



ระบบการส่งมอบใบประกาศนียบัตรด้วย Hash

สมาชิก

นาย รนพัฒน์ เอี่ยมสำอางค์ 651290

นาย วุฒิชัย ໂສระชาติ 651424

นาย ปัณณวิชญ์ พวงทอง 651699

ວັດຖຸປະສົງຄໍຂອງໂຄຮ່ານ

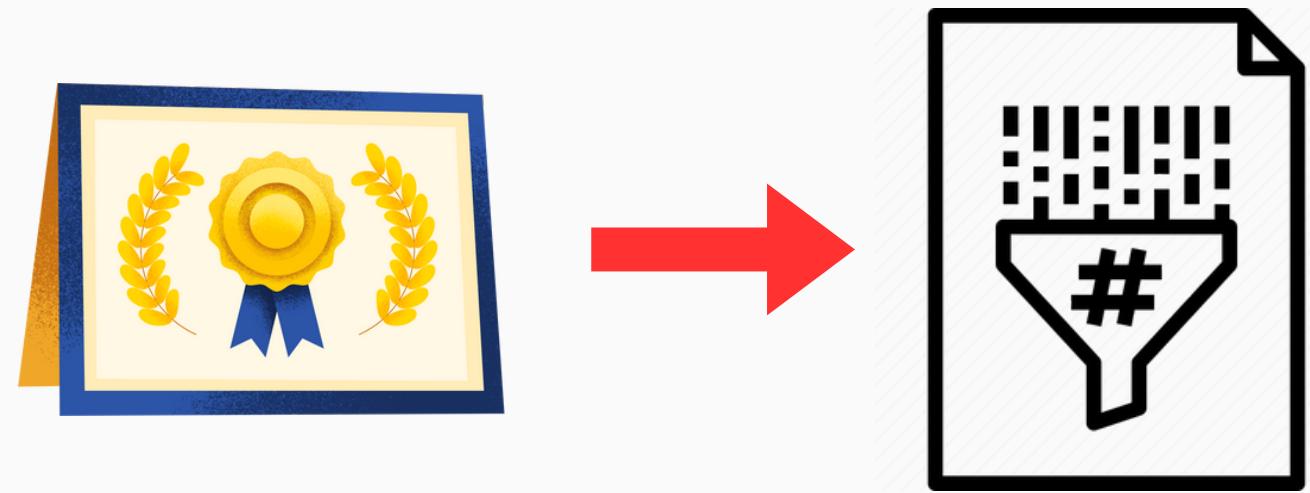


ເພື່ອເສນອຮະບບທີ່ໃຊ້ໃນການສັງມູນບັນຍາໃນຮູບແບບຂອງສິນກຣັພຢົດຈິງກໍລ ຊຶ່ງເປັນອົກກາງເລື່ອກໃນການເກີບຮັກເຫຼາໃນປະກາສນີຍບັຕາດິຈິງກໍລ ທີ່ຜູ້ຮັບປະກາສນີຍບັຕາສາມາດຈັດເກີບໄດ້ອ່າງດາວໂຫຍມ ເກີດການສູ້ລູ່ຫາຍຫຼືການປຸລອມແປລ່ງ ຮວມດື່ຳຜູ້ຮັບສາມາດກຳນົດຮຽນໄດ້ທຸກທີ່ຖຸກເວລາເພີ່ມແລ້ວ ອົບເກອຣີເນີຕໃນການເຫັນໃຊ້ງານ

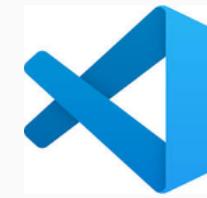


ขอบเขต และพังก์ชัน

- พัฒนาระบบที่ใช้ในการเก็บข้อมูลและแสดงข้อมูลของผู้รับใบประกาศนียบัตร
- พัฒนาระบบที่ใช้ในการส่งมอบใบประกาศนียบัตรในรูปแบบ Hash



