GridSense

A California-focused home energy optimizer

Team: Khushi Gandhi, Soham Kulkarni
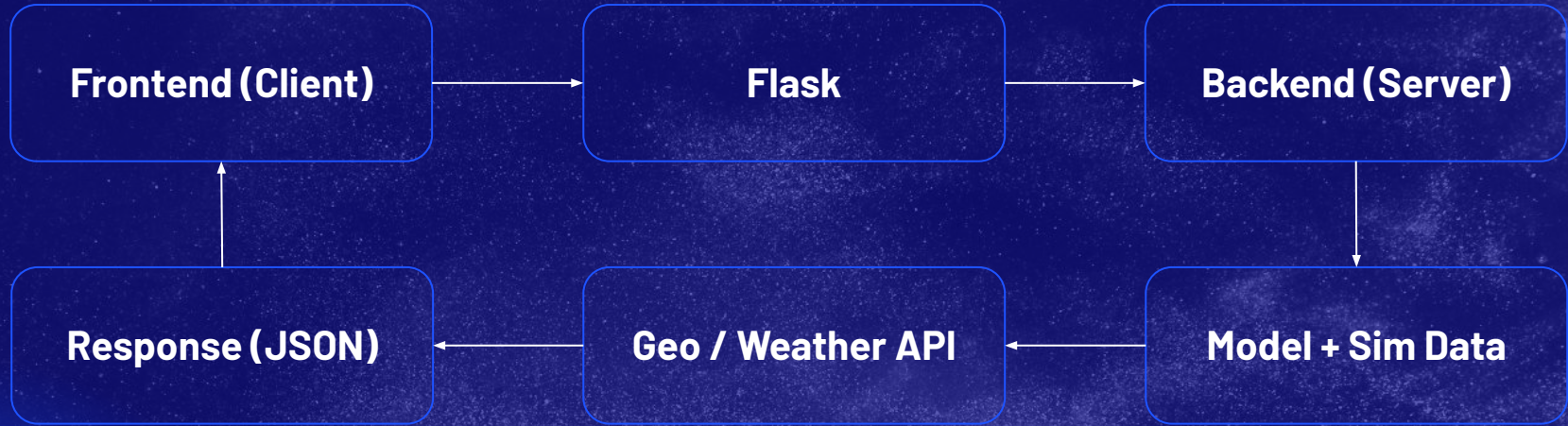
Problem Statement

THE CHALLENGE

- High electricity bills

- Energy usage is hard to track

- Low awareness of carbon footprint

- Lack of actionable, personalized insights

THE GRIDSENSE SOLUTION

- User inputs: ZIP code, home size, residents, appliance usage, HVAC, EV/pool, time-of-use patterns

- Backend calculates: predicted kWh, utility bill, carbon footprint

- Outputs: smart recommendations, eco-score, badges & leaderboard

System Architecture & Tech Stack

| Frontend (Client) | → | Flask | → | Backend (Server) |

| Response (JSON) | ← | Geo / Weather API | ← | Model + Sim Data |

Technology Stack: Python, Flask, React.js, Typescript, Vite, Tailwind CSS

Technical Understanding

Backend

- Energy consumption model (Scikit-learn)
- Simulated California household dataset
- HVAC & appliance kWh calculations
- Carbon footprint & equivalents computation
- Smart recommendations engine
- Eco score, badges, and leaderboard logic
- Utility rate lookup by provider

Frontend

- Responsive energy input form for home, appliances etc.
- Utility provider dropdown (PG&E, SCE, SDG&E)
- ZIP-to-climate enrichment via Nominatim geocoding API
- Dashboard panels
- API client for sending user input to backend and normalizing JSON responses
- Dynamic React components update dashboard in real-time with backend predictions and analytics