**Documentación: Sistema de Optimización de Rutas**

**Empresa de Paquetería**  
*Versión 1.0*

**1. Descripción del Proyecto**

Sistema en Python para gestionar y optimizar rutas de entrega usando una **lista simplemente enlazada**.

* **Objetivo**: Encontrar la ruta más corta y gestionar destinos eficientemente.
* **Algoritmo clave**: Bubble Sort para ordenar rutas por distancia.

**2. Estructura del Código**

**Clases Principales**

| **Clase** | **Función** |
| --- | --- |
| Ruta | Almacena destino (str) y distancia (float). |
| Nodo | Contiene una Ruta y un puntero al siguiente nodo. |
| ListaRutas | Gestiona la lista enlazada (agregar, eliminar, buscar, ordenar, etc.). |

**Métodos Clave**

| **Método** | **Función** |
| --- | --- |
| agregar\_ruta() | Añade una ruta al inicio de la lista. |
| buscar\_ruta\_mas\_cercana() | Ordena la lista y retorna la ruta con menor distancia. |
| guardar\_a\_archivo() | Guarda rutas en JSON/CSV/TXT. |
| cargar\_desde\_archivo() | Carga rutas desde archivo. |

**3. Pruebas de Funcionamiento**

**Caso 1: Agregar y Mostrar Rutas**

lista = ListaRutas()

lista.agregar\_ruta(Ruta("Bogotá", 150))

lista.agregar\_ruta(Ruta("Medellín", 300))

lista.imprimir\_rutas()

**Salida**:

Bogotá (150 km) -> Medellín (300 km) -> None

**Caso 2: Buscar Ruta Más Cercana**

ruta\_cercana = lista.buscar\_ruta\_mas\_cercana()

print(f"Ruta más cercana: {ruta\_cercana}")

**Salida**:

Ruta más cercana: Bogotá (150 km)

**Caso 3: Persistencia en Archivo**

lista.guardar\_a\_archivo("rutas.json") # Guarda

lista\_nueva = ListaRutas()

lista\_nueva.cargar\_desde\_archivo("rutas.json") # Carga

lista\_nueva.imprimir\_rutas()

**Contenido de**rutas.json:

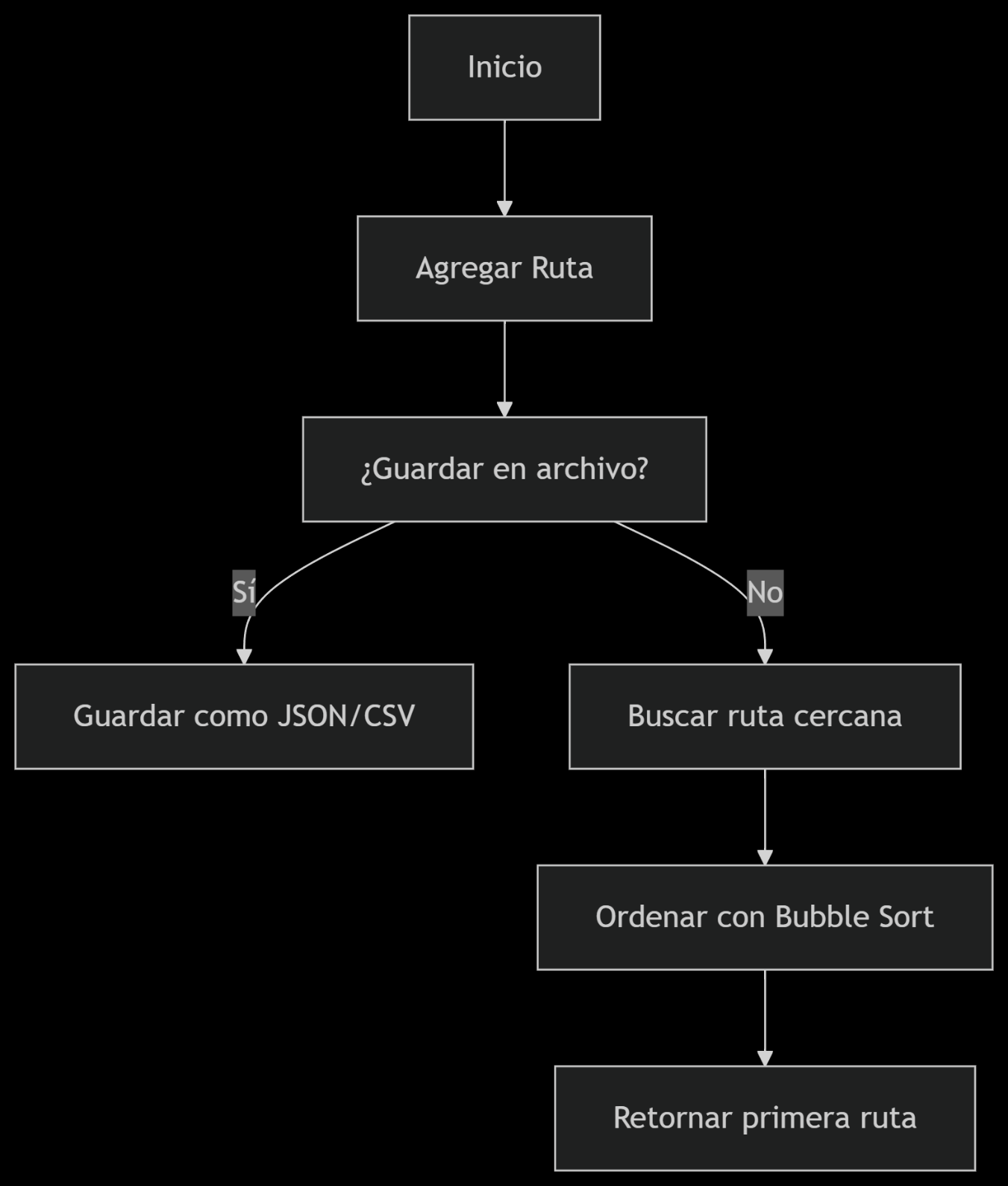
[

{"destino": "Bogotá", "distancia": 150},

{"destino": "Medellín", "distancia": 300}

]

**4. Diagrama de Flujo**



**5. Instrucciones de Uso**

1. **Ejecutar el programa**:

python3 sistema\_rutas.py

1. **Menú interactivo**:

1. Agregar ruta

2. Mostrar rutas

3. Buscar ruta más cercana

4. Guardar/Cargar desde archivo

1. **Ejemplo de entrada**:

Destino: Cartagena

Distancia: 250

**6. Posibles Errores y Soluciones**

| **Error** | **Solución** |
| --- | --- |
| ValueError al ingresar texto | Validar entrada con try-except. |
| Archivo no encontrado | Verificar ruta o crear archivo vacío. |
| Lista vacía al cargar | Revisar formato del archivo (JSON/CSV). |