Network Viewer

Static web demo to visualize JSON graphs in 3D and simulate a basic attack via random node removal.

Idunn Freya

Mar 2026

ESTADO PUBLISHABLE (DEMO) **INICIO** Dec 2025

VERSIÓN v1.0.0 **REPO** idunnmaria/visor-resiliencia-redes

SUMMARY

Network Viewer is a backendless technical demo to load a graph in JSON, render it in 3D, and simulate a simple attack (random node removal) to observe connectivity and fragmentation changes. It includes ready-to-run examples (test in under a minute) and optional scripts to generate/attack networks.

Software & Systems

Static web app + auxiliary scripts

HTML/CSS/JS + Three.js · Python + NetworkX

TECH STACK

BACKEND	N/A (static app)
FRONTEND	HTML · CSS · JavaScript (vanilla) · Three.js (CDN)
INFRASTRUCTURE	Static hosting / GitHub Pages (optional)
TESTING	Manual smoke tests (browser + scripts)

GOALS

- File-based JSON input (no copy/paste)
- 3D network visualization (nodes + edges)
- Basic attack simulation (random node removal) and observable changes
- Unified UI (shared theme + header)

- JSON validation with readable errors (future improvement)
- Drag & drop + “Load example” (future improvement)

ARCHITECTURE

Project split into: - **Static app** (HTML) with a unified UI via `assets/theme.css`. - **Optional scripts** (Python) to generate networks and produce attacked versions. - **Included examples** under `examples/` to test without installing anything.

MODULE / APP · DESCRIPTION · STATUS

MODULE	DESCRIPTION	STATUS
`index.html`	Home/intro + links to viewer and simulator	✓
`visor_viewer.html`	3D viewer + file-based JSON input	✓
`simulador_resiliencia.html`	Simulator (initialize/ simulate/reset) + file upload	✓
`assets/theme.css`	Shared theme + UI components	✓
`scripts/data_generator.py`	Network data generator (examples)	✓
`scripts/attack_simulator.py`	Random node removal + basic metrics	✓
`examples/`	Ready-to-run example JSON files	✓

MILESTONES

Mar 2026

Unified UI (theme + header) and file-based JSON loading.

Mar 2026

Simulator rebuilt with a single UI (no legacy overlays).

Mar 2026

Examples included under `examples/` for instant testing.

TECHNICAL DECISIONS & NOTES

DECISIÓN

Static app (no backend). Maximum portability and minimal maintenance; ideal for a portfolio demo.

.

DECISIÓN

Simple attack model. Random node removal to observe connectivity/fragmentation without claiming advanced simulation.

.

NOTA

JSON formats. Tolerant support for both “new” (`nodes/edges`) and “legacy” (`data/nodes/connections`) formats.

ATENCIÓN

CDN dependencies. Fine for a demo; for “production-grade” delivery consider pinned versions + SRI or bundling.

INSTALLATION & DEPLOYMENT

Run the demo locally

```bash python3 -m http.server 8000 # open: # http://localhost:8000/index.html
```