

A Financial Safety-Net Builder for Indonesian Households

Project Documentation

Project Summary

Kifusuru is a single-page, single-file web app that helps low-income households build a realistic financial safety net — one that doesn't depend on the standard, often unrealistic, advice to “save three months of expenses.”

It walks a user through four steps:

- Know your number — three sliders (income, essential monthly costs, cash on hand) calculate a live “runway”: how many days of expenses your current cash could cover.
- Stress-test your net — pick a real shock (vehicle breakdown, medical bill, appliance failure, income loss) or hit “surprise me,” and see exactly what it would do to that cash.
- Weave your net — choose from 9 accessible, evidence-based strategies localized for Indonesia (arisan savings circles, social-assistance screening via cekbansos, koperasi/BMT loans, Pegadaian pawning, employer relief funds, automatic e-wallet savings, flexible gig income, bill-hardship negotiation, micro emergency funds). Each selection visibly re-weaves one patch of an SVG rope net from a frayed, dashed state into a solid, gold “woven” state.
- Your plan — a resilience score, a checklist with one concrete first move per strategy, and a “copy my plan” button.

Everything runs on the client side. No data is stored, transmitted, or persisted anywhere — closing the browser tab clears it.

Development Process

This was built by working iteratively with Claude and GitHub Copilot as a programming assistant. The paragraph below describes the architecture.

Development happened in three passes, each corresponding to a commit in the accompanying repository:

- Initial prototype (English/US) — designed around a “weave a safety net” metaphor: a cash-runway calculator, a shock stress-test grounded in 2025 Federal Reserve SHED survey data, nine real-world resilience strategies, and a plan/checklist generator.
- Indonesia localization — the same architecture, relocalized: currency to Rupiah, the headline statistic to Indonesia’s OJK/GoodStats emergency-fund survey, and all nine strategies swapped for Indonesia-specific equivalents (arisan, koperasi/BMT, Pegadaian, cekbansos, etc.), with the full UI translated to Bahasa Indonesia and the app renamed “Kifusuru.”

Architecturally, the whole app is one HTML file with a single shared JavaScript state object. Every input — the three sliders, shock selection, strategy toggles — funnels through one `refreshAll ()` function that keeps four things in sync: the sticky resilience-score HUD, the runway bar, a programmatically generated SVG net (a 3×3 grid of patches, one per strategy, built at runtime from jittered node coordinates so the mesh reads as hand-tied rope rather than a rigid grid), and the exportable action plan. There is no building step, no bundler, and no backend.

Tools Used

- Claude (Anthropic, Claude Sonnet 5) wrote and iteratively revised all code and copy, via Claude.ai’s chat interface using its built-in code-execution and file-creation tools.
- Claude’s web search tool — used to source and verify current statistics and institution details (OJK & GoodStats emergency-fund survey; Kemensos cekbansos portal; Pegadaian; OJK Kontak 157; hospital cost ranges) before writing them into the app
- HTML5 / CSS3 / vanilla JavaScript — no frameworks, no external JS libraries, no build tooling
- Google Fonts (CDN) — Fraunces, IBM Plex Sans, IBM Plex Mono
- Browser-native Web APIs — dynamically generated inline SVG, the Clipboard API (with a textarea fallback for sandboxed contexts), CSS custom properties and transitions
- Node.js (`node --check`) and Python’s `html.parser` — development-time-only scripts used to validate JS syntax and HTML tag balance before each delivery; not shipped with the app

- Git — used to create an inspectable, honest commit history. Source Disclosure — How to Verify This Was Actually Built This Way
-

This project was built entirely inside a chat conversation with Claude, using Claude’s built-in code-execution and file-creation tools to write, test, and iteratively edit one HTML/CSS/JS file. No no-code builder, hosted IDE, or separate prototyping platform was involved — the “environment” was the chat itself, plus a sandboxed Linux shell Claude used to run `node --check` and an `HTML-tag-balance` script before each delivery.

The development conversation was used during creation. The complete source code and commit history are now publicly available in the project's GitHub repository, allowing judges and reviewers to inspect the development process and changes.

The project has already been published to a public GitHub repository and deployed publicly, so reviewers can access both the live website and the source code.

The project has already been published to a public GitHub repository and deployed publicly, so reviewers can access both the live website and the source code.

```
git branch -M main
```

```
git push -u origin main
```

Or, for a zero-setup public link instead of GitHub: paste `index.html` into a GitHub Gist or CodePen — both give a shareable, publicly viewable URL in under a minute.

Note: this repository was assembled by Claude in a sandboxed environment with no internet access, so it could not push to a remote itself. The commands above are the last step, and they are for the users to run.

Updated AI & Development Stack

- Claude Sonnet 5 – primary development and architecture
- Google Gemini API (model `gemini 3 flash`) – AI features and intelligent assistance integrated via API key to make a chatbot
- GitHub Copilot – code completion and productivity
- Vercel – deployment and hosting
- HTML5, CSS3, Vanilla JavaScript
- Git for version control
- ChatGPT – image generation idea sampling

Data Sources

Every statistic and institution named in the app was checked against a live web source before being written in. Grouped by what they support:

Headline statistic — 70% of Indonesians have no emergency fund

- [Kompas.com: “Krisis Tabungan di RI, 70 Persen Warga Indonesia Hidup Tanpa Dana Darurat”](#) (28 Jul 2025), citing OJK data and the 2024 GoodStats survey

Social-assistance screening — PKH, BPNT, PBI-JK

- [cekbansos.kemensos.go.id](#) — official Ministry of Social Affairs portal
- [Kompas TV: “Cara Cek Bansos PKH-BPNT Juli 2026”](#)
- [Readers.id: “Kemensos Permudah Cek Bansos PKH dan BPNT 2026”](#) — BPNT/Rp200,000 per month detail, PBI-JK detail

OJK Kontak 157

- [OJK: infographic on combating illegal online loans](#)
- [kontak157.ojk.go.id](#) — official Kontak 157 consumer-complaint portal
- [OJK: Kontak 157 operating hours](#)

Pegadaian

- [Pegadaian: “Alternatif Pinjam Uang Tanpa Bunga di Pegadaian”](#)
- [Pegadaian: “Syarat dan Jenis Pinjaman di Pegadaian”](#)

Hospital cost range (used for the “medical bill” shock card)

- [Tuwaga.id: “Biaya Rawat Inap di Rumah Sakit Tanpa BPJS”](#)
- [Roojai.co.id: “Berapa Biaya Rawat Inap Rumah Sakit Tanpa BPJS?”](#)

2026 minimum wage (background context for slider defaults)

- [Bisnis.com: “Daftar UMP 2026 Tertinggi dan Terendah di Indonesia”](#)
- [Pajakku.com: “Daftar UMP 2026 Terlengkap di 38 Provinsi Indonesia”](#)

Honest gap: the cost ranges shown on the “vehicle breakdown” and “home/appliance repair” shock cards are illustrative estimates. No single authoritative Indonesian source was found for those two figures, so they should not be treated as official data — unlike every other figure above, which traces back to a live source.