Great — you’ve uploaded both:

* 🗺️ A base image: HoustonMap.png
* 📄 A KML file: test\_pipeline.kml

Let’s start with a **simple Python Flask app** that:

1. Lets a user upload a .kml file.
2. Overlays that KML data on a map (using Leaflet + [leaflet-kml](https://github.com/shramov/leaflet-plugins) plugin).
3. Optionally displays the HoustonMap.png image as the map background (or we can use a Leaflet base layer like OpenStreetMap if preferred).

**✅ Step 1: Project Structure**

kml\_map\_app/

├── app.py

├── templates/

│ └── index.html

├── static/

│ ├── HoustonMap.png

│ ├── leaflet.js

│ ├── leaflet.css

│ └── leaflet.kml.js

├── uploads/

└── test\_pipeline.kml

**🐍 app.py — Simple Flask Server**

from flask import Flask, request, render\_template, send\_from\_directory

import os

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

app.config['UPLOAD\_FOLDER'] = 'uploads'

# Ensure upload folder exists

os.makedirs(app.config['UPLOAD\_FOLDER'], exist\_ok=True)

@app.route('/')

def index():

return render\_template('index.html')

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

def upload():

file = request.files['kmlfile']

if file and file.filename.endswith('.kml'):

filepath = os.path.join(app.config['UPLOAD\_FOLDER'], file.filename)

file.save(filepath)

return render\_template('index.html', kml\_file=file.filename)

return 'Invalid file format. Please upload a .kml file.'

@app.route('/uploads/<filename>')

def uploaded\_file(filename):

return send\_from\_directory(app.config['UPLOAD\_FOLDER'], filename)

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

app.run(debug=True, port=5050)

**🌐 templates/index.html — Map with Upload and KML Display**

<!DOCTYPE html>

<html>

<head>

<title>KML Viewer</title>

<link rel="stylesheet" href="{{ url\_for('static', filename='leaflet.css') }}">

<style>

#map { height: 600px; width: 100%; }

</style>

</head>

<body>

<h2>KML File Uploader & Viewer</h2>

<form method="POST" enctype="multipart/form-data" action="/upload">

<input type="file" name="kmlfile" accept=".kml">

<input type="submit" value="Upload">

</form>

<div id="map"></div>

<script src="{{ url\_for('static', filename='leaflet.js') }}"></script>

<script src="{{ url\_for('static', filename='leaflet.kml.js') }}"></script>

<script>

var map = L.map('map').setView([29.7604, -95.3698], 8); // Houston center

// Optionally add your own map image

L.imageOverlay('{{ url\_for("static", filename="HoustonMap.png") }}',

[[27.5, -98.5], [31.5, -92.5]] // Rough bounding box; adjust as needed

).addTo(map);

{% if kml\_file %}

fetch("{{ url\_for('uploaded\_file', filename=kml\_file) }}")

.then(res => res.text())

.then(kmlText => {

const parser = new DOMParser();

const kml = parser.parseFromString(kmlText, 'text/xml');

const track = new L.KML(kml);

map.addLayer(track);

map.fitBounds(track.getBounds());

});

{% endif %}

</script>

</body>

</html>

**📦 Dependencies**

To run this, install Flask:

pip install flask

And copy the required Leaflet and leaflet.kml JavaScript/CSS files into static/ (or use CDN versions if you prefer).

**🚀 To Run It:**

python app.py

Then go to http://localhost:5050, upload test\_pipeline.kml, and you should see it rendered over Houston!

Would you like me to package this whole thing into a .zip or help you fine-tune the map bounding box for HoustonMap.png?