ຖុបកី គេរីអងមើនកំិច្ច និងទានវិជ្ជកំើយុខ



- Visual Studio Code
- Metamask ឱ្យប្រើប្រាស់
- Python

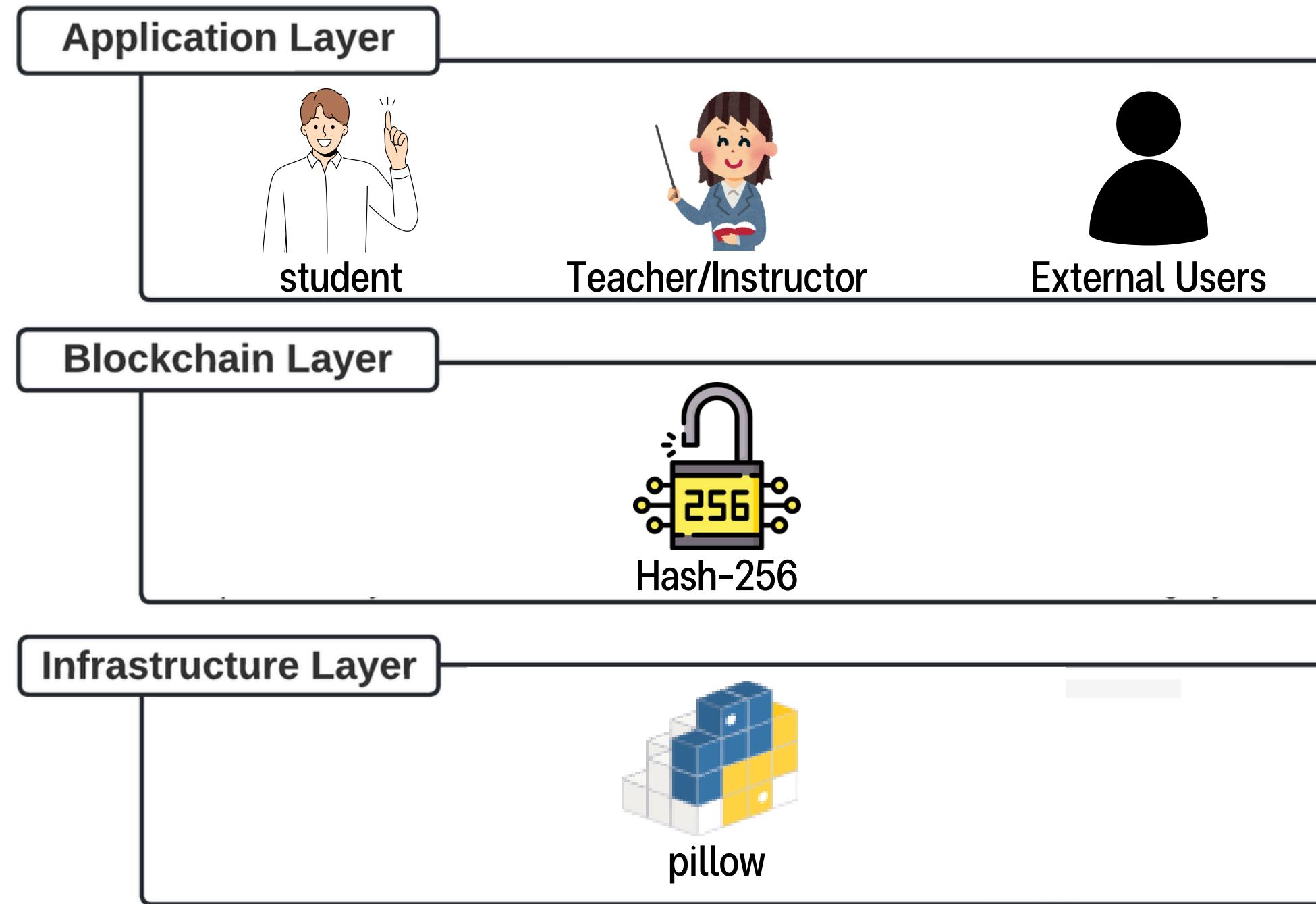


การวิเคราะห์และออกแบบโครงงาน

ระบบนี้มุ่งเน้นการเพิ่มความปลอดภัย ความน่าเชื่อถือ และการตรวจสอบได้ของใบประกาศนียบัตร โดยใช้ Hash Functions ในการสร้างลายนิ้วมือดิจิทัล (Digital Fingerprint) เพื่อป้องกันการปลอมแปลงและช่วยตรวจสอบความถูกต้องของเอกสาร

System architecture

แอปพลิเคชันนี้ใช้ สถาปัตยกรรมแบบชั้น (Layered Architecture) ซึ่งองค์ประกอบที่มีพังก์ชันคล้ายกันจะถูกจัดกลุ่มไว้ในแต่ละชั้นแนวนอน และแต่ละชั้นมีบทบาทเฉพาะภาระในแอปพลิเคชัน สถาปัตยกรรมของระบบประกอบด้วยสามชั้นหลัก ดังนี้



การสาธิตการทำงานของโครงงาน

(Project demonstration)



