	21
4.1. Kết quả đạt được.....	21
4.1.1. Về lý thuyết	21
4.1.2. Về thực hành.....	21
4.1.3. Về kỹ năng.....	21
4.2. Hạn chế	21
4.3. Hướng phát triển.....	22
TÀI LIỆU THAM KHẢO	23
PHỤ LỤC	24

DANH MỤC HÌNH

Hình 3.1 Cửa sổ cài đặt thư viện Tkinter	4
Hình 3.2 Cửa sổ cài đặt thư viện Pillow.....	5
Hình 3.3 Cửa sổ cài đặt thư viện Qrcode	6
Hình 3.4 Cửa sổ cài đặt thư viện OpenCV	6
Hình 3.5 Cửa sổ cài đặt thư viện Numpy	7
Hình 3.6 Cửa sổ cài đặt thư viện Pyzbar	8
Hình 3.7 Cửa sổ cài đặt thư viện GoogleTrans	8
Hình 3.8 Cửa sổ cài đặt thư viện Pyttsx3	9
Hình 3.9 Cửa sổ cài đặt thư viện Speech Recognition	10
Hình 3.10 Cửa sổ cài đặt thư viện Pygame	10
Hình 3.11 Hàm tạo QRCode	11
Hình 3.12 Hàm quét và đọc QRCode.....	12
Hình 3.13 Hàm đọc file nhập vào frame input	13
Hình 3.14 Hàm nhận dạng giọng nói, chuyển thành văn bản	13
Hình 3.15 Hàm dịch văn bản đã nhập	13
Hình 3.16 Hàm chuyển văn bản thành giọng nói	14
Hình 3.17 Hàm vẽ các vật thể trong game	14
Hình 3.18 Hàm kiểm tra va chạm.....	14
Hình 3.19 Hàm tạo hiệu ứng và hàm di chuyển của chim Bird	15
Hình 3.20 Hàm tính điểm và hiển thị điểm	15
Hình 3.21 Hàm xử lý	16
Hình 3.22 Giao diện Menu chính	17
Hình 3.23 Giao diện tạo mã QRCode.....	17
Hình 3.24 Giao diện Camera nhận dạng mã.....	18
Hình 3.25 Giao diện đọc QR và hiển thị dữ liệu đọc được	18
Hình 3.26 Giao diện dịch nội dung	19
Hình 3.27 Giao diện trò chơi	20

PHẦN 1: GIỚI THIỆU

1.1. Giới thiệu đề tài

Ngày nay cùng với sự phát triển của công nghệ, người sử dụng đòi hỏi nó cần phải là một ứng dụng đa chức năng để dễ dàng hơn trong việc thao tác hằng ngày.

Nhận thấy được vấn đề đó, nhóm sinh viên đã chọn đề tài “*Xây dựng chương trình tạo mã, quét mã QRCode, BarCode, Dịch máy, Game cơ bản với Python*” nhằm xây dựng một chương trình gồm nhiều chức năng như: Hỗ trợ đọc mã QRCode/BarCode, hỗ trợ dịch thuật, hỗ trợ giải trí. Chương trình được xây dựng bởi ngôn ngữ lập trình Python cho phép người sử dụng có thể tạo các mã QRCode/BarCode tại hộp thoại nhập dữ liệu và quét mã bằng cách sử dụng camera của máy tính hoặc các camera rời bên ngoài. Hỗ trợ dịch tất cả các loại ngôn ngữ khác nhau sử dụng thư viện của Google Translate. Ngoài ra, chương trình còn đáp ứng nhu cầu giải trí của người sử dụng bằng trò chơi Flappy Bird sau những giờ làm việc căng thẳng.

1.2. Nội dung thực hiện

- Cài đặt các thư viện cần thiết để xây dựng chương trình.
- Xây dựng giao diện.
- Tạo các nút lệnh thao tác.
- Xây dựng các thuật toán xử lý.

1.3. Mục tiêu của đề tài

Xây dựng thành công chương trình tạo mã, quét mã QRCode/BarCode, Dịch máy, Game cơ bản sử dụng ngôn ngữ lập trình Python.

PHẦN 2: CƠ SỞ LÝ THUYẾT

2.1. Ngôn ngữ lập trình Python

2.1.1. Giới thiệu

Python là một ngôn ngữ lập trình thông dịch do Guido van Rossum tạo ra năm 1990. Python hoàn toàn tạo kiểu động và dùng cơ chế cấp phát bộ nhớ tự động do vậy nó tương tự như Perl, Ruby, Scheme, Smalltalk,... Python được phát triển trong một dự án mã nguồn mở, do tổ chức phi lợi nhuận Python Software Foundation quản lý.

Theo đánh giá của Eric S. Raymond, Python là ngôn ngữ có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học lập trình. Cấu trúc của Python còn cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu. Ban đầu, Python được phát triển để chạy trên nền Unix. Nhưng rồi theo thời gian, nó đã phát triển sang mọi hệ điều hành từ MS-DOS đến Mac OS, OS/2, Windows, Linux và các hệ điều hành khác thuộc họ Unix. Mặc dù sự phát triển của Python có sự đóng góp của rất nhiều cá nhân, nhưng Guido van Rossum vẫn được xem là tác giả chính của Python. Ông giữ vai trò chủ chốt trong việc định hướng phát triển của Python.

2.1.1. Đặc điểm

Python được thiết kế với tư tưởng giúp người học dễ đọc, dễ hiểu và dễ nhớ; vì thế ngôn ngữ Python có hình thức rất sáng sủa, cấu trúc rõ ràng, thuận tiện cho người mới học. Cấu trúc của Python cho phép người sử dụng viết mã lệnh với số lần gõ phím tối thiểu, nói cách khác thì so với các ngôn ngữ lập trình khác, chúng ta có thể sử dụng ít dòng code hơn để viết ra một chương trình trong Python.

Ban đầu, Python được phát triển để chạy trên nền Unix, vì thế nó là mã nguồn mở. Sau này qua thời gian phát triển, Python mở rộng và hiện nay đã hỗ trợ hầu hết các nền tảng khác như Window hay MacOS.

Python là một ngôn ngữ lập trình đa mẫu hình, nó hỗ trợ hoàn toàn mẫu lập trình hướng đối tượng và lập trình cấu trúc; ngoài ra về mặt tính năng, Python cũng hỗ trợ lập trình hàm và lập trình hướng khía cạnh. Nhờ vậy mà Python có thể làm được rất nhiều thứ, sử dụng trong nhiều lĩnh vực khác nhau.

2.1.2. Ứng dụng

Python là ngôn ngữ được ứng dụng đa dạng trong các lĩnh vực.

- **Làm Web với các Framework của Python:** Django và Flask là 2 framework phổ biến hiện nay dành cho các lập trình viên Python để tạo ra các website.
- **Tool tự động hóa:** các ứng dụng như từ điển, crawl dữ liệu từ website, tool giúp tự động hóa công việc được các lập trình viên ưu tiên lựa chọn Python để viết nhờ tốc độ code nhanh của nó.

