from flask import Flask, request, jsonify

import requests

import os

from datetime import datetime

app = Flask(\_\_name\_\_)

# Verificamos que las variables de entorno estén bien definidas

SUPABASE\_URL = os.getenv("SUPABASE\_URL")

SUPABASE\_KEY = os.getenv("SUPABASE\_KEY")

if not SUPABASE\_URL or not SUPABASE\_KEY:

print("❌ ERROR: Las variables de entorno de Supabase no están configuradas correctamente.")

@app.route('/')

def home():

return "✅ Servidor Flask en Heroku funcionando correctamente 🚀", 200

@app.route('/ubicacion', methods=['POST'])

def recibir\_ubicacion():

data = request.json

print("📍 Datos recibidos:", data) # Log para ver la data en los logs de Heroku

if not data:

return jsonify({"error": "No se recibieron datos"}), 400

evento = data.get("evento")

zona = data.get("zona")

lat = data.get("lat")

lon = data.get("lon")

timestamp = data.get("tst")

print("Evento:", evento)

print("Zona:", zona)

print("Lat:", lat)

print("Lon:", lon)

print("Timestamp:", timestamp)

if lat is None or lon is None:

print("⚠️ Error: Falta latitud o longitud en la data.")

return jsonify({"error": "latitud o longitud faltante"}), 400

fecha = datetime.fromtimestamp(timestamp).isoformat() if timestamp else None

payload = {

"latitud": lat,

"longitud": lon,

"timestamp": fecha,

"evento": evento,

"zona": zona

}

headers = {

"apikey": SUPABASE\_KEY,

"Authorization": f"Bearer {SUPABASE\_KEY}",

"Content-Type": "application/json",

"Prefer": "return=representation"

}

try:

response = requests.post(

f"{SUPABASE\_URL}/rest/v1/ubicaciones",

json=payload,

headers=headers

)

print("🔄 Respuesta de Supabase:", response.status\_code, response.text)

if response.status\_code >= 400:

return jsonify({"error": "Error al insertar en Supabase", "detalle": response.text}), response.status\_code

# return jsonify({"status": "ok", "supabase\_response": response.json()}), 201

return jsonify({

"status": "ok",

"supabase\_status": response.status\_code,

"supabase\_text": response.text

}), 201

except Exception as e:

print("❌ Error al conectar con Supabase:", str(e))

return jsonify({"error": "Error al conectar con Supabase"}), 500

if \_\_name\_\_ == '\_\_main\_\_':

app.run(host='0.0.0.0', port=int(os.environ.get("PORT", 5000)), debug=True)