ขั้นตอนการทำงาน การติดตั้ง และโครงสร้างของ Code

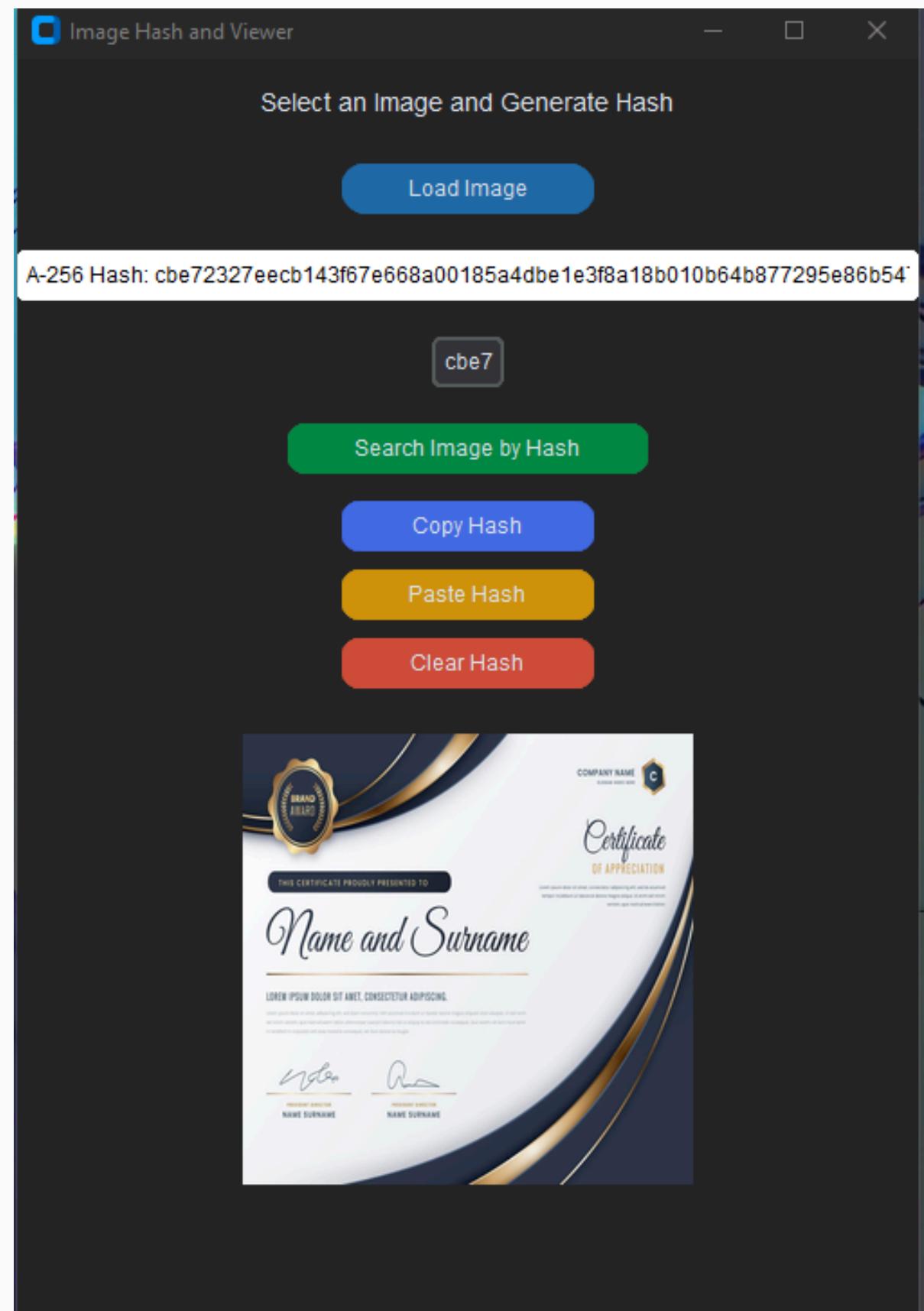
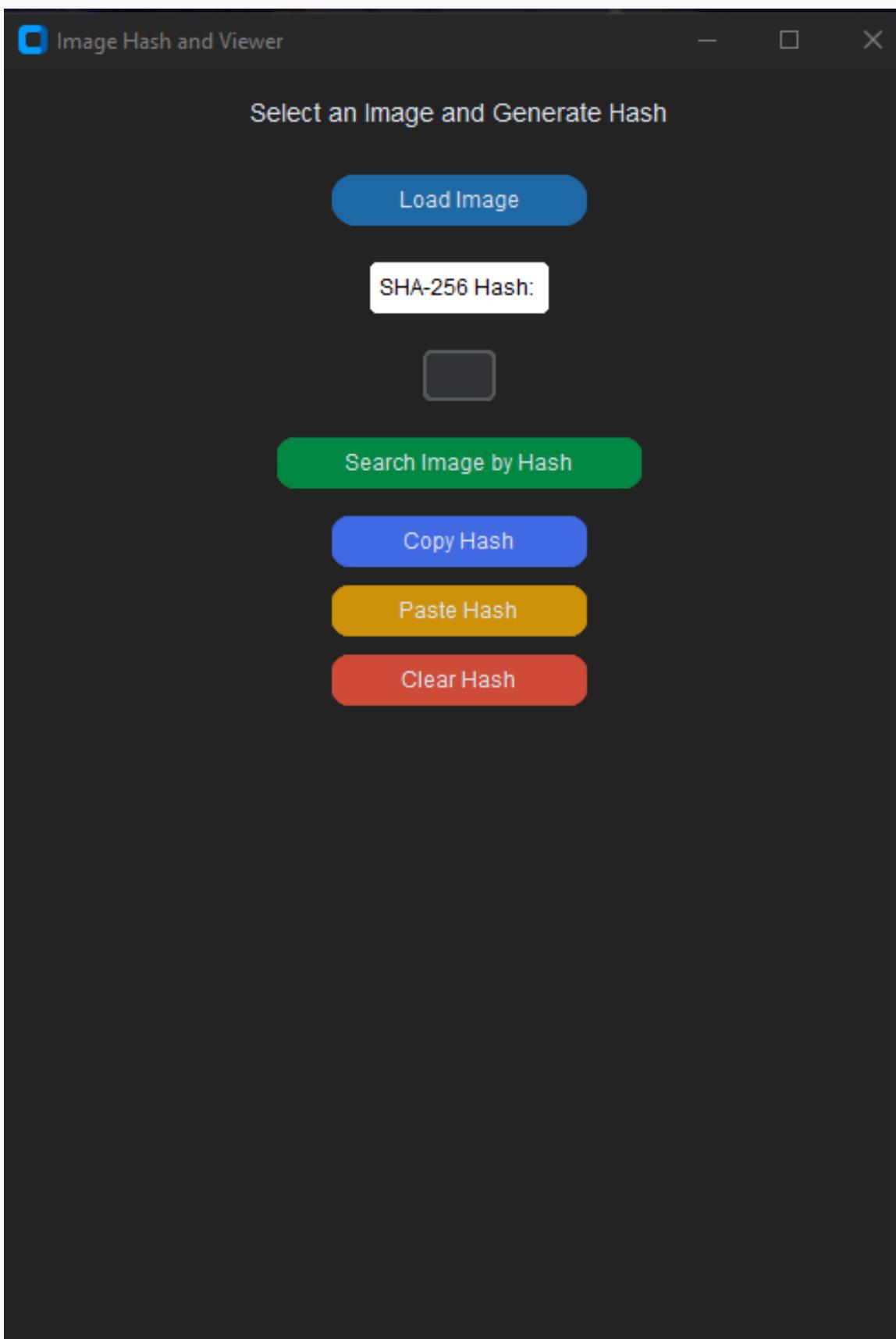
- ดาวน์โหลดไฟล์โปรเจกต์
- Copy Path โฟลเดอร์รูปภาพ แล้วเปลี่ยนกับ Path เดิมของโปรเจกต์
- เปิด Terminal และดาวน์โหลด Pillow ด้วยคำสั่ง

