

Prompt — Build a VAR Dashboard (Claude in VS Code)

Copy everything below the line into Claude Code, in a folder containing `dashboard_data/`, `dashboard_best_practices.md`, and (optionally) an `assets/` folder with your college seal or logo.

Tip: before building, you can ask Claude to adapt this prompt to your college — your name, your colors, and whichever data files you actually have — and save the adapted version over this one.

Build a public-facing **Vision Aligned Reporting (VAR) dashboard** for the college whose data is in `dashboard_data/`. VAR is the California Community Colleges' framework for reporting student services, student outcomes, and equity against the system's Vision 2030 goals.

Before writing any code, read `dashboard_best_practices.md` and follow it strictly — it is the design authority for this project. Then read `dashboard_data/README.md`, `metric_definitions.json`, and `data_notes.json` to understand every file and its caveats. The site presents only what is in `dashboard_data/` — nothing else, and nothing from files the sections below don't name.

Identify the college from the data files (`data_notes.json` names it). If the repo doesn't say, use the college name given in your instructions — and if you have neither, ask before building.

What to build

A single self-contained static site: one `index.html`, no CDNs, no external fonts, no chart libraries — hand-rolled SVG/CSS charts with the data inlined (a small build script that reads `dashboard_data/` and writes the finished page is ideal). It must work opened from a file and dropped onto Netlify.

Four sections behind a centered pill toggle, plus a footer on every view:

1. **Vision 2030: At a Glance** — KPI tiles for the four Vision 2030 outcomes: awards conferred, transfers to four-year, median units at completion (vs. the 60-unit minimum), and workforce — which has no local data and is an honest labeled placeholder pointing to the state's published figure. Timely completion (within 3 years) rides along as a sub-stat. Trends vs. prior year.
2. **Completion & Momentum** — awards by type and year, median units trend with the 60-unit reference line, transfers to four-year trend. The college's VAR narrative (from `narrative.json`) sits beside the charts it talks about — narrative next to numbers is the point of VAR, not an appendix.
3. **Student Services** — the heart of VAR: students served (headline number + by program), service contacts by category, direct aid. Contacts, dollars, and hours are different units — separate charts, labeled, never summed.
4. **Equity Gap Analysis** — outcome rates by student group shown as **lollipop charts**: each group a dot on a stem, measured against a vertical line marking the collegewide rate, switchable across the metrics in `equity_outcomes.csv`; and the participation panel from `equity_participation_share.csv` — per program, each group's share of students served vs. its share of the college, with the proportionality index and its plain-language verdict shown as a badge.

Footer (all views): four closed dropdowns — Methodology, Data Protection, Sources, Glossary of Terms — built from `data_notes.json` and `metric_definitions.json`, including any data-quality indicators and disclaimers those files carry.

A single academic-year selector scopes the tiles and per-year charts; trend charts always show all years with the selected year marked; equity pools cohorts and says so.

Hard rules

- Aggregates only, exactly as shipped — never re-derive or un-suppress.
- Suppressed cells (<10) render visibly (hatched, with an explanation on hover) — never silently dropped. Blank means "not yet observable," never 0.
- Branding comes from the college: look for an `assets/` folder with a seal or logo (compress it for the header) and brand colors. Pick one dark brand hue for chrome/headings and one warm accent, then derive and validate chart colors from them per the best-practices doc. No assets and no guidance from the user? Use a restrained two-hue institutional palette, a simple circular monogram, and say so in your summary.
- Every value must be reachable without hovering (labels or a "view as table" option), and the page must hold up at phone width.

Finish

Open the result in a browser and actually look at every section — label collisions, balance, masked-cell rendering, an early data year — before calling it done. Then start a local host serving the site folder: check whether port 8000 is free and use it if so (otherwise pick a nearby open port), open the URL in the browser, and share it. Finish by summarizing what you built and any data quirks you hit in a few plain sentences.