- **Khoa học máy tính:** Trong Python có rất nhiều thư viện quan trọng phục vụ cho ngành khoa học máy tính như: OpenCV cho xử lý ảnh và machine learning, Scipy và Numpys cho lĩnh vực toán học, đại số tuyến tính, Pandas, ...
- **Lĩnh vực IoT:** Python có thể viết được các ứng dụng cho nền tảng nhúng, đồng thời cũng được lựa chọn cho việc xử lý dữ liệu lớn. Vì thế Python là một ngôn ngữ quen thuộc trong lĩnh vực Internet kết nối vạn vật
- **Làm game:** Pygame là một bộ module Python cross-platform được thiết kế để viết game cho cả máy tính và các thiết bị di động

2.2. Mã QRCode/BarCode

QR Code (mã QR) là viết tắt của Quick response code (Tạm dịch: Mã phản hồi nhanh), hoặc có thể gọi là Mã vạch ma trận (Matrix-barcode) hay Mã vạch 2 chiều (2D). Đây là một dạng thông tin được mã hóa để hiển thị sao cho máy có thể đọc được.

QR Code xuất hiện lần đầu tiên vào năm 1994, được tạo ra bởi Denso Wave (công ty con của Toyota). QR Code bao gồm những chấm đen và ô vuông mẫu trên nền trắng, có thể chứa những thông tin như URL, thời gian, địa điểm của sự kiện, mô tả, giới thiệu một sản phẩm nào đó,...

QR Code cho phép quét và đọc mã nhanh hơn bằng các thiết bị như máy đọc mã vạch hoặc điện thoại có camera với ứng dụng cho phép quét mã, vô cùng tiện lợi cho người dùng Barcode là thuật ngữ tiếng Anh được dịch sang tiếng Việt là "Mã vạch". Đây là một công nghệ phổ biến hiện nay được sử dụng để thu thập và nhận dạng dữ liệu thông qua một mã số, chữ số của một đối tượng nhất định nào đó. Đồng thời, mã vạch này có độ nhỏ lớn khác nhau và có khoảng trống song song xen kẽ nhau. Chúng được sắp xếp theo một quy tắc mã hóa để các máy quét và máy đọc mã vạch có thể nhận dạng và đọc được thông tin.

Nói một cách đơn giản hơn thì Barcode chính là sự thể hiện thông tin dưới hình dạng có thể thấy được từ mắt thường trên bề mặt sản phẩm, hàng hóa mà máy móc có thể đọc được.

2.3. Dịch máy

Dịch máy, nói một cách đơn giản là việc sử dụng phần mềm để dịch – từ văn bản hoặc giọng nói – từ ngôn ngữ này sang ngôn ngữ khác. Bằng cách sử dụng các thuật toán, mẫu và mô hình ngôn ngữ lấy từ cơ sở dữ liệu lớn của các bản dịch hiện có, nó có thể đề xuất bản dịch cho các chuyên gia ngôn ngữ hoặc trong một số trường hợp, tự động dịch số lượng lớn văn bản mà không cần sự tham gia của con người. Đối với ngữ cảnh, các yếu tố phần mềm nằm trong danh mục chủ đề (ví dụ: Y tế, pháp lý, khoa học), những nguồn thông tin trên mạng và bảng chú giải thuật ngữ.

Có nhiều kiểu dịch máy với các mức độ phức tạp khác nhau. Bản dịch có sự hỗ trợ của máy tính, bản dịch có sự hỗ trợ của con người và bản dịch tương tác. Chúng không giống như dịch máy, mỗi công cụ đều có các đặc điểm và bộ công cụ riêng biệt. Có ba loại dịch máy: Dịch máy thống kê (SMT), dịch máy dựa trên quy tắc và dịch máy nơ-ron.

PHẦN 3: XÂY DỰNG CHƯƠNG TRÌNH

3.1. Đặt vấn đề

Hiện nay, các ứng dụng hỗ trợ làm việc chưa có nhiều chức năng tích hợp. Tuy nhiên, yêu cầu về thời gian, thao tác thực hiện sao cho tối ưu là quan trọng nhất. Nếu mỗi lần làm một việc khác nhau cần phải mở một ứng dụng khác nhau sẽ làm tiêu tốn rất nhiều thời gian và dẫn đến việc có rất nhiều thao tác thừa lặp đi lặp lại trong quá trình làm việc.

Nhận thấy rõ được tầm quan trọng của những ứng dụng đa chức năng có khả năng tối ưu thời gian của người sử dụng khi mỗi công việc phải mở một ứng dụng khác nhau, từ đó Chương trình với chức năng: Tạo mã, quét mã QRCode, BarCode, hỗ trợ dịch, giải trí với game flappy bird ra đời góp phần hỗ trợ người sử dụng tối ưu thời gian và thao tác cho công việc của mình.

3.2. Phương pháp giải quyết vấn đề

Hiện nay ngôn ngữ lập trình Python vô cùng phổ biến, do đó khi xây dựng bằng ngôn ngữ lập trình này sẽ có rất nhiều phương pháp để giải quyết các vấn đề được đặt ra. Trong đề tài này sử dụng thư viện có sẵn trong Python để giải quyết các vấn đề liên quan đến chương trình.

3.3. Thiết kế, cài đặt thư viện và xây dựng các hàm xử lý

3.3.1. Cài đặt các thư viện

- Thư viện “Tkinter”

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 Tk GUI.

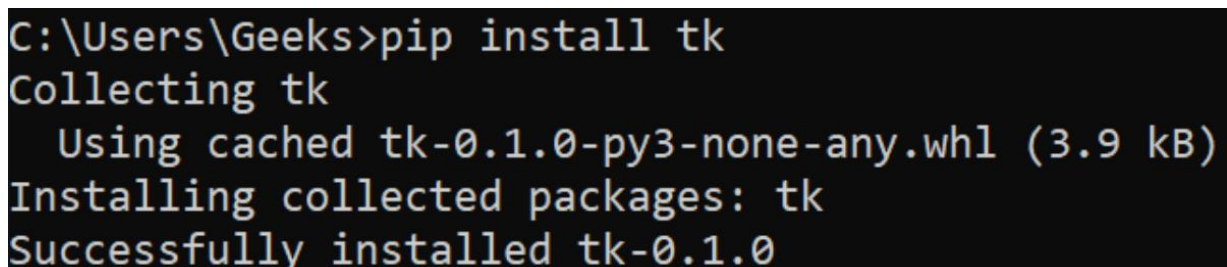
Bước 1: Kiểm tra xem thư viện googletrans đã được cài đặt hay chưa, dùng lệnh:

```
import tkinter as tk
```

Nếu báo lỗi ModuleNotFoundError: No module named 'tkinter', tức là thư viện tkinter chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install tk
```



```
C:\Users\Geeks>pip install tk
Collecting tk
  Using cached tk-0.1.0-py3-none-any.whl (3.9 kB)
Installing collected packages: tk
Successfully installed tk-0.1.0
```

Hình 3.1 Cửa sổ cài đặt thư viện Tkinter

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
from tkinter import *
```

- *Thư viện “PIL”*

Thư viện hình ảnh PIL là một mô-đun xử lý hình ảnh được phát triển cho Python. Nó cung cấp các công cụ xử lý hình ảnh giúp tạo, chỉnh sửa và xuất các tệp hình ảnh

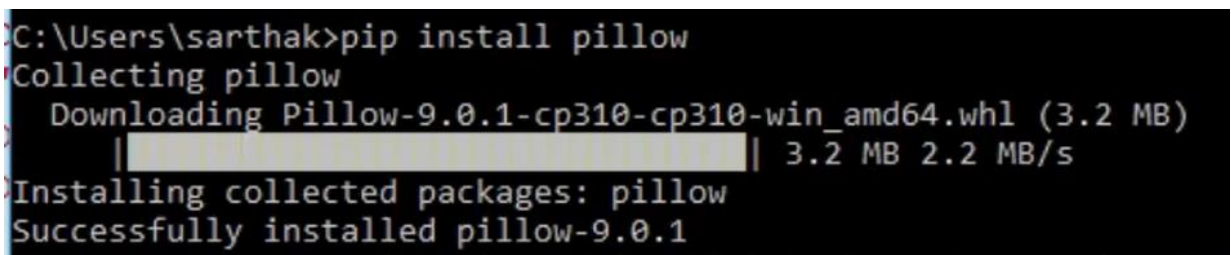
Bước 1: Kiểm tra xem thư viện PIL đã được cài đặt hay chưa, dùng lệnh:

```
import PIL
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'PIL'`, tức là thư viện PIL chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install Pillow
```