pip install --upgrade customtkinter Pillow pyperclip

- รันโปรเจก



ผลลัพธ์การทำงานของระบบ



```
File Edit Selection View Go Run ... < > folder
EXPLORER
OPEN EDITORS
test.py index.py indexComplete.py Papering
indexComplete.py > show_image
def display_image_hash(image_path):
    image_hash_entry.delete(0, 'end')
    image_hash_entry.insert(0, image_hash)

# Function to display the image in the Tkinter window
def show_image(image_path):
    try:
        img = Image.open(image_path)
        img = img.resize(250, 250, Image.Resampling.LANCZOS) # Resize image to fit in the window
        img_tk = ImageTk.PhotoImage(img)
        image_label.configure(image=img_tk)
        image_label.image = img_tk # Keep a reference to the image
    except Exception as e:
        messagebox.showerror("Error", f"Error opening image: {e}")

# Function to search for an image using its hash
def search_image_by_hash():
    user_hash = image_hash_entry.get()

    if not user_hash:
        messagebox.showerror("Error", "Please enter a hash value.")
        return

    found_image = None

    # Loop through all images in the directory to find a match
    for image_name in os.listdir(image_directory):
        image_path = os.path.join(image_directory, image_name)

        if os.path.isfile(image_path) and image_name.lower().endswith(("png", "jpg", "jpeg", "gif")):
            image_hash = image_to_hash(image_path)

    PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
C:\Users\drago\OneDrive\Documents\Python\Python312\lib\site-packages\customtkinter\windows\widgets\core_widget_classes\ctk_base_class.py:179: UserWarning: CTkLabel Warning: Given image is not CTkImage but <class 'PIL.ImageTk.PhotoImage'>. Image can not be scaled on HighDPI displays, use CTkImage instead.
    warnings.warn(f"type(self).__name__) Warning: Given image is not CTkImage but (type(image)). Image can not be scaled on HighDPI displays, use CTkImage instead\n.\n")
Ln 37, Col 37 Spaces: 4 UTF-8 CRLF () Python 3.12.5 64-bit
```

