

ALGORITMO PRINCIPAL

- 1- INICIO
- 2- LLAMAR AL MODULO HAY_INTERNET() ← resultado
- 3- SI resultado = VERDADERO ENTONCES:
 - LLAMAR AL MODULO OBTENER_COORDENADAS_INTERACTIVAS() ← lat, lon, ubicacion
 - LLAMAR AL MODULO PEDIR_POTENCIA() ← potencia_kw
- 4- perdidas ← 14
- 5- SI lat = NULO O lon = NULO ENTONCES
 - IMPRIMIR "No se pudo convertir la dirección en coordenadas."
 - SINO
 - IMPRIMIR "No hay conexión a internet. Usando datos locales de respaldo"
 - LLAMAR AL MODULO CARGAR_JSON_LOCAL() ← datos_locales
- 6- SI datos_locales = NULO ENTONCES
 - IMPRIMIR "No se pudo cargar el archivo local."
- 7- FIN_SI
- 8- ubicacion ← "Estandar"
- 9- lat ← datos_locales["inputs"]["location"]["latitude"]
- 10- lon ← datos_locales["inputs"]["location"]["longitude"]
- 11- PEDIR_POTENCIA()←potencia_kw
- 12- perdidas ← 14
- 13- LLAMAR AL MODULO HAY_INTERNET() → resultado
- 14- SI resultado = VERDADERO ENTONCES:
 - IMPRIMIR "Conexión detectada, consultando PVGIS"
 - LLAMAR AL MODULO OBTENER_DATOS_PVGIS(lat, lon, potencia_kw, perdidas) ← datos
 - SINO
 - IMPRIMIR "Sin conexión, cargando nuevamente los datos locales"
 - LLAMAR CARGAR_JSON_LOCAL()← datos
- 15- FIN_SI
- 16- SI datos ≠ NULO ENTONCES
 - LLAMAR EXTRAER_ENERGIA_MENSUAL(datos) ← energia_mensual
 - IMPRIMIR "Energía mensual estimada (kWh):"
 - PARA CADA mes, energia EN energia_mensual HACER
 - IMPRIMIR mes + ": " + energía
 - FIN_PARA
- SINO
- 17- IMPRIMIR "No se pudieron obtener los datos del sistema"

18- FIN_SI

19- LLAMAR AL MODULO EXPORTAR_A_EXCEL(energia_mensual, ubicacion,
"informe_solar.xlsx")

20- FIN

ALGORITMO DE LOS MODULOS

Algoritmo Modulo para dirección a coordenadas

- 1-** Inicio
- 2-** Preparar los datos para hacer llamada a API
 - Prepara URL
 - Prepara Parametros
 - Prepara Headers
- 3-** Realizar llamada a API y verificar respuesta
 - Respuesta= realizar_llamada_API (URL, PARAMETROS, HEADERS)
 - Evaluar resultado
- 4-** Si hay respuesta (respuesta):
 - Retornar respuesta
- Sino
 - Retornar None
- 5-** Fin del modulo

Algoritmo Modulo para obtener datosPVGIS

- 1-** Inicio
- 2-** Preparar datos para llamada a API PVGIS
 - Preparar URL
 - Preparar PARAMS
- 3-** Realizar llamada a API y verificar respuesta
 - Respuesta= realizar_ llamada_API_PVGIS(URL, PARAMS)
- 4-** Evaluar resultado
- 6-** Si hay_respuesta(respuesta):
 - Retornar respuesta
- Sino
 - Retornar none
- 7-** Fin del modulo

Algoritmo Modulo para extraer energía mensual

- 1- Inicio
- 2- EXTRAER_ENERGIA_MENSUAL(DATA_JSON):
- 3- MESES = DATA_JSON['OUTPUTS']['MONTHLY']['FIXED']
- 4- ENERGIA_MENSUAL = {} {llave: valor}
- 5- Para cada diccionario en MESES:
 - Si el diccionario contiene las claves necesarias:
 - MES = diccionario['month']
 - ENERGIA = diccionario['E_m']
 - ENERGIA_MENSUAL[MES] = ENERGIA
- 6- Retornar ENERGIA_MENSUAL
- 7- Fin del modulo