Hình 3.2 Cửa sổ cài đặt thư viện Pillow

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
from PIL import ImageTK
```

3.3.1.1. Tạo mã, quét mã QRCode/BarCode

- *Thư viện “QRCode”*

QRCode là một loại mã vạch ma trận là nhãn quang học có thể đọc được bằng máy có chứa thông tin về mặt hàng mà nó được gắn vào. Trên thực tế, mã QR thường chứa dữ liệu cho bộ định vị, số nhận dạng hoặc trình theo dõi trở đến một trang web hoặc ứng dụng.

Bước 1: Kiểm tra xem thư viện QRCode đã được cài đặt hay chưa, dùng lệnh:

```
import qrcode
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'qrcode'`, tức là thư viện qrcode chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install qrcode
```

```
(qr_code_venv) E:\code_learn\QRCode>pip install qrcode
Collecting qrcode
  Downloading https://files.pythonhosted.org/packages/42/87/4a3a77e59ab7493d64da1f69bf1
Collecting colorama; platform_system == "Windows" (from qrcode)
  Downloading https://files.pythonhosted.org/packages/44/98/5b86278fbbf250d239ae0ecb724
Collecting six (from qrcode)
  Using cached https://files.pythonhosted.org/packages/ee/ff/48bde5c0f013094d729fe4b031
Installing collected packages: colorama, six, qrcode
Successfully installed colorama-0.4.4 qrcode-6.1 six-1.15.0
```

Hình 3.3 Cửa sổ cài đặt thư viện Qrcode

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
import qrcode
```

- *Thư viện “OpenCV (CV2)”*

OpenCV viết tắt cho Open Source Computer Vision Library. OpenCV là thư viện nguồn mở hàng đầu cho Computer Vision và Machine Learning, và hiện có thêm tính năng tăng tốc GPU cho các hoạt động theo real-time.

Bước 1: Kiểm tra xem thư viện OpenCV đã được cài đặt hay chưa, dùng lệnh:

```
import cv2
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'cv2'`, tức là thư viện opencv chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install opencv-python
```

```
(qr_code_venv) E:\code_learn\QRCode>pip install qrcode
Collecting qrcode
  Downloading https://files.pythonhosted.org/packages/42/87/4a3a77e59ab7493d64da1f69bf1
Collecting colorama; platform_system == "Windows" (from qrcode)
  Downloading https://files.pythonhosted.org/packages/44/98/5b86278fbbf250d239ae0ecb724
Collecting six (from qrcode)
  Using cached https://files.pythonhosted.org/packages/ee/ff/48bde5c0f013094d729fe4b031
Installing collected packages: colorama, six, qrcode
Successfully installed colorama-0.4.4 qrcode-6.1 six-1.15.0
```

Hình 3.4 Cửa sổ cài đặt thư viện OpenCV

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
import cv2
```

- *Thư viện “Numpy”*

NumPy là một gói Python là viết tắt của Numerical Python. Đây là thư viện cốt lõi cho scientific computing, nó chứa một đối tượng mảng n chiều mạnh mẽ, cung cấp các công cụ để tích hợp C, C ++, v.v. Nó cũng hữu ích trong đại số tuyến tính, random number capability, NumPy Array cũng có thể được sử dụng như multi-dimensional container hiệu quả cho dữ liệu chung.

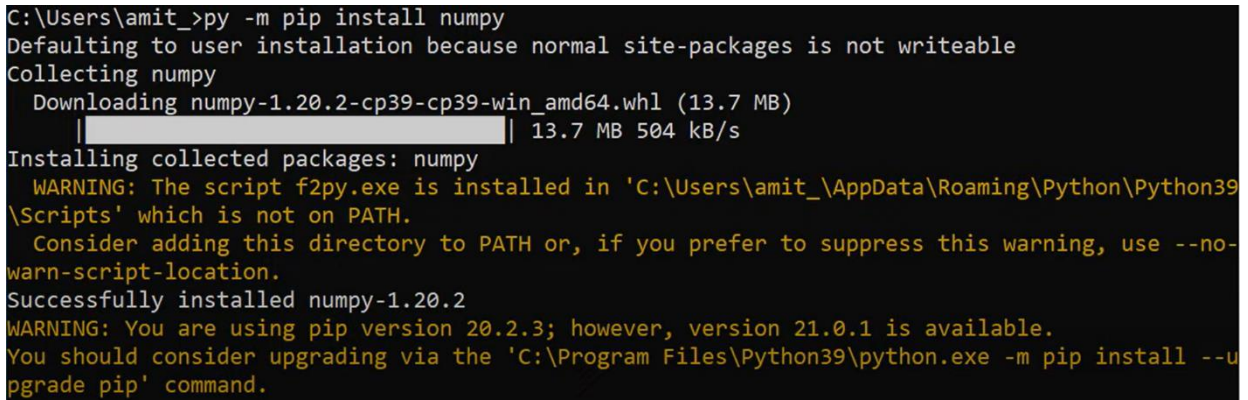
Bước 1: Kiểm tra xem thư viện Numpy đã được cài đặt hay chưa, dùng lệnh:

```
import numpy as np
```

Nếu báo lỗi ModuleNotFoundError: No module named 'Numpy', tức là thư viện numpy chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install numpy
```



```
C:\Users\amit>py -m pip install numpy
Defaulting to user installation because normal site-packages is not writeable
Collecting numpy
  Downloading numpy-1.20.2-cp39-cp39-win_amd64.whl (13.7 MB)
    |#####| 13.7 MB 504 kB/s
Installing collected packages: numpy
  WARNING: The script f2py.exe is installed in 'C:\Users\amit\AppData\Roaming\Python\Python39\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed numpy-1.20.2
WARNING: You are using pip version 20.2.3; however, version 21.0.1 is available.
You should consider upgrading via the 'C:\Program Files\Python39\python.exe -m pip install --upgrade pip' command.
```

Hình 3.5 Cửa sổ cài đặt thư viện Numpy

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
import numpy as np
```

- *Thư viện “Pyzbar”*

Để có thể giải mã QRCode và Barcode, có thể sử dụng các thư viện mã nguồn mở, trong đó Zbar là một thư viện đa nền tảng, khá gọn nhẹ và cho kết quả nhận dạng tốt. Có thể nhận dữ liệu trực tiếp từ Camera hoặc từ file ảnh, sau đó sẽ được đưa qua bộ giải mã (decoder) để xử lý và nhận dạng.

Bước 1: Kiểm tra xem thư viện Numpy đã được cài đặt hay chưa, dùng lệnh:

```
import pyzbar
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'pyzbar'`, tức là thư viện `pyzbar` chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install pyzbar
```

```
C:\Users\hiluo>pip install pyzbar
Requirement already satisfied: pyzbar in c:\users\hiluo\appdata\local\programs\python\python311\lib\site-packages (0.1.9)
```

Hình 3.6 Cửa sổ cài đặt thư viện *Pyzbar*

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh :

