1. Project Overview: Smart Urban Green Placement System (SUPGS)

Urban highways and signals suffer from high CO_2 intensity. Traditional planting often fails due to infrastructure damage (deep roots) or safety risks (visibility). My system, **SUPGS**, uses a data-driven approach to optimize urban greenery.

- **Problem:** High traffic idling CO_2 and infrastructure damage by unscientific tree placement.
- **Solution:** A GPS-based recommendation engine that selects plants based on road type, traffic load, and safety filters.
- **Safety Integration:** Automatically excludes deep-rooted species like Banyan for city roads to protect underground utilities.

2. Quantitative Impact Analysis (Per 1 KM Stretch)

Based on my cross-checked calculations, the model estimates the following sequestration:

Species/Component	Quantity/km	Est. CO2 Absorption (Yearly)
Areca Palm	1000	~20.0 Tons
Snake Plant	2000	~10.0 Tons
Ashoka Tree	667	~16.7 Tons
Teak Tree (Highways)	667	~26.7 Tons
Total mitigation	-----	**73.4 tones CO2/year**

Effectiveness: This model can reduce CO_2 exposure by **10% to 15.9%** per kilometer, significantly improving pedestrian air quality. ($73.4/460\times100$)

3. Software Prototype (Python)
I have developed a Python prototype to automate these calculations. While I used AI tools for code synthesis and data collection, the logic, safety parameters, and mathematical verification are my original work.
Code is in last sir

```
# SUPGS: Smart Urban Green Placement System
# Developer: Kishan & AI Assistant

def calculate_green_plan(road_length_km, road_type,
traffic_density):
    """
    road_type: 'highway', 'city_main', 'flyover_pillar'
    traffic_density: 'high', 'medium', 'low'
    """

    # 1. CO2 Emission Factors (Approx. kg/km/year
based on density)
    emissions = {
        'high': 500000,   # 500 Tons/km/year
        'medium': 300000, # 300 Tons/km/year
        'low': 150000     # 150 Tons/km/year
    }

    total_emission = emissions[traffic_density] *
road_length_km
```

```
    # 2. Plant Database: (CO2 absorption in kg/unit/
year, min_spacing_meters)
    # Safety Check: Deep root trees excluded for
city_main
    plants = {
        'Areca Palm': {'co2': 20, 'spacing': 1.0, 'safe_for':
['highway', 'city_main']},
        'Snake Plant': {'co2': 5, 'spacing': 0.5, 'safe_for':
['city_main', 'flyover_pillar']},
        'Ashoka Tree': {'co2': 25, 'spacing': 3.0, 'safe_for':
['highway', 'city_main']},
        'Moss Panel': {'co2': 100, 'spacing': 10.0, 'safe_for':
['flyover_pillar']} # per panel
    }

    print(f"--- SUPGS Analysis Report for
{road_length_km} km ({road_type}) ---")
    print(f"Estimated Local CO2 Load: {total_emission/
1000} Tons/year\n")
    print("Recommended Planting Strategy:")

    total_absorbed = 0
```

```
    for plant, data in plants.items():
        if road_type in data['safe_for']:
            # Calculate quantity based on spacing
            qty = int((road_length_km * 1000) / data['spacing'])
            absorption = (qty * data['co2']) / 1000 # convert
to tons
            total_absorbed += absorption


            print(f"- {plant}: {qty} units (Spacing:
{data['spacing']}m) | Impact: {absorption:.2f} Tons CO2/
yr")


    impact_percent = (total_absorbed / total_emission) *
100


    print("-" * 50)
    print(f"TOTAL CO2 MITIGATION:
{total_absorbed:.2f} Tons/year")
    print(f"SYSTEM EFFECTIVENESS:
{impact_percent:.2f}% reduction in local exposure")
    print("-" * 50)
    print("SAFETY GUIDELINE: Ensure 2m distance
from road edge & 5m visibility at turns.")
```

```python
# --- Testing the System ---

# Example: 2 km Highway with High Traffic

calculate_green_plan(road_length_km=2,

road_type='highway', traffic_density='high')


# Example: 1 km City Road with Medium Traffic

# calculate_green_plan(road_length_km=1,

road_type='city_main', traffic_density='medium')
```