Algoritmo Modulo para grafica de energía mensual

- 1- Inicio
- 2- Preparar datos para la gráfica
 - meses = obtener_claves(ENERGIA_MENSUAL) Eje X
 - valores = obtener_valores(ENERGIA_MENSUAL) Eje Y
- 3- Crear y configurar gráfico
 - crear_grafico_barras()
 - configurar_eje_x(meses)
 - configurar_eje_y("Energía")
- 4- Generar gráfico de barras
 - para cada i en rango(longitud(meses)):
 - dibujar_barra(meses[i], valores[i])
- 5- Mostrar o guardar el gráfico
- 6- mostrar_grafico()
- 7- Fin del modulo

Algoritmo Modulo para obtener coordenadas

- 1- Inicio
- 2- Verificar la condición inicial
- 3- Si TRUE:
 - Obtener dirección de entrada
 - DIRECCION = obtener_direccion_usuario()
- 4- Realizar llamada a API para geocodificación
- 5- DATA = llamar_API_Nominatim(DIRECCION)
- 6- Procesar resultados
- 7- LAT, LON, UBI = ELEGIR_DIRECCION(DATA)
- 8- Retornar coordenadas
- 9- Retornar LAT, LON, UBI
- 10- Fin del modulo

Algoritmo Modulo para cargar JSON

- 1- Inicio
- 2- Abrir archivo JSON local
- 3- archivo = abrir_archivo("Pruebas.json", "r")
 - Leer y cargar datos JSON
 - contenido = leer_contenido(archivo)
 - datos_json = parsear_json(contenido)
- 4- Cerrar archivo
- 5- cerrar_archivo(archivo)
- 6- Retornar datos cargados
 - Retornar datos_json
- 7- Fin del modulo

Algoritmo Modulo para Internet

- 1- Inicio
- 2- Verificar conexión a internet
- 3- Conexión= verdadero entonces:
 - Retornar verdadero
 - Sino
 - Retornar falso
 - Fin si
- 4- Fin del modulo

Algoritmo Modulo para exportar a excel

- 1- Inicio
- 2- Calcular estadísticas básicas
 - media \leftarrow CalcularMedia(ENERGIA_MENSUAL)
 - mediana \leftarrow CalcularMediana(ENERGIA_MENSUAL)
 - moda \leftarrow CalcularModa(ENERGIA_MENSUAL)
 - desviacion \leftarrow CalcularDesviacion(ENERGIA_MENSUAL)
- 3- Exportar datos a archivo Excel
- 4- AbrirArchivoExcel("Reporte.xlsx")
 - Imprimir ("Consumo mensual", ENERGIA_MENSUAL)
 - Imprimir ("Media", media)
 - Imprimir ("Mediana", mediana)
 - Imprimir ("Moda", moda)
 - Imprimir Desviación estándar", desviacion)
- 5- GuardarArchivoExcel()
- 6- Fin del modulo

Algoritmo Modulo para pedir potencia

- 1- Inicio
- 2- Declaración de variables:
Potencia:Real
Entrada: Cadena
es_valido: Logico
- 3- es_valido ←FALSO
- 4- Mientras es_valido ← FALSO Hacer
 Imprimir "Ingrese la potencia del sistema en kW (mayor a 0): "
5- Si EsNumero(entrada) Entonces
 potencia <- ConvertirANumero(entrada)
 Si potencia > 0 Entonces
 es_valido <- VERDADERO
 Sino
 Imprimir"La potencia debe ser mayor que cero."
 FinSi
Sino
 Imprimir "Entrada no válida. Intente de nuevo."
FinSi
- 6- FinMientras
- 7- Imprimir "Potencia ingresada: ", potencia
- 8- Fin del modulo