```
from pyzbar.pyzbar import decode
```

3.3.1.2. Dịch máy

- Thư viện “*GoogleTrans*”

Python `googletrans` là một module để dịch văn bản. Nó sử dụng API Google Translate Ajax để phát hiện ngôn ngữ và dịch văn bản sang ngôn ngữ mong muốn.

Bước 1: Kiểm tra xem thư viện `googletrans` đã được cài đặt hay chưa, dùng lệnh:

```
import googletrans
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'googletrans'`, tức là thư viện `googletrans` chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install googletrans
```

```
C:\Users\gf63>pip install googletrans
Collecting googletrans
  Downloading googletrans-3.0.0.tar.gz (17 kB)
Collecting httpx==0.13.3
  Downloading httpx-0.13.3-py3-none-any.whl (55 kB)
  | 55 kB 860 kB/s
Collecting hstspreload
  Downloading hstspreload-2020.12.22-py3-none-any.whl (994 kB)
  | 994 kB 2.2 MB/s
Collecting sniffio
  Downloading sniffio-1.2.0-py3-none-any.whl (10 kB)
Collecting idna==2.*
  Downloading idna-2.10-py2.py3-none-any.whl (58 kB)
  | 58 kB 3.0 MB/s
```

Hình 3.7 Cửa sổ cài đặt thư viện *GoogleTrans*

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
From googletrans import Translator
```

- *Thư viện “Pyttsx3”*

Pyttsx3 là một thư viện chuyển đổi văn bản thành giọng nói trong Python. Không giống như các thư viện thay thế, nó hoạt động ngoại tuyến và tương thích với cả Python 2 và 3.

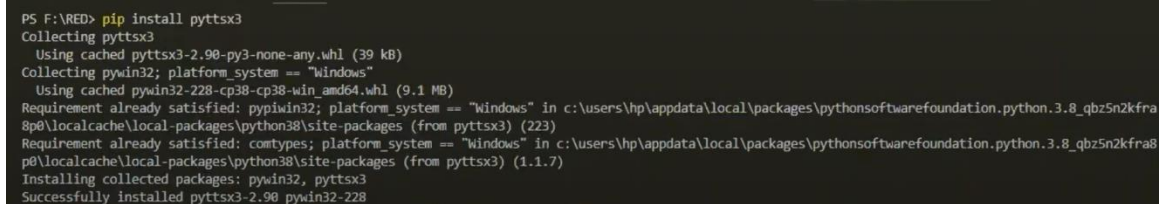
Bước 1: Kiểm tra xem thư viện Pyttsx3 đã được cài đặt hay chưa, dùng lệnh:

```
import pyttsx3
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'pyttsx3'`, tức là thư viện `pyttsx3` chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install pyttsx3
```



```
PS F:\RED> pip install pyttsx3
Collecting pyttsx3
  Using cached pyttsx3-2.90-py3-none-any.whl (39 kB)
Collecting pywin32; platform_system == "Windows"
  Using cached pywin32-228-cp38-cp38-win_amd64.whl (9.1 MB)
Requirement already satisfied: pypiwin32; platform_system == "Windows" in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.8_qbz5n2kfra8p0\localcache\local-packages\python38\site-packages (from pyttsx3) (223)
Requirement already satisfied: ctypes; platform_system == "Windows" in c:\users\hp\appdata\local\packages\pythonsoftwarefoundation.python.3.8_qbz5n2kfra8p0\localcache\local-packages\python38\site-packages (from pyttsx3) (1.1.7)
Installing collected packages: pywin32, pyttsx3
Successfully installed pyttsx3-2.90 pywin32-228
```

Hình 3.8 Cửa sổ cài đặt thư viện Pyttsx3

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

```
import pyttsx3
```

- *Thư viện “Speech Recognition”*

Speech Recognition là một thư viện nhận dạng giọng nói trong Python.

Bước 1: Kiểm tra xem thư viện *Speech Recognition* đã được cài đặt hay chưa, dùng lệnh:

```
import speech_recognition
```

Nếu báo lỗi `ModuleNotFoundError: No module named 'speechrecognition'`, tức là thư viện `speech_recognition` chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

```
pip install speech_recognition
```

```
(venv) C:\Users\cmano\PycharmProjects\Speech>pip install SpeechRecognition
Collecting SpeechRecognition
  Using cached SpeechRecognition-3.8.1-py2.py3-none-any.whl (32.8 MB)
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.8.1

(venv) C:\Users\cmano\PycharmProjects\Speech
```

Hình 3.9 Cửa sổ cài đặt thư viện Speech Recognition

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

Import speech_recognition

3.3.1.3. Game Flappy Bird

- Thư viện “Pygame”

Pygame là một bộ mô-đun Python đa nền tảng được thiết kế để viết trò chơi điện tử. Nó bao gồm đồ họa máy tính và thư viện âm thanh được thiết kế để sử dụng với ngôn ngữ lập trình Python.

Bước 1: Kiểm tra xem thư viện Pygame đã được cài đặt hay chưa, dùng lệnh:

import pygame

Nếu báo lỗi `ModuleNotFoundError: No module named 'pygame'`, tức là thư viện pygame chưa được cài đặt, tiếp tục bước 2.

Bước 2: Cài đặt thư viện bằng lệnh:

pip install

```
C:\Users\sarthak>pip install pygame
Collecting pygame
  Downloading pygame-2.1.2-cp310-cp310-win_amd64.whl (8.4 MB)
    | 8.4 MB 1.3 MB/s
Installing collected packages: pygame
Successfully installed pygame-2.1.2
WARNING: You are using pip version 21.2.4; however, version 22.0.3 is available.
You should consider upgrading via the 'C:\Users\sarthak\AppData\Local\Programs\Python\Python310\python.exe -m pip install --upgrade pip' command.
```

Hình 3.10 Cửa sổ cài đặt thư viện Pygame

Bước 3: Sau khi quá trình cài đặt hoàn tất, để sử dụng thư viện này cần thêm vào file source code câu lệnh:

import pygame

3.3.2. Các hàm xử lý tiêu biểu

3.3.2.1. Tạo mã, quét mã QRCode/BarCode

- Hàm tạo QRCode

```
def CreateQRCode():  
  
    Input_Length = Input.get("1.0", "end-1c")  
  
    if Input_Length == '':  
        messagebox.showerror('Input Error', "Please enter the characters!")  
    else:  
        #Create - L:Low, M:Medium, Q:Quarter, H:High  
        qr = qrcode.QRCode(version = 4, error_correction = qrcode.ERROR_CORRECT_M, box_size=5, border=2) #Tạo đối tượng QR  
        qr.add_data(Input_Length) #Thêm dữ liệu  
        qr.make(fit = True) #Tận dụng khung hình để tạo ra mã QRcode tối ưu không gian tròn  
  
        CreateImage = qr.make_image(fill_Color = 'black', back_color = 'white') #Màu của QRCode  
        CreateImage.save('QRCode/MyQRCode.png') #Lưu thành tên MyQRCode.png  
  
        QR_Output = Text(frameMain, width = 250, height = 250, borderwidth= 0, relief= GROOVE, font = ('verdana', 15))  
        QR_Output.place(anchor='center', x = 450, y = 385)  
  
        ImageShow = ImageTk.PhotoImage(Image.open("MyQRCode.png"))  
        ImageLabel = ttk.Label(QR_Output, image=ImageShow)  
        ImageLabel.image = ImageShow  
        ImageLabel.pack()  
  
        Input.delete(1.0, 'end')
```

Hình 3.11 Hàm tạo QRCode

Trước tiên chương trình sẽ kiểm tra dữ liệu người dùng nhập vào, nếu dữ liệu rỗng sẽ hiển thị thông báo yêu cầu nhập dữ liệu.

