📝 Carbon Footprint Estimator – Documentation

🔍 Overview

This web-based MVP estimates the carbon footprint of various consumer products in specific localities across Bengaluru, India. The tool combines:

- 🌍 Geographic context
- 📦 Product-based emissions
- 🌱 Actionable sustainability tips
- 🗺️ An interactive map visualization

It demonstrates how everyday purchasing decisions could be made more environmentally aware using lightweight technology and public data proxies.

💡 Problem Statement

Consumers lack visibility into the carbon impact of their daily purchases — especially when **locality-specific factors** like transportation, energy mix, adoption rates, and recycling infrastructure come into play.

Our solution bridges this gap by providing an **instant estimate of carbon emissions** per product-locality combination and offering **tips to reduce environmental impact**.

🛠️ Architecture & Stack

Frontend

- **HTML + CSS (Dark Theme)**: Clean two-panel layout for product interaction and map visualization.
- **Vanilla JavaScript**: DOM logic, event handling, autocomplete with arrow key support, and map integration.

Map Integration

- **Leaflet.js**: Renders a dark-themed interactive map with circular markers.
- **CARTO Tiles**: Dark basemap for sharp visual contrast.

🧮 Updated Calculation Logic

Formula:

```
Estimated CO₂ (kg) = Population × (Multiplier ÷ 10) × Product Footprint
```

- ***Population***: Approximate headcount of people living in the selected locality.
- ***Multiplier (Adoption Rate)***: Simulates product popularity or tech penetration in the area (e.g., 8.2/10 = 82%).
- ***Product Footprint***: Average per-unit carbon footprint, e.g., Laptop = 500 kg CO_2.

✅ **Example**:
Laptop in Indiranagar =
`150,000 (people) × 0.78 × 500 = 58,500,000 kg CO₂` → 58,500 metric tons CO_2

⚠️ *Note*: This is a **dummy logic** for hackathon/MVP purposes. Values are **not from lifecycle analysis datasets** but demonstrate realistic regional variation patterns.

📦 Product List

1. Laptop
2. T-shirt
3. Mobile Phone
4. Jeans
5. Television

Each product comes with **5 randomized eco-tips** that display upon result generation.

📍 Locality Coverage

10 granular Bengaluru localities are currently supported:

- Whitefield
- Kaggadasapura
- Nayandahalli
- HSR Layout
- Malleshwaram
- Electronic City
- Indiranagar

- Marathahalli
- Koramangala
- Jayanagar

Each area has a hardcoded **population** and **multiplier** (0–10 scale) representing product adoption rate or infrastructure differences.

🧠 Features

🖊️ Autocomplete with DSA-style Navigation

- As users type in either input box, matching options appear instantly.
- **Arrow key navigation** and **Enter key selection** are fully supported.
- Enhances accessibility and demonstrates scalability via predictive UX.

📍 Dynamic Map + Heat Circle

- The typed locality is **geocoded** using Nominatim (OpenStreetMap).
- A colored **circular marker** indicates emission intensity:
  - 🔴 **Red**: High (> 100 million kg CO_2)
  - 🟠 **Orange**: Moderate (50–100 million)
  - 🟢 **Green**: Low (< 50 million)

🌱 Sustainability Tips

- Once product + locality is selected and result is calculated, a **random eco tip** (from a curated list) is shown for that product.

🎨 UI/UX Design

- 🖤 **Dark theme** for reduced screen fatigue and modern feel.
- 💚 Eco-visual palette with greens to support sustainability messaging.
- 🔲 Two-Panel Layout:
  - **Left Panel**: Inputs + results + tips
  - **Right Panel**: Real-time map
- No scrolling needed — responsive, centered, and smooth.

🧪 MVP Strengths

- ✅ **Lightweight**: Entirely frontend-based, runs in browser — no hosting/server required.
- ✅ **Scalable**: Architecture supports adding more data (localities, products, APIs).
- ✅ **Extendable**: Easily plug in real APIs or lifecycle carbon datasets in the future.
- ✅ **DSA Layer**: Smart suggestion + navigation logic shows readiness for large-scale usage.
- ✅ **Hackathon-Ready**:
  - Innovation ✅
  - Usability ✅
  - Technical implementation ✅
  - Real-world applicability ✅