การอธิบายส่วนของ Source code ที่สำคัญ

```
import hashlib
import os
import customtkinter as ctk
from tkinter import filedialog, messagebox
import pyperclip # Used for copying to the clipboard
from PIL import Image, ImageTk

# Function to calculate SHA-256 hash of an image
def image_to_hash(image_path):
    with open(image_path, 'rb') as image_file:
        image_data = image_file.read()
        sha256_hash = hashlib.sha256(image_data).hexdigest()
    return sha256_hash

# Function to open the file dialog and load an image
def load_image():
    global image_path
    # Ask user to select an image file
    file_path = filedialog.askopenfilename(filetypes=[("Image Files", "*.*")])

    if file_path:
        image_path.set(file_path) # Update the StringVar
        # Display the image hash
        display_image_hash(file_path)
        show_image(file_path)
```



```
# Function to copy the hash to the clipboard
def copy_hash():
    hash_value = image_hash_entry.get()
    if hash_value:
        pyperclip.copy(hash_value) # Copy the hash to clipboard
        messagebox.showinfo("Success", "Hash copied to clipboard!")
    else:
        messagebox.showerror("Error", "No hash to copy.")

# Function to paste the hash from the clipboard
def paste_hash():
    hash_value = pyperclip.paste() # Get text from clipboard
    if hash_value:
        image_hash_entry.delete(0, 'end')
        image_hash_entry.insert(0, hash_value) # Paste it into the entry field
    else:
        messagebox.showerror("Error", "No hash found in clipboard.")

# Function to clear the hash in the entry field
def clear_hash():
    image_hash_entry.delete(0, 'end') # Clear the text field
    hash_label.configure(text=f"SHA-256 Hash: ")
    image_label.configure(image=None)
```



```
def search_image_by_hash():
    user_hash = image_hash_entry.get()

    if not user_hash:
        messagebox.showerror("Error", "Please enter a hash value.")
        return

    found_image = None

    # Loop through all images in the directory to find a match
    for image_name in os.listdir(image_directory):
        image_path = os.path.join(image_directory, image_name)

        if os.path.isfile(image_path) and image_name.lower().endswith(('png', 'jpg', 'jpeg', 'gif')):
            image_hash = image_to_hash(image_path)

            if image_hash == user_hash:
                found_image = image_path
                break

    if found_image:
        show_image(found_image)
        messagebox.showinfo("Success", f"Image found: {os.path.basename(found_image)}")
    else:
        messagebox.showerror("Not Found", "No image found with the specified hash.")
```



การอธิบายส่วนของ Source code ที่สำคัญ

1. คนแรก
2. คนสอง
3. คนสาม



ເບື້ອງຮັດກາຣດຳເນີນໂຄຣທານ

ກາຣແບ່ງຫານ

- ມີ 3 ມັນຕົກລັກໆ 1. Main Code 2. Data Base 3. Samrt Contract

ກາຣກຳຈານຮ່ວມກັນ

- ກຳຈານດ້ວຍກັນຕລອດກາຣກຳຈານເພື່ອຄ່ອຍຊ່ວຍເຫຼືອກັນແລະສ້ວສາຣກັນຕລອດກາຣກຳຈານ

ປັບປຸງຫາແລະອຸປະສົງຄົກືພົບ

- ຜ່ານແຮກໄດ້ເຂີຍໂປຣແກຣມ ແລະອອກແບບໜ້າເວີບເສັ້ນແລ້ວມີ Database ທີ່ເປັນກາຣຈໍາລອງ Hash ເສັ້ນ ແຕ່ອຍາກໃຊ້ກັບ Smart Contarct ເລີຍລວງທຳດູແຕ່ທຳໄມ່ສໍາເຮົງແລະໄດ້ແກ້ໄຂ code ມາກເກີນໄປຈົນກລັບມາ ເປັນເໝັ້ນເດີມໄມ້ໄດ້ແລ້ວຈຶ່ງທຳໃໝ່ Project ລ່າມໄປ 1 ຮອບ

ແນວທາງແກ້ໄຂ

- ຄວຣສໍາລອງໄຟລືໄວ້ກ່ອນທີ່ຈະກຳວະໄຮໃໝ່ໆ ກັນພາດ



thank you