Nếu đã có dữ liệu nhập, chương trình sẽ tiến hành tạo đối tượng qr. Và nhận dữ liệu từ dữ liệu nhập vào.

Tạo hình ảnh QR với màu đen và khung màu trắng. Lưu file ở Thư mục QRCode với tên tập tin là MyQRCode.png.

Cuối cùng, tạo một vùng hiển thị QR và hiển thị mã QR đã được tạo ra Frame.

- *Hàm quét và đọc QRCode*

```
def ReadQrCode():
    Input.delete(1.0, 'end')
    #Scan and Read
    Camera = cv2.VideoCapture(0) #Phát hiện camera của máy tính
    while True:
        _, frame = Camera.read()
        Key_pressed = cv2.waitKey(1)
        File_open = open('QRCode/QRCode_logs.txt', 'a') #Mở tập tin Qrcode_logs và cho phép ghi đè

        for code in decode(frame): #Tạo biến code thuộc decode
            art = np.array([code.polygon], np.int32) #Tạo mảng 1 chiều
            art = art.reshape((-1, 1, 2)) #1 dòng, 2 cột (seft, shape, order)
            cv2.polylines(frame, [art], True, (255, 0, 255), 2) #Nối kín các đỉnh với nhau

        cv2.imshow('QRCode Scanner', frame)

        if Key_pressed == 13: #Khi bấm phím ENTER sẽ tiến hành đọc QR
            data = code.data.decode('UTF-8')
            File_open.write(data + '\n')
            Input.insert('end', data)
            Input.insert('end', '\n')

        if Key_pressed == 27: #Khi ấn phím ESC thì thoát
            break

    File_open.close()
    Camera.release()
    cv2.destroyAllWindows()
```

Hình 3.12 Hàm quét và đọc QRCode

Trước tiên chương trình sẽ xóa tất các nội dung trong khung nhập dữ liệu. Vì khi gọi chức năng quét và đọc QRCode thì nơi nhập dữ liệu sẽ là nơi hiển thị nội dung của QRCode quét được.

Chương trình Phát hiện camera của máy tính qua hàm VideoCapture có trong OpenCV.

Vòng lặp while thực hiện mở tập tin lưu dữ liệu quét được, thực hiện tạo một mảng một chiều và tạo vùng phát hiện mã, sau đó vẽ lại vùng đó bằng hàm polylines để phát hiện mã ở nhiều góc cách khác nhau bằng cách vẽ và nối các đỉnh lại với nhau.

Sau khi phát hiện được mã, người dùng ấn phím Enter trên bàn phím để thực hiện đọc mã và hiển thị ra frame cũng như lưu vào file.

Bấm phím ESC để thoát khỏi chương trình.

3.3.2.2. Dịch máy

- Hàm đọc file

```
def ReadFile(): #Đọc file từ tập tin
    Clear()
    f = open("Translate/intPut.txt", encoding= 'utf8')
    readText = f.read()
    f.close()
    Text_Input.insert('end', readText)
```

Hình 3.13 Hàm đọc file nhập vào frame input

Hàm này sẽ nhập nội dung của file input dạng plain text có sẵn trong thư mục source để nhập vào ô input.

- Hàm nhận dạng giọng nói từ microphone thành văn bản

```
def Microphone(): #Nhập từ microphone
    with speech_recognition.Microphone() as mic:
        audio = ear.listen(mic)
    try:
        Text_Input.insert('end', ear.recognize_google(audio, language="vi-VI"))
    except:
        Text_Input.insert('end', "Error! Error, can't hear clearly")
```

Hình 3.14 Hàm nhận dạng giọng nói, chuyển thành văn bản

Hàm này sẽ thực hiện nhiệm vụ ghi nhận lại âm thanh từ microphone của người dùng khi người dùng có nhu cầu nhập văn bản cần dịch bằng lời nói. Và chỉ hoạt động khi người dùng click vào nút “Microphone”.

- Hàm dịch văn bản

```
def Translate(): #Dịch
    lang_length = Text_Input.get("1.0", "end-1c")
    choose = choose_language.get()

    if lang_length == '':
        messagebox.showerror('Input Error', "Please enter the characters to translate!")
    else:
        Text_Output.delete(1.0, 'end')
        translator = Translator()
        outPut = translator.translate(lang_length, dest=choose)
        Text_Output.insert('end', outPut.text)
```

Hình 3.15 Hàm dịch văn bản đã nhập

Hàm này có nhiệm vụ phát hiện ngôn ngữ được nhập vào Frame Input và chuyển nó thành ngôn ngữ đã được lựa chọn tại Combobox ngôn ngữ cần phiên dịch.

- Hàm chuyển văn bản thành giọng nói

```
def Speech(): #Phát âm
    Text_Speech.say(Text_Output.get("1.0", "end-1c"))
    Text_Speech.runAndWait()
```

Hình 3.16 Hàm chuyển văn bản thành giọng nói

Chức năng của hàm này là sẽ chuyển nội dung đã được dịch của input thành giọng nói nếu người dùng có nhu cầu nghe nó.

3.3.2.3. Game Flappy Bird

- Hàm vẽ các vật thể

```
def Draw_Floor(): #Vẽ cỏ bên dưới
    screen.blit(floor,(floor_x_pos,650))
    screen.blit(floor,(floor_x_pos+432,650))

def Create_Pipe(): #Tạo các đường ống
    random_pipe_pos = random.choice(pipe_height)
    bottom_pipe = pipe_surface.get_rect(midtop =(500,random_pipe_pos))
    top_pipe = pipe_surface.get_rect(midtop =(500,random_pipe_pos-650))
    return bottom_pipe, top_pipe

def Draw_Pipe(pipes): #Vẽ các đường ống
    for pipe in pipes:
        if pipe.bottom ≥ 600 :
            screen.blit(pipe_surface,pipe)
        else:
            flip_pipe = pygame.transform.flip(pipe_surface,False,True)
            screen.blit(flip_pipe,pipe)
```

Hình 3.17 Hàm vẽ các vật thể trong game

Chức năng của các hàm này là sẽ vẽ ra các đám cỏ ở dưới chân, các đường ống để chim bay qua.

- Hàm kiểm tra va chạm

```
def Check_Collision(pipes): #Phát hiện và kiểm tra va chạm
    for pipe in pipes:
        if bird_rect.colliderect(pipe):
            hit_sound.play()
            return False
    if bird_rect.top ≤ -75 or bird_rect.bottom ≥ 650:
        return False
    return True
```

Hình 3.18 Hàm kiểm tra va chạm

Chức năng của hàm này là sẽ kiểm tra chuyển động của chim, nếu nó va chạm vào các vật thể quy định thì sẽ tính là thua.

- *Hàm di chuyển và hiệu ứng*

```
def Move_Pipe(pipes): #Di chuyển của các đường ống
    for pipe in pipes :
        pipe.centerx -= 5
    return pipes

def Rotate_Bird(bird1): #Di chuyển của Bird
    new_bird = pygame.transform.rotozoom(bird1, -bird_movement * 1, 1)
    return new_bird

def Bird_Animation(): #Hiệu ứng của Bird
    new_bird = bird_list[bird_index]
    new_bird_rect = new_bird.get_rect(center = (100,bird_rect.centery))
    return new_bird, new_bird_rect
```

Hình 3.19 Hàm tạo hiệu ứng và hàm di chuyển của chim Bird

Chức năng của hàm này là tạo ra các hiệu ứng của chim như: hiệu ứng cánh khi bay lên, bay xuống, ... Ngoài ra nó còn tạo ra các đối tượng ống di chuyển khi game bắt đầu.

