Fragility Map Project — Concept and Execution Plan

Summary

A real-time, WebGL-powered visualization revealing the Earth's physical, infrastructural, and digital fragility. Combines dynamic global data feeds (NOAA, NASA, Cloudflare, USGS) with an interactive globe and analytical overlays. Designed to engage casual users, analysts, and media while offering ad revenue, sponsorship, and licensing opportunities.

Core Experience

1. Interactive 3D globe visualizing natural and human infrastructure fragility.
2. Topographic 'lens' hover interaction showing terrain, bathymetry, and fault data.
3. Dynamic color overlays for hazard intensity, energy flow, and connectivity loss.
4. Minimalist analytical layer below: Tufte-inspired charts and sparklines summarizing live metrics.

Audience and Value

1. Casual visitors — hypnotic real-time visual, shareable, ambient site.
2. Analysts & infrastructure planners — insight into global interdependencies and outage patterns.
3. Media & educators — licensed visuals and embeddable components.

Technical Stack

1. Frontend: Three.js or Deck.gl for globe; D3.js for analytical charts.
2. Backend: Node.js + Supabase/Postgres for caching API data.
3. Realtime APIs: NOAA, NASA, Copernicus, USGS, Cloudflare Radar, RIPE Atlas, ENTSO-E, IRENA, WorldPop.
4. Hosting: Vercel + CDN caching; CRON or n8n workflows for data ingestion.

Data Fusion Model

fragility = hazard_intensity × exposure × (1 - resilience_score).

Hazards include storms, quakes, floods. Exposure = population density and infrastructure concentration. Resilience derived from redundancy, energy reserve, and governance indices.

Visual Features

1. Hover lens: switches to terrain or alternate data mode using hillshade textures.
2. Optional topographic layers: elevation, slope, aspect, bathymetry, or contour mode.
3. Multi-lens compare: Topo vs Population, or Topo vs Power Grid.
4. Smooth fade transitions and shader blending for responsive feel.

Analytical Layer (Tufte influence)

1. Compact sparklines for hazard and outage frequency.
2. Slopegraphs comparing pre/post-event metrics.
3. Small multiples by region for monthly variation.
4. Annotation-based storytelling with minimal ink.

Monetization Opportunities

1. Ad-supported ambient site (Windy/Earth.nullschool model).
2. API access and professional dashboards.
3. Media licensing and sponsor partnerships.
4. Educational integration and museum installations.

Difficulty & Timeline

1. Solo MVP (1 region): 8 weeks.
2. Global prototype: 4 months.
3. Polished public site: 6–10 months.
4. Overall success likelihood: ~55% (95% for MVP launch).

Next Steps

1. Define data scope (natural hazards vs infra + socio-economic).
2. Select rendering library and prototype globe + hover lens.
3. Integrate 3–4 real-time APIs for initial live data pipeline.
4. Build Tufte-style analytical strip below hero visualization.
5. Test public launch with SEO + Reddit/ProductHunt exposure.