**Documentación Técnica de la API Interna**

**Módulo modelo**

**lugares: dict**

Diccionario que almacena los nodos y sus coordenadas geográficas.

**reconstruir\_grafo()**

Reconstruye el grafo G conectando todos los nodos con distancias geodésicas.

**dijkstra(grafo, inicio, fin)**

Calcula la ruta más corta entre dos nodos usando Dijkstra.

* **Parámetros**: grafo, inicio, fin
* **Retorna**: camino (list[str]), distancia (float)

**Módulo controlador**

**calcular\_ruta(origen, destino, intermedios)**

Calcula la ruta óptima considerando origen, destino y puntos intermedios.

* **Retorna**: (camino: list[str], distancia: float)

**agregar\_nodo(nombre, lat, lon)**

Agrega un nuevo nodo.

* **Retorna**: (True, None) o (False, "mensaje de error")

**eliminar\_nodo(nombre)**

Elimina un nodo si existe.

**editar\_nodo(nombre\_original, nuevo\_nombre, lat, lon)**

Edita la información de un nodo.

* **Retorna**: (True, None) o (False, "mensaje de error")

**Módulo mapa.py**

**mostrar\_mapa(camino, distancia\_total)**

Genera y abre un archivo HTML con la ruta trazada sobre un mapa interactivo usando folium.

**Módulo vista.py**

Implementa una interfaz gráfica con Tkinter que permite:

* CRUD de nodos
* Cálculo de rutas con visualización del resultado
* Control dinámico para puntos intermedios

**iniciar\_interfaz()**

Lanza la GUI principal.