- *Hàm tính điểm và hiển thị điểm*

```
def Score_Display(game_state): #Hiển thị điểm
    if game_state == 'main game':
        score_surface = game_font.render(str(int(score)),True,(255,255,255))
        score_rect = score_surface.get_rect(center = (216,100))
        screen.blit(score_surface,score_rect)

    if game_state == 'game_over':
        score_surface = game_font.render(f'Score: {int(score)}',True,(255,255,255))
        score_rect = score_surface.get_rect(center = (216,50)) #Score postion
        screen.blit(score_surface,score_rect)

        high_score_surface = game_font.render(f'High Score: {int(high_score)}',True,(255,255,255))
        high_score_rect = high_score_surface.get_rect(center = (216,120)) #High Score postion
        screen.blit(high_score_surface,high_score_rect)

def Update_Score(score,high_score): #Kiểm tra và cập nhật điểm cao nhất
    if score > high_score:
        high_score = score
    return high_score
```

Hình 3.20 Hàm tính điểm và hiển thị điểm

Chức năng của hàm này là hiển thị điểm ra màn hình khi người chơi bắt đầu trò chơi và cập nhật lại điểm khi người chơi đạt một mốc mới, điểm mới.

- *Hàm xử lý suốt trò chơi*

```

while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
        if event.type == pygame.KEYDOWN:
            if event.key == pygame.K_SPACE and game_active:
                bird_movement = 0
                bird_movement -= 7
                flap_sound.play()
            if event.key == pygame.K_SPACE and game_active==False:
                game_active = True
                pipe_list.clear()
                bird_rect.center = (100,384)
                bird_movement = 0
                score = 0

        if event.type == spawnpipe:
            pipe_list.extend(Create_Pipe())
        if event.type == birdflap:
            if bird_index < 2:
                bird_index += 1
            else:
                bird_index =0
            bird, bird_rect = Bird_Animation()

    screen.blit(bg,(0,0))
    if game_active:
        #Bird
        bird_movement += gravity
        rotated_bird = Rotate_Bird(bird)
        bird_rect.centery += bird_movement
        screen.blit(rotated_bird,bird_rect)
        game_active= Check_Collision(pipe_list)

        #Pipe
        pipe_list = Move_Pipe(pipe_list)
        Draw_Pipe(pipe_list)
        score += 0.01
        Score_Display('main game')
        score_sound_countdown -= 1
        if score_sound_countdown ≤ 0:
            score_sound.play()
            score_sound_countdown = 100
    else:
        screen.blit(game_over_surface,game_over_rect)
        high_score = Update_Score(score,high_score)
        Score_Display('game_over')

    #Bot
    floor_x_pos -= 1
    Draw_Floor()
    if floor_x_pos ≤ -432:
        floor_x_pos =0

    pygame.display.update()
    clock.tick(120)

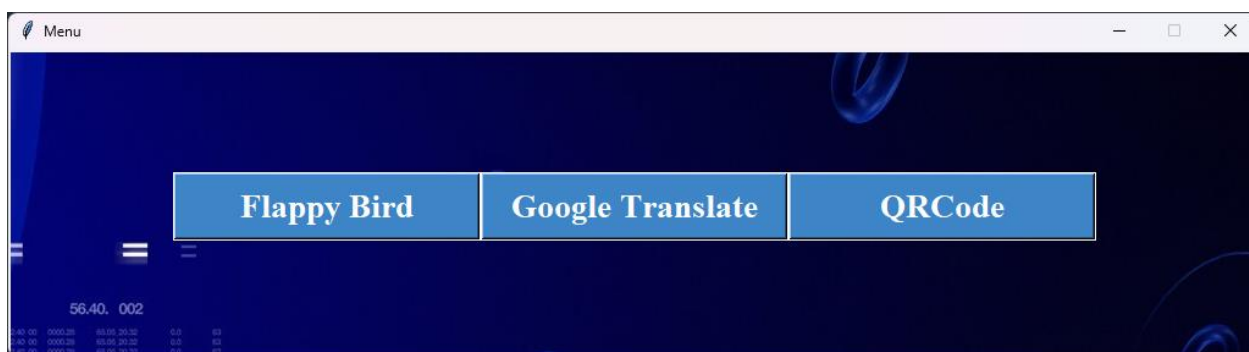
```

Hình 3.21 Hàm xử lý

Nó thực hiện nhiệm vụ xuyên suốt khi người chơi bắt đầu chơi game, thực hiện phát hiện các sự kiện phím như: Bấm phím để thoát, bấm phím để chim bay lên và cập nhật hoạt ảnh cho chim bird, điểm cho người chơi, ...

3.4. Giới thiệu Demo và hướng dẫn sử dụng

3.4.1. Menu chính



Hình 3.22 Giao diện Menu chính

Khi chương trình khởi động, Menu sẽ hiển thị và cho người dùng chọn chức năng cần thiết, hiện tại có 3 chức năng: Flappy Bird, Google Translate và QRCode. Khi bấm vào một chức năng nào đó sẽ hiển thị giao diện của chức năng đó.

3.4.2. Tạo mã, quét mã QRCode/BarCode



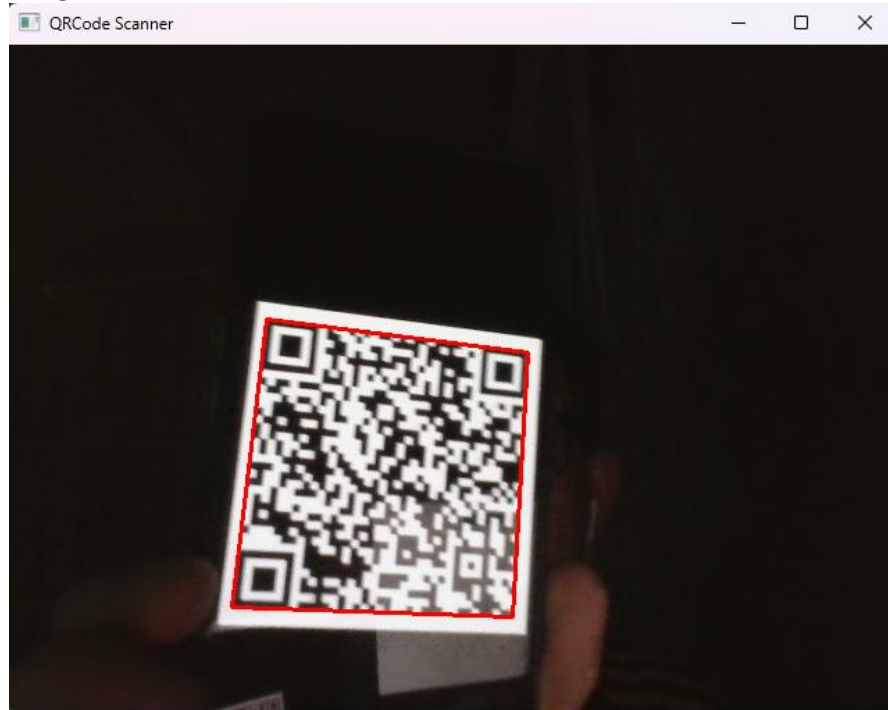
Hình 3.23 Giao diện tạo mã QRCode

Chức năng tạo mã QR:

Dữ liệu hiện tại chỉ có thể chuyển đổi thành QRCode, dữ liệu sẽ được nhập từ Frame và sau khi xử lý, QRCode sẽ hiển thị ở dưới Frame.

