



RETO



PROGRAMACIÓN III



INTEGRANTES

Rene Alejandro Osorio González

Karen Yamileth Jiménez Galicia

Edgar Guillermo Chinchilla Chinchilla



CÓDIGO FUENTE DEL PROGRAMA.

Documentación del proceso de investigación (qué APIs se investigaron y por qué se eligió una en particular).

Dos compañeros utilizaron Claude y Copilot, y este fue el prom que usamos para consultar

En copilot

Estoy trabajando en un programa que tome una foto, y luego detecte si en esa foto hay una persona o no. Pero para esto necesito trabajar con una api que me haga este trabajo, y que sea compatible con Python.

Quiero que me des una lista de 10 APIs y servicios, ordenadas de manera que la primera sea la mejor.

En Claude

Estoy trabajando en un proyecto para un hackathon donde necesito implementar detección de personas usando inteligencia artificial. Me han sugerido varias APIs y servicios:

Clarifai

Roboflow

Google Cloud Vision API

Azure Computer Vision

Hugging Face con YOLO

¿Podrías ordenarlas de la menos recomendable a la más recomendable para un proyecto de hackathon con tiempo limitado?

Además, ¿podrías darme un breve ejemplo de código para implementar la opción más recomendada en Python?

Explicación de cómo funciona el programa

Toma la foto, la envía a la API, la API de hugging fase la procesa y luego devuelve la respuesta.

Capturas de pantalla o videos demostrativos del funcionamiento del programa.

Prompts utilizados para interactuar con LLMs (Language Models) o cualquier otra herramienta de IA.

Prom 1:

Hola, hoy vamos a hacer un reto de programación, ¡haz de cuenta que es un mini hackaton!, obviamente queremos ganarnos el primer lugar, somos un equipo de tres personas.

Necesito que seas nuestro consultor y que nos des una dirección de como vamos a trabajar.

Para la parte de la programación usaremos TRAE, un IDE con inteligencia artificial, al cual le pasaremos los prompts ya resumidos con las ideas e instrucciones que tus nos vas a dar.

Por ende, quiero que te comportes como un developer senior apasionado por ayudar a los jovenes, y eres un juez en el hackaton. Recuerda que queremos el primer lugar. ¿Entiendes todo para darte el contexto del reto?

Prom 2:

Desarrollo de un Programa con Inteligencia Artificial para Detección de Personas

Objetivo General:

El objetivo de este ejercicio es que los estudiantes desarrollen habilidades de investigación, programación y uso de herramientas de inteligencia artificial (IA) para resolver un problema práctico.

No importa si no tienen el conocimiento suficiente de programación, el objetivo es aprender a apoyarse con la IA para resolver problemas.

Instrucciones:

Equipo de Trabajo:

1. Los estudiantes deben trabajar en parejas.

Cada pareja debe documentar su proceso y resultados en un archivo README.md o equivalente.

2. Requisitos del Programa:

Captura de Imagen: Desarrollar un programa que tome una foto a través de la cámara web de la computadora.

Uso de API de IA: Investigar y seleccionar una API de inteligencia artificial que permita subir la imagen capturada y determinar si hay una persona en la foto.

Resultados Esperados: El programa debe mostrar el resultado de la detección (si hay una persona o no).

Opcional: Ampliar el programa para que describa otros objetos identificados en la imagen.

Investigación:

Deben investigar diferentes plataformas que ofrezcan APIs gratuitas o de prueba para la detección de personas y objetos. Algunas opciones incluyen:

Clarifai

Roboflow

Google Cloud Vision API

Azure Computer Vision

Hugging Face

Documentación:

Cada equipo debe entregar lo siguiente:

Código fuente del programa.

Documentación del proceso de investigación (qué APIs se investigaron y por qué se eligió una en particular).

Explicación de cómo funciona el programa.

Capturas de pantalla o videos demostrativos del funcionamiento del programa.

Prompts utilizados para interactuar con LLMs (Language Models) o cualquier otra herramienta de IA.

Entrega:

Suban su repo, código fuente y anotaciones a canvas

nota: ahí dice parejas, pero en realidad somos tres

IMPORTANTE: Trabajaremos con Python

Prom 3:

Prompt en CLAUDE 3.7 SONET

Quiero que me ayudes a crear un programa en Python que:

1. Capture una imagen desde la cámara web.
2. Guarde esa imagen localmente como captura.jpg.
3. Envíe la imagen a la API gratuita de Hugging Face para detectar si hay una persona.
4. Imprima "Persona detectada" si encuentra una, o "No se detectó persona".

Por favor, dime qué librerías debo instalar y genera el código paso a paso.

Prom 4:

pip install opencv-python requests pillow

Prom 5:

TRAE con Claude 3.7 ya nos dio el código con estas dependencias

pip install opencv-python requests pillow

ahora necesitamos saber como generar la API y lo necesario para usar hugging face

El código generado por python fue este:

```
import cv2
import requests
import os
from PIL import Image
import io
import base64

def capturar_imagen():
    """Captura una imagen desde la cámara web y la guarda como 'captura.jpg'"""
    # Inicializar la cámara web (0 es generalmente la cámara predeterminada)
    camara = cv2.VideoCapture(0)

    # Verificar si la cámara se abrió correctamente
    if not camara.isOpened():
        print("Error: No se pudo acceder a la cámara web.")
        return False

    # Capturar un fotograma
    ret, fotograma = camara.read()

    # Liberar la cámara
    camara.release()

    if not ret:
        print("Error: No se pudo capturar la imagen.")
        return False

    # Guardar la imagen
    cv2.imwrite('captura.jpg', fotograma)
    print("Imagen guardada como 'captura.jpg'")
    return True

def detectar_persona(ruta_imagen):
    """Envía la imagen a la API de Hugging Face para detectar personas"""
    # Cargar la imagen
    with open(ruta_imagen, "rb") as file:
        imagen = file.read()
```

```

# Codificar la imagen en base64
imagen_codificada = base64.b64encode(imagen).decode('utf-8')

# API de Hugging Face para detección de objetos (usando un modelo de
detección de objetos)
API_URL = "https://api-inference.huggingface.co/models/facebook/detr-resnet-
50"

# Reemplaza con tu token de API de Hugging Face
# Puedes obtener uno gratuito en https://huggingface.co/settings/tokens
API_TOKEN = "hf_XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" # Reemplaza con tu token

headers = {
    "Authorization": f"Bearer {API_TOKEN}",
    "Content-Type": "application/json"
}

# Preparar los datos para la API
payload = {
    "inputs": {
        "image": imagen_codificada
    }
}

# Enviar la solicitud a la API
response = requests.post(API_URL, headers=headers, json=payload)

# Verificar si la solicitud fue exitosa
if response.status_code == 200:
    resultado = response.json()

    # Buscar si hay una persona en los resultados
    for deteccion in resultado:
        if deteccion["label"] == "person":
            return True

    return False
else:
    print(f"Error en la API: {response.status_code}")
    print(response.text)
    return None

def main():
    # Capturar la imagen
    if capturar_imagen():
        # Detectar si hay una persona en la imagen
        resultado = detectar_persona("captura.jpg")

```

```
if resultado is True:
    print("Persona detectada")
elif resultado is False:
    print("No se detectó persona")
else:
    print("No se pudo realizar la detección")

if __name__ == "__main__":
    main()
```

QUE TRABAJO CADA INTEGRANTE

- Rene Alejandro Osorio González ----- Desarrollador
- Karen Yamileth Jiménez Galicia -----Generador Api
- Edgar Guillermo Chinchilla Chinchilla -----Documentación