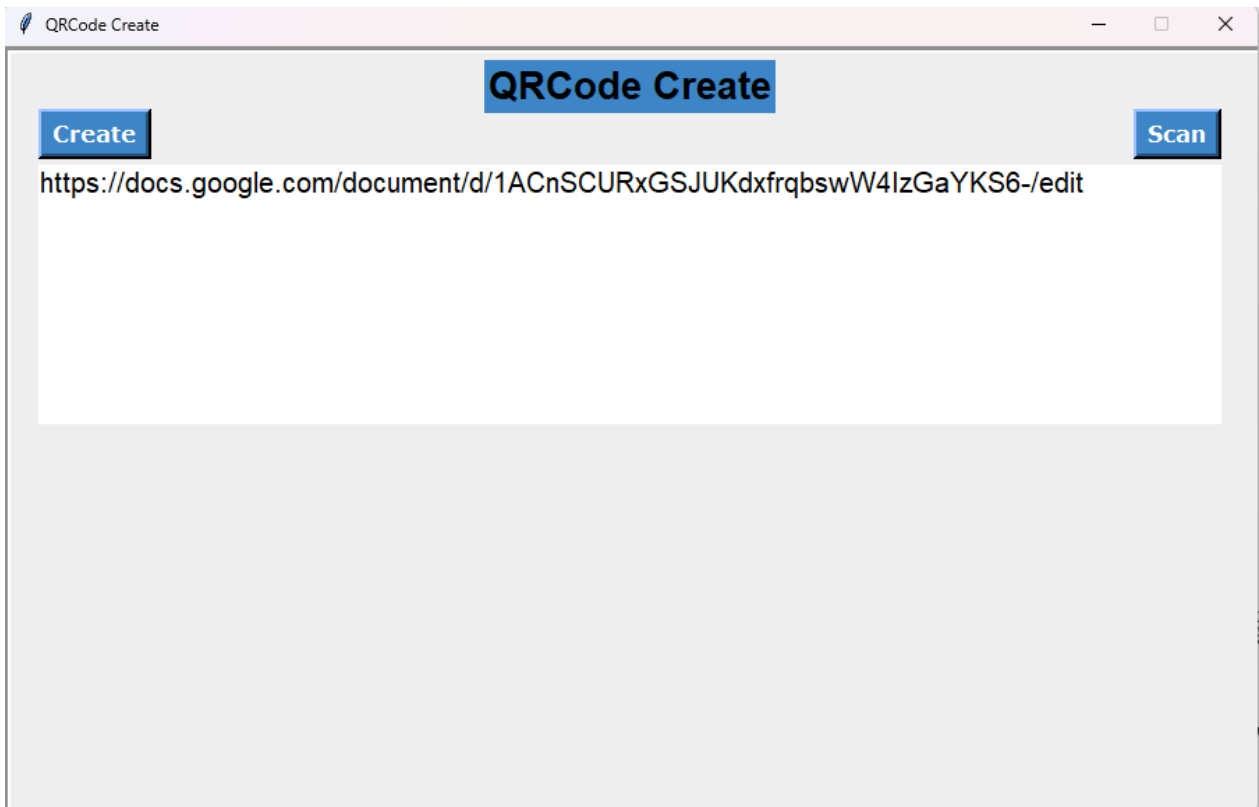
Tập tin lưu trữ mã QRCode sẽ được lưu lại trong thư mục QRCode của dự án.

Chức năng Scan mã:



Hình 3.24 Giao diện Camera nhận dạng mã

Chương trình sẽ sử dụng camera của thiết bị để nhận dạng QRCode mà người dùng cần quét, sau khi nhận dạng được chương trình sẽ vẽ lại một khung nhận dạng mã đó.

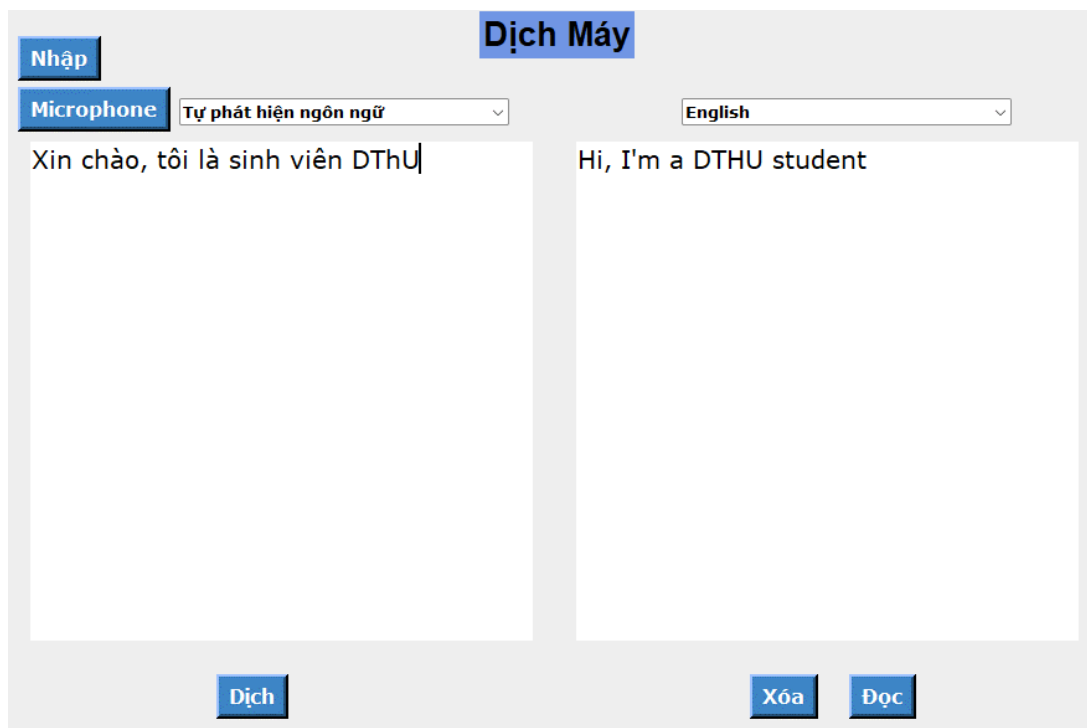


Hình 3.25 Giao diện đọc QR và hiển thị dữ liệu đọc được

Để đọc mã chỉ cần ấn phím ENTER. Dữ liệu quét được sẽ hiển thị trên khung Frame và lưu vào tập tin QRCode_logs.txt trong thư mục QRCode của dự án.

Để thoát chương trình ấn phím ESC.

3.4.3. Dịch máy



Hình 3.26 Giao diện dịch nội dung

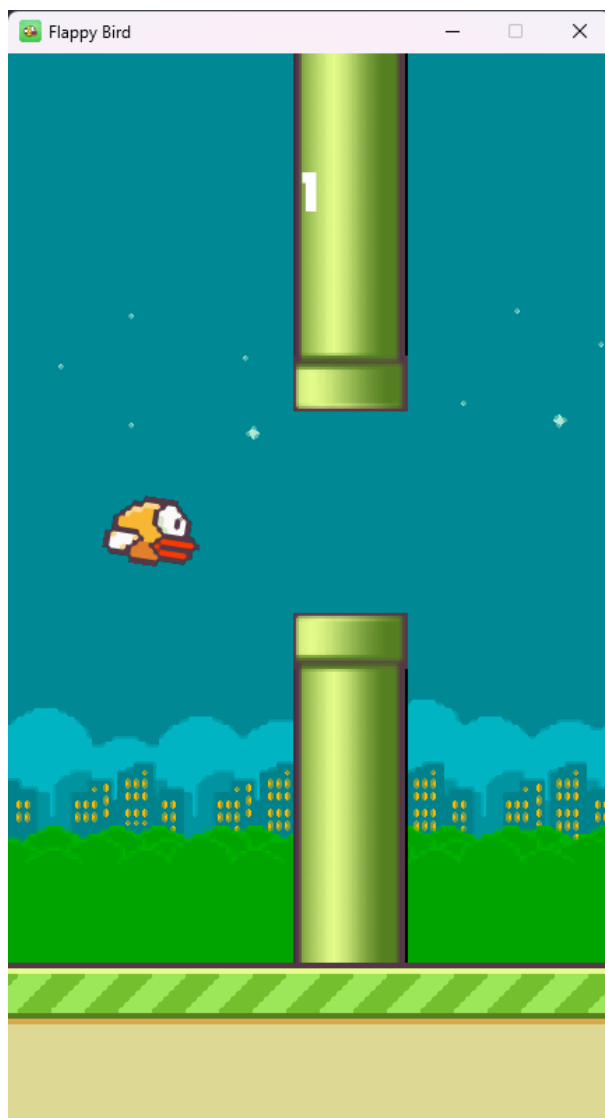
Người dùng có thể nhập trực tiếp nội dung cần dịch lên khung Frame, chương trình sẽ tự phát hiện ngôn ngữ đã nhập và để dịch được cần phải chọn ngôn ngữ cần dịch sang ở combobox bên phải.

Đối với nhập từ Microphone, đảm bảo microphone được bật, không gian không có tiếng ồn lớn, không bị gây nhiễu và phải có internet. Bấm vào nút Microphone để gọi microphone.

Đối với nhập từ tập tin, người dùng bấm vào nút “Nhập” để nhập nội dung từ tập tin, tập tin được đặt trong thư mục Translate/Input.txt của dự án.

Đối với tính năng Đọc (Chỉ đối với ngôn ngữ English), người dùng sau khi dịch văn bản. Muốn nghe nội dung văn bản theo ngôn ngữ của người bản xứ, chỉ cần bấm vào nút “Đọc” là có thể nghe được.

3.4.4. Game Flappy Bird



Hình 3.27 Giao diện trò chơi

Khi trò chơi bắt đầu, người dùng bấm phím Space để điều khiển chim Bird di chuyển, nếu chim va chạm vào cột hoặc di chuyển quá cao, quá thấp sẽ bị tính là thua.

Màn hình sẽ hiển thị điểm đạt được và điểm cao nhất khi kết thúc màn chơi.

PHẦN 4: KẾT LUẬN

4.1. Kết quả đạt được

4.1.1. Về lý thuyết

Củng cố và nâng cao được kiến thức về ngôn ngữ lập trình Python và cách sử dụng các thư viện trong Python như các thư viện:

- OpenCV
- Pygame
- Tkinter
- GoogleTrans
- Speech_Recognition
- Pyttsx3
- PIL
- Qrcode
- Pyzbar
- Numby

4.1.2. Về thực hành

Xây dựng thành công chương trình đa chức năng như:

- Hỗ trợ tạo mã QRCode, quét mã QRCode/Barcode bằng Camera và lưu lại dữ liệu đọc được.
- Dịch văn bản nhập từ bàn phím, nhập từ microphone, nhập từ tập tin, chuyển văn bản thành giọng nói, tự động phát hiện ngôn ngữ, dịch đa dạng ngôn ngữ.
- Giải trí với game Flappy Bird.

4.1.3. Về kỹ năng

Qua việc xây dựng thành công chương trình đa chức năng trên, đã phần nào góp phần cải thiện được khả năng lập trình, xử lý những vấn đề phát sinh trong quá trình thực hiện và sử dụng thành thạo hơn phần mềm soạn thảo mã nguồn Visual Studio Code và nâng cao kiến thức về ngôn ngữ lập trình Python.

4.2. Hạn chế

Các vấn đề hạn chế còn tồn đọng ít nhiều ảnh hưởng đến người dùng như:

Chương trình này được xây dựng dựa trên các thư viện có sẵn của ngôn ngữ lập trình Python. Một số thư viện nếu không có kết nối mạng thì sẽ không thể sử dụng được.

4.3. Hướng phát triển

Cải thiện giao diện Menu chính, thêm hiệu ứng khi người dùng bấm vào.

Cải thiện tốc độ của trò chơi Flappy Bird.

Cải thiện khả năng xử lý của chương trình Dịch máy.

Tối ưu khả năng hoàn thiện mã QRCode khi tạo mã.

TÀI LIỆU THAM KHẢO

- [1] Lê Minh Thư, Nguyễn Quốc Anh, Nguyễn Thị Mỹ Dung, Nguyễn Thị Thanh Thảo 05/2022, *Bài giảng Lập trình python và ứng dụng*.
- [2] PyPi: Python community, 14/06/2020, <https://pypi.org/project/googletrans/>, 29/12/2022.
- [3] PyPi: Python community, 07/07/2020, <https://pypi.org/project/pyttsx3>, 29/12/2022.
- [4] PyPi: Python community, 19/07/2020, <https://pypi.org/project/PyAudio>, 30/12/2022.
- [5] PyPi: Python community, 04/12/2022, <https://pypi.org/project/SpeechRecognition>, 10/12/2022.
- [6] Python: Python Software Foundation, 10/9/2022, <https://docs.python.org/3/library/tkinter.html>, 29/12/2022.
- [7] PyPi: Python community, 27/12/2021, <https://pypi.org/project/pygame/>, 01/01/2022.
- [8] PyPi: Python community, 30/12/2022, <https://pypi.org/project/opencv-python/> 01/01/2023.

PHỤ LỤC

Hướng dẫn cài đặt

Cài đặt phần mềm Visual Studio Code

Tải bộ cài tại: <https://code.visualstudio.com/docs?dv=win>

Bước 1. Sau khi tải xong, chạy file VSCodeUserSetup.exe.

Bước 2. Nhấp vào Next để cài đặt. Tiếp theo đồng ý điều khoản sử dụng.

Bước 3: Lựa chọn nơi cài đặt (Nên để mặc định) sau đó nhấn Next.

Bước 4. Các bước tiếp theo tiếp tục nhấn Next cho tới khi hoàn tất. Trong quá trình này, nên tích chọn vào 2 chức năng: (1) Add “*Open with Code*” action to Windows Explorer file context menu và (2) Add “*Open with Code*” action to Windows Explorer directory context menu. Việc này giúp bạn có thể click chuột phải vào thư mục sẽ có lựa chọn mở bằng VS Code.

Bước 5. Cài đặt hoàn tất, mở phần mềm để sử dụng.

Cài đặt môi trường Python 3.11

Tải bộ cài tại: <https://www.python.org/downloads>

Bước 1. Sau khi tải xong, chạy file python-3.11.1.exe.

Bước 2. Tick vào ô Add Python 3.11.1 to PATH và chọn Install.

Bước 3: Khi cửa sổ hiển thị Setup was successful là ta đã cài đặt thành công môi trường Python. Bấm Close để tắt. Cài đặt môi trường Python hoàn tất.